Relaxation Bounds on the Minimum Pseudo-Weight of Linear Block Codes

Panu Chaichanavong
Marvell Semiconductor, Inc.
Sunnyvale, CA 94089, USA
Email: panuc@marvell.com

Paul H. Siegel
Center for Magnetic Recording Research
University of California, San Diego
La Jolla, CA 92093, USA
Email: psiegel@ucsd.edu

***Abstract*— Just as the Hamming weight spectrum of a linear block code sheds light on the performance of a maximum likelihood decoder, the pseudo-weight spectrum provides insight into the performance of a linear programming decoder. Using properties of polyhedral cones, we find the pseudo-weight spectrum of some short codes. We also present two general lower bounds on the minimum pseudo-weight. The first bound is based on the column weight of the parity-check matrix. The second bound is computed by solving an optimization problem. In some cases, this bound is more tractable to compute than previously known bounds and thus can be applied to longer codes.**

I. Introduction

Inspired by the success of iterative message-passing decoding, there have been numerous efforts to understand its behavior [11], [4], [5]. Recently, Koetter and Vontobel [6] presented an analysis of iterative decoding based on graph-covering. This analysis explains why the notion of pseudo-codeword arises so naturally in iterative decoding. They also showed that the set of pseudo-codewords can be described as a polytope, which they called the fundamental polytope.

At the same time, Feldman [3] introduced a decoding algorithm based on linear programming (LP). This decoder was successfully applied to low-density parity-check (LDPC) codes and many turbo-like codes. It turns out that this decoding method is closely related to the analysis by Koetter and Vontobel. In particular, the feasible region of Feldman's linear program agrees with the fundamental polytope.

For a given channel, a pseudo-weight can be defined for each pseudo-codeword. The pseudo-weight spectrum relates to the performance of an LP decoder in very much the same way as the Hamming weight spectrum does to the performance of a maximum likelihood decoder. Thus it is of interest to find the pseudo-weight spectrum of a code. For very short codes, this might be achieved by employing a technique related to the dual polyhedral cone given in [2]. Some examples of pseudo-weight spectrum calculation for the additive white Gaussian noise channel will be demonstrated in Section IV. For longer codes, computing the entire pseudo-weight spectrum becomes intractable and we have to judge the performance of a code from bounds on the minimum pseudo-weight. Some techniques to compute lower bounds on minimum pseudo-weight were presented in [10]. In Section V, we will discuss two new lower bounds. One is based on the column weights of the parity-check matrix. The other is computed by solving an optimization problem.

II. Linear Programming Decoding

Let C be a binary linear code of length n. Such a code is a linear subspace of $\mathbb{F}_2^n$. In this paper, we will also view C as a subset of $\mathbb{R}^n$. Suppose that a codeword y is transmitted through a binary-input memoryless channel and r is the output of the channel. The log-likelihood ratio γ is defined as

$$\gamma_i = \ln\left(\frac{\Pr(r_i \mid y_i = 0)}{\Pr(r_i \mid y_i = 1)}\right).$$

Any codeword $x \in C$ that minimizes the cost $\gamma^T x$ is a maximum-likelihood (ML) codeword [3]. Thus ML decoding is equivalent to solving the problem:

minimize	$\gamma^T x$
subject to	$x \in C$.

Letting H be a parity-check matrix of C, the feasible set of this problem can be relaxed to a polytope $P = \{x \in \mathbb{R}^n \mid Bx \leq b,\ 0 \leq x_i \leq 1\}$, where the matrix B and the vector b are determined from H as follows [3]. For a row h of H, let $U(h)$ be the support of h, i.e., the set of positions of 1 in h. Then $Bx \leq b$ consists of the following inequalities: for each row h of H and for each set $V \subseteq U(h)$ such that $|V|$ odd,

$$\sum_{i \in V} x_i - \sum_{i \in U(h) \setminus V} x_i \leq |V| - 1. \qquad (1)$$

Now, the problem is transformed to a linear program. This approach, introduced by Feldman [3], is called *linear programming (LP) decoding*. The polytope P is called the *fundamental polytope* by Koetter and Vontobel [6] (Fig. 1). It has the property that a 0-1 vector is in the polytope if and only if it is a codeword of C. Thus, if a 0-1 vector is a solution to the linear program, it must be an ML codeword. However, unlike in ML decoding, the output of the LP decoder may not be a 0-1 vector, in which case the decoder simply declares an error.

