

Linear-Programming Decoding of Nonbinary Linear Codes

Mark F. Flanagan, *Member, IEEE*, Vitaly Skachek, *Member, IEEE*, Eimear Byrne, and Marcus Greferath

Abstract—A framework for linear-programming (LP) decoding of nonbinary linear codes over rings is developed. This framework facilitates LP-based reception for coded modulation systems which use direct modulation mapping of coded symbols. It is proved that the resulting LP decoder has the “maximum-likelihood (ML) certificate” property. It is also shown that the decoder output is the lowest cost pseudocodeword. Equivalence between pseudocodewords of the linear program and pseudocodewords of graph covers is proved. It is also proved that if the modulator–channel combination satisfies a particular symmetry condition, the codeword error rate performance is independent of the transmitted codeword. Two alternative polytopes for use with LP decoding are studied, and it is shown that for many classes of codes these polytopes yield a complexity advantage for decoding. These polytope representations lead to polynomial-time decoders for a wide variety of classical nonbinary linear codes. LP decoding performance is illustrated for the [11,6] ternary Golay code with ternary phase-shift keying (PSK) modulation over additive white Gaussian noise (AWGN), and in this case it is shown that the performance of the LP decoder is comparable to codeword-error-rate-optimum hard-decision-based decoding. LP decoding is also simulated for medium-length ternary and quaternary low-density parity-check (LDPC) codes with corresponding PSK modulations over AWGN.

Index Terms—Coded modulation, linear-programming decoding, low-density parity-check (LDPC) codes, pseudocodewords.

I. INTRODUCTION

LOW-density parity-check (LDPC) codes [1] have become very popular in recent years due to their excellent performance under *sum-product* (SP) decoding (or *message-passing* decoding). The primary research focus in this area to date has been on *binary* LDPC codes. Finite-length analysis of such LDPC codes under SP decoding is a difficult task. An approach to such an analysis was proposed in [2] based on the consideration of so-called *pseudocodewords* and their *pseudoweights*,

Manuscript received April 25, 2008; revised March 31, 2009. Current version published August 19, 2009. This work was supported in part by the Claude Shannon Institute for Discrete Mathematics, Coding and Cryptography (Science Foundation Ireland under Grant 06/MI/006). The material in this paper was presented in part at the 7th International ITG Conference on Source and Channel Coding (SCC), Ulm, Germany, January 2008, and in part at the IEEE International Symposium on Information Theory (ISIT), Toronto, ON, Canada, July 2008.

M. F. Flanagan is with the School of Electrical, Electronic and Mechanical Engineering, University College Dublin, Belfield, Dublin 4, Ireland (e-mail: mark.flanagan@ieee.org).

V. Skachek, E. Byrne, and M. Greferath are with the Claude Shannon Institute and the School of Mathematical Sciences, University College Dublin, Belfield, Dublin 4, Ireland (e-mail: vitaly.skachek@ucd.ie; ebyrne@ucd.ie; marcus.greferath@ucd.ie).

Communicated by I. Sason, Associate Editor for Coding Theory.

Color versions of Figures 4 and 5 in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TIT.2009.2025571

defined with respect to a structure called the *computation tree*. By replacing this set of pseudocodewords with another set defined with respect to cover graphs of the Tanner graph (here called *graph-cover pseudocodewords*), the analysis was found to be significantly more tractable while still yielding accurate experimental results [3]–[5].

In [6] and [7], the decoding of *binary* LDPC codes using *linear-programming* (LP) decoding was proposed, and many important connections between LP decoding and classical message-passing decoding were established. In particular, it was shown that the LP decoder is inhibited by a set of pseudocodewords corresponding to points in the LP relaxation polytope with rational coordinates (here called *linear-programming pseudocodewords*), and that the set of these pseudocodewords is equivalent to the set of graph-cover pseudocodewords. This represents a major result as it indicates that essentially the same phenomenon determines performance of LDPC codes under both LP and SP decoding.

For high-data-rate communication systems, bandwidth-efficient signaling schemes are required which necessitate the use of higher order (nonbinary) modulation. Of course, within such a framework it is desirable to use state-of-the-art error-correcting codes. Regarding the combination of LDPC coding and higher order modulation, bit-interleaved coded modulation (BICM) [8] is a high-performance method which cascades the operations of binary coding, interleaving and higher order constellation mapping. Here however the problem of system analysis is exacerbated by the complication of joint design of binary code, interleaver, and constellation mapping; this becomes even more difficult when feedback is included from the decoder to the demodulator [9].

Alternatively, higher order modulation may be achieved in conjunction with coding by the use of nonbinary codes whose symbols map directly to modulation signals. A study of such codes over rings, for use with phase-shift keying (PSK) modulation, was performed in [10], with particular focus on the ring of integers modulo 8. Nonbinary LDPC codes over fields have been investigated with direct mapping to binary [11] and nonbinary [12]–[15] modulation signals; in all of this work, SP decoding (with respect to the nonbinary alphabet) was assumed. Recently, some progress has been made on the topic of analysis of such codes; in particular, pseudocodewords of nonbinary codes were defined and some bounds on the pseudoweights were derived [16].

In this work, we extend the approach in [7] towards coded modulation, in particular to codes over rings mapped to nonbinary modulation signals. As was done in [7], we show that the problem of decoding may be formulated as an LP problem for the nonbinary case. We also show that an appropriate relax-

ation of the LP leads to a solution which has the “maximum-likelihood (ML) certificate” property, i.e., if the LP outputs a codeword, then it must be the ML codeword. Moreover, we show that if the LP output is integral, then it must correspond to the ML codeword. We define the *graph-cover pseudocodewords* of the code, and the *linear-programming pseudocodewords* of the code, and prove the equivalence of these two concepts. This shows that the links between LP decoding on the relaxation polytope and message-passing decoding on the Tanner graph generalize to the nonbinary case. Of course, while we use the term “nonbinary” throughout this paper, our framework includes the binary framework as a special case.

For coded modulation systems using ML decoding, the concept of *geometric uniformity* [17] was introduced as a condition which, if satisfied, guarantees codeword error rate (WER) performance independent of the transmitted codeword (this condition was used for design of the coded modulation systems in [10]). An analogous symmetry condition was defined in [18] for binary codes over GF(2) with SP decoding; this was later extended to nonbinary codes over GF(q) by invoking the concept of *coset LDPC codes* [13], [14]. We show that for the present framework, there exists a symmetry condition under which the WER performance is independent of the transmitted codeword. This provides a condition somewhat akin to geometric uniformity for the present framework. It is noteworthy that the same symmetry condition has recently been shown to yield codeword-independent decoder performance in the context of SP decoding [19] and also in the context of ML decoding [20]. In particular, this identifies a “natural” mapping for nonbinary codes mapped to PSK modulation, where LP, SP, or ML decoding is used with direct modulation mapping of coded symbols.

For the binary framework, alternative polytope representations were studied which gave a complexity advantage in certain scenarios [6], [7], [21]–[23]. Analogous to these works, we define two alternative polytope representations, which offer a smaller number of variables and constraints for many classes of nonbinary codes. We compare these representations with the original polytope, and show that both of them have equal error-correcting performance to the original LP relaxation. Both of these representations lead to polynomial-time decoders for a wide variety of classical nonbinary linear codes.

To demonstrate performance, LP decoding is simulated for the ternary Golay code mapped to ternary PSK over additive white Gaussian noise (AWGN), and the LP decoder is seen to perform approximately as well as codeword-error-rate optimum hard-decision decoding, and approximately 1.5 dB from the union bound for codeword-error-rate optimum soft-decision decoding.

The paper is organized as follows. Section II introduces general settings and notation. The nonbinary decoding problem is formulated as an LP problem in Section III, and basic properties of the decoding polytope are studied in Section IV. A sufficient condition for codeword-independent performance of the decoder is presented in Section V. *Linear-programming pseudocodewords* are defined in Section VI, and their properties are discussed. Their equivalence to the *graph-cover pseudocodewords* is shown in Section VII. Two alternative polytope representations are presented in Sections VIII and IX, both of which

have equivalent performance to the original but may provide lower complexity decoding. Simulation results are presented in Section X for some example coded modulation systems. Finally, some directions for future research are proposed in Section XI.

II. GENERAL SETTINGS

We consider codes over finite rings (this includes codes over finite fields, but may be more general). Denote by \mathfrak{R} a ring with q elements, by 0 its additive identity, and let $\mathfrak{R}^- = \mathfrak{R} \setminus \{0\}$. Let \mathcal{C} be a code of length n over the ring \mathfrak{R} , defined by

$$\mathcal{C} = \{\mathbf{c} \in \mathfrak{R}^n : \mathbf{c}\mathcal{H}^T = \mathbf{0}\} \quad (1)$$

where \mathcal{H} is an $m \times n$ matrix (with entries from \mathfrak{R}) called the *parity-check matrix* of the code \mathcal{C} . Obviously, the code \mathcal{C} may admit more than one parity-check matrix; however, we will consider that the parity-check matrix \mathcal{H} is fixed in this paper.

Linearity of the code \mathcal{C} follows directly from (1). Also, the *rate* of the code \mathcal{C} is defined as $R(\mathcal{C}) = \log_q(|\mathcal{C}|)/n$ and is equal to the number of information symbols per coded symbol. The code \mathcal{C} may then be referred to as an $[n, \log_q(|\mathcal{C}|)]$ linear code over \mathfrak{R} .

Denote the set of column indices and the set of row indices of \mathcal{H} by $\mathcal{I} = \{1, 2, \dots, n\}$ and $\mathcal{J} = \{1, 2, \dots, m\}$, respectively. We use notation \mathcal{H}_j for the j th row of \mathcal{H} , where $j \in \mathcal{J}$. Denote by $\text{supp}(\mathbf{c})$ the support of a vector \mathbf{c} . For each $j \in \mathcal{J}$, let $\mathcal{I}_j = \text{supp}(\mathcal{H}_j)$ and $d_j = |\mathcal{I}_j|$, and let $d = \max_{j \in \mathcal{J}} \{d_j\}$.

Given any $\mathbf{c} \in \mathfrak{R}^n$, we say that parity check $j \in \mathcal{J}$ is *satisfied* by \mathbf{c} if and only if

$$\mathbf{c}\mathcal{H}_j^T = \sum_{i \in \mathcal{I}_j} c_i \cdot \mathcal{H}_{j,i} = 0. \quad (2)$$

For $j \in \mathcal{J}$, define the single parity-check code \mathcal{C}_j over \mathfrak{R} by

$$\mathcal{C}_j = \left\{ (b_i)_{i \in \mathcal{I}_j} : \sum_{i \in \mathcal{I}_j} b_i \cdot \mathcal{H}_{j,i} = 0 \right\}.$$

Note that while the symbols of the codewords in \mathcal{C} are indexed by \mathcal{I} , the symbols of the codewords in \mathcal{C}_j are indexed by \mathcal{I}_j . We define the projection mapping for parity-check $j \in \mathcal{J}$ by

$$\mathbf{x}_j(\mathbf{c}) = (c_i)_{i \in \mathcal{I}_j}.$$

Then, given any $\mathbf{c} \in \mathfrak{R}^n$, we may say that parity-check $j \in \mathcal{J}$ is satisfied by \mathbf{c} if and only if

$$\mathbf{x}_j(\mathbf{c}) \in \mathcal{C}_j \quad (3)$$

since (2) and (3) are equivalent. Also, it is easily seen that $\mathbf{c} \in \mathcal{C}$ if and only if all parity checks $j \in \mathcal{J}$ are satisfied by \mathbf{c} . In this case, we say that \mathbf{c} is a *codeword* of \mathcal{C} .

We shall take an example which shall be used to illustrate concepts throughout this paper. Consider the $[4, 2]$ linear code over $\mathfrak{R} = \mathbb{Z}_3$ with parity-check matrix

$$\mathcal{H} = \begin{pmatrix} 1 & 2 & 2 & 1 \\ 2 & 0 & 1 & 2 \end{pmatrix}. \quad (4)$$

Here $\mathcal{I}_1 = \{1, 2, 3, 4\}$, $\mathcal{I}_2 = \{1, 3, 4\}$, and the two single parity-check codes \mathcal{C}_1 and \mathcal{C}_2 , of length $d_1 = 4$ and $d_2 = 3$, respectively, are given by

$$\mathcal{C}_1 = \{(b_1 \ b_2 \ b_3 \ b_4) : b_1 + 2b_2 + 2b_3 + b_4 = 0\}$$

and

$$\mathcal{C}_2 = \{(b_1 \ b_3 \ b_4) : 2b_1 + b_3 + 2b_4 = 0\}.$$

III. DECODING AS A LINEAR-PROGRAMMING PROBLEM

Assume that the codeword $\bar{\mathbf{c}} = (\bar{c}_1, \bar{c}_2, \dots, \bar{c}_n) \in \mathcal{C}$ has been transmitted over a q -ary input memoryless channel, and a corrupted word $\mathbf{y} = (y_1, y_2, \dots, y_n) \in \Sigma^n$ has been received. Here Σ denotes the set of channel output symbols; we assume that this set either has finite cardinality, or is equal to \mathbb{R}^l or \mathbb{C}^l for some integer $l \geq 1$. In practice, this channel may represent the combination of modulator and physical channel. We assume hereafter that all information words are equally probable, and so all codewords are transmitted with equal probability.

It was suggested in [6] to represent each symbol as a binary vector of length $|\mathfrak{R}^-|$, where the entries in the vector are indicators of a symbol taking on a particular value. Below, we show how this representation may lead to a generalization of the framework of [7] to the case of nonbinary coding. This generalization is nontrivial since, while such a representation converts the nonbinary code into a binary code, this binary code is not linear and therefore the analysis in [6], [7] is not directly applicable.

For use in the following derivation, we shall define the mapping

$$\boldsymbol{\xi} : \mathfrak{R} \longrightarrow \{0, 1\}^{q-1} \subset \mathbb{R}^{q-1}$$

by

$$\boldsymbol{\xi}(\alpha) = \mathbf{x} = (x^{(\gamma)})_{\gamma \in \mathfrak{R}^-}$$

such that, for each $\gamma \in \mathfrak{R}^-$

$$x^{(\gamma)} = \begin{cases} 1, & \text{if } \gamma = \alpha \\ 0, & \text{otherwise.} \end{cases}$$

We note that the mapping $\boldsymbol{\xi}$ is one-to-one, and its image is the set of binary vectors of length $q-1$ with Hamming weight 0 or 1. Building on this, we also define

$$\boldsymbol{\Xi} : \mathfrak{R}^n \longrightarrow \{0, 1\}^{(q-1)n} \subset \mathbb{R}^{(q-1)n}$$

according to

$$\boldsymbol{\Xi}(\mathbf{c}) = (\boldsymbol{\xi}(c_1) \mid \boldsymbol{\xi}(c_2) \mid \dots \mid \boldsymbol{\xi}(c_n)).$$

We note that $\boldsymbol{\Xi}$ is also one-to-one.

Now, for vectors $\mathbf{f} \in \mathbb{R}^{(q-1)n}$, we adopt the notation

$$\mathbf{f} = (\mathbf{f}_1 \mid \mathbf{f}_2 \mid \dots \mid \mathbf{f}_n)$$

where

$$\forall i \in \mathcal{I}, \mathbf{f}_i = (f_i^{(\alpha)})_{\alpha \in \mathfrak{R}^-}.$$

Also, we may use this notation to write the inverse of $\boldsymbol{\Xi}$ as

$$\boldsymbol{\Xi}^{-1}(\mathbf{f}) = (\boldsymbol{\xi}^{-1}(\mathbf{f}_1), \boldsymbol{\xi}^{-1}(\mathbf{f}_2), \dots, \boldsymbol{\xi}^{-1}(\mathbf{f}_n)).$$

We also define a function $\boldsymbol{\lambda} : \Sigma \longrightarrow (\mathbb{R} \cup \{\pm\infty\})^{q-1}$ by

$$\boldsymbol{\lambda} = (\lambda^{(\alpha)})_{\alpha \in \mathfrak{R}^-}$$

where, for each $y \in \Sigma$, $\alpha \in \mathfrak{R}^-$

$$\lambda^{(\alpha)}(y) = \log \left(\frac{p(y|0)}{p(y|\alpha)} \right)$$

and $p(y|c)$ denotes the channel output probability (density) conditioned on the channel input. We extend this to a map on Σ^n by defining $\boldsymbol{\Lambda}(\mathbf{y}) = (\boldsymbol{\lambda}(y_1) \mid \boldsymbol{\lambda}(y_2) \mid \dots \mid \boldsymbol{\lambda}(y_n))$.

The codeword-error-rate-optimum receiver operates according to the *maximum a posteriori probability* (MAP) decision rule

$$\begin{aligned} \hat{\mathbf{c}} &= \arg \max_{\mathbf{c} \in \mathcal{C}} p(\mathbf{c}|\mathbf{y}) \\ &= \arg \max_{\mathbf{c} \in \mathcal{C}} \frac{p(\mathbf{y}|\mathbf{c})p(\mathbf{c})}{p(\mathbf{y})}. \end{aligned}$$

Here $p(\cdot)$ denotes probability if Σ has finite cardinality, and probability density if Σ has infinite cardinality.

By assumption, the *a priori* probability $p(\mathbf{c})$ is uniform over codewords, and $p(\mathbf{y})$ is independent of \mathbf{c} . Therefore, the decision rule reduces to ML decoding

$$\begin{aligned} \hat{\mathbf{c}} &= \arg \max_{\mathbf{c} \in \mathcal{C}} p(\mathbf{y}|\mathbf{c}) \\ &= \arg \max_{\mathbf{c} \in \mathcal{C}} \prod_{i=1}^n p(y_i|c_i) \\ &= \arg \max_{\mathbf{c} \in \mathcal{C}} \sum_{i=1}^n \log(p(y_i|c_i)) \\ &= \arg \min_{\mathbf{c} \in \mathcal{C}} \sum_{i=1}^n \log \left(\frac{p(y_i|0)}{p(y_i|c_i)} \right) \\ &= \arg \min_{\mathbf{c} \in \mathcal{C}} \sum_{i=1}^n \boldsymbol{\lambda}(y_i) \boldsymbol{\xi}(c_i)^T \\ &= \arg \min_{\mathbf{c} \in \mathcal{C}} \boldsymbol{\Lambda}(\mathbf{y}) \boldsymbol{\Xi}(\mathbf{c})^T \end{aligned}$$

where we have made use of the memoryless property of the channel, and of the fact that if $c_i = \alpha \in \mathfrak{R}^-$, then $\boldsymbol{\lambda}(y_i) \boldsymbol{\xi}(c_i)^T = \lambda^{(\alpha)}(y_i)$. This is then equivalent to

$$\hat{\mathbf{c}} = \boldsymbol{\Xi}^{-1}(\hat{\mathbf{f}})$$

where

$$\hat{\mathbf{f}} = \arg \min_{\mathbf{f} \in \mathcal{K}(\mathcal{C})} \boldsymbol{\Lambda}(\mathbf{y}) \mathbf{f}^T \quad (5)$$

and $\mathcal{K}(\mathcal{C})$ represents the convex hull of all points $\mathbf{f} \in \mathbb{R}^{(q-1)n}$ which correspond to codewords, i.e.,

$$\mathcal{K}(\mathcal{C}) = H_{\text{conv}} \{ \boldsymbol{\Xi}(\mathbf{c}) : \mathbf{c} \in \mathcal{C} \}.$$

Therefore, it is seen that the ML decoding problem reduces to the minimization of a linear objective function (or cost function) over a polytope in $\mathbb{R}^{(q-1)n}$. The number of variables and constraints for this linear program is exponential in n , and it is

therefore too complex for practical implementation. To circumvent this problem, we formulate a relaxed LP problem, as shown next.

