On the Height Profile of Analog Error-Correcting Codes

Ron M. Roth Ziyuan Zhu Changcheng Yuan Paul H. Siegel Anxiao Jiang

***Abstract*—In recent work, it has been shown that maintaining reliability in analog vector–matrix multipliers can be modeled as the following coding problem. Vectors in $\mathbb{R}^k$ are encoded into codewords of a linear $[n, k, d]$ code $\mathcal{C}$ over $\mathbb{R}$. For prescribed positive reals $\delta < \Delta$, additive errors of magnitude $\leq \delta$ are tolerable and need no handling, yet outlying errors of magnitude $> \Delta$ are to be located or detected. The trade-off between the ratio Δ/δ and the number of outlying errors that can be handled is determined by the *height profile* of $\mathcal{C}$; as such, the height profile provides a finer description of the error handling capability of $\mathcal{C}$, compared to the minimum distance d, which only determines the number of correctable errors. This work contains a further study of the notion of the height profile. Several characterizations of the height profile are presented, thereby yielding methods for computing it. The starting point is formulating this computation as an optimization problem that is solved by a set of linear programs. This, in turn, leads to a combinatorial characterization of the height profile as a maximum (or max–min) over a certain finite set of codewords of $\mathcal{C}$. Moreover, this characterization is shown to have a simple geometric interpretation when the columns of the generator matrix of $\mathcal{C}$ all have the same L_2 norm. Through examples of several code families, it is demonstrated how the results herein can be used to compute the height profile explicitly.**

***Index Terms*—Fault-tolerant computing, least absolute deviations, linear codes over the real field, linear programming, Platonic solids, spherical codes, unit-norm tight frame, vector–matrix multiplication.**

I. Introduction

Vector–matrix multiplication (VMM) is a very common computational task found in many applications, and there are quite a few hardware implementations of analog VMM accelerators, particularly designs that are based on memristive crossbars [1],[6],[7],[17],[19]. Such devices, however, are susceptible to various types of errors. As shown in recent work [14],[15], maintaining reliability in such devices can be modeled as the following coding problem. Blocks in $\mathbb{R}^k$ are mapped one-to-one to codewords of a linear $[n, k, d]$ code $\mathcal{C}$ over $\mathbb{R}$. For prescribed positive reals $\delta < \Delta$, additive errors whose values are within the interval $[-\delta, \delta]$ are tolerable and need no handling; on the other hand, outlying errors, whose values are outside the interval $[-\Delta, \Delta]$, are to be located or detected. The correction capability of the code $\mathcal{C}$ is determined by the following parameters: (i) the number τ of outlying errors that can be located, (ii) the number $\tau + \sigma$ of outlying errors that can be detected, and (iii) the ratio Δ/δ (which should be as small as possible).

Specifically, in the coding model presented in [14], the error-free result of the VMM is a codeword $\boldsymbol{c} \in \mathcal{C}$, and the read vector $\boldsymbol{y} \in \mathbb{R}^n$ is expressed as a sum

$$\boldsymbol{y} = \boldsymbol{c} + \boldsymbol{\varepsilon} + \boldsymbol{e},$$

where $\boldsymbol{\varepsilon} \in [-\delta, \delta]^n$ is the vector of tolerable errors and $\boldsymbol{e} = (e_j)_j \in \mathbb{R}^n$ is the vector of outlying errors which, in turn, is assumed to have Hamming weight $\mathsf{w}(\boldsymbol{e}) \leq \tau + \sigma$. A (τ, σ)-*decoder for* $(\mathcal{C}, \Delta : \delta)$ is a function $\mathcal{D}$ which gets $\boldsymbol{y}$ as an input and returns a set of locations (or a decoding failure indicator "e") such that the following two conditions hold.

(D1) If $\mathsf{w}(\boldsymbol{e}) \leq \tau$ then $\mathcal{D}(\boldsymbol{y}) \neq$ "e" and no false alarms are allowed, i.e.,

$$e_j = 0 \quad \Longrightarrow \quad j \notin \mathcal{D}(\boldsymbol{y}).$$

(D2) If $\mathcal{D}(\boldsymbol{y}) \neq$ "e" then no misses are allowed, i.e.,

$$|e_j| > \Delta \quad \Longrightarrow \quad j \in \mathcal{D}(\boldsymbol{y}).$$

In the classical coding setting, the inequality $2\tau + \sigma < d$ is necessary and sufficient in order to correct any τ errors and detect any $\tau + \sigma$ errors. Herein, a finer parametrization of $\mathcal{C}$ is required that also determines the smallest attainable ratio Δ/δ; this is done by means of the *height profile* of the code, which is a list of positive (possibly infinite) real values $(\mathsf{h}_m(\mathcal{C}))_m$, whose precise definition will be recalled below. In terms of the height profile, we then get the following necessary and sufficient relationship between the parameters τ, σ, and Δ/δ for having a (τ, σ)-decoder for $(\mathcal{C}, \Delta : \delta)$:

$$\frac{\Delta}{\delta} \geq 2(\mathsf{h}_{2\tau+\sigma}(\mathcal{C}) + 1)$$

(see Theorem 1 in [14] and Theorem 2 in [15]).

The aim of this work is providing several characterizations of the height profile of a linear code $\mathcal{C}$. We mention that the notion of the height profile has been the subject of

This work was supported in part by NSF Grants Nos. CCF-2212437, CCF-2416361, and CCF-2416362, by Grant No. 2023786 from the United-States–Israel Binational Science Foundation (BSF), and by Grant No. 2369/24 from the Israel Science Foundation. Parts of this work were presented at the IEEE International Symposium on Information Theory, Guangzhou, China (July 2026).

Ron M. Roth is with the Computer Science Department, Technion, Haifa, Israel. The work of R.M. Roth was done in part while visiting the Center for Memory and Recording Research, UC San Diego (email: ronny@cs.technion.ac.il).

Ziyuan Zhu is with the Department of Electrical and Computer Engineering, University of California San Diego, La Jolla, CA, USA (email: ziz050@ucsd.edu).

Changcheng Yuan is with the Department of Computer Science and Engineering, Texas A&M University, College Station, TX, USA (email: ericycc@tamu.edu).

Paul H. Siegel is with the Department of Electrical and Computer Engineering, University of California San Diego, La Jolla, CA, USA (email: psiegel@ucsd.edu).

Anxiao Jiang is with the Department of Computer Science and Engineering, Texas A&M University, College Station, TX, USA (email: ajiang@tamu.edu).

several recent papers, including the construction of codes with favorable height properties and proving the optimality of the height in certain cases; see [9],[10],[11],[12],[18],[22].

We next introduce some notation and definitions. For integers $\ell \leq n$, we denote by $[\ell : n\rangle$ the integer subset $\{z \in \mathbb{Z} : \ell \leq z < n\}$. When $\ell = 0$ we use the shorthand notation $[n\rangle$, and we will typically use $[n\rangle$ to index the entries of vectors in $\mathbb{R}^n$. Similarly, the entries of an $r \times n$ matrix $H = (H_{i,j})$ will be indexed by $(i,j) \in [r\rangle \times [n\rangle$. The ith row of H will be denoted by $[H]_i$, and for a subset $\mathcal{I} \subseteq [n\rangle$, the notation $(H)_{\mathcal{I}}$ will stand for the $r \times |\mathcal{I}|$ submatrix of H that is formed by the columns that are indexed by $\mathcal{I}$ (this notation extends to vectors as well, seen as single-row matrices). We denote the support of a real vector $\boldsymbol{x}$ by $\mathsf{Supp}(\boldsymbol{x})$ and its L_p norm by $\|\boldsymbol{x}\|_p$, with $\|\boldsymbol{x}\|_\infty$ standing for the largest magnitude (i.e., absolute value) of any entry of $\boldsymbol{x}$. The unit (Euclidean) hypersphere in $\mathbb{R}^k$, namely, the set $\left\{\boldsymbol{x} \in \mathbb{R}^k : \|\boldsymbol{x}\|_2 = 1\right\}$, will be denoted by $\mathbb{S}^{k-1}$.