III. Error Analysis

The fundamental polytope has a symmetry property that allows us to assume without loss of generality that the all-zeros codeword is transmitted, provided that the channel is a binary-input output-symmetric channel [3]. Roughly speaking, the fundamental polytope "looks" the same from every codeword. Therefore we assume that the all-zeros codeword is transmitted and remove from the linear program all inequality constraints that are not active at the origin. (An inequality $f(x) \leq \alpha$ is active at a point x^* if $f(x^*) = \alpha$.) We obtain a new linear program, which we will call LPCONE:

minimize	$\gamma^T x$
subject to	$x \in K = \{x \in \mathbb{R}^n \mid Ax \leq 0,\ x_i \geq 0\}$,

where A is the submatrix of B corresponding to zero-entries of b. The feasible set K is a polyhedral cone and it is called the *fundamental cone* [6] (Fig. 1).

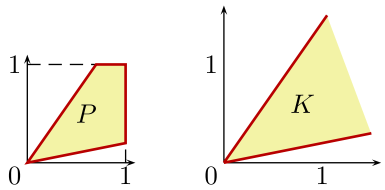

Fig. 1. Conceptual pictures of the fundamental polytope and the fundamental cone

Assuming that the zero codeword is transmitted, the probability of error of LPCONE is the same as that of the linear program in the previous section. To see this, suppose that the zero vector is a solution to the original linear program. Then $\gamma^T x \geq 0$ for all $x \in P$. It can be shown that a vector x is in K if and only if $\alpha x \in P$ for some $\alpha > 0$. It follows that the zero vector is also a solution to LPCONE. The converse is immediate since $P \subset K$. Hence, it is sufficient to consider LPCONE to evaluate the performance of the LP decoder. For this reason, we will mainly consider LPCONE instead of the original linear program.

To compute the probability of error, we need to find the set K^* such that the zero vector is a solution to the linear program if and only if $\gamma \in K^*$. To describe the set K^*, we proceed as follows. Let $W = \{w_1, \ldots, w_m\}$ be the set of "generators" of the cone K (Fig. 2), i.e., a vector x is in K if and only if x can be written as a nonnegative linear combination of the generators: $x = \alpha_1 w_1 + \cdots + \alpha_m w_m$, where $\alpha_1, \ldots, \alpha_m$ are some nonnegative real numbers. The zero vector is a solution to the linear program if and only if $\gamma^T x \geq 0$ for all $x \in K$. It can be shown that this condition is equivalent to $\gamma^T x \geq 0$ for all $x \in W$. Hence, the decoding is successful if and only if the log-likelihood ratio γ is in

$$K^* = \{z \in \mathbb{R}^n \mid z^T x \geq 0 \text{ for all } x \in W\} \tag{2}$$

(Fig. 2). The set K^* is called the *dual cone* of K [1].

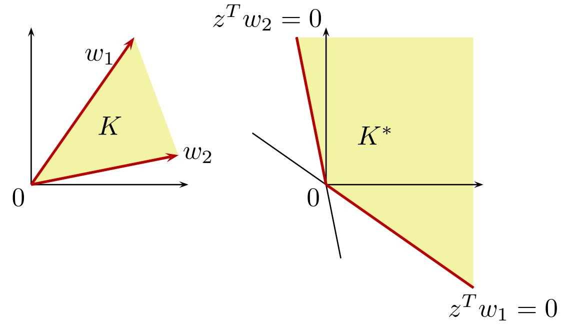

Fig. 2. The generators of the cone K and the dual cone K^*

IV. Pseudo-Weight on the AWGN Channel

For ML decoding over a memoryless channel, the probability of error of a code is largely determined by its Hamming weight spectrum. For iterative and LP decoding, it has been observed that the notion of "pseudo-weight" is more appropriate for determining the probability of error [11], [4], [6]. The definition of pseudo-weight varies with the channel; we will study only the pseudo-weight on the additive white Gaussian noise (AWGN) channel.