The solution we seek for \mathbf{f} (i.e., the desired LP output) is

$$\mathbf{f} = \Xi(\bar{\mathbf{c}}) = (\xi(\bar{c}_1) \mid \xi(\bar{c}_2) \mid \dots \mid \xi(\bar{c}_n)). \quad (6)$$

Note that (6) implies that the solution we seek for each \mathbf{f}_i ($i \in \mathcal{I}$) is an indicator function which “points” to the i th transmitted symbol \bar{c}_i , i.e.,

$$\forall i \in \mathcal{I} : f_i^{(\alpha)} = \begin{cases} 1, & \text{if } \alpha = \bar{c}_i \\ 0, & \text{otherwise.} \end{cases}$$

We now introduce auxiliary variables whose constraints, along with those of the elements of \mathbf{f} , will form the relaxed LP problem. We denote these auxiliary variables by

$$w_{j,\mathbf{b}}, \quad \text{for } j \in \mathcal{J}, \mathbf{b} \in \mathcal{C}_j$$

and we denote the vector containing these variables as

$$\mathbf{w} = (\mathbf{w}_j)_{j \in \mathcal{J}}, \quad \text{where } \mathbf{w}_j = (w_{j,\mathbf{b}})_{\mathbf{b} \in \mathcal{C}_j} \quad \forall j \in \mathcal{J}.$$

The solution we seek for these variables is

$$\forall j \in \mathcal{J} : w_{j,\mathbf{b}} = \begin{cases} 1, & \text{if } \mathbf{b} = \mathbf{x}_j(\bar{\mathbf{c}}) \\ 0, & \text{otherwise.} \end{cases} \quad (7)$$

Note that the solution we seek for each \mathbf{w}_j ($j \in \mathcal{J}$) is an indicator function which “points” to the j th transmitted local codeword $\mathbf{x}_j(\bar{\mathbf{c}})$. Based on (7), we impose the constraints

$$\forall j \in \mathcal{J}, \forall \mathbf{b} \in \mathcal{C}_j, \quad w_{j,\mathbf{b}} \geq 0 \quad (8)$$

and

$$\forall j \in \mathcal{J}, \quad \sum_{\mathbf{b} \in \mathcal{C}_j} w_{j,\mathbf{b}} = 1. \quad (9)$$

Finally, we note that the solution we seek (given by the combination of (6) and (7)) satisfies the further constraints

$$\forall j \in \mathcal{J}, \quad \forall i \in \mathcal{I}_j, \quad \forall \alpha \in \mathfrak{R}^-, \quad f_i^{(\alpha)} = \sum_{\mathbf{b} \in \mathcal{C}_j, b_i = \alpha} w_{j,\mathbf{b}}. \quad (10)$$

It is interesting to note that from (8) and (9), each vector \mathbf{w}_j (for $j \in \mathcal{J}$) may be interpreted as a probability distribution for the local codeword $\mathbf{b} \in \mathcal{C}_j$, in which case each \mathbf{f}_i (for $i \in \mathcal{I}$) has a natural interpretation (via (10)) as the corresponding probability distribution for the i th coded symbol $c_i \in \mathfrak{R}$. The following example illustrates the connection (10) between \mathbf{f} and \mathbf{w} .

Example 3.1: Consider the example [4, 2] code over \mathbb{Z}_3 defined by the parity-check matrix (4). The second row \mathcal{H}_2 of the parity-check matrix corresponds to the parity-check equation

$$2b_1 + b_3 + 2b_4 = 0$$

over \mathbb{Z}_3 . Here $\mathbf{b} = (b_1 \ b_3 \ b_4) \in \mathcal{C}_2$. Assume that the values of $w_{2,\mathbf{b}}$ for $\mathbf{b} \in \mathcal{C}_2$ are as given in the following table:

$b_1 b_3 b_4$	$w_{2,\mathbf{b}}$	$b_1 b_3 b_4$	$w_{2,\mathbf{b}}$	$b_1 b_3 b_4$	$w_{2,\mathbf{b}}$
000	0.01	102	0.05	201	0.15
011	0.04	110	0.07	212	0.32
022	0.05	121	0.08	220	0.23

Then, some of the values of $f_i^{(\alpha)}$ are as follows:

$$\begin{aligned} f_1^{(2)} &= 0.15 + 0.32 + 0.23 = 0.7 \\ f_2^{(1)} &= 0.04 + 0.07 + 0.32 = 0.43 \\ f_3^{(2)} &= 0.05 + 0.05 + 0.32 = 0.42. \end{aligned}$$

Constraints (8)–(10) may be interpreted as the statement that for all $j \in \mathcal{J}$, the vector $\mathbf{f}_j = (\mathbf{f}_i)_{i \in \mathcal{I}_j}$ lies in the convex hull $\mathcal{K}(\mathcal{C}_j)$. Constraints (8)–(10) form a polytope which we denote by \mathcal{Q} . The minimization of the objective function (5) over \mathcal{Q} forms the relaxed LP decoding problem. This LP is defined by $O(qn + q^d m)$ variables and $O(qn + q^d m)$ constraints, and therefore, the number of variables and of constraints scales as approximately q^d .

We note that the further constraints

$$\forall j \in \mathcal{J}, \quad \forall \mathbf{b} \in \mathcal{C}_j, \quad w_{j,\mathbf{b}} \leq 1 \quad (11)$$

$$\forall i \in \mathcal{I}, \quad \forall \alpha \in \mathfrak{R}^-, \quad 0 \leq f_i^{(\alpha)} \leq 1 \quad (12)$$

and

$$\forall i \in \mathcal{I}, \quad \sum_{\alpha \in \mathfrak{R}^-} f_i^{(\alpha)} \leq 1. \quad (13)$$

follow from the constraints (8)–(10), for any $(\mathbf{f}, \mathbf{w}) \in \mathcal{Q}$.

Now we may define the decoding algorithm, which works as follows. The decoder solves the LP problem of minimizing the objective function (5) subject to the constraints (8)–(10). If $\mathbf{f} \in \{0, 1\}^{(q-1)n}$, the output is the codeword $\Xi^{-1}(\mathbf{f})$ (we shall prove in the next section that this output is indeed a codeword). This codeword may then be the correct one (we call this “correct decoding”) or an incorrect one (we call this “incorrect decoding”). If $\mathbf{f} \notin \{0, 1\}^{(q-1)n}$, the decoder reports a “decoding failure.” Note that in this paper, we say that the decoder makes a *codeword error* when the decoder output is not equal to the transmitted codeword (this could correspond to a “decoding failure,” or to an “incorrect decoding”).

The time complexity of an LP solver depends on the number of variables and constraints in the LP problem. The *simplex method* is a popular and practically efficient algorithm for solving LP problems. However, its worst case time complexity has been shown to be exponential in the number of variables. There are other known LP solvers, such as solvers that are based on interior-point methods [24, Ch. 11], which have time complexity polynomial in the number of variables and constraints. For more detail the reader may also refer to [25]. We note, however, that the standard iterative decoding algorithms (such as the min-sum or sum-product algorithms) have time complexity which is linear in the block length of the code, and therefore significantly outperform the LP decoder in terms of efficiency.

IV. POLYTOPE PROPERTIES

The analysis in this section is a direct generalization of the results in [7].

Definition 4.1: An *integral point* in a polytope is a point with all *integer* coordinates.

Proposition 4.1:

- 1) Let $(\mathbf{f}, \mathbf{w}) \in \mathcal{Q}$, and $f_i^{(\alpha)} \in \{0, 1\}$ for every $i \in \mathcal{I}$, $\alpha \in \mathfrak{R}^-$. Then $\Xi^{-1}(\mathbf{f}) \in \mathcal{C}$.
- 2) Conversely, for every codeword $\mathbf{c} \in \mathcal{C}$, there exists \mathbf{w} such that (\mathbf{f}, \mathbf{w}) is an integral point in \mathcal{Q} with $\mathbf{f} = \Xi(\mathbf{c})$.

Proof:

- 1) Suppose $(\mathbf{f}, \mathbf{w}) \in \mathcal{Q}$, and $f_i^{(\alpha)} \in \{0, 1\}$ for every $i \in \mathcal{I}$, $\alpha \in \mathfrak{R}^-$.

Let $\mathbf{c} = \Xi^{-1}(\mathbf{f})$; by (13), this is well defined. Now, fix some $j \in \mathcal{J}$ and define $\mathbf{t} = \mathbf{x}_j(\mathbf{c})$. Note that from these definitions it follows that for any $i \in \mathcal{I}$, $\alpha \in \mathfrak{R}^-$, $f_i^{(\alpha)} = 1$ if and only if $t_i = \alpha$. Now let $\mathbf{r} \in \mathcal{C}_j$, $\mathbf{r} \neq \mathbf{t}$. Since \mathbf{r} and \mathbf{t} are distinct, there must exist $\alpha \in \mathfrak{R}^-$ and $l \in \mathcal{I}_j$ such that either $r_l = \alpha$ and $t_l \neq \alpha$, or $t_l = \alpha$ and $r_l \neq \alpha$. We examine these two cases separately.

- If $r_l = \alpha$ and $t_l \neq \alpha$, then by (10)

$$f_l^{(\alpha)} = 0 = \sum_{\mathbf{b} \in \mathcal{C}_j, b_l = \alpha} w_{j,\mathbf{b}}.$$

Therefore, $w_{j,\mathbf{b}} = 0$ for all $\mathbf{b} \in \mathcal{C}_j$ with $b_l = \alpha$, and in particular $w_{j,\mathbf{r}} = 0$.

- If $t_l = \alpha$ and $r_l \neq \alpha$, then by (9) and (10)

$$\begin{aligned} 0 &= 1 - f_l^{(\alpha)} \\ &= \sum_{\mathbf{b} \in \mathcal{C}_j} w_{j,\mathbf{b}} - \sum_{\mathbf{b} \in \mathcal{C}_j, b_l = \alpha} w_{j,\mathbf{b}} \\ &= \sum_{\mathbf{b} \in \mathcal{C}_j, b_l \neq \alpha} w_{j,\mathbf{b}}. \end{aligned}$$

Therefore, $w_{j,\mathbf{b}} = 0$ for all $\mathbf{b} \in \mathcal{C}_j$ with $b_l \neq \alpha$, and in particular $w_{j,\mathbf{r}} = 0$.

It follows that $w_{j,\mathbf{r}} = 0$ for all $\mathbf{r} \in \mathcal{C}_j$, $\mathbf{r} \neq \mathbf{t}$. But by (9) this implies that $\mathbf{t} \in \mathcal{C}_j$ (and that $w_{j,\mathbf{t}} = 1$). Applying this argument for every $j \in \mathcal{J}$ implies $\mathbf{c} \in \mathcal{C}$.

- 2) For $\mathbf{c} \in \mathcal{C}$, we let $\mathbf{f} = \Xi(\mathbf{c})$. For each parity check $j \in \mathcal{J}$, we let $\mathbf{t} = \mathbf{x}_j(\mathbf{c}) \in \mathcal{C}_j$ and then set

$$\forall j \in \mathcal{J} : w_{j,\mathbf{b}} = \begin{cases} 1, & \text{if } \mathbf{b} = \mathbf{t} \\ 0, & \text{otherwise.} \end{cases}$$

It is easily checked that the resulting point (\mathbf{f}, \mathbf{w}) is integral and satisfies constraints (8)–(10). \square

The following proposition assures the so-called *ML certificate* property.

Proposition 4.2: Suppose that the decoder outputs a codeword $\mathbf{c} \in \mathcal{C}$. Then, \mathbf{c} is the ML codeword.

The proof of this proposition is straightforward. The reader can refer to a similar proof for the binary case in [7].

V. CODEWORD-INDEPENDENT DECODER PERFORMANCE

In this section, we state and prove a theorem on decoder performance, namely, that under a certain symmetry condition, the probability of codeword error is independent of the transmitted codeword. The proof generalizes the corresponding proof for the binary case which may be found in [6], [7].

A. Symmetry Condition

For each $\beta \in \mathfrak{R}$, there exists a bijection

$$\tau_\beta : \Sigma \longrightarrow \Sigma$$

such that the channel output probability (density) conditioned on the channel input satisfies

$$p(y|\alpha) = p(\tau_\beta(y)|\alpha - \beta) \quad (14)$$

$\forall y \in \Sigma$, $\alpha \in \mathfrak{R}$. When Σ is equal to \mathbb{R}^l or \mathbb{C}^l for $l \geq 1$, the mapping τ_β is assumed to be isometric with respect to Euclidean distance in Σ , for every $\beta \in \mathfrak{R}$.

Note that the symmetry condition above is very similar to that introduced in [20] which guarantees codeword-independent performance under ML decoding.

Theorem 5.1: Under the stated symmetry condition, the probability of codeword error is independent of the transmitted codeword.

Proof: We shall prove the theorem for the case where Σ has infinite cardinality; the case of discrete Σ may be handled similarly. Fix some codeword $\mathbf{c} \in \mathcal{C}$, $\mathbf{c} \neq \mathbf{0}$. We wish to prove that

$$\Pr(\text{Err}|\mathbf{c}) = \Pr(\text{Err}|\mathbf{0})$$

where $\Pr(\text{Err}|\mathbf{c})$ denotes the probability of codeword error given that the codeword \mathbf{c} was transmitted.

Now

$$\Pr(\text{Err}|\mathbf{c}) = \Pr(\mathbf{y} \in B(\mathbf{c})|\mathbf{c})$$

where

$$\begin{aligned} B(\mathbf{c}) &= \{\mathbf{y} \in \Sigma^n : \exists (\mathbf{f}, \mathbf{w}) \in \mathcal{Q}, \mathbf{f} \neq \Xi(\mathbf{c}) \\ &\quad \text{with } \Lambda(\mathbf{y})\mathbf{f}^T \leq \Lambda(\mathbf{y})\Xi(\mathbf{c})^T\}. \end{aligned}$$

Here $B(\mathbf{c})$ is the set of all received words which may cause codeword error, given that \mathbf{c} was transmitted. Recall that the elements of $\Lambda(\mathbf{y})$ are given by

$$\lambda^{(\alpha)}(y_i) = \log \left(\frac{p(y_i|0)}{p(y_i|\alpha)} \right) \quad (15)$$

for $i \in \mathcal{I}$, $\alpha \in \mathfrak{R}^-$. Also

$$\Pr(\text{Err}|\mathbf{0}) = \Pr(\mathbf{y} \in B(\mathbf{0})|\mathbf{0})$$

where

$$\begin{aligned} B(\mathbf{0}) &= \{\tilde{\mathbf{y}} \in \Sigma^n : \exists (\tilde{\mathbf{f}}, \tilde{\mathbf{w}}) \in \mathcal{Q}, \tilde{\mathbf{f}} \neq \Xi(\mathbf{0}) \\ &\quad \text{with } \Lambda(\tilde{\mathbf{y}})\tilde{\mathbf{f}}^T \leq \Lambda(\tilde{\mathbf{y}})\Xi(\mathbf{0})^T\}. \end{aligned}$$

So we write

$$\Pr(\text{Err}|\mathbf{c}) = \int_{\mathbf{y} \in B(\mathbf{c})} p(\mathbf{y}|\mathbf{c}) d\mathbf{y} \quad (16)$$

and

$$\Pr(\text{Err}|\mathbf{0}) = \int_{\tilde{\mathbf{y}} \in B(\mathbf{0})} p(\tilde{\mathbf{y}}|\mathbf{0}) d\tilde{\mathbf{y}}. \quad (17)$$

Now, setting $\alpha = \beta$ in the symmetry condition (14) yields

$$p(y|\beta) = p(\tau_\beta(y)|0) \quad (18)$$

for any $y \in \Sigma$, $\beta \in \mathfrak{R}$.

We now define $\mathbf{G} : \Sigma^n \rightarrow \Sigma^n$ and $\tilde{\mathbf{y}}$ as follows:

$$\tilde{\mathbf{y}} = \mathbf{G}(\mathbf{y}), \quad \text{s.t. } \forall i \in \mathcal{I} : \tilde{y}_i = \tau_\beta(y_i), \text{ where } \beta = c_i.$$

We note that \mathbf{G} is a bijection from the set Σ^n to itself, and that if $\mathbf{y}, \mathbf{z} \in \Sigma^n$, and $\beta = c_i$ then

$$\|y_i - z_i\|^2 = \|\tau_\beta(y_i) - \tau_\beta(z_i)\|^2$$

and so

$$\|\mathbf{G}(\mathbf{y}) - \mathbf{G}(\mathbf{z})\|^2 = \|\mathbf{y} - \mathbf{z}\|^2$$

i.e., \mathbf{G} is isometric with respect to Euclidean distance in Σ^n .

We prove that the integral (16) may be transformed to (17) via the substitution $\tilde{\mathbf{y}} = \mathbf{G}(\mathbf{y})$. First, we have

$$\begin{aligned} p(\mathbf{y}|\mathbf{c}) &= \prod_{i \in \mathcal{I}} p(y_i|c_i) \\ &= \prod_{\beta \in \mathfrak{R}} \prod_{i \in \mathcal{I}, c_i = \beta} p(y_i|\beta) \\ &= \prod_{\beta \in \mathfrak{R}} \prod_{i \in \mathcal{I}, c_i = \beta} p(\tau_\beta(y_i)|0) \\ &= \prod_{\beta \in \mathfrak{R}} \prod_{i \in \mathcal{I}, c_i = \beta} p(\tilde{y}_i|0) \\ &= \prod_{i \in \mathcal{I}} p(\tilde{y}_i|0) \\ &= p(\tilde{\mathbf{y}}|\mathbf{0}). \end{aligned}$$

Since \mathbf{G} is isometric with respect to Euclidean distance in Σ^n , it follows that the Jacobian determinant of the transformation is equal to unity. Therefore, to complete the proof, we need only show that

$$\mathbf{y} \in B(\mathbf{c}) \quad \text{if and only if} \quad \tilde{\mathbf{y}} \in B(\mathbf{0}).$$

We begin by relating the elements of $\Lambda(\mathbf{y})$ to the elements of $\Lambda(\tilde{\mathbf{y}})$. Let $i \in \mathcal{I}$, $\alpha \in \mathfrak{R}^-$. Suppose $c_i = \beta \in \mathfrak{R}$. We then have

$$\begin{aligned} \lambda^{(\alpha)}(y_i) &= \log \left(\frac{p(y_i|0)}{p(y_i|\alpha)} \right) \\ &= \log \left(\frac{p(\tau_\beta(y_i)|-\beta)}{p(\tau_\beta(y_i)|\alpha-\beta)} \right) \\ &= \log \left(\frac{p(\tilde{y}_i|-\beta)}{p(\tilde{y}_i|\alpha-\beta)} \right). \end{aligned}$$

This yields

$$\lambda^{(\alpha)}(y_i) = \begin{cases} \lambda^{(\alpha)}(\tilde{y}_i), & \text{if } \beta = 0 \\ -\lambda^{(-\alpha)}(\tilde{y}_i), & \text{if } \alpha = \beta \\ \lambda^{(\alpha-\beta)}(\tilde{y}_i) - \lambda^{(-\beta)}(\tilde{y}_i), & \text{otherwise.} \end{cases}$$

Next, for any point $(\mathbf{f}, \mathbf{w}) \in \mathcal{Q}$, we define a new point $(\tilde{\mathbf{f}}, \tilde{\mathbf{w}})$ as follows. For $\beta = c_i$ and all $i \in \mathcal{I}$, $\alpha \in \mathfrak{R}^-$

$$\tilde{f}_i^{(\alpha)} = \begin{cases} 1 - \sum_{\gamma \in \mathfrak{R}^-} f_i^{(\gamma)}, & \text{if } \alpha = -\beta \\ f_i^{(\alpha+\beta)}, & \text{otherwise.} \end{cases} \quad (19)$$

For all $j \in \mathcal{J}$, $\mathbf{r} \in \mathcal{C}_j$ we define

$$\tilde{w}_{j,\mathbf{r}} = w_{j,\mathbf{b}}$$

where

$$\mathbf{b} = \mathbf{r} + \mathbf{x}_j(\mathbf{c}).$$

Next, we prove that for every $(\mathbf{f}, \mathbf{w}) \in \mathcal{Q}$, the new point $(\tilde{\mathbf{f}}, \tilde{\mathbf{w}})$ lies in \mathcal{Q} and thus is a feasible solution for the LP. Constraints (8) and (9) obviously hold from the definition of $\tilde{\mathbf{w}}$. To verify (10), we let $j \in \mathcal{J}$, $i \in \mathcal{I}_j$, and $\alpha \in \mathfrak{R}^-$. We also let $\beta = c_i$. We now check two cases.

- If $\alpha = -\beta$

$$\begin{aligned} \tilde{f}_i^{(\alpha)} &= 1 - \sum_{\gamma \in \mathfrak{R}^-} f_i^{(\gamma)} \\ &= \sum_{\mathbf{b} \in \mathcal{C}_j} w_{j,\mathbf{b}} - \sum_{\gamma \in \mathfrak{R}^-} \sum_{\mathbf{b} \in \mathcal{C}_j, b_i = \gamma} w_{j,\mathbf{b}} \\ &= \sum_{\mathbf{b} \in \mathcal{C}_j, b_i = 0} w_{j,\mathbf{b}} \\ &= \sum_{\mathbf{r} \in \mathcal{C}_j, r_i = \alpha} \tilde{w}_{j,\mathbf{r}}. \end{aligned}$$

- If $\alpha \neq -\beta$

$$\begin{aligned} \tilde{f}_i^{(\alpha)} &= f_i^{(\alpha+\beta)} = \sum_{\mathbf{b} \in \mathcal{C}_j, b_i = \alpha+\beta} w_{j,\mathbf{b}} \\ &= \sum_{\mathbf{r} \in \mathcal{C}_j, r_i = \alpha} \tilde{w}_{j,\mathbf{r}}. \end{aligned}$$

Therefore, $(\tilde{\mathbf{f}}, \tilde{\mathbf{w}}) \in \mathcal{Q}$, i.e., $(\tilde{\mathbf{f}}, \tilde{\mathbf{w}})$ is a feasible solution for the LP. We write $(\tilde{\mathbf{f}}, \tilde{\mathbf{w}}) = \tilde{\mathbf{L}}(\mathbf{f}, \mathbf{w})$. We also note that the mapping $\tilde{\mathbf{L}}$ is a bijection from \mathcal{Q} to itself; this is easily shown by verifying the inverse

$$f_i^{(\alpha)} = \begin{cases} 1 - \sum_{\gamma \in \mathfrak{R}^-} \tilde{f}_i^{(\gamma)}, & \text{if } \alpha = \beta \\ \tilde{f}_i^{(\alpha-\beta)}, & \text{otherwise} \end{cases} \quad (20)$$

for all $i \in \mathcal{I}$, $\alpha \in \mathfrak{R}^-$, and

$$w_{j,\mathbf{b}} = \tilde{w}_{j,\mathbf{r}}$$

where

$$\mathbf{r} = \mathbf{b} - \mathbf{x}_j(\mathbf{c})$$

for all $j \in \mathcal{J}$, $\mathbf{b} \in \mathcal{C}_j$.