Given a nonzero vector $\boldsymbol{x} = (x_j)_{j\in[n\rangle} \in \mathbb{R}^n$, a *sorting permutation* of $\boldsymbol{x}$ is a permutation $\pi : [n\rangle \to [n\rangle$ that sorts[1] the entries of $\boldsymbol{x}$ according to descending magnitudes:

$$|x_{\pi(0)}| \geq |x_{\pi(1)}| \geq \ldots \geq |x_{\pi(n-1)}|.$$

Given an integer $m \in [n\rangle$, the *m-height* of $\boldsymbol{x}$, denoted $\mathsf{h}_m(\boldsymbol{x})$, is defined by

$$\mathsf{h}_m(\boldsymbol{x}) = \left|\frac{x_{\pi(0)}}{x_{\pi(m)}}\right|$$

(where $\mathsf{h}_m(\boldsymbol{x}) = \infty$ when $x_{\pi(m)} = 0$). We also define the set[2]

$$\mathcal{P}_m(\boldsymbol{x}) = \left\{j \in [n\rangle : |x_j| = |x_{\pi(m)}|\right\}.$$

The m-height of a linear $[n,k,d]$ code[3] $\mathcal{C}$ over $\mathbb{R}$ is defined by

$$\mathsf{h}_m(\mathcal{C}) = \max_{\boldsymbol{c}\in\mathcal{C}\setminus\{\boldsymbol{0}\}} \mathsf{h}_m(\boldsymbol{c}),$$

and a nonzero codeword of $\mathcal{C}$ is called *m-extremal* if its m-height equals $\mathsf{h}_m(\mathcal{C})$. The list $(\mathsf{h}_m(\mathcal{C}))_{m\in[n\rangle}$ is called the *height profile* of $\mathcal{C}$. The minimum distance d of $\mathcal{C}$ is related to its height profile by

$$d = \min\left\{m \in [n+1\rangle : \mathsf{h}_m(\mathcal{C}) = \infty\right\} \tag{1}$$

(where we formally define $\mathsf{h}_n(\mathcal{C}) = \infty$). The minimum-weight codewords of $\mathcal{C}$ are its codewords $\boldsymbol{c}$ whose Hamming weight, $\mathsf{w}(\boldsymbol{c})$, equals d.

For a subset $\mathcal{I} \subseteq [n\rangle$ we let $\overline{\mathcal{I}}$ be the complement set $[n\rangle \setminus \mathcal{I}$. The ambient set $[n\rangle$ will always be the index set of the coordinates of a linear code and will always be understood from the context.

Given a linear $[n,k]$ code $\mathcal{C}$ over $\mathbb{R}$ and a subset $\mathcal{I} \subseteq [n\rangle$, we let $(\mathcal{C})_{\mathcal{I}}$ denote the linear code that is obtained by puncturing $\mathcal{C}$ on $\overline{\mathcal{I}}$:

$$(\mathcal{C})_{\mathcal{I}} = \left\{(\boldsymbol{c})_{\mathcal{I}} : \boldsymbol{c} \in \mathcal{C}\right\}.$$

We also define $\mathcal{C}|_{\mathcal{I}}$ to be the set of all codewords $\boldsymbol{c} \in \mathcal{C}$ whose projections $(\boldsymbol{c})_{\mathcal{I}}$ are in the hypercube $[-1,1]^{|\mathcal{I}|}$, i.e.,

$$\mathcal{C}|_{\mathcal{I}} = \left\{\boldsymbol{c} = (c_j)_{j\in[n\rangle} \in \mathcal{C} : |c_j| \leq 1 \text{ for all } j \in \mathcal{I}\right\}. \tag{2}$$

As said earlier, our goal in this work is to find several characterizations of the height profile of a linear code, thereby presenting methods for calculating it. Now, the relationship (1) implies that computing the height profile of a generic linear code is at least as hard as computing its minimum distance; still, since the codes of interest are typically very structured, our characterizations provide tools for analyzing the height profile of such codes (as we demonstrate through several examples).

Our starting point will be casting the problem of computing the m-height of a linear code as a maximization over the outputs of a set of linear programs (Section II). This approach was recently used in [9] when searching for codes with favorable height profiles.[4] The algorithm that we present here is simpler; moreover, through the dual programs, we obtain a necessary condition on the structure of m-extremal codewords (this condition will turn out to be useful in subsequent sections).

In Section III, we present several characterizations of the m-height of a linear code as a maximum (or max–min) over a certain *finite* set of codewords. These characterizations, in turn, allow us to obtain exact expressions for the height profile of several (structured) families of codes.

In the context of searching for codes with favorable height profiles, almost all constructions so far in the literature are codes with parity-check matrices whose columns are all unit vectors. In Section IV, we focus on such codes and their duals. In particular, for the dual codes, we show that our condition on the structure of m-extremal codewords has a particularly simple geometric interpretation. This, in turn, allows us to completely characterize the m-extremal codewords in several cases of interest.

II. Height via linear programming

The following result is the first among several characterizations that we provide for the m-height of a linear code. It will serve as a basis for characterizing the m-height of a linear code as the largest among the solutions of a certain set of linear programs. The approach of computing the m-height of codes through linear programming was first introduced in [9] and the algorithm to be presented in Figure 1 below can be seen as a simplified version of the algorithm in [9]. Our new algorithm was used to obtain the improved table in [23].

Theorem 1. *Let $\mathcal{C}$ be a linear $[n,k,d]$ code over $\mathbb{R}$ and let $m \in [d\rangle$. Then*

$$\mathsf{h}_m(\mathcal{C}) = \max_{\mathcal{S}} \max_{\boldsymbol{x}\in\mathcal{C}|_{\overline{\mathcal{S}}}} \|\boldsymbol{x}\|_\infty, \tag{3}$$

where $\mathcal{S}$ ranges over all m-subsets of $[n\rangle$.

[1] A sorting permutation is not unique if $\boldsymbol{x}$ has multiple entries with the same magnitude, yet this will not pose any impediment in the sequel.

[2] Note that $\mathcal{P}_m(\boldsymbol{x})$ is uniquely defined, even when $\boldsymbol{x}$ has several sorting permutations.

[3] Hereafter, we always assume that the code is nontrivial, i.e., $k > 0$.

[4] Tables such as [9, Table III] and [23] can be used as a yardstick to determine how good the height profile is, as we currently do not know of negative results—i.e., general lower bounds—on the entries of the height profile (lower bounds in special cases have been recently obtained in [12] and [10]).

Proof. We first show that $\mathsf{h}_m(\mathcal{C})$ is bounded from above by the right-hand side of (3). Let $\boldsymbol{c} = (c_j)_{j\in[n\rangle}$ be an m-extremal codeword in $\mathcal{C}$ and let $\pi(\cdot)$ be a sorting permutation of $\boldsymbol{c}$; without loss of generality we can assume that $|c_{\pi(m)}| = 1$, in which case

$$\mathsf{h}_m(\boldsymbol{c}) = \|\boldsymbol{c}\|_\infty.$$

Since there are no more than m entries in $\boldsymbol{c}$ of magnitude greater than 1, there exists at least one m-subset $\mathcal{S} \subseteq [n\rangle$ such that $\boldsymbol{c} \in \mathcal{C}|_{\overline{\mathcal{S}}}$. Hence,

$$\begin{aligned}\mathsf{h}_m(\mathcal{C}) &= \mathsf{h}_m(\boldsymbol{c}) = \|\boldsymbol{c}\|_\infty \\ &\le \max_{\mathcal{S}} \max_{\boldsymbol{x}\in\mathcal{C}|_{\overline{\mathcal{S}}}} \|\boldsymbol{x}\|_\infty.\end{aligned}$$