Before stating the definition of pseudo-weight, we will first extend the discussion in the previous section for the AWGN channel. We hope that by doing so the intuition behind the definition of pseudo-weight will be more apparent.

Consider the discrete-time AWGN channel in Fig. 3. Each bit of the codeword y is modulated to $+1$ and -1 and then corrupted by additive white Gaussian noise with variance σ^2. The received vector is denoted by r. It can be shown that the log-likelihood ratio γ is given by $\gamma_i = (2/\sigma^2) r_i$. We recall from the previous section that LP decoding is successful if and only if $\gamma \in K^*$, the dual of the fundamental cone. Since scaling the cost function by a positive scalar does not change the solution to a linear program, the decoding is successful if and only if $r \in K^*$.

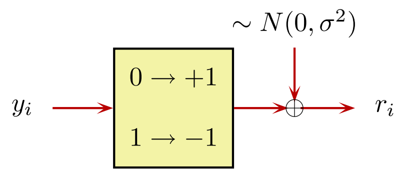

Fig. 3. Binary-input AWGN channel

We recall that $W = \{w_1, \ldots, w_m\}$ is the set of generators for the fundamental cone. The dual cone K^* can be described by the hyperplanes $z^T w_i = 0$ as in (2). The transmitted vector corresponding to the all-zeros codeword is the all-ones vector, which will be denoted by $\mathbf{1}$. The Euclidean distance from the all-ones vector to the hyperplane i is $\mathbf{1}^T w_i / \|w_i\|$, where $\|\cdot\|$ denotes the Euclidean norm (Fig. 4). If the noise perturbs the transmitted vector by this distance in the direction perpendicular to the hyperplane, the LP decoder will fail.

By way of comparison, consider ML decoding. In ML decoding, the Voronoi region for the all-zeros codeword is defined by the hyperplanes separating the all-ones vector and the transmitted vectors of the other codewords. Let y be a codeword with Hamming weight d. The distance from the

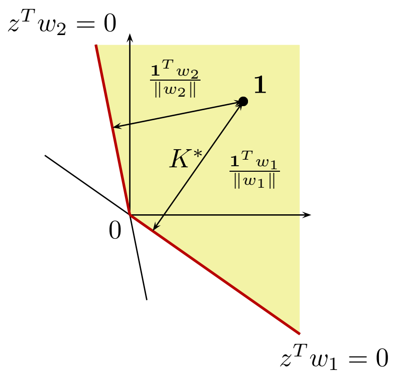

Fig. 4. Distances from the all-ones vector $\mathbf{1}$ to the hyperplanes defining the dual cone K^*

all-ones vector $\mathbf{1}$ to the hyperplane separating $\mathbf{1}$ and the transmitted vector corresponding to y is $\sqrt{d}$. This relationship between the Euclidean distance to the hyperplane and the Hamming weight of y motivates the definition of *pseudo-weight* of a vector x for the AWGN channel [11], [4], [6]:

$$p(x) = \left(\frac{\mathbf{1}^T x}{\|x\|} \right)^2 .$$

If x is a 0-1 vector, its pseudo-weight is equal to its Hamming weight.

Given a generator of K, computing its pseudo-weight is trivial. However, given a cone, finding its generators is a very complex task. A straightforward way is to add an equality constraint to LPCONE, such as $\mathbf{1}^T x = 1$, so that the feasible set becomes bounded. Then change the problem into the standard form and find all "basic feasible solutions," which correspond to the corner points of the feasible region. This involves $\binom{N}{M}$ ways of choosing "basic variables," where N and M are the number of variables and the number of equality constraints of the modified linear program. (For more details, refer to any linear optimization book, e.g., [7].)

A simpler way to find the set of generators of a cone is presented in [2]. However, the complexity is still very high and thus the algorithm can only be applied to very short codes. Using this algorithm, we computed the histograms of the pseudo-weights of the $(7, 4)$ and $(15, 11)$ Hamming codes, shown in Fig. 5.