We now prove that for every $(\mathbf{f}, \mathbf{w}) \in \mathcal{Q}$, $(\tilde{\mathbf{f}}, \tilde{\mathbf{w}}) = \tilde{\mathbf{L}}(\mathbf{f}, \mathbf{w})$ satisfies

$$\Lambda(\mathbf{y})\mathbf{f}^T - \Lambda(\mathbf{y})\Xi(\mathbf{c})^T = \Lambda(\tilde{\mathbf{y}})\tilde{\mathbf{f}}^T - \Lambda(\tilde{\mathbf{y}})\Xi(\mathbf{0})^T. \quad (21)$$

We achieve this by proving

$$\lambda(y_i)\mathbf{f}_i^T - \lambda(y_i)\boldsymbol{\xi}(c_i)^T = \lambda(\tilde{y}_i)\tilde{\mathbf{f}}_i^T - \lambda(\tilde{y}_i)\boldsymbol{\xi}(0)^T \quad (22)$$

for every $i \in \mathcal{I}$. We may then obtain (21) by summing (22) over $i \in \mathcal{I}$. Let $\beta = c_i$. We consider two cases.

- If $\beta = 0$, (22) becomes

$$\lambda(y_i)\mathbf{f}_i^T = \lambda(\tilde{y}_i)\tilde{\mathbf{f}}_i^T$$

which holds since $\lambda^{(\alpha)}(\tilde{y}_i) = \lambda^{(\alpha)}(y_i)$ and $\tilde{f}_i^{(\alpha)} = f_i^{(\alpha)}$ for all $\alpha \in \mathfrak{R}^-$ in this case.

- If $\beta \neq 0$

$$\begin{aligned} & \lambda(y_i)\mathbf{f}_i^T - \lambda(y_i)\boldsymbol{\xi}(c_i)^T \\ &= \sum_{\gamma \in \mathfrak{R}^-} \lambda^{(\gamma)}(y_i)f_i^{(\gamma)} - \lambda^{(\beta)}(y_i) \\ &= \sum_{\substack{\gamma \in \mathfrak{R}^- \\ \gamma \neq \beta}} \left(\lambda^{(\gamma-\beta)}(\tilde{y}_i) - \lambda^{(-\beta)}(\tilde{y}_i) \right) f_i^{(\gamma)} \\ &\quad - \lambda^{(-\beta)}(\tilde{y}_i)f_i^{(\beta)} + \lambda^{(-\beta)}(\tilde{y}_i) \\ &= \sum_{\substack{\alpha \in \mathfrak{R}^- \\ \alpha \neq -\beta}} \lambda^{(\alpha)}(\tilde{y}_i)f_i^{(\alpha+\beta)} \\ &\quad + \lambda^{(-\beta)}(\tilde{y}_i) \left(1 - \sum_{\gamma \in \mathfrak{R}^-} f_i^{(\gamma)} \right) \\ &= \sum_{\alpha \in \mathfrak{R}^-} \lambda^{(\alpha)}(\tilde{y}_i)\tilde{f}_i^{(\alpha)} \\ &= \lambda(\tilde{y}_i)\tilde{\mathbf{f}}_i^T - \lambda(\tilde{y}_i)\boldsymbol{\xi}(0)^T \end{aligned}$$

where we have made use of the substitution $\alpha = \gamma - \beta$ in the third line. Therefore, (22) holds, proving (21).

Finally, we note that it is easy to show, using (19) and (20), that $\mathbf{f} = \Xi(\mathbf{c})$ if and only if $\tilde{\mathbf{f}} = \Xi(\mathbf{0})$. Putting together these results, we may make the following statement. Suppose we are given $\mathbf{y}, \tilde{\mathbf{y}} \in \Sigma^n$ with $\tilde{\mathbf{y}} = \mathbf{G}(\mathbf{y})$. Then the point $(\mathbf{f}, \mathbf{w}) \in \mathcal{Q}$ satisfies $\mathbf{f} \neq \Xi(\mathbf{c})$ and $\mathbf{\Lambda}(\mathbf{y})\mathbf{f}^T \leq \mathbf{\Lambda}(\mathbf{y})\Xi(\mathbf{c})^T$ if and only if the point $(\tilde{\mathbf{f}}, \tilde{\mathbf{w}}) = \mathbf{L}(\mathbf{f}, \mathbf{w}) \in \mathcal{Q}$ satisfies $\tilde{\mathbf{f}} \neq \Xi(\mathbf{0})$ and $\mathbf{\Lambda}(\tilde{\mathbf{y}})\tilde{\mathbf{f}}^T \leq \mathbf{\Lambda}(\tilde{\mathbf{y}})\Xi(\mathbf{0})^T$. This statement, along with the fact that both \mathbf{G} and \mathbf{L} are bijective, proves that

$$\mathbf{y} \in B(\mathbf{c}) \quad \text{if and only if} \quad \tilde{\mathbf{y}} \in B(\mathbf{0}). \quad \square$$

We next provide, with details, some examples of modulator–channel combinations for which the symmetry conditions hold.

Example 5.1: Discrete memoryless q -ary symmetric channel. Here we denote the ring elements by $\mathfrak{R} = \{a_0, a_1, \dots, a_{q-1}\}$. Also, $\Sigma = \{s_0, s_1, \dots, s_{q-1}\}$, where the channel output probability conditioned on the channel input satisfies, for each $t, k \in \{0, 1, \dots, q-1\}$

$$p(s_t|a_k) = \begin{cases} (1-p), & \text{if } t = k \\ \frac{p}{(q-1)}, & \text{otherwise} \end{cases}$$

where p represents the probability of transmission error. Here we may define the mapping τ_β for each $\beta \in \mathfrak{R}$ according to

$$\tau_\beta(s_t) = s_\ell, \quad \text{where } a_\ell = a_t - \beta$$

for all $t \in \{0, 1, \dots, q-1\}$. It is easy to check that these mappings are bijective and satisfy the symmetry condition.

Example 5.2: Orthogonal modulation over AWGN. Here $\Sigma = \mathbb{R}^q$, and denoting the ring elements by $\mathfrak{R} = \{a_0, a_1, \dots, a_{q-1}\}$, the modulation mapping may be written without loss of generality as

$$\mathcal{M}: \mathfrak{R} \longrightarrow \mathbb{R}^q$$

such that, for each $k = 0, 1, \dots, q-1$

$$\mathcal{M}(a_k) = \mathbf{x} = (x^{(0)}, x^{(1)}, \dots, x^{(q-1)})$$

where

$$x^{(t)} = \begin{cases} 1, & \text{if } t = k \\ 0, & \text{otherwise.} \end{cases}$$

Here we may define the mapping τ_β for each $\beta \in \mathfrak{R}$ according to (where $\mathbf{y} = (y^{(0)}, y^{(1)}, \dots, y^{(q-1)}) \in \mathbb{R}^q$, $\mathbf{z} = (z^{(0)}, z^{(1)}, \dots, z^{(q-1)}) \in \mathbb{R}^q$)

$$\tau_\beta(\mathbf{y}) = \mathbf{z}$$

such that for each $l \in \{0, 1, \dots, q-1\}$

$$z^{(l)} = y^{(k)}, \quad \text{where } a_k = a_l + \beta.$$

It is easily checked that these mappings are bijective and isometric, and satisfy the symmetry condition.

Example 5.3: q -ary PSK modulation over AWGN.

Here $\Sigma = \mathbb{C}$, and again denoting the ring elements by $\mathfrak{R} = \{a_0, a_1, \dots, a_{q-1}\}$, the modulation mapping may be written without loss of generality as

$$M: \mathfrak{R} \mapsto \mathbb{C}$$

such that

$$M(a_k) = \exp\left(\frac{i2\pi k}{q}\right) \quad (23)$$

for $k = 0, 1, \dots, q-1$ (here $i = \sqrt{-1}$). Here (18), together with the rotational symmetry of the q -ary PSK constellation, motivates us to define, for every $\beta = a_k \in \mathfrak{R}$

$$\tau_\beta(x) = \exp\left(\frac{-i2\pi k}{q}\right) \cdot x, \quad \forall x \in \mathbb{C}. \quad (24)$$

Next, we also impose the condition that \mathfrak{R} under addition is a cyclic group. To see why we impose this condition, let $\alpha = a_k \in \mathfrak{R}$ and $\beta = a_l \in \mathfrak{R}$. By the symmetry condition we must have

$$p(y_i|\alpha + \beta) = p(\tau_{\alpha+\beta}(y_i)|0)$$

and also

$$\begin{aligned} p(y_i|\alpha + \beta) &= p(\tau_\beta(y_i)|\alpha) \\ &= p(\tau_\alpha(\tau_\beta(y_i))|0). \end{aligned}$$

In order to equate these two expressions, we impose the condition $\tau_{\alpha+\beta}(x) = \tau_\alpha(\tau_\beta(x))$ for all $x \in \mathbb{C}$, $\alpha, \beta \in \mathfrak{R}$. Letting $\alpha + \beta = a_p \in \mathfrak{R}$, and using (24) yields

$$\exp\left(\frac{-i2\pi k}{q}\right) \cdot \exp\left(\frac{-i2\pi l}{q}\right) = \exp\left(\frac{-i2\pi p}{q}\right)$$

and thus $p \equiv (k+l) \pmod{q}$.

Therefore, we must have

$$a_k + a_l = a_{(k+l) \bmod q} \quad (25)$$

for all $a_k, a_l \in \mathfrak{R}$. This implies that \mathfrak{R} , under addition, is a cyclic group.

It is easy to check that the condition that \mathfrak{R} under addition is cyclic, encapsulated by (25), along with the modulation mapping (23), satisfies the symmetry condition, where the appropriate mappings τ_β are given by (24). This means that codeword-independent performance is guaranteed for such systems using nonbinary codes with PSK modulation. This applies to AWGN, flat-fading wireless channels, and orthogonal frequency-division multiplexing (OFDM) systems transmitting over frequency-selective channels with sufficiently long cyclic prefix.

VI. LINEAR PROGRAMMING PSEUDOCODEWORDS

Definition 6.1: A linear-programming pseudocodeword (LP pseudocodeword) of the code \mathcal{C} , with parity-check matrix \mathcal{H} , is a pair (\mathbf{h}, \mathbf{z}) where $\mathbf{h} \in \mathbb{R}^{(q-1)n}$ and

$$\mathbf{z} = (z_{j,\mathbf{b}})_{j \in \mathcal{J}, \mathbf{b} \in \mathcal{C}_j}$$

where $z_{j,\mathbf{b}}$ is a nonnegative integer for all $j \in \mathcal{J}$, $\mathbf{b} \in \mathcal{C}_j$, such that the following constraints are satisfied:

$$\forall j \in \mathcal{J}, \forall i \in \mathcal{I}_j, \forall \alpha \in \mathfrak{R}^-, h_i^{(\alpha)} = \sum_{\mathbf{b} \in \mathcal{C}_j, b_i = \alpha} z_{j,\mathbf{b}} \quad (26)$$

and

$$\forall j \in \mathcal{J}, \sum_{\mathbf{b} \in \mathcal{C}_j} z_{j,\mathbf{b}} = M \quad (27)$$

where M is a nonnegative integer independent of j .

It follows from (26) that $h_i^{(\alpha)}$ is a nonnegative integer for all $i \in \mathcal{I}$, $\alpha \in \mathfrak{R}^-$. We note that the further constraints

$$\forall j \in \mathcal{J}, \forall \mathbf{b} \in \mathcal{C}_j, z_{j,\mathbf{b}} \leq M \quad (28)$$

$$\forall i \in \mathcal{I}, \forall \alpha \in \mathfrak{R}^-, 0 \leq h_i^{(\alpha)} \leq M \quad (29)$$

and

$$\forall i \in \mathcal{I}, \sum_{\alpha \in \mathfrak{R}^-} h_i^{(\alpha)} \leq M \quad (30)$$

follow from the constraints (26) and (27).

For each $i \in \mathcal{I}$, we also define

$$h_i^{(0)} = M - \sum_{\alpha \in \mathfrak{R}^-} h_i^{(\alpha)}. \quad (31)$$

By (30), $h_i^{(0)}$ is a nonnegative integer for all $i \in \mathcal{I}$. Now, for any $j \in \mathcal{J}$, $i \in \mathcal{I}_j$ we have

$$\begin{aligned} h_i^{(0)} &= M - \sum_{\alpha \in \mathfrak{R}^-} h_i^{(\alpha)} \\ &= \sum_{\mathbf{b} \in \mathcal{C}_j} z_{j,\mathbf{b}} - \sum_{\alpha \in \mathfrak{R}^-} \sum_{\mathbf{b} \in \mathcal{C}_j, b_i = \alpha} z_{j,\mathbf{b}} \\ &= \sum_{\mathbf{b} \in \mathcal{C}_j, b_i = 0} z_{j,\mathbf{b}} \end{aligned}$$

where we have used (26) and (27).

Corresponding to the LP pseudocodeword (\mathbf{h}, \mathbf{z}) defined above, we define the *normalized LP pseudocodeword* as the vector obtained by scaling of (\mathbf{h}, \mathbf{z}) by a factor $1/M$. We also define the $n \times q$ LP pseudocodeword matrix

$$\mathbf{H} = (h_i^{(\alpha)})_{i \in \mathcal{I}; \alpha \in \mathfrak{R}^-}.$$

The *normalized LP pseudocodeword matrix* is defined as $(1/M) \cdot \mathbf{H}$.

Note that if we interpret $\{z_{j,\mathbf{b}}/M\}$ (for each $j \in \mathcal{J}$) as a probability distribution for the local codeword $\mathbf{b} \in \mathcal{C}_j$, then the i th row of the normalized LP pseudocodeword matrix (for $i \in \mathcal{I}$) can be interpreted as the corresponding probability distribution for the i th coded symbol $c_i \in \mathfrak{R}$. This idea of interpreting pseudocodewords as probability distributions was used in [3] for the binary case.

Example 6.1: As an illustration, we provide an LP pseudocodeword for the example [4, 2] code over \mathbb{Z}_3 defined by the parity-check matrix (4). The reader may check that

$$(h_1^{(1)}, h_2^{(1)}, h_3^{(1)}, h_4^{(1)}) = (2 \ 2 \ 2 \ 2) \quad (32)$$

and

$$(h_1^{(2)}, h_2^{(2)}, h_3^{(2)}, h_4^{(2)}) = (2 \ 2 \ 0 \ 0) \quad (33)$$

together with

$$z_{1,\mathbf{b}} = \begin{cases} 2, & \text{if } \mathbf{b} = (2 \ 1 \ 1 \ 0) \\ 2, & \text{if } \mathbf{b} = (1 \ 2 \ 0 \ 1) \\ 0, & \text{otherwise} \end{cases} \quad (34)$$

and

$$z_{2,\mathbf{b}} = \begin{cases} 2, & \text{if } \mathbf{b} = (2 \ 0 \ 1) \\ 2, & \text{if } \mathbf{b} = (1 \ 1 \ 0) \\ 0, & \text{otherwise} \end{cases} \quad (35)$$

satisfy (26) and (27), where $M = 4$ in (27). We also obtain from (31)

$$(h_1^{(0)}, h_2^{(0)}, h_3^{(0)}, h_4^{(0)}) = (0 \ 0 \ 2 \ 2).$$

Therefore, (32)–(35) define an LP pseudocodeword, with pseudocodeword matrix

$$\mathbf{H} = \begin{pmatrix} 0 & 2 & 2 \\ 0 & 2 & 2 \\ 2 & 2 & 0 \\ 2 & 2 & 0 \end{pmatrix}. \quad (36)$$

The corresponding normalized LP pseudocodeword matrix is then given by

$$\frac{1}{4} \cdot \mathbf{H} = \begin{pmatrix} 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 \end{pmatrix}. \quad (37)$$

Here the probabilistic interpretation of this normalized LP pseudocodeword matrix corresponds to an equiprobable distribution of symbols from $\{1, 2\}$ for the first two symbols in the codeword, and an equiprobable distribution of symbols from $\{0, 1\}$ for the last two symbols in the codeword.

Theorem 6.1: Assume that the all-zero codeword was transmitted.

- 1) If the LP decoder makes a codeword error, then there exists some LP pseudocodeword (\mathbf{h}, \mathbf{z}) , $\mathbf{h} \neq \mathbf{0}$, such that $\mathbf{\Lambda}(\mathbf{y})\mathbf{h}^T \leq 0$.
- 2) If there exists some LP pseudocodeword (\mathbf{h}, \mathbf{z}) , $\mathbf{h} \neq \mathbf{0}$, such that $\mathbf{\Lambda}(\mathbf{y})\mathbf{h}^T < 0$, then the LP decoder makes a codeword error.

Proof: The proof follows the lines of its counterpart in [7].

- 1) Let (\mathbf{f}, \mathbf{w}) be the point in \mathcal{Q} which minimizes $\mathbf{\Lambda}(\mathbf{y})\mathbf{f}^T$. Suppose there is a codeword error; then $\mathbf{f} \neq \mathbf{0}$, and we must have $\mathbf{\Lambda}(\mathbf{y})\mathbf{f}^T \leq 0$.

Next, we construct the LP pseudocodeword (\mathbf{h}, \mathbf{z}) as follows. Since the LP has rational coefficients, all elements of the vectors \mathbf{f} and \mathbf{w} must be rational. Let M denote their lowest common denominator; since $\mathbf{f} \neq \mathbf{0}$, we may have $M > 0$. Now set $h_i^{(\alpha)} = M \cdot f_i^{(\alpha)}$ for all $i \in \mathcal{I}$, $\alpha \in \mathbb{R}^-$, and set $z_{j,\mathbf{b}} = M \cdot w_{j,\mathbf{b}}$ for all $j \in \mathcal{J}$ and $\mathbf{b} \in \mathcal{C}_j$.

By (8)–(10), (\mathbf{h}, \mathbf{z}) is an LP pseudocodeword and $\mathbf{h} \neq \mathbf{0}$ since $\mathbf{f} \neq \mathbf{0}$. Also, $\mathbf{\Lambda}(\mathbf{y})\mathbf{f}^T \leq 0$ implies $\mathbf{\Lambda}(\mathbf{y})\mathbf{h}^T \leq 0$.

- 2) Now, suppose that an LP pseudocodeword (\mathbf{h}, \mathbf{z}) with $\mathbf{h} \neq \mathbf{0}$ satisfies $\mathbf{\Lambda}(\mathbf{y})\mathbf{h}^T < 0$. Since $\mathbf{h} \neq \mathbf{0}$, we have $M > 0$ in (27). Now, set $f_i^{(\alpha)} = h_i^{(\alpha)}/M$ for all $i \in \mathcal{I}$, $\alpha \in \mathbb{R}^-$, and set $w_{j,\mathbf{b}} = z_{j,\mathbf{b}}/M$ for all $j \in \mathcal{J}$ and $\mathbf{b} \in \mathcal{C}_j$. It is straightforward to check that (\mathbf{f}, \mathbf{w}) satisfies all the constraints of the polytope \mathcal{Q} . Also, $\mathbf{h} \neq \mathbf{0}$ implies $\mathbf{f} \neq \mathbf{0}$. Finally, $\mathbf{\Lambda}(\mathbf{y})\mathbf{h}^T < 0$ implies $\mathbf{\Lambda}(\mathbf{y})\mathbf{f}^T < 0$. Therefore, the LP decoder will make a codeword error. \square

VII. EQUIVALENCE BETWEEN PSEUDOCODEWORD CONCEPTS

A. Tanner Graphs and Graph-Cover Pseudocodewords

The Tanner graph of a linear code \mathcal{C} over \mathbb{R} is an equivalent characterization of the code's parity-check matrix \mathcal{H} . The Tanner graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ has vertex set $\mathcal{V} = \{u_1, u_2, \dots, u_n\} \cup \{v_1, v_2, \dots, v_m\}$, and there is an edge between u_i and v_j if and only if $\mathcal{H}_{j,i} \neq 0$. This edge is labeled with the value $\mathcal{H}_{j,i}$. We denote by $\mathcal{N}(v)$ the set of neighbors of a vertex $v \in \mathcal{V}$.