Next, we show that $\mathsf{h}_m(\mathcal{C})$ is at least the right-hand side of (3). Let $\mathcal{S}$ and $\boldsymbol{c} \in \mathcal{C}|_{\overline{\mathcal{S}}}$ be where the maximum in (3) is attained (since $\boldsymbol{x} \mapsto \|\boldsymbol{x}\|_\infty$ is continuous and is maximized in (3) over a compact set, the maximum indeed exists). Also, let $\pi(\cdot)$ be a sorting permutation of $\boldsymbol{c}$. Since $\boldsymbol{c} \in \mathcal{C}|_{\overline{\mathcal{S}}}$, there are no more than m entries in $\boldsymbol{c}$ of magnitude greater than 1; in particular, $|c_{\pi(m)}| \le 1$. Hence,

$$\begin{aligned}\mathsf{h}_m(\mathcal{C}) &\ge \mathsf{h}_m(\boldsymbol{c}) = \left|\frac{c_{\pi(0)}}{c_{\pi(m)}}\right| \ge |c_{\pi(0)}| = \|\boldsymbol{c}\|_\infty \\ &= \max_{\mathcal{S}} \max_{\boldsymbol{x}\in\mathcal{C}|_{\overline{\mathcal{S}}}} \|\boldsymbol{x}\|_\infty.\end{aligned}$$

□

Remark 1. Theorem 1 holds also when $m \ge d$. By (1), the left-hand side of (3) equals ∞ in this case. As for the right-hand side, let $\boldsymbol{c}$ be any minimum-weight codeword of $\mathcal{C}$ and let $\mathcal{S}$ be any m-subset of $[n\rangle$ that is a superset of $\mathsf{Supp}(\boldsymbol{c})$. Since all the scalar multiples of $\boldsymbol{c}$ belong to $\mathcal{C}|_{\overline{\mathcal{S}}}$, the inner maximum in (3) for this $\mathcal{S}$ is unbounded. □

A. Primal linear program

Theorem 1 implies the algorithm in Figure 1 for computing the m-height of a linear $[n, k, d]$ code $\mathcal{C}$ which is specified by its $k \times n$ generator matrix $G = (\boldsymbol{g}_0\ \boldsymbol{g}_1\ \ldots\ \boldsymbol{g}_{n-1})$. The algorithm enumerates over all m-subsets $\mathcal{S} \subseteq [n\rangle$ and over all indexes $i \in \mathcal{S}$. For each pair $(\mathcal{S}, i)$, the following linear program computes a codeword $\boldsymbol{c} = (c_j)_{j\in[n\rangle} = \boldsymbol{u}G$ in $\mathcal{C}|_{\overline{\mathcal{S}}}$ for which the entry c_i is maximized:

$$\boxed{\begin{array}{l}\text{Maximize} \\ \qquad \boldsymbol{u}\cdot\boldsymbol{g}_i \\ \text{over } \boldsymbol{u} \in \mathbb{R}^k, \text{ subject to} \\ \qquad \left\{\begin{array}{rcl} \boldsymbol{u}\,(G)_{\overline{\mathcal{S}}} & \le & \mathbf{1} \\ \boldsymbol{u}\,(G)_{\overline{\mathcal{S}}} & \ge & -\mathbf{1}\end{array}\right.\end{array}} \tag{4}$$

(the notation $\mathbf{1}$ in (4) stands for the all-1 row vector in $\mathbb{R}^{n-m}$, and the inequalities therein are componentwise). The return value of the algorithm is the largest c_i found across all pairs $(\mathcal{S}, i)$.

The algorithm requires $\binom{n}{m} \cdot m$ applications of the linear program (4), each involving k variables that satisfy $2(n-m)$ linear inequality constraints. This is compared with the algorithm described in [9, §III.B], which requires $2^m \cdot \binom{n}{m} \cdot m \cdot (n-m)$ applications of a linear program in k variables that satisfy $2n-3$ linear constraints. As such, the algorithm herein can be seen as a simplified (and more efficient) version of the algorithm in [9].

Input:
- $k \times n$ generator matrix $G = (\boldsymbol{g}_0\ \boldsymbol{g}_1\ \ldots\ \boldsymbol{g}_{n-1})$ of $\mathcal{C}$
- $m \in [1 : d\rangle$

$h \leftarrow 0$
for every m-subset $\mathcal{S} \subseteq [n\rangle$ **do**
  for $i \in \mathcal{S}$ **do**
    $c_i \leftarrow \max_{\boldsymbol{u}} \boldsymbol{u}\cdot\boldsymbol{g}_i$ as computed by (4)
    $h \leftarrow \max\{h, c_i\}$
Output: $\mathsf{h}_m(\mathcal{C}) = h$

Fig. 1. Computation of the m-height of a linear $[n, k]$ code $\mathcal{C}$ using linear programming.

Remark 2. The augmented form (as in [13, p. 11]) of the linear program (4), which is suitable for the simplex method, involves $2(n-m)$ nonnegative slack variables that satisfy $2(n-m)-k = n+r-2m$ linear equality constraints, where $r = n-k$ is the redundancy of $\mathcal{C}$ (the vector $\boldsymbol{u}$ in (4) can be expressed in terms of the slack variables and therefore can be eliminated from the augmented form). □

Remark 3. The algorithm in Figure 1 works also when we use a different generator matrix for each pair $(\mathcal{S}, i)$. Moreover, it will work also if we permute the coordinates of $\mathcal{C}$, as long as we keep track of the positions that are indexed by $\mathcal{S}$ and i after the permutation. Now, when $m < d$ we have $n - m \ge n-d+1 \ge k$, which means that the matrix $(G)_{\overline{\mathcal{S}}}$ in (4) has full rank k and, so, there is a k-subset $\mathcal{I} \subseteq \overline{\mathcal{S}}$ such that $(G)_{\mathcal{I}}^{-1}(G)_{\overline{\mathcal{S}}}$ contains a $k \times k$ identity matrix. Using $(G)_{\mathcal{I}}^{-1}GP$ in the inner loop in Figure 1, where P is an $n \times n$ permutation matrix that maps $\mathcal{I}$ to $[k\rangle$, results in a modified linear program where $2k$ out of the $2(n-m)$ constraints in (4) become independent of $(\mathcal{S}, i)$ and take the form

$$\left\{\begin{array}{rcl}(\boldsymbol{u})_{[k\rangle} & \le & \mathbf{1} \\ (\boldsymbol{u})_{[k\rangle} & \ge & -\mathbf{1}.\end{array}\right.$$

□

B. Dual linear program

An alternate characterization of the m-height of a linear code can be obtained by considering the dual of the linear program (4) (see [2, Ch. 6]) and [13, Ch. 4]).

Theorem 2. *Let $\mathcal{C}$ be a linear $[n, k, d]$ code over $\mathbb{R}$, let G be a $k \times n$ generator matrix of $\mathcal{C}$ over $\mathbb{R}$, and let $m \in [1 : d\rangle$. Then*

$$\mathsf{h}_m(\mathcal{C}) = \max_{\mathcal{S}} \max_{i\in\mathcal{S}} \min_{\boldsymbol{e}\in\mathcal{E}_{\mathcal{S},i}} \|\boldsymbol{e}\|_1, \tag{5}$$

where $\mathcal{S}$ is as in Theorem 1 and[5]

$$\mathcal{E}_{\mathcal{S},i} = \left\{\boldsymbol{e} \in \mathbb{R}^{n-m} \,:\, (G)_{\overline{\mathcal{S}}}\,\boldsymbol{e} = \boldsymbol{g}_i\right\}. \tag{6}$$

[5]The affine space $\mathcal{E}_{\mathcal{S},i}$ depends only on $\mathcal{C}$, $\mathcal{S}$, and i; in particular, it does not depend on the selected generator matrix G.