V. BOUNDS ON MINIMUM PSEUDO-WEIGHT

Tanner [8] gave several lower bounds on the minimum Hamming weight of linear block codes. One of them, called the *bit-oriented bound*, is a function of the column and row weights of the parity-check matrix H and the eigenvalues of $H^T H$. Another one, called the *optimization distance bound*, is computed by solving an optimization problem. Two lower bounds on minimum pseudo-weight were presented in [10]. One of them is similar to the bit-oriented bound of Tanner [8]. The other, called the *LP-based bound*, is computed by solving a linear program derived from the fundamental cone.

In this section, we will present two lower bounds on minimum pseudo-weight. Before doing so, we prove two propositions which are useful in establishing the bounds.

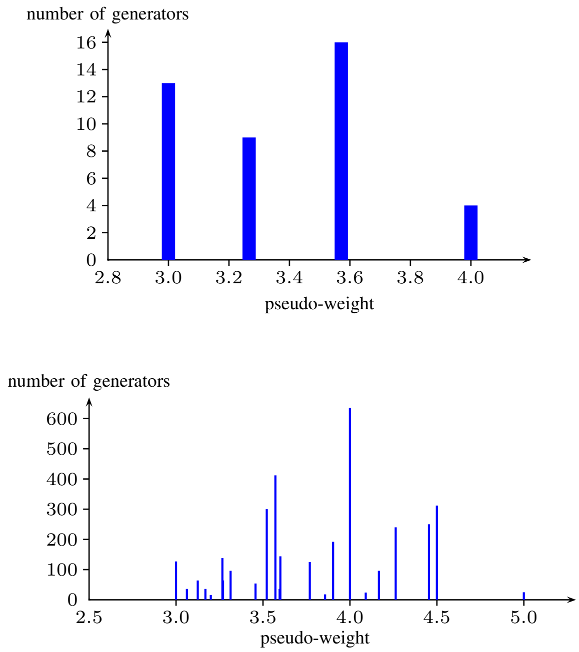

Fig. 5. Pseudo-weight spectra of the $(7, 4)$ and the $(15, 11)$ Hamming codes

Proposition 1: Let K be a polyhedral cone and W its set of generators. Then

$$\min_{x \in W} p(x) = \min_{x \in K} p(x) = \min_{x \in K,\ \mathbf{1}^T x = 1} p(x).$$

The proof of Proposition 1 is given in the appendix. Since $p(x) = ((\mathbf{1}^T x)/\|x\|)^2$, it follows from Proposition 1 that the problem of minimizing the pseudo-weight over W becomes the following non-convex problem, which we call MAXNORM:

$$\begin{array}{ll} \text{maximize} & \|x\|^2 \\ \text{subject to} & x \in K, \\ & \mathbf{1}^T x = 1. \end{array}$$

Our lower bounds are obtained by relaxing this difficult problem to an easier one, particularly the one in Proposition 2 below. Since the feasible set of MAXNORM is contained in the feasible set of the relaxed problem, $\|x^*\|^2 \leq \|x'\|^2$, where x^* and x' are the solutions to MAXNORM and the relaxed problem, respectively. Therefore the minimum pseudo-weight, which equals $1/\|x^*\|^2$, is lower bounded by $1/\|x'\|^2$.

Proposition 2: Let α_i, $1 \leq i \leq n$, be nonnegative real numbers. Consider the optimization problem:

$$\begin{array}{ll} \text{maximize} & \|x\|^2 \\ \text{subject to} & x \in \mathbb{R}^n \\ & 0 \leq x_i \leq \alpha_i \text{ for all } 1 \leq i \leq n, \\ & \mathbf{1}^T x = 1. \end{array}$$

Suppose that $\sum_{i=1}^n \alpha_i \geq 1$ and α_i are ordered such that $\alpha_1 \geq \alpha_2 \geq \cdots \geq \alpha_n$. Let j be the first index such that $\alpha_1 + \cdots + \alpha_j \geq 1$. Then the maximum of the objective function is $\alpha_1^2 + \cdots + \alpha_{j-1}^2 + (1 - \alpha_1 - \cdots - \alpha_{j-1})^2$.