For any word $\mathbf{c} = (c_1, c_2, \dots, c_n) \in \mathbb{R}^n$, the Tanner graph allows an equivalent graphical statement of the condition $\mathbf{c} \in \mathcal{C}_j$ for each $j \in \mathcal{J}$, as follows. The variable vertex u_i is labeled with the value c_i for each $i \in \mathcal{I}$. Equation (2) (or (3)) is then equivalent to the condition that for vertex v_j , the sum, over all vertices in $\mathcal{N}(v_j)$, of the vertex labels multiplied by the corresponding

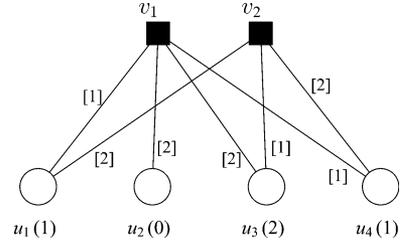


Fig. 1. Tanner graph for the example $[4, 2]$ code over \mathbb{Z}_3 . Edge labels are shown in square brackets, and vertex labels in parentheses. For each parity check j , the sum, over all vertices in $\mathcal{N}(v_j)$ of the vertex labels multiplied by the corresponding edge labels is zero; therefore, all parity checks are satisfied.

edge labels is zero. This graphical means of checking whether a parity check is satisfied by $\mathbf{c} \in \mathbb{R}^n$ will be useful when defining graph-cover pseudocodewords later in this section.

To illustrate this concept, Fig. 1 shows the Tanner graph for the codeword $\mathbf{c} = (1 \ 0 \ 2 \ 1)$ of the example $[4, 2]$ code over \mathbb{Z}_3 defined by the parity-check matrix (4). In Fig. 1, edge labels are shown in square brackets, and vertex labels in parentheses. The reader may check that for each parity check $j = 1, 2$, the sum, over all vertices in $\mathcal{N}(v_j)$, of the vertex labels multiplied by the corresponding edge labels is zero.

We next define what is meant by a finite cover of a Tanner graph.

Definition 7.1: ([4]) A graph $\tilde{\mathcal{G}} = (\tilde{\mathcal{V}}, \tilde{\mathcal{E}})$ is a *finite cover* of the Tanner graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ if there exists a mapping $\Pi : \tilde{\mathcal{V}} \rightarrow \mathcal{V}$ which is a graph homomorphism (Π takes adjacent vertices of $\tilde{\mathcal{G}}$ to adjacent vertices of \mathcal{G}), such that for every vertex $v \in \mathcal{G}$ and every $\tilde{v} \in \Pi^{-1}(v)$, the neighborhood $\mathcal{N}(\tilde{v})$ of \tilde{v} (including edge labels) is mapped bijectively to $\mathcal{N}(v)$.

Definition 7.2: ([4]) A cover of the graph \mathcal{G} is said to have degree M , where M is a positive integer, if $|\Pi^{-1}(v)| = M$ for every vertex $v \in \mathcal{V}$. We refer to such a cover graph as an *M-cover* of \mathcal{G} .

Fix some positive integer M . Let $\tilde{\mathcal{G}} = (\tilde{\mathcal{V}}, \tilde{\mathcal{E}})$ be an M -cover of the Tanner graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ representing the code \mathcal{C} with parity-check matrix \mathcal{H} . The vertices in the set $\Pi^{-1}(u_i)$ are called *copies* of u_i and are denoted $\{u_{i,1}, u_{i,2}, \dots, u_{i,M}\}$, where $i \in \mathcal{I}$. Similarly, the vertices in the set $\Pi^{-1}(v_j)$ are called *copies* of v_j and are denoted $\{v_{j,1}, v_{j,2}, \dots, v_{j,M}\}$, where $j \in \mathcal{J}$.

Less formally, given a code \mathcal{C} with parity-check matrix \mathcal{H} and corresponding Tanner graph \mathcal{G} , an M -cover of \mathcal{G} is a graph whose vertex set consists of M copies of u_i and M copies of v_j , such that for each $j \in \mathcal{J}$, $i \in \mathcal{I}_j$, the M copies of u_i and the M copies of v_j are connected in an arbitrary one-to-one fashion, with edges labeled by the value $H_{j,i}$.

For any $M \geq 1$, a *graph-cover pseudocodeword* is a labeling of vertices of the M -cover graph with values from \mathbb{R} such that all parity checks are satisfied. We denote the label of $u_{i,\ell}$ by $p_{i,\ell}$ for each $i \in \mathcal{I}$, $\ell = 1, 2, \dots, M$, and we may then write the graph-cover pseudocodeword in vector form as

$$\mathbf{p} = (p_{1,1}, p_{1,2}, \dots, p_{1,M}, p_{2,1}, p_{2,2}, \dots, p_{2,M}, \dots, p_{n,1}, p_{n,2}, \dots, p_{n,M}).$$

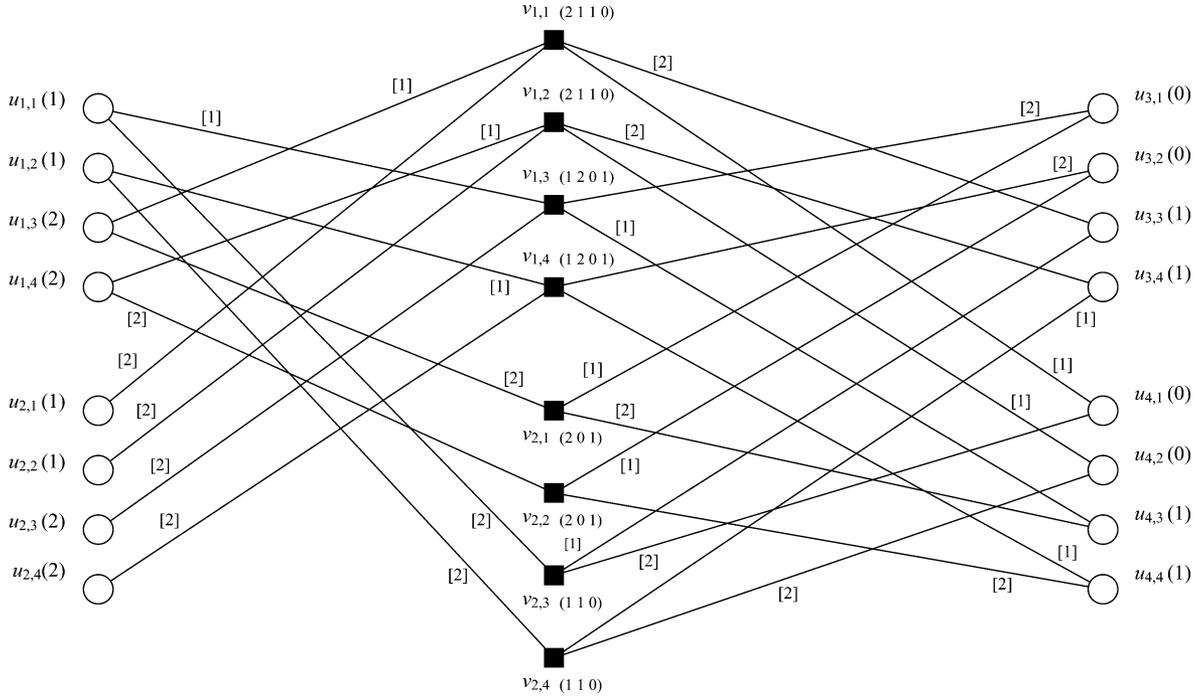


Fig. 2. Cover graph of degree 4 and corresponding graph-cover pseudocodeword for the example $[4, 2]$ code over \mathbb{Z}_3 with parity-check matrix given by (4). Edge labels are shown in square brackets, and vertex labels in parentheses. This graph-cover pseudocodeword corresponds to the LP pseudocodeword described by (32)–(35) via the correspondence described in the proof of Theorem 7.1.

It is easily seen that \mathbf{p} belongs to a linear code $\tilde{\mathcal{C}}$ of length Mn over \mathfrak{R} , defined by an $Mm \times Mn$ parity-check matrix $\tilde{\mathcal{H}}$. To construct $\tilde{\mathcal{H}}$, for $1 \leq i^*, j^* \leq M$ and $i \in \mathcal{I}, j \in \mathcal{J}$, we let $i' = (i-1)M + i^*, j' = (j-1)M + j^*$, and so

$$\tilde{\mathcal{H}}_{j',i'} = \begin{cases} \mathcal{H}_{j,i}, & \text{if } u_{i,i^*} \in \mathcal{N}(v_{j,j^*}) \\ 0, & \text{otherwise.} \end{cases}$$

It may be seen that $\tilde{\mathcal{G}}$ is the Tanner graph of the code $\tilde{\mathcal{C}}$ corresponding to the parity-check matrix $\tilde{\mathcal{H}}$.

We also define the $n \times q$ graph-cover pseudocodeword matrix

$$\mathcal{P} = \left(m_i^{(\alpha)} \right)_{i \in \mathcal{I}; \alpha \in \mathfrak{R}}$$

where

$$m_i^{(\alpha)} = |\{\ell \in \{1, 2, \dots, M\} : p_{i,\ell} = \alpha\}| \geq 0$$

for $i \in \mathcal{I}, \alpha \in \mathfrak{R}$, i.e., $m_i^{(\alpha)}$ is equal to the number of copies of u_i which are labeled with α , for each $i \in \mathcal{I}, \alpha \in \mathfrak{R}$. The normalized graph-cover pseudocodeword matrix is defined as $(1/M) \cdot \mathcal{P}$. This matrix representation is similar to that defined in [16]. Note that the i th row of the normalized graph-cover pseudocodeword matrix (for $i \in \mathcal{I}$) can be viewed as a probability distribution for the i th coded symbol $c_i \in \mathfrak{R}$, in a similar manner to the case of the normalized LP pseudocodeword matrix.

Another representation, which we shall use in Section IX, is the graph-cover pseudocodeword vector $\mathbf{m} = (\mathbf{m}_i)_{i \in \mathcal{I}}$ where $\mathbf{m}_i = (m_i^{(\alpha)})_{\alpha \in \mathfrak{R}}$ for each $i \in \mathcal{I}$. Correspondingly, the normalized graph-cover pseudocodeword vector is given by $(1/M) \cdot \mathbf{m} \in \mathbb{R}^{(q-1)n}$.

It is easily seen that for any $\mathbf{c} \in \mathcal{C}$, the labeling of $u_{i,\ell}$ by the value c_i for all $i \in \mathcal{I}, \ell = 1, 2, \dots, M$ trivially yields a graph-

cover pseudocodeword for all M -covers of \mathcal{G} , $M \geq 1$. However, nontrivial graph-cover pseudocodewords exist in general.

Example 7.1: To illustrate these concepts, a graph-cover pseudocodeword is shown in Fig. 2 for the example $[4, 2]$ code over \mathbb{Z}_3 defined by the parity-check matrix (4). Here, the degree of the cover graph is $M = 4$, and we have

$$\mathbf{p} = (1 \ 1 \ 2 \ 2 \mid 1 \ 1 \ 2 \ 2 \mid 0 \ 0 \ 1 \ 1 \mid 0 \ 0 \ 1 \ 1)$$

and the parity-check matrix of the code $\tilde{\mathcal{C}}$ is given by

$$\tilde{\mathcal{H}} = \begin{pmatrix} 0010 & 2000 & 0020 & 1000 \\ 0001 & 0200 & 0002 & 0100 \\ 1000 & 0020 & 2000 & 0010 \\ 0100 & 0002 & 0200 & 0001 \\ \hline 0020 & 0000 & 1000 & 0020 \\ 0002 & 0000 & 0100 & 0002 \\ 2000 & 0000 & 0010 & 2000 \\ 0200 & 0000 & 0001 & 0200 \end{pmatrix}.$$

Also, the graph-cover pseudocodeword matrix corresponding to \mathbf{p} is

$$\mathcal{P} = \begin{pmatrix} 0 & 2 & 2 \\ 0 & 2 & 2 \\ 2 & 2 & 0 \\ 2 & 2 & 0 \end{pmatrix} \quad (38)$$

and the normalized graph-cover pseudocodeword matrix is

$$\frac{1}{4} \cdot \mathcal{P}.$$

The graph-cover pseudocodeword vector corresponding to \mathbf{p} is

$$\mathbf{m} = (2\ 2 \mid 2\ 2 \mid 2\ 0 \mid 2\ 0)$$

and the normalized graph-cover pseudocodeword vector is

$$\frac{1}{4} \cdot \mathbf{m}.$$

B. Equivalence Between LP Pseudocodewords and Graph-Cover Pseudocodewords

In this subsection, we show the equivalence between the set of LP pseudocodewords and the set of graph-cover pseudocodewords. The result is summarized in the following theorem.

Theorem 7.1: Let \mathcal{C} be a linear code over the ring \mathfrak{R} with parity-check matrix \mathcal{H} and corresponding Tanner graph \mathcal{G} . Then, there exists an LP pseudocodeword (\mathbf{h}, \mathbf{z}) with pseudocodeword matrix \mathbf{H} if and only if there exists a graph-cover pseudocodeword for some M -cover of \mathcal{G} with the same pseudocodeword matrix.

Proof:

- 1) Let (\mathbf{h}, \mathbf{z}) be an LP pseudocodeword of \mathcal{C} , and let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be the Tanner graph associated with the parity-check matrix \mathcal{H} . We construct an M -cover $\tilde{\mathcal{G}} = (\tilde{\mathcal{V}}, \tilde{\mathcal{E}})$, where $M = \sum_{\mathbf{b} \in \mathcal{C}_j} z_{j,\mathbf{b}}$, and corresponding graph-cover pseudocodeword, as follows. We begin with the vertex set, which consists of M copies of $u_i, i \in \mathcal{I}$, and M copies of $v_j, j \in \mathcal{J}$. Then we proceed as follows.
 - Label $h_i^{(\alpha)}$ copies of u_i with the value α , for each $i \in \mathcal{I}, \alpha \in \mathfrak{R}$. By (31), all copies of u_i are labeled.
 - Label $z_{j,\mathbf{b}}$ copies of v_j with the value \mathbf{b} , for every $j \in \mathcal{J}, \mathbf{b} \in \mathcal{C}_j$. By (27), all copies of v_j are labeled.
 - Next, let $T_i^{(\alpha)}$ denote the set of copies of u_i labeled with the value α , for $i \in \mathcal{I}, \alpha \in \mathfrak{R}$. Also, for all $i \in \mathcal{I}, j \in \mathcal{J}, \alpha \in \mathfrak{R}$, let $R_{i,j}^{(\alpha)}$ denote the set of copies of v_j whose label satisfies $b_i = \alpha$. The vertices in $T_i^{(\alpha)}$ and the vertices in $R_{i,j}^{(\alpha)}$ are then connected by edges in an arbitrary one-to-one fashion, for every $j \in \mathcal{J}, i \in \mathcal{I}_j, \alpha \in \mathfrak{R}$. All of these edges are labeled with the value $H_{j,i}$. First, we note that this is possible because

$$\begin{aligned} |T_i^{(\alpha)}| &= h_i^{(\alpha)} \\ &= \sum_{\mathbf{b} \in \mathcal{C}_j, b_i = \alpha} z_{j,\mathbf{b}} \\ &= |R_{i,j}^{(\alpha)}| \end{aligned}$$

for every $j \in \mathcal{J}, i \in \mathcal{I}_j, \alpha \in \mathfrak{R}$. Here we have used (26). Second, we note that all checks are satisfied by this labeling. For $j \in \mathcal{J}$, consider any copy of v_j with label \mathbf{b} . By construction of the graph, the sum, over all vertices in $\mathcal{N}(v_j)$, of the vertex labels multiplied by the corresponding edge labels is

$$\sum_{i \in \mathcal{I}_j} b_i \cdot \mathcal{H}_{j,i},$$

which is zero because $\mathbf{b} \in \mathcal{C}_j$. Therefore, this vertex labeling yields a graph-cover pseudocodeword of the code \mathcal{C} with parity-check matrix \mathcal{H} .

- 2) Now suppose that there exists a graph-cover pseudocodeword corresponding to some M -cover of the Tanner graph \mathcal{G} of \mathcal{C} . Then we proceed as follows.

- **Step 1:** for every $i \in \mathcal{I}$, and for every $\alpha \in \mathfrak{R}^-$, we define $h_i^{(\alpha)}$ to be the number of copies of u_i labeled with the value α .
- **Step 2:** for every copy of $v_j, j \in \mathcal{J}$, label the copy with the word \mathbf{b} , where b_i is equal to the label on the neighboring copy of $u_i, i \in \mathcal{I}_j$. Then, for every $j \in \mathcal{J}, \mathbf{b} \in \mathcal{C}_j$, we define $z_{j,\mathbf{b}}$ to be the number of copies of v_j labeled with the word \mathbf{b} .

Step 2 ensures that $z_{j,\mathbf{b}}$ are nonnegative integers for all $j \in \mathcal{J}$ and $\mathbf{b} \in \mathcal{C}_j$, and that (27) holds. Also, to show that (26) holds, we reason as follows. The right-hand side of (26) counts the number of copies of v_j whose labels \mathbf{b} satisfy $b_i = \alpha$. By Step 2, this is equal to the number of copies of u_i labeled with α , which by Step 1 is equal to the left-hand side of (26). Therefore, (\mathbf{h}, \mathbf{z}) is an LP pseudocodeword of the code \mathcal{C} with parity-check matrix \mathcal{H} . \square

As an illustration of the correspondences described in this proof, consider the example $[4, 2]$ code over \mathbb{Z}_3 defined by the parity-check matrix (4). First, note that the LP pseudocodeword of (32)–(35) and the graph-cover pseudocodeword of Fig. 2 have the same pseudocodeword matrix, via (36) and (38). Indeed, the reader may check that each pseudocodeword may be derived from the other using the correspondences described in the proof of Theorem 7.1.

The next corollary follows immediately from Theorem 7.1.

Corollary 7.2: Let \mathcal{C} be a linear code over the ring \mathfrak{R} with parity-check matrix \mathcal{H} and corresponding Tanner graph \mathcal{G} . Then, there exists a (normalized) LP pseudocodeword (\mathbf{h}, \mathbf{z}) if and only if there exists a graph-cover pseudocodeword for some M -cover of \mathcal{G} with (normalized) graph-cover pseudocodeword vector \mathbf{h} .

Note that this corollary contains two different equivalences, one for normalized objects and the other for nonnormalized ones.

VIII. ALTERNATIVE POLYTOPE REPRESENTATION

In this section, we present an alternative polytope for use with linear-programming decoding. This polytope may be regarded as a generalization of the “high-density polytope” defined in [7]. As we show in this section, the new polytope may under some circumstances yield a complexity advantage over the polytope of Section III. In the sequel, we will analyze the properties of this polytope.

First, we introduce some convenient notation and definitions. Recall that the ring \mathfrak{R} contains $q - 1$ nonzero elements; correspondingly, for vectors $\mathbf{k} \in \mathbb{N}^{q-1}$, we adopt the notation

$$\mathbf{k} = (k_\alpha)_{\alpha \in \mathfrak{R}^-}.$$

Now, for any $j \in \mathcal{J}$, we define the mapping

$$\begin{aligned} \kappa_j : \mathcal{C}_j &\longrightarrow \mathbb{N}^{q-1} \\ \mathbf{b} &\mapsto \kappa_j(\mathbf{b}) \end{aligned}$$

defined by

$$(\kappa_j(\mathbf{b}))_\alpha = |\{i \in \mathcal{I}_j : b_i \cdot \mathcal{H}_{j,i} = \alpha\}|$$

for all $\alpha \in \mathbb{R}^-$. We may then characterize the image of κ_j , which we denote by \mathcal{T}_j , as

$$\mathcal{T}_j = \left\{ \mathbf{k} \in \mathbb{N}^{q-1} : \sum_{\alpha \in \mathbb{R}^-} \alpha \cdot k_\alpha = 0 \quad \text{and} \quad \sum_{\alpha \in \mathbb{R}^-} k_\alpha \leq d_j \right\}$$

for each $j \in \mathcal{J}$, where, for any $k \in \mathbb{N}$, $\alpha \in \mathbb{R}$

$$\alpha \cdot k = \begin{cases} 0, & \text{if } k = 0 \\ \alpha + \dots + \alpha, & \text{if } k > 0 \text{ (} k \text{ terms in sum).} \end{cases}$$

Note that κ_j is not a bijection, in general. We say that a local codeword $\mathbf{b} \in \mathcal{C}_j$ is \mathbf{k} -constrained over \mathcal{C}_j if $\kappa_j(\mathbf{b}) = \mathbf{k}$.

Next, for any index set $\Gamma \subseteq \mathcal{I}$, we introduce the following definitions. Let $N = |\Gamma|$. We define the single parity-check code, over vectors indexed by Γ , by

$$\mathcal{C}_\Gamma = \left\{ \mathbf{a} = (a_i)_{i \in \Gamma} \in \mathbb{R}^N : \sum_{i \in \Gamma} a_i = 0 \right\}. \quad (39)$$

Also, define a mapping $\kappa_\Gamma : \mathcal{C}_\Gamma \rightarrow \mathbb{N}^{q-1}$ by

$$(\kappa_\Gamma(\mathbf{a}))_\alpha = |\{i \in \Gamma : a_i = \alpha\}|,$$

and define, for $\mathbf{k} \in \mathcal{T}_j$,

$$\mathcal{C}_\Gamma^{(\mathbf{k})} = \{\mathbf{a} \in \mathcal{C}_\Gamma : \kappa_\Gamma(\mathbf{a}) = \mathbf{k}\}.$$

Below, we define a new polytope for decoding. Recall that $\mathbf{y} = (y_1, y_2, \dots, y_n) \in \Sigma^n$ stands for the received (corrupted) word. In the sequel, we make use of the following variables.