Proof. The dual of the linear program (4) (which results in the same output value) takes the form:

$$\boxed{\begin{array}{l}\text{Minimize}\\ \qquad \mathbf{1}\cdot(\boldsymbol{y}+\boldsymbol{z})\\ \text{over } \boldsymbol{y},\boldsymbol{z}\in\mathbb{R}^{n-m}\text{, subject to}\\ \qquad \begin{cases} (G)_{\overline{\mathcal{S}}}(\boldsymbol{y}-\boldsymbol{z}) & = & \boldsymbol{g}_i\\ \boldsymbol{y} & \geq & \mathbf{0}\\ \boldsymbol{z} & \geq & \mathbf{0}, \end{cases}\end{array}} \tag{7}$$

where $\boldsymbol{y}$ and $\boldsymbol{z}$ are two column vectors of variables that correspond, respectively, to the two sets of inequalities in (4). The set of feasible solutions consists of all pairs of nonnegative vectors $\boldsymbol{y}$ and $\boldsymbol{z}$ whose difference, $\boldsymbol{y}-\boldsymbol{z}$, is a vector $\boldsymbol{e}$ in $\mathcal{E}_{\mathcal{S},i}$. Fixing any such vector $\boldsymbol{e}$, the expression $\mathbf{1}\cdot(\boldsymbol{y}+\boldsymbol{z})$ is minimized when

$$\boldsymbol{y}=\max\{\boldsymbol{e},\mathbf{0}\} \quad \text{and} \quad \boldsymbol{z}=-\min\{\boldsymbol{e},\mathbf{0}\},$$

where the maximum and minimum are applied componentwise. For this choice we get $\mathbf{1}\cdot(\boldsymbol{y}+\boldsymbol{z})=\|\boldsymbol{e}\|_1$. We conclude that the solution to the linear program (7) is the minimum of $\|\boldsymbol{e}\|_1$ taken over $\boldsymbol{e}\in\mathcal{E}_{\mathcal{S},i}$. The outer enumeration over $\mathcal{S}$ and i, as in Figure 1, in turn, yields (5). □

The $k\times(n-m)$ matrix $(G)_{\overline{\mathcal{S}}}$ is a generator matrix of the punctured code $(\mathcal{C})_{\overline{\mathcal{S}}}$. As pointed out in Remark 3, when $m<d$ we have $\operatorname{rank}\left((G)_{\overline{\mathcal{S}}}\right)=k$ and, so, $(\mathcal{C})_{\overline{\mathcal{S}}}$ is a linear $[n-m,k]$ code.[6] In view of this, the affine space $\mathcal{E}_{\mathcal{S},i}$ in (6) is a coset of the dual code $((\mathcal{C})_{\overline{\mathcal{S}}})^{\perp}$, which is a linear $[n-m,r-m]$ code with $(G)_{\overline{\mathcal{S}}}$ as its parity-check matrix, where $r=n-k$ is the redundancy of $\mathcal{C}$. Specifically, $\mathcal{E}_{\mathcal{S},i}$ consists of all vectors $\boldsymbol{e}\in\mathbb{R}^{n-m}$ whose syndrome, with respect to the parity-check matrix $(G)_{\overline{\mathcal{S}}}$, equals $\boldsymbol{g}_i$. The inner minimum in (5) is the smallest L_1 norm of any vector in this coset, i.e., the minimization is, in effect, a maximum-likelihood decoder for a memoryless channel where the error values obey a Laplace distribution.

Letting B be an $(r-m)\times(n-m)$ parity-check matrix of $(\mathcal{C})_{\overline{\mathcal{S}}}$ (and a generator matrix of $((\mathcal{C})_{\overline{\mathcal{S}}})^{\perp}$), the inner minimization in (5) can thus be recast in the form

$$\min_{\boldsymbol{v}\in\mathbb{R}^{r-m}}\|\boldsymbol{a}-\boldsymbol{v}B\|_1, \tag{8}$$

for some given vector $\boldsymbol{a}\in\mathbb{R}^{n-m}$, where both B and $\boldsymbol{a}$ depend on $\mathcal{C}$, $\mathcal{S}$, and i. Specifically, $\boldsymbol{a}$ is any vector in the coset $\mathcal{E}_{\mathcal{S},i}$, and $\boldsymbol{v}$ in (8) is related to $\boldsymbol{e}$ in (5) by

$$\boldsymbol{e}=\boldsymbol{a}-\boldsymbol{v}B.$$

The minimization problem in (8), in turn, is known as *least absolute deviations (LAD) regression* [2]. It is a convex optimization problem in $r-m$ variables and can be solved, *inter alia*, using linear programming, by introducing a vector $\widehat{\boldsymbol{e}}$ of $n-m$ auxiliary variables (see [2, p. 158]):

$$\boxed{\begin{array}{l}\text{Minimize}\\ \qquad \mathbf{1}\cdot\widehat{\boldsymbol{e}}\\ \text{over } (\boldsymbol{v},\widehat{\boldsymbol{e}})\in\mathbb{R}^{r-m}\times\mathbb{R}^{n-m}\text{, subject to}\\ \qquad \begin{cases} \boldsymbol{v}B+\widehat{\boldsymbol{e}} & \geq & \boldsymbol{a}\\ \boldsymbol{v}B-\widehat{\boldsymbol{e}} & \leq & \boldsymbol{a}. \end{cases}\end{array}} \tag{9}$$

Note that $\widehat{\boldsymbol{e}}$ is related to $\boldsymbol{y}$ and $\boldsymbol{z}$ in (7) by $\widehat{\boldsymbol{e}}=\boldsymbol{y}+\boldsymbol{z}$ and its entries are the magnitudes of the entries of $\boldsymbol{e}$ in (5).

Remark 4. The linear program (9) involves $n+r-2m$ variables that satisfy $2(n-m)$ linear inequality constraints, and its augmented form involves $2(n-m)$ nonnegative slack variables that satisfy $2(n-m)-(n+r-2m)=k$ linear equality constraints (the same as the linear program (7)). □

We end this section by presenting a structural property of m-extremal codewords.

Lemma 3. *Let $\mathcal{C}$ be a linear $[n,k,d]$ code over $\mathbb{R}$ with a dual code of minimum distance*[7] *$d^{\perp}\geq 2$. For $m\in[1:d\rangle$, let $\boldsymbol{c}=(c_j)_{j\in[n\rangle}$ be any m-extremal codeword in $\mathcal{C}$ and let $\pi(\cdot)$ be a sorting permutation of $\boldsymbol{c}$. Then,*

$$|c_{\pi(m)}|=|c_{\pi(m+1)}|=\ldots=|c_{\pi(m+d^{\perp}-2)}|\neq 0$$

(i.e., $\mathcal{P}_m(\boldsymbol{c})\supseteq\left[m:m+d^{\perp}-1\right\rangle$ and, so, $|\mathcal{P}_m(\boldsymbol{c})|\geq d^{\perp}-1$).

Proof. Since $m<d$ we have $c_{\pi(m)}\neq 0$; without loss of generality we assume hereafter that $|c_{\pi(m)}|=1$. Fix a $k\times n$ generator matrix G of $\mathcal{C}$ and consider the linear program (4) when $i\leftarrow\pi(0)$ and

$$\mathcal{S}\leftarrow\{\pi(t)\,:\,t\in[m\rangle\}\,.$$

The maximum then is attained at the vector $\boldsymbol{u}$ that (uniquely) satisfies $\boldsymbol{c}=\boldsymbol{u}G$. The result will follow once we show that there exists a $(d^{\perp}-1)$-subset $\mathcal{I}$ of $\overline{\mathcal{S}}$ such that, for this maximizing $\boldsymbol{u}$,

$$(\boldsymbol{c})_{\mathcal{I}}=\boldsymbol{u}\,(G)_{\mathcal{I}}\in\{\pm 1\}^{d^{\perp}-1}.$$

Suppose to the contrary that such a subset $\mathcal{I}$ does not exist, namely, there exists a subset $\mathcal{J}\subseteq\overline{\mathcal{S}}$ of size $|\mathcal{J}|>|\overline{\mathcal{S}}|-d^{\perp}+1$ such that at any coordinate $j\in\mathcal{J}$ of $\boldsymbol{u}\,(G)_{\overline{\mathcal{S}}}$, both inequalities in (4) are strict. Referring to the proof of Theorem 2, consider the minimizing vectors $\boldsymbol{y}$ and $\boldsymbol{z}$ of the dual linear program (7). By the complementary slackness theorem [13, p. 96], both $\boldsymbol{y}$ and $\boldsymbol{z}$ have zeroes at all the positions that are indexed by $\mathcal{J}$. Consequently, $\boldsymbol{e}=\boldsymbol{y}-\boldsymbol{z}$ has zeroes at those positions as well, which means that

$$\mathsf{w}(\boldsymbol{e})\leq|\overline{\mathcal{S}}|-|\mathcal{J}|<d^{\perp}-1. \tag{10}$$