Proof: Let x satisfy $\mathbf{1}^T x = 1$ and $0 \leq x_i \leq \alpha_i$ for all $1 \leq i \leq n$. Let k be the smallest index such that $x_k < \alpha_k$. Let m be the largest index such that $x_m > 0$. We define a new vector x' as follows.

Case 1: $\alpha_k - x_k \leq x_m$. Let $x'_k = \alpha_k$ and $x'_m = x_m - \alpha_k + x_k$.

Case 2: $\alpha_k - x_k > x_m$. Let $x'_k = x_k + x_m$ and $x'_m = 0$.

For the other indices $t \neq k, t \neq m$, let $x'_t = x_t$. It can be shown that $\mathbf{1}^T x' = 1$ and $0 \leq x'_i \leq \alpha_i$. Moreover, we claim that $\|x'\| \geq \|x\|$. We repeat this assignment until $x' = x$. (The algorithm terminates since either k is increased or m is decreased in the next iteration.) The final vector x^* satisfies $x^*_v = \alpha_v$ for $1 \leq v \leq j-1$, $x^*_j = 1 - \alpha_1 - \cdots - \alpha_{j-1}$, and $x^*_v = 0$ for $j+1 \leq v \leq n$, and the proposition follows.

To prove the claim, consider the two cases.

Case 1: $\alpha_k - x_k \leq x_m$.

$$\begin{aligned}\|x'\|^2 - \|x\|^2 &= (x'_k)^2 + (x'_m)^2 - x_k^2 - x_m^2 \\ &= \alpha_k^2 + (x_m - \alpha_k + x_k)^2 - x_k^2 - x_m^2 \\ &= 2(\alpha_k - x_m)(\alpha_k - x_k) \geq 0.\end{aligned}$$

Case 2: $\alpha_k - x_k > x_m$.

$$\|x'\|^2 - \|x\|^2 = (x_k + x_m)^2 - x_k^2 - x_m^2 \geq 0.$$

■

A. Bound from Column Weight

If the Tanner graph of a parity-check matrix H has no cycle of length four, it is well known that the minimum Hamming distance is lower bounded by the minimum column weight of H plus one. This is true for the minimum pseudo-weight as well.

Theorem 3: Suppose that any two columns of the parity-check matrix H have at most one 1 in the same position. Then the minimum pseudo-weight is lower bounded by the minimum column weight plus one.

Proof: Let m^* be the minimum column weight of H. The basic idea of the proof is to relax MAXNORM to the problem in Proposition 2 where $\alpha_i = 1/(m^*+1)$. Then the theorem will follow.

Consider the i-th column of H, which is denoted by c_i. Let m be its column weight. Let $q_1, \ldots, q_m$ be the positions of 1 in c_i. Since any two columns of H have at most one 1 in the same position, the support of the q_j-th row can be written as $R_j \cup \{i\}$, where $R_j \cap R_k = \emptyset$ for $1 \leq k \leq m$, $k \neq j$.

Let $x \in K$ and $\mathbf{1}^T x = 1$. We recall that K is defined by the inequalities (1) that are active at the origin. (These are the inequalities with $|V| = 1$.) Therefore x satisfies $x_i - \sum_{k \in R_j} x_k \leq 0$ for all $1 \leq j \leq m$. Since the sets R_j are pairwise disjoint,

$$1 = \sum_{l=1}^{n} x_l \geq x_i + \sum_{j=1}^{m} \sum_{k \in R_j} x_k \geq x_i + \sum_{j=1}^{m} x_i = (m+1)x_i.$$

Hence $x_i \leq 1/(m^*+1)$ for all $1 \leq i \leq n$. Then the theorem follows from Proposition 2. ■

B. Relaxation Bounds

Suppose that we choose a set $S \supseteq K$ and relax MAXNORM to the following:

maximize	$\|x\|^2$
subject to	$x \in S$,
	$\mathbf{1}^T x = 1$.

The set S should be as small as possible; however, the new problem should be easy to solve. A good choice for S is the hyper-rectangle

$$S = \{x \in \mathbb{R}^n \mid 0 \leq x_i \leq \alpha_i\},$$

where α_i is the maximum of the objective function of the following linear program:

maximize	x_i
subject to	$x \in K$,
	$\mathbf{1}^T x = 1$.