- For all $i \in \mathcal{I}$ and all $\alpha \in \mathbb{R}^-$, we have a variable $f_i^{(\alpha)}$. This variable is an indicator of the event $y_i = \alpha$.
- For all $j \in \mathcal{J}$ and $\mathbf{k} \in \mathcal{T}_j$, we have a variable $\sigma_{j,\mathbf{k}}$. Similarly to its counterpart in [7], this variable indicates the contribution to parity check j of \mathbf{k} -constrained local codewords over \mathcal{C}_j .
- For all $j \in \mathcal{J}$, $i \in \mathcal{I}_j$, $\mathbf{k} \in \mathcal{T}_j$, $\alpha \in \mathbb{R}^-$, we have a variable $z_{i,j,\mathbf{k}}^{(\alpha)}$. This variable indicates the portion of $f_i^{(\alpha)}$ assigned to \mathbf{k} -constrained local codewords over \mathcal{C}_j .

Motivated by these variable definitions, for all $j \in \mathcal{J}$ we impose the following set of constraints:

$$\forall i \in \mathcal{I}_j, \forall \alpha \in \mathbb{R}^-, \quad f_i^{(\alpha)} = \sum_{\mathbf{k} \in \mathcal{T}_j} z_{i,j,\mathbf{k}}^{(\alpha)} \quad (40)$$

$$\sum_{\mathbf{k} \in \mathcal{T}_j} \sigma_{j,\mathbf{k}} = 1 \quad (41)$$

$$\forall \mathbf{k} \in \mathcal{T}_j, \forall \alpha \in \mathbb{R}^-, \quad \sum_{i \in \mathcal{I}_j, \beta \in \mathbb{R}^-, \beta \mathcal{H}_{j,i} = \alpha} z_{i,j,\mathbf{k}}^{(\beta)} = k_\alpha \cdot \sigma_{j,\mathbf{k}} \quad (42)$$

$$\forall i \in \mathcal{I}_j, \forall \mathbf{k} \in \mathcal{T}_j, \forall \alpha \in \mathbb{R}^-, \quad z_{i,j,\mathbf{k}}^{(\alpha)} \geq 0 \quad (43)$$

$$\forall i \in \mathcal{I}_j, \forall \mathbf{k} \in \mathcal{T}_j, \quad \sum_{\alpha \in \mathbb{R}^-} \sum_{\beta \in \mathbb{R}^-, \beta \mathcal{H}_{j,i} = \alpha} z_{i,j,\mathbf{k}}^{(\beta)} \leq \sigma_{j,\mathbf{k}}. \quad (44)$$

We note that the further constraints

$$\forall i \in \mathcal{I}, \forall \alpha \in \mathbb{R}^-, \quad 0 \leq f_i^{(\alpha)} \leq 1 \quad (45)$$

$$\forall j \in \mathcal{J}, \forall \mathbf{k} \in \mathcal{T}_j, \quad 0 \leq \sigma_{j,\mathbf{k}} \leq 1 \quad (46)$$

and

$$\forall j \in \mathcal{J}, \forall i \in \mathcal{I}_j, \forall \mathbf{k} \in \mathcal{T}_j, \forall \alpha \in \mathbb{R}^-, \quad z_{i,j,\mathbf{k}}^{(\alpha)} \leq \sigma_{j,\mathbf{k}} \quad (47)$$

follow from constraints (40)–(44). We denote by \mathcal{U} the polytope formed by constraints (40)–(44).

Let $T = \max_{j \in \mathcal{J}} |\mathcal{T}_j|$. Then, upper bounds on the number of variables and constraints in this LP are given by $n(q-1) + m(d(q-1)+1)T$ and $m(d(q-1)+1) + m((d+1)(q-1)+d)T$, respectively. Since $T \leq \binom{d+q-1}{d}$, the number of variables and constraints are $O(mq \cdot d^q)$, which, for many families of codes, is significantly lower than the corresponding complexity for polytope \mathcal{Q} .

For notational simplicity in proofs in this section, it is convenient to define a new set of variables as follows:

$$\forall j \in \mathcal{J}, \forall i \in \mathcal{I}_j, \forall \mathbf{k} \in \mathcal{T}_j, \forall \alpha \in \mathbb{R}^-, \quad \tau_{i,j,\mathbf{k}}^{(\alpha)} = \sum_{\beta \in \mathbb{R}^-, \beta \mathcal{H}_{j,i} = \alpha} z_{i,j,\mathbf{k}}^{(\beta)}. \quad (48)$$

Then constraints (42) and (44) may be rewritten as

$$\forall j \in \mathcal{J}, \mathbf{k} \in \mathcal{T}_j, \forall \alpha \in \mathbb{R}^-, \quad \sum_{i \in \mathcal{I}_j} \tau_{i,j,\mathbf{k}}^{(\alpha)} = k_\alpha \cdot \sigma_{j,\mathbf{k}} \quad (49)$$

and

$$\forall j \in \mathcal{J}, \forall i \in \mathcal{I}_j, \forall \mathbf{k} \in \mathcal{T}_j, \quad 0 \leq \sum_{\alpha \in \mathbb{R}^-} \tau_{i,j,\mathbf{k}}^{(\alpha)} \leq \sigma_{j,\mathbf{k}}. \quad (50)$$

Note that the variables τ do not form part of the LP description, and therefore do not contribute to its complexity. However, these variables will provide a convenient notational shorthand for proving results in this section.

We will prove that optimizing the cost function (5) over this new polytope is equivalent to optimizing over \mathcal{Q} . First, we state the following proposition, which will be necessary to prove this result.

Proposition 8.1: Let $M \in \mathbb{N}$ and $\mathbf{k} \in \mathbb{N}^{q-1}$. Also, let $\Gamma \subseteq \mathcal{I}$. Assume that for each $\alpha \in \mathbb{R}^-$, we have a set of nonnegative integers $\mathcal{X}^{(\alpha)} = \{x_i^{(\alpha)} : i \in \Gamma\}$ and that together these satisfy the constraints

$$\sum_{i \in \Gamma} x_i^{(\alpha)} = k_\alpha M \quad (51)$$

for all $\alpha \in \mathbb{R}^-$, and

$$\sum_{\alpha \in \mathbb{R}^-} x_i^{(\alpha)} \leq M \quad (52)$$

for all $i \in \Gamma$.

Then, there exist nonnegative integers $\{w_{\mathbf{a}} : \mathbf{a} \in \mathcal{C}_\Gamma^{(\mathbf{k})}\}$ such that

$$1) \quad \sum_{\mathbf{a} \in \mathcal{C}_\Gamma^{(\mathbf{k})}} w_{\mathbf{a}} = M. \quad (53)$$

2) For all $\alpha \in \mathbb{R}^-$, $i \in \Gamma$

$$x_i^{(\alpha)} = \sum_{\mathbf{a} \in \mathcal{C}_\Gamma^{(\mathbf{k})}, a_i = \alpha} w_{\mathbf{a}}. \quad (54)$$

The proof of this proposition appears in the Appendix. We now prove the main result.

Theorem 8.2: The set $\bar{\mathcal{U}} = \{\mathbf{f} : \exists \boldsymbol{\sigma}, \mathbf{z} \text{ s.t. } (\mathbf{f}, \boldsymbol{\sigma}, \mathbf{z}) \in \mathcal{U}\}$ is equal to the set $\bar{\mathcal{Q}} = \{\mathbf{f} : \exists \mathbf{w} \text{ s.t. } (\mathbf{f}, \mathbf{w}) \in \mathcal{Q}\}$. Therefore, optimizing the linear cost function (5) over \mathcal{U} is equivalent to optimizing over \mathcal{Q} .

Proof:

1) Suppose, $(\mathbf{f}, \mathbf{w}) \in \mathcal{Q}$. For all $j \in \mathcal{J}$, $\mathbf{k} \in \mathcal{T}_j$, we define

$$\sigma_{j,\mathbf{k}} = \sum_{\mathbf{b} \in \mathcal{C}_j, \boldsymbol{\kappa}_j(\mathbf{b}) = \mathbf{k}} w_{j,\mathbf{b}},$$

and for all $j \in \mathcal{J}$, $i \in \mathcal{I}_j$, $\mathbf{k} \in \mathcal{T}_j$, $\alpha \in \mathbb{R}^-$, we define

$$z_{i,j,\mathbf{k}}^{(\alpha)} = \sum_{\mathbf{b} \in \mathcal{C}_j, \boldsymbol{\kappa}_j(\mathbf{b}) = \mathbf{k}, b_i = \alpha} w_{j,\mathbf{b}}.$$

It is straightforward to check that constraints (43) and (44) are satisfied by these definitions.

For every $j \in \mathcal{J}$, $i \in \mathcal{I}_j$, $\alpha \in \mathbb{R}^-$, we have by (10)

$$\begin{aligned} f_i^{(\alpha)} &= \sum_{\mathbf{b} \in \mathcal{C}_j, b_i = \alpha} w_{j,\mathbf{b}} \\ &= \sum_{\mathbf{k} \in \mathcal{T}_j} \sum_{\mathbf{b} \in \mathcal{C}_j, \boldsymbol{\kappa}_j(\mathbf{b}) = \mathbf{k}, b_i = \alpha} w_{j,\mathbf{b}} \\ &= \sum_{\mathbf{k} \in \mathcal{T}_j} z_{i,j,\mathbf{k}}^{(\alpha)} \end{aligned}$$

and thus constraint (40) is satisfied.

Next, for every $j \in \mathcal{J}$, we have by (9)

$$\begin{aligned} 1 &= \sum_{\mathbf{b} \in \mathcal{C}_j} w_{j,\mathbf{b}} \\ &= \sum_{\mathbf{k} \in \mathcal{T}_j} \sum_{\mathbf{b} \in \mathcal{C}_j, \boldsymbol{\kappa}_j(\mathbf{b}) = \mathbf{k}} w_{j,\mathbf{b}} \\ &= \sum_{\mathbf{k} \in \mathcal{T}_j} \sigma_{j,\mathbf{k}} \end{aligned}$$

and thus constraint (41) is satisfied.

Finally, for every $j \in \mathcal{J}$, $\mathbf{k} \in \mathcal{T}_j$, $\alpha \in \mathbb{R}^-$

$$\begin{aligned} &\sum_{i \in \mathcal{I}_j, \beta \in \mathbb{R}^-, \beta \mathcal{H}_{j,i} = \alpha} z_{i,j,\mathbf{k}}^{(\beta)} \\ &= \sum_{i \in \mathcal{I}_j, \beta \in \mathbb{R}^-, \beta \mathcal{H}_{j,i} = \alpha} \sum_{\mathbf{b} \in \mathcal{C}_j, \boldsymbol{\kappa}_j(\mathbf{b}) = \mathbf{k}, b_i = \beta} w_{j,\mathbf{b}} \\ &= \sum_{\mathbf{b} \in \mathcal{C}_j, \boldsymbol{\kappa}_j(\mathbf{b}) = \mathbf{k}} \sum_{i \in \mathcal{I}_j, b_i \mathcal{H}_{j,i} = \alpha} w_{j,\mathbf{b}} \\ &= \sum_{\mathbf{b} \in \mathcal{C}_j, \boldsymbol{\kappa}_j(\mathbf{b}) = \mathbf{k}} k_\alpha \cdot w_{j,\mathbf{b}} \\ &= k_\alpha \cdot \sigma_{j,\mathbf{k}}. \end{aligned}$$

Thus, constraint (42) is also satisfied. This completes the proof of the first part of the theorem.

2) Now assume $(\mathbf{f}, \boldsymbol{\sigma}, \mathbf{z})$ is a vertex of the polytope \mathcal{U} , and so all variables are rational, as are the variables $\boldsymbol{\tau}$. Next, fix some $j \in \mathcal{J}$, $\mathbf{k} \in \mathcal{T}_j$, and consider the sets

$$\mathcal{X}_0^{(\alpha)} = \left\{ \frac{\tau_{i,j,\mathbf{k}}^{(\alpha)}}{\sigma_{j,\mathbf{k}}} : i \in \mathcal{I}_j \right\}.$$

for $\alpha \in \mathbb{R}^-$. By constraint (50), for each $\alpha \in \mathbb{R}^-$, all the values in the set $\mathcal{X}_0^{(\alpha)}$ are rational numbers between 0 and 1. Let μ be the lowest common denominator of all the numbers in all the sets $\mathcal{X}_0^{(\alpha)}$, $\alpha \in \mathbb{R}^-$. Let

$$\mathcal{X}^{(\alpha)} = \left\{ \mu \cdot \frac{\tau_{i,j,\mathbf{k}}^{(\alpha)}}{\sigma_{j,\mathbf{k}}} : i \in \mathcal{I}_j \right\}$$

for each $\alpha \in \mathbb{R}^-$. The sets $\mathcal{X}^{(\alpha)}$ consist of integers between 0 and μ . By constraint (49), we must have that for every $\alpha \in \mathbb{R}^-$, the sum of the elements in $\mathcal{X}^{(\alpha)}$ is equal to $k_\alpha \mu$. By constraint (50), we have

$$\sum_{\alpha \in \mathbb{R}^-} \mu \cdot \frac{\tau_{i,j,\mathbf{k}}^{(\alpha)}}{\sigma_{j,\mathbf{k}}} \leq \mu$$

for all $i \in \mathcal{I}_j$.

We now apply the result of Proposition 8.1 with $\Gamma = \mathcal{I}_j$, $M = \mu$, and with the sets $\mathcal{X}^{(\alpha)}$ defined as above (here $N = d_j$). Set the variables $\{w_{\mathbf{a}} : \mathbf{a} \in \mathcal{C}_\Gamma^{(\mathbf{k})}\}$ according to Proposition 8.1.

Next, for $\mathbf{k} \in \mathcal{T}_j$, we show how to define the variables $\{w'_{\mathbf{b}} : \mathbf{b} \in \mathcal{C}_j, \boldsymbol{\kappa}_j(\mathbf{b}) = \mathbf{k}\}$. Initially, we set $w'_{\mathbf{b}} = 0$ for all $\mathbf{b} \in \mathcal{C}_j$, $\boldsymbol{\kappa}_j(\mathbf{b}) = \mathbf{k}$. Observe that the values $\mu \cdot z_{i,j,\mathbf{k}}^{(\beta)} / \sigma_{j,\mathbf{k}}$ are nonnegative integers for every $i \in \mathcal{I}$, $j \in \mathcal{J}$, $\mathbf{k} \in \mathcal{T}_j$, $\beta \in \mathbb{R}^-$.

For every $\mathbf{a} \in \mathcal{C}_\Gamma^{(\mathbf{k})}$, we define $w_{\mathbf{a}}$ words $\mathbf{b}^{(1)}, \mathbf{b}^{(2)}, \dots, \mathbf{b}^{(w_{\mathbf{a}})} \in \mathcal{C}_j$. Assume some ordering on the elements $\beta \in \mathbb{R}^-$ satisfying $\beta \mathcal{H}_{j,i} = a_i$, namely, $\beta_1, \beta_2, \dots, \beta_{\ell_0}$ for some positive integer ℓ_0 . For $i \in \mathcal{I}_j$, $\mathbf{b}_i^{(\ell)}$ ($\ell = 1, 2, \dots, w_{\mathbf{a}}$) is defined as follows: $\mathbf{b}_i^{(\ell)}$ is equal to β_1 for the first $\mu \cdot z_{i,j,\mathbf{k}}^{(\beta_1)} / \sigma_{j,\mathbf{k}}$ words $\mathbf{b}^{(1)}, \mathbf{b}^{(2)}, \dots, \mathbf{b}^{(w_{\mathbf{a}})}$; $\mathbf{b}_i^{(\ell)}$ is equal to β_2 for the next $\mu \cdot z_{i,j,\mathbf{k}}^{(\beta_2)} / \sigma_{j,\mathbf{k}}$ words; and so on. For every $\mathbf{b} \in \mathcal{C}_j$, we define

$$w'_{\mathbf{b}} = \left| \left\{ i \in \{1, 2, \dots, w_{\mathbf{a}}\} : \mathbf{b}^{(i)} = \mathbf{b} \right\} \right|.$$

Finally, for every $\mathbf{b} \in \mathcal{C}_j$, $\boldsymbol{\kappa}_j(\mathbf{b}) = \mathbf{k}$, we define

$$w_{j,\mathbf{b}} = \frac{\sigma_{j,\mathbf{k}}}{\mu} \cdot w'_{\mathbf{b}}.$$

Using Proposition 8.1

$$\sum_{\mathbf{a} \in \mathcal{C}_\Gamma^{(\mathbf{k})}, a_i = \alpha} w_{\mathbf{a}} = \mu \cdot \frac{\tau_{i,j,\mathbf{k}}^{(\alpha)}}{\sigma_{j,\mathbf{k}}} = \sum_{\beta: \beta \mathcal{H}_{j,i} = \alpha} \mu \cdot \frac{z_{i,j,\mathbf{k}}^{(\beta)}}{\sigma_{j,\mathbf{k}}}$$

and so all $\mathbf{b}^{(1)}, \mathbf{b}^{(2)}, \dots, \mathbf{b}^{(w_{\mathbf{a}})}$ (for all $\mathbf{a} \in \mathcal{C}_\Gamma^{(\mathbf{k})}$) are well defined. It is also straightforward to see that $\mathbf{b}^{(\ell)} \in \mathcal{C}_j$ for

$\ell = 1, 2, \dots, w_{\mathbf{a}}$. Next, we check that the newly defined $w_{j,\mathbf{b}}$ satisfy (8)–(10) for every $j \in \mathcal{J}, \mathbf{b} \in \mathcal{C}_j$. It is easy to see that $w_{j,\mathbf{b}} \geq 0$; therefore (8) holds. By Proposition 8.1 we obtain

$$\sigma_{j,\mathbf{k}} = \sum_{\mathbf{b} \in \mathcal{C}_j, \kappa_j(\mathbf{b}) = \mathbf{k}} w_{j,\mathbf{b}}$$

for all $j \in \mathcal{J}, \mathbf{k} \in \mathcal{T}_j$, and

$$\tau_{i,j,\mathbf{k}}^{(\alpha)} = \sum_{\mathbf{b} \in \mathcal{C}_j, \kappa_j(\mathbf{b}) = \mathbf{k}, b_i \mathcal{H}_{j,i} = \alpha} w_{j,\mathbf{b}}$$

for all $j \in \mathcal{J}, i \in \mathcal{I}_j, \mathbf{k} \in \mathcal{T}_j, \alpha \in \mathbb{R}^-$. Let $\beta \mathcal{H}_{j,i} = \alpha$. Since

$$\tau_{i,j,\mathbf{k}}^{(\alpha)} = \sum_{\beta: \beta \mathcal{H}_{j,i} = \alpha} z_{i,j,\mathbf{k}}^{(\beta)}$$

by the definition of $w_{j,\mathbf{b}}$ it follows that

$$\begin{aligned} & \sum_{\mathbf{b} \in \mathcal{C}_j, \kappa(\mathbf{b}) = \mathbf{k}, b_i = \beta} w_{j,\mathbf{b}} \\ &= \frac{z_{i,j,\mathbf{k}}^{(\beta)}}{\tau_{i,j,\mathbf{k}}^{(\alpha)}} \cdot \sum_{\mathbf{b} \in \mathcal{C}_j, \kappa(\mathbf{b}) = \mathbf{k}, b_i \mathcal{H}_{j,i} = \alpha} w_{j,\mathbf{b}} \\ &= z_{i,j,\mathbf{k}}^{(\beta)}, \end{aligned}$$

where the first equality is due to the definition of the words $\mathbf{b}^{(\ell)}, \ell = 1, 2, \dots, w_{\mathbf{a}}$.

By constraint (41) we have, for all $j \in \mathcal{J}$

$$\begin{aligned} 1 &= \sum_{\mathbf{k} \in \mathcal{T}_j} \sigma_{j,\mathbf{k}} \\ &= \sum_{\mathbf{k} \in \mathcal{T}_j} \sum_{\mathbf{b} \in \mathcal{C}_j, \kappa_j(\mathbf{b}) = \mathbf{k}} w_{j,\mathbf{b}} \\ &= \sum_{\mathbf{b} \in \mathcal{C}_j} w_{j,\mathbf{b}} \end{aligned}$$

thus satisfying (9).

Finally, by constraint (40) we obtain, for all $j \in \mathcal{J}, i \in \mathcal{I}_j, \beta \in \mathbb{R}^-$

$$\begin{aligned} f_i^{(\beta)} &= \sum_{\mathbf{k} \in \mathcal{T}_j} z_{i,j,\mathbf{k}}^{(\beta)} \\ &= \sum_{\mathbf{k} \in \mathcal{T}_j} \sum_{\mathbf{b} \in \mathcal{C}_j, \kappa_j(\mathbf{b}) = \mathbf{k}, b_i = \beta} w_{j,\mathbf{b}} \\ &= \sum_{\mathbf{b} \in \mathcal{C}_j, b_i = \beta} w_{j,\mathbf{b}}, \end{aligned}$$

thus satisfying (10). \square

IX. CASCADED POLYTOPE REPRESENTATION

In this section we show that the ‘‘cascaded polytope’’ representation described in [21]–[23], can be extended to nonbinary codes in a straightforward manner. In the following, we elaborate on the details.