On the other hand, by (7) we also have

$$(G)_{\overline{\mathcal{S}}}\,\boldsymbol{e}=\boldsymbol{g}_i. \tag{11}$$

Combining (10) and (11) we conclude that there are $d^{\perp}-1$ linearly dependent columns in G. However, this is impossible,

[6] Conversely, when $m\geq d$, for any m-subset $\mathcal{S}\subseteq[n\rangle$ that is a superset of the support of a minimum-weight codeword of $\mathcal{C}$ we have $\operatorname{rank}\left((G)_{\overline{\mathcal{S}}}\right)<k$. Yet $\operatorname{rank}(G)=k$ and, so, there is an index $i\in\mathcal{S}$ such that $\boldsymbol{g}_i$ is not in the linear span of the columns of $(G)_{\overline{\mathcal{S}}}$. In this case $\mathcal{E}_{\mathcal{S},i}$ is empty, which translates into an infinite value in the minimum in (5).

[7] The case $d^{\perp}=1$ corresponds to a code $\mathcal{C}$ that has a coordinate which is identically 0 across all codewords. Shortening $\mathcal{C}$ on that coordinate does not affect its m-height.

since G is a parity-check matrix of the dual code $\mathcal{C}^\perp$ and, as such, every $d^\perp - 1$ columns in G are linearly independent. $\square$

III. Combinatorial characterization of the height

In this section, we present characterizations of the m-height of a linear code as a maximum (or a maximum–minimum) taken over an explicit subset of finitely many codewords. In essence, these codewords correspond to a subset of the vertices of the union of the feasible regions of the linear program (4) (respectively, the dual linear program (7)) as $\mathcal{S}$ ranges over all the m-subsets of $[n\rangle$ and i ranges over the elements in each $\mathcal{S}$.

A. Primal characterization

The next lemma adds to Lemma 3 in that it implies that there is always an m-extremal codeword $\boldsymbol{c} \in \mathcal{C}$ such that $|\mathcal{P}_m(\boldsymbol{c})| \geq k$. (Note that for the special case of MDS codes, we get from Lemma 3 that $|\mathcal{P}_m(\boldsymbol{c})| \geq k$ for any m-extremal codeword $\boldsymbol{c}$ in $\mathcal{C}$.)

Lemma 4. *Let $\mathcal{C}$ be a linear $[n,k,d]$ code over $\mathbb{R}$ and let $m \in [1:d\rangle$. Suppose that $\boldsymbol{c}$ is an m-extremal codeword in $\mathcal{C}$ with a largest set $\mathcal{T} = \mathcal{P}_m(\boldsymbol{c})$ among all m-extremal codewords in $\mathcal{C}$. Then for any $k \times n$ generator matrix G of $\mathcal{C}$ over $\mathbb{R}$,*

$$\operatorname{rank}\left((G)_{\mathcal{T}}\right) = k$$

(in particular, $|\mathcal{T}| \geq k$).

Equivalently, in any parity-check matrix of $\mathcal{C}$ over $\mathbb{R}$, the columns in $(H)_{\overline{\mathcal{T}}}$ are linearly independent.

Proof. Let $\pi(\cdot)$ be a sorting permutation of $\boldsymbol{c}$; we assume hereafter in the proof (without loss of generality) that $|c_{\pi(m)}| = 1$.

Suppose to the contrary that $\operatorname{rank}\left((G)_{\mathcal{T}}\right) < k$ (or that for a parity-check matrix H of $\mathcal{C}$, the columns in $(H)_{\overline{\mathcal{T}}}$ are linearly dependent). Then there exists a nonzero codeword $\boldsymbol{x} = (x_j)_{j\in[n\rangle} \in \mathcal{C}$ such that $\mathsf{Supp}(\boldsymbol{x}) \subseteq \overline{\mathcal{T}}$, i.e.,

$$\mathsf{Supp}(\boldsymbol{x}) \cap \mathcal{T} = \emptyset. \tag{12}$$

For $\varepsilon \in \mathbb{R}$, define

$$\boldsymbol{c}_\varepsilon = (c_{\varepsilon,j})_{j\in[n\rangle} = \boldsymbol{c} + \varepsilon \cdot \boldsymbol{x}.$$

We distinguish between two cases.

Case 1: $\mathsf{Supp}(\boldsymbol{x}) \cap \mathcal{P}_0(\boldsymbol{c}) \neq \emptyset$. Let $i \in \mathsf{Supp}(\boldsymbol{x}) \cap \mathcal{P}_0(\boldsymbol{c})$; by possibly negating $\boldsymbol{x}$ we can assume that $c_{\pi(i)}$ and $x_{\pi(i)}$ have the same sign. Let $\varepsilon \in \mathbb{R}$ be in the range

$$0 < \varepsilon < \min_{j\in\mathsf{Supp}(\boldsymbol{x})} \left|\frac{c_j}{x_j}\right|.$$

By (12), for every $j \in [n\rangle$ we have:

$$\begin{array}{rcl} |c_j| = 1 & \iff & |c_{\varepsilon,j}| = 1, \\ |c_j| > 1 & \iff & |c_{\varepsilon,j}| > 1, \\ |c_j| < 1 & \iff & |c_{\varepsilon,j}| < 1. \end{array}$$

Hence,

$$\mathcal{P}_m(\boldsymbol{c}_\varepsilon) = \mathcal{T}$$

(and all the entries in $(\boldsymbol{c}_\varepsilon)_{\mathcal{T}}$ are ± 1). On the other hand,

$$|c_{\pi(i)} + \varepsilon \cdot x_{\pi(i)}| > |c_{\pi(i)}| = |c_{\pi(0)}|,$$

which implies that $\mathsf{h}_m(\boldsymbol{c}_\varepsilon) > \mathsf{h}_m(\boldsymbol{c})$, thereby contradicting the m-extremality of $\boldsymbol{c}$.

Case 2: $\mathsf{Supp}(\boldsymbol{x}) \cap \mathcal{P}_0(\boldsymbol{c}) = \emptyset$. Let ε be of smallest magnitude such that $(\boldsymbol{c}_\varepsilon)_{\overline{\mathcal{T}}}$ contains an entry of magnitude 1; explicitly,

$$\varepsilon = \pm \min_{j\in\mathsf{Supp}(\boldsymbol{x})} \left|\frac{1-|c_j|}{x_j}\right|, \tag{13}$$

with the sign determined to match that of $c_{j'} \cdot (1-|c_{j'}|)/x_{j'}$, where j' is an index j at which the minimum in (13) is attained.

On the one hand, for every $j \in [n\rangle$:

$$\begin{array}{rcl} |c_j| = 1 & \implies & |c_{\varepsilon,j}| = 1, \\ |c_j| > 1 & \implies & |c_{\varepsilon,j}| \geq 1, \\ |c_j| < 1 & \implies & |c_{\varepsilon,j}| \leq 1. \end{array}$$

These, in turn, imply that

$$\mathcal{T} \subseteq \mathcal{P}_m(\boldsymbol{c}_\varepsilon).$$

Moreover, the containment is strict, since there exists at least one index, namely, $j' \in \overline{\mathcal{T}}$, where $|c_{\varepsilon,j'}| = 1$ while $|c_{j'}| \neq 1$.