Without loss of generality, assume that $\alpha_1 \geq \alpha_2 \geq \cdots \geq \alpha_n$. Let j be the first index such that $\alpha_1 + \cdots + \alpha_j \geq 1$. From Proposition 2,

$$\text{min pseudo-weight} \geq \frac{1}{\sum_{i=1}^{j-1} \alpha_i^2 + (1 - \sum_{i=1}^{j-1} \alpha_i)^2}$$

We call this the *first-order bound*.

A more complex choice for S is

$$S = \{x \in \mathbb{R}^n \mid 0 \leq x_i \leq \alpha_i,\ x_i + x_j \leq \beta_{i,j}\},$$

where α_i are computed in the same way as above, and $\beta_{i,j}$ are computed by replacing the objective function of the above linear program by $x_i + x_j$. Unfortunately, the problem is hard to solve for this set S. We approach the problem by partitioning the feasible region into n sub-regions based on the maximum entry of x. We obtain n sub-problems; the sub-problem k, $1 \leq k \leq n$, is the following:

maximize	$\|x\|^2$	
subject to	$0 \leq x_i \leq \alpha_i$	for all $1 \leq i \leq n$,
	$x_i + x_j \leq \beta_{i,j}$	for all $1 \leq i \leq n$, $1 \leq j \leq n$,
	$\mathbf{1}^T x = 1$,	
	$x_k \geq x_i$	for all $1 \leq i \leq n$.

Then we relax each sub-problem by omitting the inequalities $x_i + x_j \leq \beta_{i,j}$ whenever $i \neq k$ and $j \neq k$.

maximize	$\|x\|^2$	
subject to	$0 \leq x_i \leq \alpha_i$	for all $1 \leq i \leq n$,
	$x_k + x_i \leq \beta_{k,i}$	for all $1 \leq i \leq n$,
	$\mathbf{1}^T x = 1$,	
	$x_k \geq x_i$	for all $1 \leq i \leq n$.

If we fix x_k, then the relaxed sub-problem has the same form as when S is a rectangle, which we can solve by Proposition 2. Thus we can compute the maximum of the relaxed sub-problem for a fixed x_k. By varying x_k and computing the corresponding maximum, we can solve the relaxed sub-problem k. Taking the maximum of $\|x\|^2$ over all relaxed sub-problems, we obtain an upper bound of $\|x\|^2$ for the original problem, which will lead to a lower bound for the minimum pseudo-weight. We call this the *second-order bound*.

We have a few remarks regarding these lower bounds:

- Any lower bound for minimum pseudo-weight is also a lower bound for minimum Hamming weight.
- To compute the first-order and the second-order bounds, we need to solve as many as n and $n(n+1)/2$ linear programs respectively. Each of these linear programs has n variables. This is in contrast to the optimization distance bound and the LP-based bound discussed earlier, which are computed by solving one linear program whose worst-case number of variables is quadratic in the codeword length.
- The number of linear programs to be solved for the first-order and the second-order bounds can be reduced if the code has some structure.

Next we compute these bounds for a class of group-structured LDPC codes presented in [9]. A particular sequence of codes in this class has constant column weight 3 and constant row weight 5, with rate approximately $2/5$. The bounds on minimum pseudo-weights for these codes are shown in Table I. The upper bound is computed by using the MATLAB optimization toolbox to find a good feasible solution to MAXNORM. Note that for the code with length 155, Vontobel and Koetter [10] have found tighter lower and upper bounds: 10.8 and 16.4 respectively.

TABLE I

LOWER AND UPPER BOUNDS ON MINIMUM PSEUDO-WEIGHTS OF TANNER'S LDPC CODES WITH RATE APPROXIMATELY $2/5$

length	first-order	second-order	upper bound
155	8.3	9.7	17.0
305	11.5	13.8	20.1
755	13.0	14.0	27.6
905	17.6	21.5	39.4

APPENDIX
PROOF OF PROPOSITION 1

In this appendix we present a proof of Proposition 1. First, we need the following lemma.