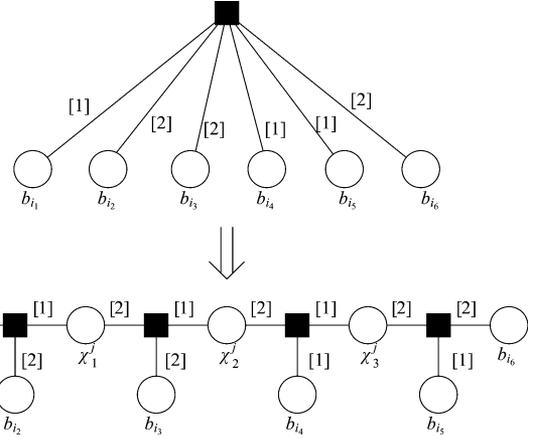


Fig. 3. Example of the Tanner graph of a local code $\mathcal{C}_j = \{(b_{i_1} \ b_{i_2} \ b_{i_3} \ b_{i_4} \ b_{i_5} \ b_{i_6}) : b_{i_1} + 2b_{i_2} + 2b_{i_3} + b_{i_4} + b_{i_5} + 2b_{i_6} = 0\}$ of length $d_j = 6$ over $\mathbb{R} = \mathbb{Z}_3$, and its transformation into the Tanner graph of the corresponding code $\mathcal{C}_j^{(X)}$. Note that the degree of each parity-check vertex in the transformed graph is equal to 3.

For $j \in \mathcal{J}$, consider the j th row \mathcal{H}_j of the parity-check matrix \mathcal{H} over \mathbb{R} , and recall that

$$\mathcal{C}_j = \left\{ (b_i)_{i \in \mathcal{I}_j} : \sum_{i \in \mathcal{I}_j} b_i \cdot \mathcal{H}_{j,i} = 0 \right\}.$$

Assume that $\mathcal{I}_j = \{i_1, i_2, \dots, i_{d_j}\}$ and denote $\mathcal{L}_j = \{1, 2, \dots, d_j - 3\}$. We introduce new variables

$$\chi^j = (\chi_i^j)_{i \in \mathcal{L}_j}$$

and denote

$$\chi = (\chi^j)_{j \in \mathcal{J}}.$$

We define a new linear code $\mathcal{C}_j^{(X)}$ of length $2d_j - 3$ by the $(d_j - 2) \times (2d_j - 3)$ parity-check matrix \mathcal{F}_j associated with the following set of parity-check equations over \mathbb{R} :

$$1) \quad b_{i_1} \mathcal{H}_{j,i_1} + b_{i_2} \mathcal{H}_{j,i_2} + \chi_1^j = 0. \quad (55)$$

$$2) \quad \text{For every } \ell = 1, 2, \dots, d_j - 4 \\ -\chi_\ell^j + b_{i_{\ell+2}} \mathcal{H}_{j,i_{\ell+2}} + \chi_{\ell+1}^j = 0. \quad (56)$$

$$3) \quad -\chi_{d_j-3}^j + b_{i_{d_j-1}} \mathcal{H}_{j,i_{d_j-1}} + b_{i_{d_j}} \mathcal{H}_{j,i_{d_j}} = 0. \quad (57)$$

We also define a linear code $\mathcal{C}^{(X)}$ of length $n + \sum_{j \in \mathcal{J}} (d_j - 3)$ defined by the $(\sum_{j \in \mathcal{J}} (d_j - 2)) \times (n + \sum_{j \in \mathcal{J}} (d_j - 3))$ parity-check matrix \mathcal{F} associated with all the sets of parity-check equations (55)–(57) (for all $j \in \mathcal{J}$). We adopt the notation $\check{\mathbf{b}} = (\mathbf{b} \mid \chi^j)$ for codewords of $\mathcal{C}_j^{(X)}$, and $\check{\mathbf{c}} = (\mathbf{c} \mid \chi)$ for codewords of $\mathcal{C}^{(X)}$.

Example 9.1: Fig. 3 presents an example of the Tanner graph of a local code

$$\mathcal{C}_j = \{(b_{i_1} \ b_{i_2} \ b_{i_3} \ b_{i_4} \ b_{i_5} \ b_{i_6}) : b_{i_1} + 2b_{i_2} + 2b_{i_3} + b_{i_4} + b_{i_5} + 2b_{i_6} = 0\}$$

of length $d_j = 6$ over $\mathbb{R} = \mathbb{Z}_3$, and the Tanner graph of the corresponding code $\mathcal{C}_j^{(X)}$ of length 9 (three extra variables were

added). The degree of every parity-check vertex in the Tanner graph of $\mathcal{C}_j^{(\chi)}$ is at most 3.

The following theorem relates the codes \mathcal{C}_j and $\mathcal{C}_j^{(\chi)}$.

Theorem 9.1: The vector $\mathbf{b} = (b_i)_{i \in \mathcal{I}_j} \in \mathbb{R}^{d_j}$ is a codeword of \mathcal{C}_j if and only if there exists a vector $\boldsymbol{\chi}^j \in \mathbb{R}^{d_j-3}$ such that $(\mathbf{b} \mid \boldsymbol{\chi}^j) \in \mathcal{C}_j^{(\chi)}$.

Proof:

1) Assume $\mathbf{b} = (b_i)_{i \in \mathcal{I}_j} \in \mathcal{C}_j$. Define

$$\chi_\ell^j = \begin{cases} -b_{i_1} \mathcal{H}_{j,i_1} - b_{i_2} \mathcal{H}_{j,i_2}, & \text{if } \ell = 1 \\ \chi_{\ell-1}^j - b_{i_{\ell+1}} \mathcal{H}_{j,i_{\ell+1}}, & \text{if } 2 \leq \ell \leq d_j - 3. \end{cases} \quad (58)$$

Then, obviously, (55) holds, and (56) holds for all $1 \leq \ell \leq d_j - 4$. Finally, (57) follows from subtraction of (55) and (56) (for each $1 \leq \ell \leq d_j - 4$) from the equation $\sum_{i \in \mathcal{I}_j} b_i \mathcal{H}_{j,i} = 0$. Therefore, $(\mathbf{b} \mid \boldsymbol{\chi}^j) \in \mathcal{C}_j^{(\chi)}$, as required.

2) Now, assume that $\mathbf{b} = (b_i)_{i \in \mathcal{I}_j}$ is such that $(\mathbf{b} \mid \boldsymbol{\chi}^j) \in \mathcal{C}_j^{(\chi)}$ for some $\boldsymbol{\chi}^j \in \mathbb{R}^{d_j-3}$, and thus (55)–(57) hold (in particular, (56) holds for all $1 \leq \ell \leq d_j - 4$). We sum all the equalities in (55)–(57) and obtain that $\sum_{i \in \mathcal{I}_j} b_i \mathcal{H}_{j,i} = 0$. Therefore, $\mathbf{b} \in \mathcal{C}_j$. \square

Note that from this theorem we may see that for every $\mathbf{b} \in \mathcal{C}_j$, there exists a unique $\boldsymbol{\chi}^j = \boldsymbol{\chi}^j(\mathbf{b})$ such that $\tilde{\mathbf{b}} = (\mathbf{b} \mid \boldsymbol{\chi}^j) \in \mathcal{C}_j^{(\chi)}$, via (58); we may therefore use the notation $\tilde{\mathbf{b}}(\mathbf{b}) = (\mathbf{b} \mid \boldsymbol{\chi}^j(\mathbf{b}))$ to denote this unique completion, where $\boldsymbol{\chi}^j(\mathbf{b}) = (\chi_i^j(\mathbf{b}))_{i \in \mathcal{L}_j}$.

It follows from Theorem 9.1 that the set of parity-check equations (55)–(57) for all $j \in \mathcal{J}$ equivalently describes the code \mathcal{C} . This description has at most $n + m \cdot (d - 3)$ variables and $m \cdot (d - 2)$ parity-check equations. However, the number of variables participating in every parity-check equation is at most three. Therefore, the total number of variables and of constraints in the corresponding LP problem (defined by constraints (8)–(10) applied to the parity-check matrix \mathcal{F}) is bounded from above by

$$(n + m(d - 3))(q - 1) + m(d - 2) \cdot q^2$$

and

$$m(d - 2)(q^2 + 3q - 2)$$

respectively.

In the sequel, we make use of some new notations, which we define next. First of all, with each parity-check equation prescribed by the matrix \mathcal{F} , we associate a pair of indices (j, ℓ) , $j \in \mathcal{J}$, $\ell = 1, 2, \dots, d_j - 2$, where j indicates the corresponding parity-check equation in \mathcal{H} , and ℓ indicates the serial number of the parity-check equation in the set of (55)–(57) corresponding to the j th row of \mathcal{H} . Denote by $\mathcal{I}_{j,\ell} \subseteq \mathcal{I}_j$ and $\mathcal{L}_{j,\ell} \subseteq \mathcal{L}_j$ the sets of indices i of variables b_i and χ_i^j , respectively, corresponding to the nonzero entries in row (j, ℓ) of \mathcal{F} . Then, each row of \mathcal{F} defines a single parity-check code $\mathcal{C}_{j,\ell}^{(\chi)}$. For any $\mathbf{g} \in \mathcal{C}_{j,\ell}^{(\chi)}$, we adopt the notation $\mathbf{g} = (\mathbf{g}^b \mid \mathbf{g}^\chi)$ where

$$\mathbf{g}^b = (g_i^b)_{i \in \mathcal{I}_{j,\ell}}; \mathbf{g}^\chi = (g_i^\chi)_{i \in \mathcal{L}_{j,\ell}}.$$

We denote by \mathcal{S} the polytope corresponding to the LP relaxation (8)–(10) for the code $\mathcal{C}^{(\chi)}$ with the parity-check matrix \mathcal{F} . Recall that codewords of $\mathcal{C}^{(\chi)}$ are denoted $\tilde{\mathbf{c}} = (\mathbf{c} \mid \boldsymbol{\chi})$. It

is natural to represent points in \mathcal{S} as $((\mathbf{f}, \mathbf{h}), \mathbf{z})$, where $\mathbf{f} = (f_i^{(\alpha)})_{i \in \mathcal{I}, \alpha \in \mathbb{R}^-}$ and $\mathbf{h} = (h_{j,i}^{(\alpha)})_{j \in \mathcal{J}, i \in \mathcal{L}_j, \alpha \in \mathbb{R}^-}$ are vectors of indicators corresponding to the entries c_i ($i \in \mathcal{I}$) in \mathbf{c} and χ_i^j ($j \in \mathcal{J}$, $i \in \mathcal{L}_j$) in $\boldsymbol{\chi}$, respectively. Here

$$\mathbf{z} = (z_{j,\ell}, \mathbf{g})_{j \in \mathcal{J}, \ell=1,2,\dots,d_j-2, \mathbf{g} \in \mathcal{C}_{j,\ell}^{(\chi)}}$$

is a vector of weights associated with each parity-check equation (j, ℓ) and each codeword $\mathbf{g} \in \mathcal{C}_{j,\ell}^{(\chi)}$.

Similarly, for each $j \in \mathcal{J}$, we denote by \mathcal{S}_j the polytope corresponding to the LP relaxation (8)–(10) for the code $\mathcal{C}_j^{(\chi)}$, defined by the parity-check matrix \mathcal{F}_j . Recall that codewords of $\mathcal{C}_j^{(\chi)}$ are denoted $\tilde{\mathbf{b}} = (\mathbf{b} \mid \boldsymbol{\chi}^j)$. Then, it is also natural to represent points in \mathcal{S}_j as $((\hat{\mathbf{f}}_j, \hat{\mathbf{h}}_j), \hat{\mathbf{z}}_j)$, where $\hat{\mathbf{f}}_j = (f_i^{(\alpha)})_{i \in \mathcal{I}_j, \alpha \in \mathbb{R}^-}$ and $\hat{\mathbf{h}}_j = (h_{j,i}^{(\alpha)})_{i \in \mathcal{L}_j, \alpha \in \mathbb{R}^-}$ are vectors of indicators corresponding to the entries b_i ($i \in \mathcal{I}_j$) in \mathbf{b} and χ_i^j ($i \in \mathcal{L}_j$) in $\boldsymbol{\chi}^j$, respectively. Moreover

$$\hat{\mathbf{z}}_j = (z_{j,\ell}, \mathbf{g})_{\ell=1,2,\dots,d_j-2, \mathbf{g} \in \mathcal{C}_{j,\ell}^{(\chi)}}$$

is a vector of weights associated with each parity-check equation (j, ℓ) and each codeword $\mathbf{g} \in \mathcal{C}_{j,\ell}^{(\chi)}$.

For each $j \in \mathcal{J}$, define the mapping Ξ_j analogously to the mapping Ξ with respect to the dimensionality of the code $\mathcal{C}_j^{(\chi)}$, namely

$$\Xi_j : \mathbb{R}^{2d_j-3} \longrightarrow \{0, 1\}^{(q-1)(2d_j-3)} \subset \mathbb{R}^{(q-1)(2d_j-3)}$$

such that for $\tilde{\mathbf{b}} = (\mathbf{b} \mid \boldsymbol{\chi}^j) \in \mathcal{C}_j^{(\chi)}$

$$\Xi_j(\tilde{\mathbf{b}}) = (\xi(b_{i_1}) \mid \xi(b_{i_2}) \mid \dots \mid \xi(b_{i_{d_j}}) \mid \xi(\chi_1^j) \mid \xi(\chi_2^j) \mid \dots \mid \xi(\chi_{d_j-3}^j)).$$

The next lemma is similar to one of the claims of Proposition 10 in [5].

Lemma 9.2: Let \mathcal{C} be a code of length n over \mathbb{R} with parity-check matrix \mathcal{H} , and let $\mathcal{Q}(\mathcal{H})$ be the corresponding polytope of the LP relaxation, i.e., the set of points (\mathbf{f}, \mathbf{w}) satisfying (8)–(10). Let $\bar{\mathcal{Q}}(\mathcal{H})$ denote the projection of \mathcal{Q} onto the \mathbf{f} variables, i.e.,

$$\bar{\mathcal{Q}}(\mathcal{H}) = \{\mathbf{f} : \exists \mathbf{w} \text{ s.t. } (\mathbf{f}, \mathbf{w}) \in \mathcal{Q}\}.$$

Denote by \mathbf{P} the set of *normalized* graph-cover pseudocodeword vectors associated with \mathcal{H} . Then, $\bar{\mathcal{Q}}(\mathcal{H}) = \bar{\mathbf{P}}$, where $\bar{\mathbf{P}}$ is the closure of \mathbf{P} under the usual (Euclidean) metric in $\mathbb{R}^{(q-1)n}$.

Proof: Generally, the proof is similar to the proof of the relevant parts of Proposition 10 in [5]. It is largely based on the equivalence between the set of graph-cover pseudocodewords and the set of LP pseudocodewords (Theorem 7.1 and Corollary 7.2). We avoid many technical details, and mention only the main ideas. The proof consists of proving two main claims.

1) $\mathbf{P} \subseteq \bar{\mathcal{Q}}(\mathcal{H})$.

Given any normalized graph-cover pseudocodeword vector $\mathbf{f} \in \mathbf{P}$, by Corollary 7.2 there must exist \mathbf{w} with $(\mathbf{f}, \mathbf{w}) \in \mathcal{Q}(\mathcal{H})$. Therefore, $\mathbf{f} \in \bar{\mathcal{Q}}(\mathcal{H})$.

2) If a point in $\bar{\mathcal{Q}}(\mathcal{H})$ has all rational entries, then it must also be in \mathbf{P} .

The proof follows the lines of the proof of Lemma 56 in [5]. Let $(\mathbf{f}, \mathbf{w}) \in \mathcal{Q}(\mathcal{H})$ be a point such that all entries in \mathbf{f} are rational. Then for all $j \in \mathcal{J}$, the vector $\hat{\mathbf{f}}_j = (\mathbf{f}_i)_{i \in \mathcal{I}_j}$ lies in the convex hull $\mathcal{K}(\mathcal{C}_j)$. For convenience in what follows,

denote the index set $\Psi = \{1, 2, \dots, (q-1)n + 1\}$. Using Carathéodory's theorem [26, p. 10], for all $j \in \mathcal{J}$ we may write $\mathbf{f} = \boldsymbol{\mu}^{(j)} \mathbf{P}^{(j)}$ where $\boldsymbol{\mu}^{(j)} = (\mu_i^{(j)})_{i \in \Psi}$ is a row vector of length $|\Psi|$ whose elements sum to unity, and $\mathbf{P}^{(j)}$ is a $|\Psi| \times (|\Psi| - 1)$ matrix such that for each $i \in \Psi$, the i th row of $\mathbf{P}^{(j)}$, denoted $\mathbf{p}_i^{(j)}$, satisfies $\mathbf{p}_i^{(j)} = \boldsymbol{\Xi}(\mathbf{c})$ for some $\mathbf{c} \in \mathfrak{R}^n$ with $\mathbf{x}_j(\mathbf{c}) \in \mathcal{C}_j$. Therefore

$$(\mathbf{f} \mathbf{1}) = \boldsymbol{\mu}^{(j)} (\mathbf{P}^{(j)} \mathbf{1})$$

where $\mathbf{1}$ denotes a vector of length $|\Psi|$ all of whose entries are equal to 1, is a $|\Psi| \times |\Psi|$ system; therefore, by Cramer's rule the solution for $\boldsymbol{\mu}^{(j)}$ has all rational entries (this argument applies for every $j \in \mathcal{J}$). Let M denote a common denominator of all variables in vectors $\boldsymbol{\mu}^{(j)}$, for $j \in \mathcal{J}$. Define $h_i^{(\alpha)} = M f_i^{(\alpha)} \in \mathbb{R}$ for each $i \in \mathcal{I}$, $\alpha \in \mathfrak{R}^-$ (it is easy to see that these variables must be nonnegative integers). Also define $\delta_i^{(j)} = M \mu_i^{(j)}$ for each $i \in \Psi$, $j \in \mathcal{J}$, and $\boldsymbol{\delta}^{(j)} = (\delta_i^{(j)})_{i \in \Psi}$. We then have

$$\mathbf{h} = \boldsymbol{\delta}^{(j)} \mathbf{P}^{(j)}. \quad (59)$$

Next define, for all $j \in \mathcal{J}$, $\mathbf{b} \in \mathcal{C}_j$

$$z_{j,\mathbf{b}} = \sum_{i \in \Psi: \mathbf{p}_i^{(j)} = \boldsymbol{\Xi}(\mathbf{c}), \mathbf{x}_j(\mathbf{c}) = \mathbf{b}} \delta_i^{(j)}.$$

By comparing appropriate entries in the vector (59), we obtain

$$\forall j \in \mathcal{J}, \forall i \in \mathcal{I}_j, \forall \alpha \in \mathfrak{R}^-, \quad h_i^{(\alpha)} = \sum_{\mathbf{b} \in \mathcal{C}_j, b_i = \alpha} z_{j,\mathbf{b}}$$

and so (\mathbf{h}, \mathbf{z}) is an LP pseudocodeword (the preceding equation yields (26), and (27) follows from the fact that the sum of the entries in $\boldsymbol{\delta}^{(j)}$ is equal to M for all $j \in \mathcal{J}$, these entries being nonnegative integers). So the construction of Theorem 7.1, part 1, yields a corresponding graph-cover pseudocodeword with graph-cover pseudocodeword vector \mathbf{h} . Therefore, the corresponding normalized graph-cover pseudocodeword vector is \mathbf{f} , and so we must have $\mathbf{f} \in \mathcal{P}$.

The claim of the lemma follows. \square

The following proposition is a counterpart of Lemma 28 in [5].

Proposition 9.3: Let \mathcal{C} be a code of length n over \mathfrak{R} with parity-check matrix \mathcal{H} . Assume that the Tanner graph represented by \mathcal{H} is a tree. Then, the projected polytope $\bar{\mathcal{Q}}(\mathcal{H})$ of the corresponding LP relaxation problem is equal to $\mathcal{K}(\mathcal{C})$.

Proof: The proof follows the lines of the proof of Lemma 28 in [5]. Let \mathcal{G} be the labeled Tanner graph of the code \mathcal{C} corresponding to \mathcal{H} . Let $\tilde{\mathcal{G}}$ be an M -cover of \mathcal{G} for some positive integer M . Since \mathcal{G} is a tree, $\tilde{\mathcal{G}}$ is a collection of M labeled trees which are copies of \mathcal{G} . Let $\tilde{\mathcal{C}}$ be a code defined by the parity-check matrix corresponding to this $\tilde{\mathcal{G}}$. We obtain that

$$\tilde{\mathcal{C}} = \left\{ \mathbf{x} \in \mathfrak{R}^{Mn} : (x_{1,m}, x_{2,m}, \dots, x_{n,m}) \in \mathcal{C}, \right. \\ \left. \text{for all } m = 1, 2, \dots, M \right\}.$$

Then, it is easy to see that the set of *normalized* graph-cover pseudocodeword vectors of \mathcal{H} , \mathcal{P} , is equal to $\mathcal{K}(\mathcal{C}) \cap \mathbb{Q}^{(q-1)n}$.