On the other hand,

$$\mathsf{h}_m(\boldsymbol{c}_\varepsilon) = |c_{\varepsilon,\pi(0)}| = |c_{\pi(0)}| = \mathsf{h}_m(\boldsymbol{c}).$$

Thus, $\boldsymbol{c}_\varepsilon$ is m-extremal and $|\mathcal{P}_m(\boldsymbol{c}_\varepsilon)| > |\mathcal{T}|$; yet this contradicts the assumption on the maximality of $|\mathcal{T}|$ among all m-extremal codewords in $\mathcal{C}$. $\square$

Remark 5. Lemma 4 holds also when $m \geq d$, except when the code $\mathcal{C}$ therein is MDS. Specifically, when $m \geq d$, for any m-extremal codeword $\boldsymbol{c} \in \mathcal{C}$ (in particular, for any minimum-weight codeword of $\mathcal{C}$) we have $c_{\pi(m)} = 0$ (and therefore $\mathsf{h}_m(\boldsymbol{c}) = \infty$), yet when $\mathcal{C}$ is MDS, none of its nonzero codewords has more than $k-1$ zero entries. $\square$

Lemma 4 implies the next theorem, which presents another characterization of the m-height of a linear code. We introduce the following notation. For a linear $[n, k{=}n{-}r]$ code $\mathcal{C}$, we let $\Upsilon(\mathcal{C})$ be the set of all k-subsets $\mathcal{I} \subseteq [n\rangle$ such that the projections of the codewords of $\mathcal{C}$ on $\mathcal{I}$ range over the whole space $\mathbb{R}^k$, namely:

$$\left\{(\boldsymbol{c})_{\mathcal{I}} \,:\, \boldsymbol{c} \in \mathcal{C}\right\} = \mathbb{R}^k.$$

Equivalently, in any $k \times n$ generator matrix G of $\mathcal{C}$, the submatrix $(G)_{\mathcal{I}}$ is nonsingular. It is easy to see that a k-subset $\mathcal{I}$ belongs to $\Upsilon(\mathcal{C})$ if and only if its complement r-subset $\overline{\mathcal{I}}$ belongs to $\Upsilon(\mathcal{C}^\perp)$.

Theorem 5. *Let $\mathcal{C}$ be a linear $[n,k,d]$ code over $\mathbb{R}$, let $m \in [1:d\rangle$, and let G be any $k \times n$ generator matrix of $\mathcal{C}$ over $\mathbb{R}$. Then*

$$\mathsf{h}_m(\mathcal{C}) = \max_{\mathcal{I}\in\Upsilon(\mathcal{C})} \max_{\boldsymbol{u}\in\{\pm1\}^k} \mathsf{h}_m\big(\boldsymbol{u}\,(G)_{\mathcal{I}}^{-1} G\big). \tag{14}$$

Also,

$$\mathsf{h}_m(\mathcal{C}) = \max_{\mathcal{I}\in\Upsilon(\mathcal{C})} \max_{\boldsymbol{u}} \left\|\boldsymbol{u}\,(G)_{\mathcal{I}}^{-1}(G)_{\overline{\mathcal{I}}}\right\|_\infty, \tag{15}$$

where $\boldsymbol{u}$ ranges over all the elements in $\{\pm1\}^k$ such that the vector $\boldsymbol{u}\,(G)_{\mathcal{I}}^{-1}(G)_{\overline{\mathcal{I}}}$ ($\in \mathbb{R}^{n-k}$) has no more than m entries of

magnitude greater than 1 *and no more than* $n-m-1$ *entries of magnitude smaller than* 1.

Proof. Define the following subset $\mathcal{C}'$ of $\mathcal{C}$:

$$\mathcal{C}' = \Big\{ \boldsymbol{c} \in \mathcal{C} \,:\, (\boldsymbol{c})_{\mathcal{I}} \in \{\pm 1\}^k \text{ for some } \mathcal{I} \in \Upsilon(\mathcal{C}) \Big\}$$

(the subset $\mathcal{I}$ may depend on $\boldsymbol{c}$). The right-hand side of (14) then equals

$$\max_{\boldsymbol{c} \in \mathcal{C}'} \mathsf{h}_m(\boldsymbol{c}).$$

Eq. (14) now follows from Lemma 4, which implies that $\mathcal{C}'$ contains at least one m-extremal codeword of $\mathcal{C}$.

Turning to Eq. (15), given some $\mathcal{I} \in \Upsilon(\mathcal{C})$, consider a codeword $\boldsymbol{c} = \boldsymbol{u}\,(G)_{\mathcal{I}}^{-1} G$. The assumed conditions on $\boldsymbol{u}$ in the inner maximization in (15) are then equivalent to requiring that $(\boldsymbol{c})_{\mathcal{I}} \in \{\pm 1\}^k$ and $\mathcal{I} \subseteq \mathcal{P}_m(\boldsymbol{c})$. Hence, the right-hand side of (15) equals

$$\max_{\boldsymbol{c} \in \mathcal{C}''} \|\boldsymbol{c}\|_\infty,$$

where

$$\mathcal{C}'' = \Big\{ \boldsymbol{c} \in \mathcal{C} \,:\, (\boldsymbol{c})_{\mathcal{I}} \in \{\pm 1\}^k \text{ for some } \mathcal{I} \in \Upsilon(\mathcal{C}) \text{ such that } \mathcal{I} \subseteq \mathcal{P}_m(\boldsymbol{c}) \Big\}.$$

Now, for $\boldsymbol{c} \in \mathcal{C}''$ we have $\mathsf{h}_m(\boldsymbol{c}) = \|\boldsymbol{c}\|_\infty$, which means that the right-hand side of (15) also equals

$$\max_{\boldsymbol{c} \in \mathcal{C}''} \mathsf{h}_m(\boldsymbol{c}).$$

Lemma 4 again implies that $\mathcal{C}''$ contains at least one m-extremal codeword of $\mathcal{C}$. □

Noting that the vectors $\boldsymbol{u}$ in the inner maximizations in both (14) and (15) can be restricted to have 1 as their first entry, the total number of terms to maximize over in the right-hand side of (14) (or (15)) is at most

$$\binom{n}{k} \cdot 2^{k-1}. \tag{16}$$

Remark 6. Eq. (15) can be expressed also in terms of a full-rank $r \times n$ parity-check matrix H of $\mathcal{C}$ (where $r = n-k$). Recall that for every r-subset $\mathcal{J} \in \Upsilon(\mathcal{C}^\perp)$, its complement k-subset $\mathcal{I} = \overline{\mathcal{J}}$ belongs to $\Upsilon(\mathcal{C})$ (and vice versa), and we have

$$(H)_{\mathcal{J}}^{-1} (H)_{\overline{\mathcal{J}}} = -\left((G)_{\mathcal{I}}^{-1} (G)_{\overline{\mathcal{I}}}\right)^\top.$$

From (15) we then get

$$\mathsf{h}_m(\mathcal{C}) = \max_{\mathcal{J} \in \Upsilon(\mathcal{C}^\perp)} \max_{\boldsymbol{u}} \left\| (H)_{\mathcal{J}}^{-1} (H)_{\overline{\mathcal{J}}}\, \boldsymbol{u}^\top \right\|_\infty, \tag{17}$$

where $\boldsymbol{u}$ ranges over all the elements in $\{\pm 1\}^k$ such that $(H)_{\mathcal{J}}^{-1} (H)_{\overline{\mathcal{J}}}\, \boldsymbol{u}^\top$ has no more than m entries of magnitude greater than 1 and no more than $n-m-1$ entries of magnitude smaller than 1. In both (15) and (17), the total number of terms to maximize over is the same (and the range of $\boldsymbol{u}$ is the same). However, the computation of (17) involves inverting $r \times r$ matrices, compared to (15) where the order of the inverted matrices is $k \times k$; hence, from a complexity viewpoint, Eq. (17) may be preferable when $k > r$. □

B. Dual characterization

We next turn to results that are conceptual duals of Lemma 4 and Theorem 5, based on the characterization (5) of the m-height of a linear code.

When $m = r$ we have $|\mathcal{E}_{\mathcal{S},i}| = 1$, in which case the minimization in (5) becomes trivial (we will deal with this case in Section III-C).

The next lemma provides an explicit solution to the minimization problem (8) (and hence to (5)) for the case $m = r-1$. This lemma is found in [2, §1.2] and seems to have its roots in the 18th century. For completeness, we include a proof in the Appendix.