Lemma 4: Let a, b, c, d be positive real numbers. Then

$$\frac{a+c}{b+d} \geq \min\left\{\frac{a}{b}, \frac{c}{d}\right\}.$$

Proof: Suppose that $a/b \leq c/d$. Then

$$\begin{aligned} \frac{c}{a} &\geq \frac{d}{b} \\ \frac{a+c}{a} &\geq \frac{b+d}{b} \\ \frac{a+c}{b+d} &\geq \frac{a}{b}. \end{aligned}$$

The case $c/d \leq a/b$ is similar. ■

Proof of Proposition 1: First, we will show that $\min_{x \in W} p(x) = \min_{x \in K} p(x)$. Since $W \subset K$, $\min_{x \in W} p(x) \geq \min_{x \in K} p(x)$. Conversely, let $x, y, z \in K$ and $\alpha, \beta > 0$ with $x = \alpha y + \beta z$. Then $\mathbf{1}^T x = \alpha \mathbf{1}^T y + \beta \mathbf{1}^T z$. By the triangle inequality, $\|x\| \leq \alpha\|y\| + \beta\|z\|$. It follows that

$$\begin{aligned} p(x) &= \left(\frac{\mathbf{1}^T x}{\|x\|}\right)^2 \geq \left(\frac{\alpha \mathbf{1}^T y + \beta \mathbf{1}^T z}{\alpha\|y\| + \beta\|z\|}\right)^2 \\ &\geq \min\left\{\left(\frac{\mathbf{1}^T y}{\|y\|}\right)^2, \left(\frac{\mathbf{1}^T z}{\|z\|}\right)^2\right\} = \min\{p(y), p(z)\}, \end{aligned}$$

where the latter inequality follows from Lemma 4. Since every $x \in K$ can be written as a nonnegative linear combination of the generators, there is a generator $w \in W$ such that $p(x) \geq p(w)$. Therefore $\min_{x \in W} p(x) \leq \min_{x \in K} p(x)$. Finally, $\min_{x \in K} p(x) = \min_{x \in K,\, \mathbf{1}^T x = 1} p(x)$ since the pseudo-weight is invariant under scaling, i.e., $p(ax) = p(x)$. ■

ACKNOWLEDGEMENT

The authors would like to thank the anonymous reviewers for many helpful comments and suggestions.

This work is supported in part by the Center for Magnetic Recording Research, the Information Storage Industry Consortium, and the NSF under Grant CCR-0219852.

REFERENCES

[1] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge University Press, 2004.

[2] E. Castillo, A. Cobo, F. Jubete, and R. E. Pruneda, *Orthogonal Sets and Polar Methods in Linear Algebra*. New York: Wiley, 1999.

[3] J. Feldman, "Decoding error-correcting codes via linear programming," Ph.D. dissertation, Massachusetts Institute of Technology, 2003.

[4] G. D. Forney Jr., R. Koetter, F. R. Kschischang, and A. Reznik, "On the effective weights of pseudocodewords for codes defined on graphs with cycles," in *Codes, systems, and graphical models*. New York: Springer, 2001, pp. 101–112.

[5] B. J. Frey, R. Koetter, and A. Vardy, "Signal-space characterization of iterative decoding," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 766–781, Feb. 2001.

[6] R. Koetter and P. O. Vontobel, "Graph-covers and iterative decoding of finite-length codes," in *Proc. Int. Conf. on Turbo Codes and Related Topics*, Brest, France, 2003, pp. 75–82.

[7] S. G. Nash and A. Sofer, *Linear and nonlinear programming*. McGraw-Hill, 1996.

[8] R. M. Tanner, "Minimum-distance bounds by graph analysis," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 808–821, Feb. 2001.

[9] R. M. Tanner, D. Sridhara, and T. Fuja, "A class of group-structured LDPC codes," in *Proc. ICSTA*, Ambleside, England, 2001.

[10] P. O. Vontobel and R. Koetter, "Lower bounds on the minimum pseudo-weight of linear codes," in *Proc. IEEE Int. Symp. Inform. Theory*, Chicago, IL, 2004, p. 70.

[11] N. Wiberg, "Codes and decoding on general graphs," Ph.D. dissertation, Linköping University, Sweden, 1996.