To this end, we apply Lemma 9.2 to see that

$$\bar{\mathcal{Q}}(\mathcal{H}) = \bar{\mathcal{P}} = \overline{\mathcal{K}(\mathcal{C}) \cap \mathbb{Q}^{(q-1)n}} = \mathcal{K}(\mathcal{C})$$

as required. \square

By taking $\mathcal{C} = \mathcal{C}_j^{(x)}$ and $\mathcal{H} = \mathcal{F}_j$ so that $\mathcal{Q}(\mathcal{H}) = \mathcal{S}_j$ (for $j \in \mathcal{J}$), we immediately obtain the following corollary.

Corollary 9.4: For $j \in \mathcal{J}$, let

$$\bar{\mathcal{S}}_j = \{(\hat{\mathbf{f}}_j, \hat{\mathbf{h}}_j) : \exists \hat{\mathbf{z}}_j \text{ s.t. } ((\hat{\mathbf{f}}_j, \hat{\mathbf{h}}_j), \hat{\mathbf{z}}_j) \in \mathcal{S}_j\}.$$

Then $\bar{\mathcal{S}}_j = \mathcal{K}(\mathcal{C}_j^{(x)})$.

The proof of the next theorem requires the following definition. Let $\mathbf{b} \in \mathcal{C}_j$, and let $\mathbf{g} \in \mathcal{C}_{j,\ell}^{(x)}$. We say that \mathbf{g} *coincides with* \mathbf{b} , writing $\mathbf{g} \bowtie \mathbf{b}$, if and only if $g_i^b = b_i$ for all $i \in \mathcal{I}_{j,\ell}$, and $g_i^x = \chi_i^j(\mathbf{b})$ for all $i \in \mathcal{L}_{j,\ell}$.

Theorem 9.5: The set $\bar{\mathcal{S}} = \{\mathbf{f} : \exists \mathbf{h}, \mathbf{z} \text{ s.t. } ((\mathbf{f}, \mathbf{h}), \mathbf{z}) \in \mathcal{S}\}$ is equal to the set $\bar{\mathcal{Q}} = \{\mathbf{f} : \exists \mathbf{w} \text{ s.t. } (\mathbf{f}, \mathbf{w}) \in \bar{\mathcal{Q}}\}$, and, therefore, optimizing the linear cost function (5) over \mathcal{S} is equivalent to optimizing over $\bar{\mathcal{Q}}$.

Proof:

1) Let $\mathbf{f} \in \bar{\mathcal{Q}}$. Then, there exists \mathbf{w} such that $(\mathbf{f}, \mathbf{w}) \in \bar{\mathcal{Q}}$. Therefore

$$\forall j \in \mathcal{J}, \forall i \in \mathcal{I}_j, \forall \alpha \in \mathfrak{R}^-, \quad f_i^{(\alpha)} = \sum_{\mathbf{b} \in \mathcal{C}_j, b_i = \alpha} w_{j,\mathbf{b}}. \quad (60)$$

In addition, the entries in \mathbf{w} satisfy (8) and (9).

We set the values of the variables $z_{j,\ell,\mathbf{g}}$ as follows:

$$\forall j \in \mathcal{J}, \forall \ell = 1, 2, \dots, d_j - 2, \forall \mathbf{g} \in \mathcal{C}_{j,\ell}^{(x)}, \\ z_{j,\ell,\mathbf{g}} = \sum_{\mathbf{b} \in \mathcal{C}_j, \mathbf{g} \bowtie \mathbf{b}} w_{j,\mathbf{b}}$$

So we have that

$$\forall j \in \mathcal{J}, \forall \ell = 1, 2, \dots, d_j - 2, \forall i \in \mathcal{I}_{j,\ell}, \forall \alpha \in \mathfrak{R}^-, \\ \sum_{\mathbf{g} \in \mathcal{C}_{j,\ell}^{(x)}, g_i^b = \alpha} z_{j,\ell,\mathbf{g}} = \sum_{\mathbf{b} \in \mathcal{C}_j, b_i = \alpha} w_{j,\mathbf{b}} = f_i^{(\alpha)} \quad (61)$$

using (60), since $\mathcal{I}_{j,\ell} \subseteq \mathcal{I}_j$ for all $\ell = 1, 2, \dots, d_j - 2$. In addition, we define the variables $h_{j,i}^{(\alpha)}$ as follows:

$$\forall j \in \mathcal{J}, \forall i \in \mathcal{L}_j, \forall \alpha \in \mathfrak{R}^-, \\ h_{j,i}^{(\alpha)} = \sum_{\mathbf{b} \in \mathcal{C}_j, \chi_i^j(\mathbf{b}) = \alpha} w_{j,\mathbf{b}}. \quad (62)$$

Note that all variables $h_{j,i}^{(\alpha)}$ are well defined. It then follows that

$$\forall j \in \mathcal{J}, \forall \ell = 1, 2, \dots, d_j - 2, \forall i \in \mathcal{L}_{j,\ell}, \forall \alpha \in \mathfrak{R}^-, \\ \sum_{\mathbf{g} \in \mathcal{C}_{j,\ell}^{(x)}, g_i^x = \alpha} z_{j,\ell,\mathbf{g}} = \sum_{\mathbf{b} \in \mathcal{C}_j, \chi_i^j(\mathbf{b}) = \alpha} w_{j,\mathbf{b}} = h_{j,i}^{(\alpha)}, \quad (63)$$

using (60), since $\mathcal{L}_{j,\ell} \subseteq \mathcal{L}_j$ for all $\ell = 1, 2, \dots, d_j - 2$. Next, we claim that

$$((\mathbf{f}, \mathbf{h}), \mathbf{z}) \in \mathcal{S}. \quad (64)$$

In order to show this, it is necessary to show (8)–(10) with respect to $((\mathbf{f}, \mathbf{h}), \mathbf{z})$ and the code $\mathcal{C}^{(x)}$. However, (8) and (9) follow easily from the definition of the variables $z_{j,\ell,g}$, and the properties of the variables $w_{j,\mathbf{b}}$. As to (10), it follows from the combination of (61) and (63).

Finally, (64) yields that $\mathbf{f} \in \bar{\mathcal{S}}$, as required.

- 2) Now, assume that $\mathbf{f} \in \bar{\mathcal{S}}$. This means that there exist \mathbf{h}, \mathbf{z} such that $((\mathbf{f}, \mathbf{h}), \mathbf{z}) \in \mathcal{S}$. Then, for all $j \in \mathcal{J}$, $((\hat{\mathbf{f}}_j, \hat{\mathbf{h}}_j), \hat{\mathbf{z}}_j) \in \mathcal{S}_j$. By Corollary 9.4, $(\hat{\mathbf{f}}_j, \hat{\mathbf{h}}_j)$ lies in $\mathcal{K}(\mathcal{C}_j^{(x)})$. Therefore

$$(\hat{\mathbf{f}}_j, \hat{\mathbf{h}}_j) = \sum_{\tilde{\mathbf{b}} \in \mathcal{C}_j^{(x)}} \beta_{j,\tilde{\mathbf{b}}} \cdot \Xi_j(\tilde{\mathbf{b}}) \quad (65)$$

where $\sum_{\tilde{\mathbf{b}} \in \mathcal{C}_j^{(x)}} \beta_{j,\tilde{\mathbf{b}}} = 1$ and $\beta_{j,\tilde{\mathbf{b}}} \geq 0$ for all $\tilde{\mathbf{b}} \in \mathcal{C}_j^{(x)}$.

For all $j \in \mathcal{J}$, $\mathbf{b} \in \mathcal{C}_j$ set the value of $w_{j,\mathbf{b}}$ as

$$w_{j,\mathbf{b}} = \beta_{j,\tilde{\mathbf{b}}(\mathbf{b})}$$

and thus

$$\sum_{\mathbf{b} \in \mathcal{C}_j} w_{j,\mathbf{b}} = 1 \quad (66)$$

and

$$w_{j,\mathbf{b}} \geq 0, \quad \text{for all } \mathbf{b} \in \mathcal{C}_j. \quad (67)$$

Then, (65) becomes

$$(\hat{\mathbf{f}}_j, \hat{\mathbf{h}}_j) = \sum_{\mathbf{b} \in \mathcal{C}_j} w_{j,\mathbf{b}} \cdot \Xi_j(\tilde{\mathbf{b}}(\mathbf{b})).$$

Comparing the first set of coordinates, we obtain that

$$\forall i \in \mathcal{I}_j, \forall \alpha \in \mathbb{R}^-, \quad f_i^{(\alpha)} = \sum_{\mathbf{b} \in \mathcal{C}_j, b_i = \alpha} w_{j,\mathbf{b}}.$$

This set of equations holds for all $j \in \mathcal{J}$. Together with (66) and (67) this means that $(\mathbf{f}, \mathbf{w}) \in \mathcal{Q}$. Therefore, $\mathbf{f} \in \bar{\mathcal{Q}}$, as required. \square

The polytope representation described in this section leads to a polynomial-time decoder for a wide variety of classical nonbinary codes (for example, the generalized Reed–Solomon codes).

X. SIMULATION STUDY

A. Comparison With ML Decoding

In this section, we compare performance of the linear-programming decoder with hard-decision and soft-decision based ML decoding. For such a comparison, a code and modulation scheme are needed which possess sufficient symmetry properties to enable derivation of analytical ML performance results. We consider encoding of six-symbol blocks according to the [11, 6] ternary Golay code, and modulation of the resulting ternary symbols with 3-PSK modulation prior to transmission

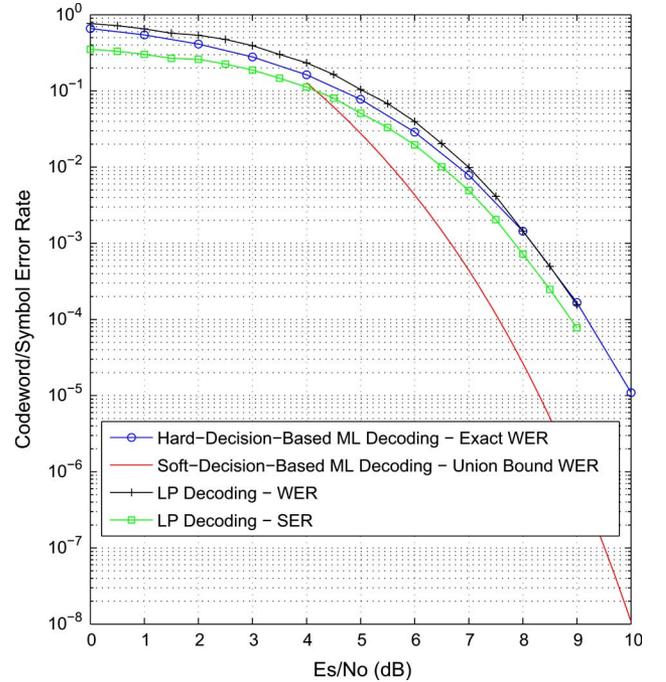


Fig. 4. Codeword error rate (WER) and symbol error rate (SER) for the [11, 6] ternary Golay code with 3-PSK modulation over the AWGN channel. The figure shows performance under LP decoding, as well as the exact result for hard-decision decoding and the union bound for soft-decision decoding.

over the AWGN channel. Fig. 4 shows the symbol error rate (SER) and codeword error rate (WER) performance of this code under LP decoding using the polytope \mathcal{Q} of Section III. Note that this is the same as its performance using the polytope \mathcal{U} of Section VIII, and its performance using the polytope \mathcal{S} of Section IX. When the decoder reports a decoding failure, the SER and WER are both taken to be 1. To quantify performance, we define the signal-to-noise ratio (SNR) per information symbol $\gamma_s = E_s/N_0$ as the ratio of the received signal energy per information symbol to the noise power spectral density. Also shown in the figure are two other performance curves for WER. The first is the exact result for ML hard-decision decoding of the ternary Golay code; since the Golay code is perfect, this is obtained from

$$\text{WER}(\gamma_s) = \sum_{\ell=3}^{11} \binom{11}{\ell} (p(\gamma_s))^\ell (1 - p(\gamma_s))^{11-\ell}$$

where $p(\gamma_s)$ represents the probability of incorrect hard decision at the demodulator and was evaluated for each value of γ_s using numerical integration. The second WER curve represents the union bound for ML soft-decision decoding. Using the symmetry of the 3-PSK constellation, this may be obtained from

$$\text{WER}(\gamma_s) < \frac{1}{2} \sum_{\mathbf{c} \in \mathcal{C}} \text{erfc} \left(\sqrt{\frac{3}{4} w_H(\mathbf{c}) R(C) \gamma_s} \right)$$

where $R(C) = 6/11$ denotes the code rate, and the Hamming weight of the codeword $\mathbf{c} \in \mathcal{C}$, $w_H(\mathbf{c})$, is distributed according to the weight enumerating polynomial [27]

$$W(x) = 1 + 132x^5 + 132x^6 + 330x^8 + 110x^9 + 24x^{11}.$$

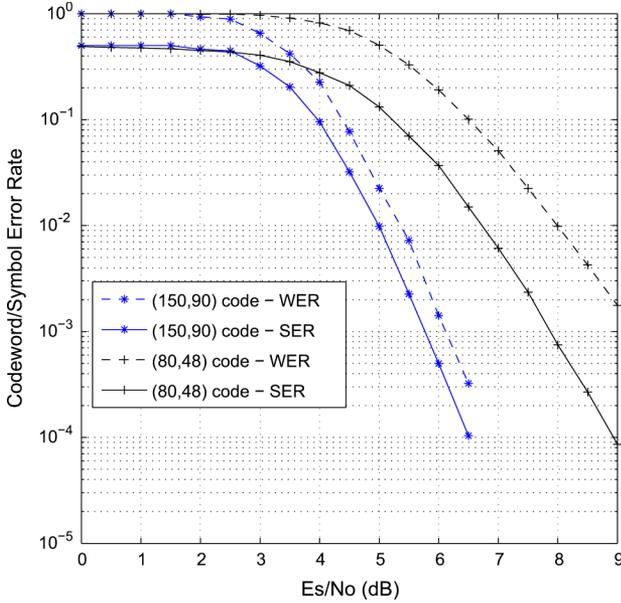


Fig. 5. Codeword error rate (WER) and symbol error rate (SER) for the [150, 90] ternary LDPC code $\mathcal{C}^{(1)}$ under ternary PSK modulation, and for the [80, 48] quaternary LDPC code $\mathcal{C}^{(2)}$ under QPSK modulation.

The performance of LP decoding is approximately the same as that of WER-optimum hard-decision decoding. The performance lies 0.1 dB from the result for ML hard-decision decoding and 1.53 dB from the union bound for WER-optimum soft-decision decoding at a WER of 10^{-4} . These results are comparable to those of a similar study conducted for the binary case in [7].

B. Low-Density Code Performance

Fig. 5 shows SER and WER simulation performance results for two LDPC codes. The first code $\mathcal{C}^{(1)}$, of length $n = 150$, is over the ring $\mathfrak{R} = \mathbb{Z}_3$, where nonbinary coded symbols are mapped directly to ternary PSK signals and transmitted over an AWGN channel, the mapping described in Example 5.3 being used for modulation. The parity-check matrix $\mathcal{H}^{(1)}$ consists of $m = 60$ rows and is equal to the right-circulant matrix

$$\mathcal{H}_{j,i}^{(1)} = \begin{cases} 1, & \text{if } i - j \in \{0, 51, 80\} \\ 2, & \text{if } i - j \in \{8, 30, 90\} \\ 0, & \text{otherwise.} \end{cases}$$

The code rate is $R(\mathcal{C}^{(1)}) = 0.6$. As expected, the performance of the low-density code $\mathcal{C}^{(1)}$ is significantly better than that of the ternary Golay code given in Fig. 4. The second code $\mathcal{C}^{(2)}$, of length $n = 80$, is over the ring $\mathfrak{R} = \mathbb{Z}_4$, where nonbinary coded symbols are mapped directly to quaternary phase-shift keying (QPSK) signals and transmitted over an AWGN channel, the mapping described in Example 5.3 again being used for modulation. The parity-check matrix $\mathcal{H}^{(2)}$ consists of $m = 32$ rows and is equal to the right-circulant matrix

$$\mathcal{H}_{j,i}^{(2)} = \begin{cases} 1, & \text{if } i - j \in \{0, 41, 48\} \\ 3, & \text{if } i - j \in \{8, 25\} \\ 0, & \text{otherwise.} \end{cases}$$

This code also has rate $R(\mathcal{C}^{(2)}) = 0.6$. The quaternary code has a higher SER and WER than the ternary code for the same E_s/N_0 ; however, it has a smaller block length and a higher spectral efficiency. In both systems, when the decoder reports a decoding failure the SER and WER are both taken to be 1.

XI. FUTURE RESEARCH

Sections VIII and IX presented two alternative polytope representations, which have a smaller number of variables and constraints than the respective standard LP representation in certain contexts. It would be interesting to further reduce the complexity of the polytope representation in order to yield more efficient decoding algorithms. Alternatively, one could try to reduce complexity of the LP solver for the nonbinary decoding problem by exploiting knowledge of the polytope structure.

The notion of *pseudodistance* for nonbinary codes was recently defined in [28], and lower bounds on the pseudodistance of nonbinary codes under q -ary PSK modulation over the AWGN channel were presented. It would be interesting to obtain lower bounds on the pseudodistance for other families of nonbinary linear codes and for other modulation schemes.

APPENDIX

PROOF OF PROPOSITION 8.1

Before proving this proposition, we give some background material on flow networks.

Flow Networks: Let $G = (V, E)$ be a directed graph, and let $\{s, t\} \subseteq V, s \neq t$. A flow network $(G(V, E), c)$ is a graph $G = (V, E)$ with a nonnegative *capacity* function $c : E \rightarrow \mathbb{R} \cup \{+\infty\}$ defined for every edge.

For a subset $V' \subseteq V$ let $V'' = V \setminus V'$. We define a cut $(V' : V'')$ induced by V' as a set of edges $\{(u, v) : u \in V', v \in V''\}$. The capacity of this cut, $c(V' : V'')$, is defined as

$$c(V' : V'') = \sum_{u \in V', v \in V''} c((u, v)).$$

For the edge $e = (u, v)$ we use the notation $e \in \text{in}(v)$ and $e \in \text{out}(u)$. We also use the notation $\mathcal{N}(v)$ to denote the set of neighbors of v , namely

$$\mathcal{N}(v) = \{u : (u, v) \in E\} \cup \{u' : (v, u') \in E\}.$$

For a set of vertices $V_0 \subseteq V$, denote

$$\mathcal{N}(V_0) = \cup_{v \in V_0} \mathcal{N}(v) \setminus V_0.$$

The flow in the graph (network) G with a source s and a sink t is defined as a function $f : E \rightarrow \mathbb{R} \cup \{+\infty\}$ that satisfies $0 \leq f(e) \leq c(e)$ for all $e \in E$, and

$$\forall v \in V \setminus \{s, t\}, \sum_{e \in E, e \in \text{in}(v)} f(e) = \sum_{e \in E, e \in \text{out}(v)} f(e).$$

The value of the flow f is defined as

$$\sum_{e \in E, e \in \text{in}(t)} f(e) = \sum_{e \in E, e \in \text{out}(s)} f(e).$$

The maximum flow in the network is defined as the flow f that attains the maximum possible value. There are several known algorithms, for instance the Ford–Fulkerson algorithm, for finding

the maximum flow in a network, the reader can refer to [29, Sec. 26.2]. It is well known that the value of the maximum flow in the network is equal to the capacity of the minimum cut induced by a vertex set V' such that $s \in V'$ and $t \notin V'$ (see [29]).

Finally, we prove the proposition.

Proof: The proof will be by induction on M . We set $w_{\mathbf{a}} = 0$ for all $\mathbf{a} \in \mathcal{C}_{\Gamma}^{(\mathbf{k})}$. We show that there exists a vector $\mathbf{a} = (a_i)_{i \in \Gamma} \in \mathcal{C}_{\Gamma}^{(\mathbf{k})}$ such that

(i) For every $i \in \Gamma$ and $\alpha \in \mathfrak{R}^-$

$$a_i = \alpha \implies x_i^{(\alpha)} > 0.$$

(ii) If for some $i \in \Gamma$, $\sum_{\alpha \in \mathfrak{R}^-} x_i^{(\alpha)} = M$, then $a_i = \alpha$ for some $\alpha \in \mathfrak{R}^-$.

Then, we “update” the values of $x_i^{(\alpha)}$'s and M as follows. For every $i \in \Gamma$ and $\alpha \in \mathfrak{R}^-$ with $a_i = \alpha$ we set $x_i^{(\alpha)} \leftarrow x_i^{(\alpha)} - 1$. In addition, we set $M \leftarrow M - 1$. We also set $w_{\mathbf{a}} \leftarrow w_{\mathbf{a}} + 1$.

It is easy to see that the “updated” values of $x_i^{(\alpha)}$'s and M satisfy

$$\sum_{i \in \Gamma} x_i^{(\alpha)} = k_{\alpha} M$$

for all $\alpha \in \mathfrak{R}^-$, and $\sum_{\alpha \in \mathfrak{R}^-} x_i^{(\alpha)} \leq M$ for all $i \in \Gamma$. Therefore, the inductive step can be applied with respect to these new values. The induction ends when the value of M is equal to zero.