Lemma 6. *Let* $\boldsymbol{a} = (a_j)_{j \in [\ell\rangle}$ *and* $\boldsymbol{b} = (b_j)_{j \in [\ell\rangle}$ *be two vectors in* $\mathbb{R}^\ell$ *where* $\boldsymbol{b} \neq \boldsymbol{0}$. *The function* $f : \mathbb{R} \to \mathbb{R}$ *that is defined by*

$$v \mapsto f(v) = \|\boldsymbol{a} - v \cdot \boldsymbol{b}\|_1$$

attains its minimum (not necessarily uniquely) at the weighted median, v_{med}, *among the values*

$$a_j / b_j, \quad j \in \mathsf{Supp}(\boldsymbol{b}),$$

with respective weights

$$|b_j|, \quad j \in \mathsf{Supp}(\boldsymbol{b}).$$

In particular, at that minimum, the vector $\boldsymbol{a} - v_{\mathrm{med}} \cdot \boldsymbol{b}$ *has at least one zero coordinate at some position in* $\mathsf{Supp}(\boldsymbol{b})$.

The following lemma is the dual counterpart of Lemma 4.

Lemma 7. *Using the notation of Theorem 2, given an* m*-subset* $\mathcal{S} \subseteq [n\rangle$ *and* $i \in \mathcal{S}$, *let* $\boldsymbol{e}'$ *be a vector in* $\mathcal{E}_{\mathcal{S},i}$ *that satisfies the following two conditions.*

(i) It attains the inner minimum in (5), i.e.,

$$\|\boldsymbol{e}'\|_1 = \min_{\boldsymbol{e} \in \mathcal{E}_{\mathcal{S},i}} \|\boldsymbol{e}\|_1. \tag{18}$$

(ii) It has the smallest support, $\mathcal{T} = \mathsf{Supp}(\boldsymbol{e}')$, *among all vectors in* $\mathcal{E}_{\mathcal{S},i}$ *that attain that minimum.*

Then the columns in $(G)_{\mathcal{T}}$ *are linearly independent (in particular,* $|\mathcal{T}| \leq k$*).*

Equivalently, writing $\overline{\mathcal{T}} = [n-m\rangle \setminus \mathcal{T}$, *for any* $(r-m) \times (n-m)$ *parity-check matrix* B *of the punctured code* $(\mathcal{C})_{\overline{\mathcal{S}}}$,

$$\operatorname{rank}\left((B)_{\overline{\mathcal{T}}}\right) = r - m.$$

The lemma follows from the fundamental theorem of linear programming, which states that the minimum in the linear program (7) is attained by a so-called basic feasible solution, namely, by vectors $\boldsymbol{y}$ and $\boldsymbol{z}$ such that the columns of $(G)_{\overline{\mathcal{S}}}$ that are indexed by $\mathsf{Supp}(\boldsymbol{y}) \cup \mathsf{Supp}(\boldsymbol{z})$ are linearly independent (see [13, p. 19]). We next present an alternate proof which is based on Lemma 6; our proof is a modification of the proof of Theorem 1 in [2, §1.2].

Proof of Lemma 7. Suppose to the contrary that the columns in $(G)_{\mathcal{T}}$ are linearly dependent (or that $\operatorname{rank}\left((B)_{\overline{\mathcal{T}}}\right) < r - m$). Then there exists a nonzero codeword $\boldsymbol{x} \in ((\mathcal{C})_{\overline{\mathcal{S}}})^\perp$ with support $\mathsf{Supp}(\boldsymbol{x}) \subseteq \mathcal{T}$. Consider the vectors

$$\boldsymbol{e}' - \varepsilon \cdot \boldsymbol{x}, \quad \varepsilon \in \mathbb{R}.$$

Their supports are all subsets of $\mathcal{T}$; moreover, these vectors are all in the same coset, $\mathcal{E}_{\mathcal{S},i}$, of $((\mathcal{C})_{\overline{\mathcal{S}}})^{\perp}$. Therefore,

$$\|\boldsymbol{e}'\|_1 \overset{(18)}{=} \min_{\boldsymbol{e}\in\mathcal{E}_{\mathcal{S},i}} \|\boldsymbol{e}\|_1 \le \min_{\varepsilon\in\mathbb{R}} \|\boldsymbol{e}' - \varepsilon\cdot\boldsymbol{x}\|_1 \le \|\boldsymbol{e}'\|_1, \quad (19)$$

which means that the inequalities in (19) are in fact equalities. By Lemma 6 it follows that there is a minimizer $\varepsilon_{\min}$ in (19) such that, at some position $h \in \mathsf{Supp}(\boldsymbol{x})$ $(\subseteq \mathcal{T})$, the vector $\boldsymbol{e}'' = \boldsymbol{e}' - \varepsilon_{\min}\cdot\boldsymbol{x}$ has a zero entry. Hence,

$$\mathsf{Supp}(\boldsymbol{e}'') \subseteq \mathcal{T}\setminus\{h\} \subsetneq \mathcal{T},$$

namely, $\boldsymbol{e}''$ attains the minima in (19) and has a smaller support than $\boldsymbol{e}'$. This, however, contradicts condition (ii). □

Lemma 7 implies the following (dual) combinatorial characterization of the m-height of a linear code.

Theorem 8. *Let $\mathcal{C}$ be a linear $[n, k{=}n{-}r, d]$ code over $\mathbb{R}$ and let $m \in [1:d\rangle$. Then the following holds.*

(i) For any $k\times n$ generator matrix $G = (\boldsymbol{g}_0\ \boldsymbol{g}_1\ \ldots\ \boldsymbol{g}_{n-1})$ of $\mathcal{C}$ over $\mathbb{R}$,

$$\mathsf{h}_m(\mathcal{C}) = \max_{\mathcal{S}} \max_{i\in\mathcal{S}} \min_{\substack{\mathcal{I}\in\Upsilon(\mathcal{C}):\\ \mathcal{I}\subseteq\overline{\mathcal{S}}}} \left\|(G)_{\mathcal{I}}^{-1}\boldsymbol{g}_i\right\|_1, \quad (20)$$

where $\mathcal{S}$ ranges over all m-subsets of $[n\rangle$.

(ii) For any $r\times n$ parity-check matrix H of $\mathcal{C}$ over $\mathbb{R}$,

$$\mathsf{h}_m(\mathcal{C}) = \max_{\mathcal{S}} \max_{i\in\mathcal{S}} \min_{\substack{\mathcal{J}\in\Upsilon(\mathcal{C}^{\perp}):\\ \mathcal{J}\supseteq\mathcal{S}}} \left\|\left[(H)_{\mathcal{J}}^{-1}\right]_i (H)_{\overline{\mathcal{J}}}\right\|_1, \quad (21)$$

where $\mathcal{S}$ is as in part (i).[8]

Proof. (i) Let $\mathcal{S}\subseteq[n\rangle$ and $i\in\mathcal{S}$ be maximizers of the outer maxima in (5). By Lemma 7, the inner minimum in (5) is attained at a vector $\boldsymbol{e}'\in\mathcal{E}_{\mathcal{S},i}$ whose support, $\mathcal{T} = \mathsf{Supp}(\boldsymbol{e}')$, is such that the columns in $(G)_{\mathcal{T}}$ are linearly independent. Since $|\overline{\mathcal{S}}| = n-m \ge n-d+1 \ge k$, the $k\times(n{-}m)$ matrix $(G)_{\overline{\mathcal{S}}}$ has full rank k and, so, we can extend $\mathcal{T}$ to a k-subset $\mathcal{I}\subseteq\overline{\mathcal{S}}$ that belongs to $\Upsilon(\mathcal{C})$. From

$$\mathsf{Supp}(\boldsymbol{e}')\subseteq\mathcal{I} \quad\text{and}\quad (G)_{\overline{\mathcal{S}}}\boldsymbol{e}' \overset{(6)}{=} \boldsymbol{g}_i$$

it follows that the vector $\boldsymbol{e}'' = (G)_{\mathcal{I}}^{-1}\boldsymbol{g}_i$, while being a subvector of $\boldsymbol{e}'$, still has $\mathcal{T}$ as its support. Hence, $\|\boldsymbol{e}''\|_1 = \|\boldsymbol{e}'\|_1$, which implies that the right-hand sides of (5) and (20) are equal.