It is straightforward to see that when the induction terminates, (53) and (54) hold with respect to the original values of the $x_i^{(\alpha)}$ and M .

Proof of Existence of \mathbf{a} That Satisfies (i): We construct a flow network $G = (V, E)$ as follows:

$$V = \{s, t\} \cup U_1 \cup U_2,$$

where

$$U_1 = \mathfrak{R}^- \quad \text{and} \quad U_2 = \Gamma.$$

Also set

$$E = \{(s, \alpha)\}_{\alpha \in \mathfrak{R}^-} \cup \{(i, t)\}_{i \in \Gamma} \cup \{(\alpha, i)\}_{x_i^{(\alpha)} > 0}.$$

We define an integer capacity function $c : E \rightarrow \mathbb{N} \cup \{+\infty\}$ as follows:

$$c(e) = \begin{cases} k_{\alpha}, & \text{if } e = (s, \alpha), \alpha \in \mathfrak{R}^- \\ 1, & \text{if } e = (i, t), i \in \Gamma \\ +\infty, & \text{if } e = (\alpha, i), \alpha \in \mathfrak{R}^-, i \in \Gamma. \end{cases} \quad (68)$$

Next, apply the Ford–Fulkerson algorithm on the network $(G(E, V), c)$ to produce a maximal flow f_{\max} . Since all the values of $c(e)$ are integer for all $e \in E$, so the values of $f_{\max}(e)$ must all be integer for every $e \in E$ (see [29]).

We will show that the minimum cut in this graph has capacity $c_{\min} = \sum_{\alpha \in \mathfrak{R}^-} k_{\alpha}$. First, consider the cut induced by the set $V' = \{s\}$. This cut has capacity $\sum_{\alpha \in \mathfrak{R}^-} k_{\alpha}$, and therefore, $c_{\min} \leq \sum_{\alpha \in \mathfrak{R}^-} k_{\alpha}$.

Assume that there is another cut, which has smaller capacity. If this smaller cut is induced by the set $V' = V \setminus \{t\}$, its capacity must be $N \geq \sum_{\alpha \in \mathfrak{R}^-} k_{\alpha}$ —it is not smaller. Therefore, without loss of generality, assume that the minimum cut is induced by

the set V' , where $V' = \{s\} \cup X' \cup Y'$, $X' \subseteq U_1$ and $Y' \subseteq U_2$. Let $X'' = U_1 \setminus X'$ and $Y'' = U_2 \setminus Y'$ (and so $V'' = \{t\} \cup X'' \cup Y''$).

Observe that there are no edges $(\alpha, i) \in E$ with $\alpha \in X'$, $i \in Y''$, because otherwise the capacity of the respective cut would be infinitely large (so it cannot be a minimum cut). Thus

$$|Y'| \geq |U_2 \cap \mathcal{N}(X')|. \quad (69)$$

Observe also that

$$\sum_{i \in \Gamma} \sum_{\alpha \in X'} x_i^{(\alpha)} = \sum_{\alpha \in X'} k_{\alpha} M$$

and

$$\sum_{\alpha \in X'} x_i^{(\alpha)} \leq \sum_{\alpha \in \mathfrak{R}^-} x_i^{(\alpha)} \leq M.$$

Therefore

$$|U_2 \cap \mathcal{N}(X')| \geq \sum_{\alpha \in X'} k_{\alpha}. \quad (70)$$

We obtain that

$$\begin{aligned} c(V' : V'') &= \sum_{\alpha \in X''} k_{\alpha} + |Y'| \\ &\geq \sum_{\alpha \in X''} k_{\alpha} + \sum_{\alpha \in X'} k_{\alpha} = \sum_{\alpha \in \mathfrak{R}^-} k_{\alpha} \end{aligned} \quad (71)$$

where the inequality is due to (69) and (70). This leads to a contradiction of the nonminimality of $c(V' : V'')$ for $V' = \{s\}$.

If we apply the Ford–Fulkerson algorithm (or a similar algorithm) on the network $(G(V, E), c)$, we obtain that the integer flow f_{\max} in G has a value of $\sum_{\alpha \in \mathfrak{R}^-} k_{\alpha}$. Observe that $f_{\max}((\alpha, i)) \in \{0, 1\}$ for all $\alpha \in \mathfrak{R}^-$ and $i \in \Gamma$. Then, for all $i \in \Gamma$, we define

$$a_i = \begin{cases} \alpha, & \text{if } f_{\max}((\alpha, i)) = 1 \text{ for some } \alpha \in U_1 \\ 0, & \text{otherwise.} \end{cases}$$

For this selection of $\mathbf{a} = (a_1, a_2, \dots, a_N)$, we have $\mathbf{a} \in \mathcal{C}_{\Gamma}^{(\mathbf{k})}$ and $a_i = \alpha$ only if $x_i^{(\alpha)} > 0$.

Proof of Existence of \mathbf{a} That Satisfies (i) and (ii) Simultaneously: We start with the following definition.

Definition A.1: The vertex $i \in U_2$ is called a *critical* vertex, if

$$\sum_{\alpha \in \mathfrak{R}^-} x_i^{(\alpha)} = M.$$

In order to have (52) satisfied after the next inductive step, we have to decrease the value of $\sum_{\alpha \in \mathfrak{R}^-} x_i^{(\alpha)}$ by (exactly) 1 for every critical vertex. This is equivalent to having $f_{\max}((i, t)) = 1$.

We have just shown that the maximum (integer) flow in G has value $\sum_{\alpha \in \mathfrak{R}^-} k_{\alpha}$. Now, we aim to show that there exists a flow f^* of the same value, which has $f^*((i, t)) = 1$ for every critical vertex i .

Suppose that there is no such flow. Then, consider the maximum flow f' , which has $f'((i, t)) = 1$ for the *maximal possible number* of the critical vertices $i \in U_2$. In the sequel, we assume

that there is a critical vertex $i_0 \in U_2$, which has $f'((i_0, t)) = 0$. We will show that the flow f' can be modified towards the flow f'' of the same value, such that for f'' the number of critical vertices $i \in U_2$ having $f''((i, t)) = 1$ is strictly larger than for f' .

Indeed, if there exists vertex $\alpha_0 \in \mathcal{N}(i)$ such that $(\alpha_0, i_1) \in E$ and $f'((\alpha_0, i_1)) = 1$ for some *noncritical* vertex i_1 , then $f'((\alpha_0, i_0)) = 0$, $f'((i_0, t)) = 0$, and $f'((i_1, t)) = 1$. We define the flow f'' as

$$f''(e) = \begin{cases} 1, & \text{if } e \in \{(\alpha_0, i_0), (i_0, t)\} \\ 0, & \text{if } e \in \{(\alpha_0, i_1), (i_1, t)\} \\ f'(e), & \text{for all other edges } e \in E. \end{cases}$$

It is easy to see that f'' is a legal flow in $(G(V, E), c)$. Moreover, it has the same value as f' , and the number of critical vertices $i \in U_2$ satisfying $f''((i, t)) = 1$ is strictly larger than for f' .

In the general case (when there is no vertex α_0 as above), we iteratively define a maximal set Z of vertices $\alpha \in U_1$ satisfying the next two rules.

- 1) For any $\alpha \in U_1$: if $(\alpha, i_0) \in E$ then $\alpha \in Z$.
- 2) For any $\alpha \in U_1$ and $i \in U_2$: if $(\alpha, i) \in E$, $f'((\alpha, i)) = 0$, and there exists $\beta \in Z$ such that $(\beta, i) \in E$, $f'((\beta, i)) = 1$, and all $i' \in U_2$ with $f'((\beta, i')) = 1$ are critical, then $\alpha \in Z$.

Consider the set Z . There are two cases.

Case 1: Every vertex α in Z satisfies

$$\forall i \in U_2 : (\alpha, i) \in E \text{ and } i \text{ is not critical} \\ \implies f'((\alpha, i)) = 0.$$

Then, for every $\alpha \in Z$ there are exactly k_α vertices i such that $(\alpha, i) \in E$ and $f'((\alpha, i)) = 1$. Define

$$T = \{i \in U_2 \text{ is critical:} \\ \exists \alpha \in Z \text{ s.t. } (\alpha, i) \in E \text{ and } f'((\alpha, i)) = 1\}.$$

We have

$$|T| = \sum_{i \in T} 1 = \sum_{\alpha \in Z} k_\alpha. \quad (72)$$

Note that $i_0 \notin T$ and recall that

$$(\beta, i_0) \in E \text{ for some } \beta \in Z. \quad (73)$$

Note also that if $\gamma \notin Z$ and $i \in T$, then there is no edge between γ and i (otherwise, $f'((\gamma, i)) = 0$, and so γ should be in Z). Therefore, $x_i^{(\gamma)} = 0$, and so

$$\sum_{\alpha \in Z} \sum_{i \in T} x_i^{(\alpha)} = \sum_{\alpha \in \mathcal{R}^-} \sum_{i \in T} x_i^{(\alpha)}. \quad (74)$$

We obtain

$$\begin{aligned} \sum_{\alpha \in Z} k_\alpha M &= \sum_{\alpha \in Z} \sum_{i \in T} x_i^{(\alpha)} > \sum_{\alpha \in Z} \sum_{i \in T} x_i^{(\alpha)} \\ &= \sum_{\alpha \in \mathcal{R}^-} \sum_{i \in T} x_i^{(\alpha)} = \sum_{i \in T} \sum_{\alpha \in \mathcal{R}^-} x_i^{(\alpha)} \\ &= \sum_{i \in T} M = \sum_{\alpha \in Z} k_\alpha M. \end{aligned}$$

Here, the first equality is due to (51), the strict inequality is due to (73), and the second equality is due to (74). The third equality is obtained by the change of the order of the summation. The fourth equality is true because all vertices in T are critical. Finally, the fifth equality is due to (72).

Therefore, this case yields a contradiction.

Case 2: There is a vertex α_0 in Z which satisfies

$$\exists j_0 \in U_2, (\alpha_0, j_0) \in E, j_0 \text{ is not critical,} \\ \text{and } f'((\alpha_0, j_0)) = 1.$$

However, by the definition of Z , there is an integer ℓ and a set of edges

$$\{(\alpha_h, j_{h+1})\}_{h=0,1,\dots,\ell} \subseteq E \\ \text{and } \{(\alpha_h, j_h)\}_{h=1,2,\dots,\ell} \subseteq E$$

such that $j_{\ell+1} = i_0$

$$\alpha_h \in Z, \quad \text{for } h = 0, 1, \dots, \ell \\ j_h \in U_2, \quad \text{for } h = 1, 2, \dots, \ell + 1,$$

and

$$f'((\alpha_h, j_{h+1})) = 0, \quad \text{for } h = 0, 1, \dots, \ell \\ f'((\alpha_h, j_h)) = 1, \quad \text{for } h = 1, 2, \dots, \ell.$$

We define the flow f'' as

$$f''(e) = \begin{cases} 1, & \text{if } e \in \{(\alpha_h, j_{h+1})\}_{h=0,1,\dots,\ell} \cup \{(j_{\ell+1}, t)\} \\ 0, & \text{if } e \in \{(\alpha_h, j_h)\}_{h=0,1,\dots,\ell} \cup \{(j_0, t)\} \\ f'(e), & \text{for all other edges } e. \end{cases}$$

This f'' is a legal flow in $(G(V, E), c)$. Moreover, it has the same value as f' , and the number of critical vertices $i \in U_2$ having $f''((i, t)) = 1$ is strictly larger than for f' .

We conclude that there exists an integer flow f^* in $(G(V, E), c)$ of value $\sum_{\alpha \in \mathcal{R}^-} k_\alpha$, such that for every critical vertex $i \in U_2$, $f^*((i, t)) = 1$. We define

$$a_i = \begin{cases} \alpha, & \text{if } f^*((\alpha, i)) = 1 \text{ for some } \alpha \in U_1 \\ 0, & \text{otherwise} \end{cases}$$

and $\mathbf{a} = (a_i)_{i \in T}$. For this selection of \mathbf{a} , we have $\mathbf{a} \in \mathcal{C}_T^{(k)}$ and the properties (i) and (ii) are satisfied. \square

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers, as well as the Associate Editor I. Sason, for their comments which improved the presentation of the paper. They would also like to thank I. Duursma, J. Feldman, R. Koetter, and O. Milenkovic for helpful discussions.

REFERENCES

- [1] R. G. Gallager, "Low-density parity-check codes," *IRE Trans. Inf. Theory*, vol. IT-8, no. 1, pp. 21–28, Jan. 1962.

- [2] N. Wiberg, "Codes and Decoding on General Graphs," Ph.D. dissertation, Linköping University, Linköping, Sweden, 1996.
- [3] G. D. Forney, 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*, ser. IMA Volumes in Mathematics and its Applications. New York: Springer, 2001, vol. 123, ch. 5, pp. 101–112.
- [4] R. Koetter, W.-C. W. Li, P. O. Vontobel, and J. L. Walker, "Characterizations of Pseudo-Codewords of LDPC Codes," Arxiv rep. arXiv:cs.IT/0508049.
- [5] P. Vontobel and R. Koetter, "Graph-cover decoding and finite-length analysis of message-passing iterative decoding of LDPC codes," 2005, Arxiv rep. arXiv:cs.IT/0512078.
- [6] J. Feldman, "Decoding Error-Correcting Codes via Linear Programming," Ph.D. dissertation, MIT, Cambridge, MA, 2003.
- [7] 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.
- [8] G. Caire, G. Taricco, and E. Biglieri, "Bit-interleaved coded modulation," *IEEE Trans. Inf. Theory*, vol. 44, no. 3, pp. 927–946, May 1998.
- [9] X. Li and J. A. Ritcey, "Bit-interleaved coded modulation with iterative decoding," *IEEE Commun. Lett.*, vol. 1, no. 6, pp. 169–171, Nov. 1997, "," in *Proc. IEEE Int. Conf. Communications (ICC)*, Sep. 1999, vol. 2, pp. 858–863.
- [10] D. Sridhara and T. E. Fuja, "LDPC codes over rings for PSK modulation," *IEEE Trans. Inf. Theory*, vol. 51, no. 9, pp. 3209–3220, Sep. 2005.
- [11] M. C. Davey and D. J. C. MacKay, "Low density parity check codes over $GF(q)$," *IEEE Commun. Lett.*, vol. 2, no. 6, pp. 165–167, Jun. 1998.
- [12] X. Li, M. R. Soleymani, J. Lodge, and P. S. Guinand, "Good LDPC codes over $GF(q)$ for bandwidth efficient transmission," in *Proc. 4th IEEE Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, Rome, Italy, Jun. 2003, pp. 95–99.
- [13] A. Bennatan and D. Burshtein, "On the application of LDPC codes to arbitrary discrete-memoryless channels," *IEEE Trans. Inf. Theory*, vol. 50, no. 3, pp. 417–438, Mar. 2004.
- [14] A. Bennatan and D. Burshtein, "Design and analysis of nonbinary LDPC codes for arbitrary discrete-memoryless channels," *IEEE Trans. Inf. Theory*, vol. 52, no. 2, pp. 549–583, Feb. 2006.
- [15] A. Bennatan, "The Application of LDPC Codes to New Problems in Communications," Ph.D. dissertation, Tel-Aviv University, Tel-Aviv, Israel, 2007.
- [16] C. A. Kelley, D. Sridhara, and J. Rosenthal, "Pseudocodeword weights for non-binary LDPC codes," in *Proc. IEEE Int. Symp. Information Theory (ISIT)*, Seattle, WA, Jul. 2006, pp. 1379–1383.
- [17] G. D. Forney, Jr., "Geometrically uniform codes," *IEEE Trans. Inf. Theory*, vol. 37, no. 5, pp. 1241–1260, Sep. 1991.
- [18] T. Richardson and R. E. Urbanke, "On the capacity of LDPC codes under message-passing decoding," *IEEE Trans. Inf. Theory*, vol. 51, no. 9, pp. 3209–3220, Sep. 2005.
- [19] M. F. Flanagan, "Codeword-independent performance of nonbinary linear codes under linear-programming and sum-product decoding," in *Proc. IEEE Int. Symp. Information Theory (ISIT)*, Toronto, ON, Canada, Jul. 2008, pp. 1503–1507.
- [20] E. Hof, I. Sason, and S. Shamai (Shitz), "Performance bounds for non-binary linear block codes over memoryless symmetric channels," *IEEE Trans. Inf. Theory*, vol. 55, no. 3, pp. 977–996, Mar. 2009.
- [21] M. Chertkov and M. Stepanov, "Pseudo-codeword landscape," in *Proc. IEEE Int. Symp. Information Theory (ISIT)*, Nice, France, Jun. 2007, pp. 1546–1550.
- [22] K. Yang, X. Wang, and J. Feldman, "Cascaded formulation of the fundamental polytope of general linear block codes," in *Proc. IEEE Int. Symp. Information Theory (ISIT)*, Nice, France, Jun. 2007, pp. 1361–1365.
- [23] K. Yang, X. Wang, and J. Feldman, "A new linear programming approach to decoding linear block codes," *IEEE Trans. Inf. Theory*, vol. 54, no. 3, pp. 1061–1072, Mar. 2008.
- [24] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.
- [25] A. Schrijver, *Theory of Linear and Integer Programming*. New York: Wiley, 1998.
- [26] A. Barvinok, *A Course in Convexity, Vol. 54 of Graduate Studies in Mathematics*. Providence, RI: Amer. Math. Soc., 2002.
- [27] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error Correcting Codes*. Amsterdam, The Netherlands: North-Holland, 1977.
- [28] V. Skachek and M. F. Flanagan, "Lower bounds on the minimum pseudodistance for linear codes with q -ary PSK modulation over AWGN," in *Proc. 5th Int. Symp. Turbo Codes and Related Topics*, Lausanne, Switzerland, Sep. 2008, pp. 426–431.
- [29] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, 2nd ed. Cambridge, MA: MIT Press, 2001.

Mark F. Flanagan (S'04–M'05) received the B.E. (1998) and Ph. D. (2005) degrees in electrical engineering from University College Dublin, Ireland.

He worked as a Project Engineer with Parthus Technologies Ltd. in the period 1998–1999. Since 2005, he has held postdoctoral research fellowships at the University of Zürich, the University of Bologna, and the University of Edinburgh. He also served as Principal Investigator on the project "Coding for Broadband wireless Applications" (COBRA) with the Digital Signal Processing research group at University College Dublin in the period 2005–2006. In 2008, he was appointed as SFI Stokes Lecturer in Electronic Engineering at University College Dublin. He is also a member of the Claude Shannon Institute for Discrete Mathematics, Coding and Cryptography, Dublin. His research interests are in the fields of information theory, wireless communications, and signal processing.

Dr. Flanagan was the recipient of a scholarship from the Institute of Advanced Studies, University of Bologna, Italy, in 2007. He is a reviewer for many leading international journals.

Vitaly Skachek (M'08) was born in Kharkov, Ukraine (former U.S.S.R.), in 1973. He received the B.A. (*cum laude*), M.Sc., and Ph.D. degrees in computer science from the Technion–Israel Institute of Technology, Haifa, in 1994, 1998, and 2007, respectively.

During 1996–2002, he held various engineering positions. During 2002–2006, he was working toward the Ph.D. degree at the Computer Science Department at the Technion. During the summer of 2004, he visited the Mathematics of Communications Department at Bell Laboratories under the DIMACS Special Focus Program in Computational Information Theory and Coding. Since 2007, he has been a Postdoctoral Fellow with the Claude Shannon Institute and the School of Mathematical Sciences, University College Dublin.

Dr. Skachek is a recipient of the Permanent Excellent Faculty Instructor award, given by Technion.

Eimear Byrne studied Mathematics in Trinity College, Dublin (Ireland), University College Dublin (Ireland), and completed the Ph.D. degree in mathematics at University College Cork (Ireland) in 2000.

She worked as a Postdoctoral Researcher at University College Cork, the National University of Ireland, Maynooth (Ireland), and the University of Notre Dame (Notre, Dame, IN). She joined the Department of Mathematics (now the School of Mathematical Sciences) at University College Dublin in 2003, where she currently holds a faculty position. She has been a member of the Claude Shannon Institute, Ireland, since 2006. Her research interests include algebraic coding theory, codes over rings, codes and graphs, almost perfect nonlinear functions, and network coding for error correction.

Dr. Byrne is on the editorial board of the journal *Advances in Mathematics of Communications*.

Marcus Greferath received the Ph.D. degree from the University of Mainz (Germany) in 1993, and the Habilitation degree from the University of Duisburg (Germany) in 2000.

He accepted an appointment at San Diego State University, San Diego, CA, in 2001 and at University College Dublin (UCD) in 2005. Currently he is with the School of Mathematical Sciences and the Claude Shannon Institute for Discrete Mathematics, Coding and Cryptography, UCD, Dublin, Ireland, where he leads the LDPC coding work group. He has published vastly in subjects including discrete mathematics (coding theory, cryptography), abstract and modern algebra (non-commutative rings and modules), geometry and ordered structures (designs, module, and lattice geometry), chemical physics (nuclear-magnetic resonance and imaging).

Dr. Greferath serves as an Editor-in-Chief for the journal *Advances in Mathematics of Communications*.