(ii) We repeat the argument in Remark 6: for every $\mathcal{J}\in\Upsilon(\mathcal{C}^{\perp})$, the complement $\mathcal{I} = \overline{\mathcal{J}}$ belongs to $\Upsilon(\mathcal{C})$, and

$$(H)_{\mathcal{J}}^{-1}(H)_{\overline{\mathcal{J}}} = -\left((G)_{\mathcal{I}}^{-1}(G)_{\overline{\mathcal{I}}}\right)^{\top}.$$

Hence, for every row index $i\in\mathcal{J}$ $(\supseteq\mathcal{S})$ of $(H)_{\mathcal{J}}^{-1}$,

$$\left\|\left[(H)_{\mathcal{J}}^{-1}\right]_i (H)_{\overline{\mathcal{J}}}\right\|_1 = \left\|(G)_{\mathcal{I}}^{-1}\boldsymbol{g}_i\right\|_1,$$

and the result follows from part (i). □

The total number of terms to maximize/minimize over in (20)–(21) is at most

$$\binom{n}{m}\cdot m\cdot\binom{n-m}{k} = m\cdot\binom{n}{r}\cdot\binom{r}{m}. \quad (22)$$

This figure is smaller than (16) for high-rate codes (when $k \ge r + \Omega(\log r)$) or when m is either small or close to r. Comparing the computations of (20) and (21), the latter is preferable when $k > r$ (see Remark 6).

Remark 7. When $m \ge d$, the inner minimizations in Eqs. (20)–(21) are over empty sets for any m-subset $\mathcal{S}$ that is a superset of the support of a minimum-weight codeword of $\mathcal{C}$; in this case the inner minimum can be seen as infinity. In contrast, we note that Eqs. (14), (15), and (17), yield finite values even when $m \ge d$, so Theorem 5 clearly does not hold in this case. □

The next corollary identifies an m-extremal codeword in case the code is MDS.

Corollary 9. *Using the notation in Theorem 8, suppose that $\mathcal{C}$ therein is MDS and let $\mathcal{S}$, i, and $\mathcal{I}$ (respectively, $\mathcal{J} = \overline{\mathcal{I}}$) be where the maximum–minimum in (20) (respectively, (21) is attained. Let $\boldsymbol{c}$ be the unique codeword in $\mathcal{C}$ that is defined by*

$$\boldsymbol{c} = (\boldsymbol{c})_{\mathcal{I}}(G)_{\mathcal{I}}^{-1}G, \quad (23)$$

where

$$(\boldsymbol{c})_{\mathcal{I}} = \operatorname{sgn}\left((G)_{\mathcal{I}}^{-1}\boldsymbol{g}_i\right)^{\top} = -\operatorname{sgn}\left(\left[(H)_{\mathcal{J}}^{-1}\right]_i (H)_{\overline{\mathcal{J}}}\right), \quad (24)$$

with $\operatorname{sgn}(\cdot)$ applied componentwise. Then $\boldsymbol{c}$ is m-extremal in $\mathcal{C}$.

Proof. As argued in the proof of Theorem 8, the inner minimum in (5) is attained at a vector $\boldsymbol{e}' = (e'_j)_{j\in\overline{\mathcal{S}}}$ with support $\mathsf{Supp}(\boldsymbol{e}')\subseteq\mathcal{I}$. We next claim the equality $\mathsf{Supp}(\boldsymbol{e}') = \mathcal{I}$. Otherwise, from $(G)_{\overline{\mathcal{S}}}\boldsymbol{e}' = \boldsymbol{g}_i$ we would have k linearly dependent columns in G, which is impossible since $\mathcal{C}$ is MDS.

The respective vectors

$$\boldsymbol{y}' = (y'_j)_j = \max\{\boldsymbol{e}', \boldsymbol{0}\} \quad\text{and}\quad \boldsymbol{z}' = (z'_j)_j = -\min\{\boldsymbol{e}', \boldsymbol{0}\}$$

(which satisfy $\boldsymbol{y}' - \boldsymbol{z}' = \boldsymbol{e}'$) are minimizers of the dual linear program (7).[9] Letting $\boldsymbol{u}'$ be a maximizer of the primal program (4), by the complementary slackness theorem we have, for every $j\in\mathcal{I}$,

$$\begin{array}{lllll} e'_j > 0 & \Longrightarrow & y'_j > 0 & \Longrightarrow & \boldsymbol{u}'\cdot\boldsymbol{g}_j = \ \ 1 \\ e'_j < 0 & \Longrightarrow & z'_j > 0 & \Longrightarrow & \boldsymbol{u}'\cdot\boldsymbol{g}_j = -1. \end{array}$$

If follows that

$$\boldsymbol{u}'(G)_{\mathcal{I}} = \operatorname{sgn}\left((G)_{\mathcal{I}}^{-1}\boldsymbol{g}_i\right)^{\top}$$

(which is a vector in $\{\pm1\}^k$). The vector $\boldsymbol{c}$ in (23)–(24), in turn, equals $\boldsymbol{u}'G$ and is thus m-extremal in $\mathcal{C}$. □

Example 1. We consider here the MDS construction from [14, §V]. Given an integer $n > 2$, let $\alpha = \pi/n$ and let ω denote the complex primitive $(2n)$th root of unity $e^{\imath\alpha}$, where $\imath = \sqrt{-1}$. We define $\mathcal{C}(n)$ to be the following linear $[n, n{-}2]$ code over $\mathbb{R}$:

$$\mathcal{C}(n) = \left\{\boldsymbol{c} = (c_j)_{j\in[n\rangle}\in\mathbb{R}^n \,:\, \textstyle\sum_{j\in[n\rangle} c_j\omega^j = 0\right\}.$$

[8] We index the rows of $(H)_{\mathcal{J}}^{-1}$ by the r-subset $\mathcal{J}$, which is the set of column indexes of $(H)_{\mathcal{J}}$.

[9] In the terminology of linear programming, the MDS property, which was used earlier, implies that the dual linear program program (7) is nondegenerate.

This code is MDS and negacyclic with a $2 \times n$ parity-check matrix $H = (\boldsymbol{h}_j)_{j\in[n\rangle}$ whose columns are given by

$$\boldsymbol{h}_j = \begin{pmatrix} \cos(j\alpha) \\ \sin(j\alpha) \end{pmatrix}, \quad j \in [n\rangle \tag{25}$$

(note that this matrix is different from the parity-check matrix used in [14, §V]). Using purely geometric arguments, it was shown in [14, Proposition 11] that

$$\mathsf{h}_2(\mathcal{C}(n)) = \frac{1}{2\sin^2(\alpha/2)} - 1. \tag{26}$$

We will re-derive this formula using Theorem 8 and also show that

$$\mathsf{h}_1(\mathcal{C}(n)) = \begin{cases} \dfrac{1}{\sin(\alpha/2)} - 1, & \text{when } n \text{ is odd} \\ \cot(\alpha/2) - 1, & \text{when } n \text{ is even.} \end{cases} \tag{27}$$

Due to the negacyclic property of $\mathcal{C}(n)$, we can assume hereafter that the maximum in (21) is attained at $i = 0$ (and at a subset $\mathcal{S}$ that contains the element 0). Let $\mathcal{J} = \{0, t\}$, for some $t \in [1 : n\rangle$. Then

$$(H)_{\mathcal{J}}^{-1} = \begin{pmatrix} 1 & \cos(t\alpha) \\ 0 & \sin(t\alpha) \end{pmatrix}^{-1} = \frac{1}{\sin(t\alpha)} \begin{pmatrix} \sin(t\alpha) & -\cos(t\alpha) \\ 0 & 1 \end{pmatrix}$$

and, therefore, for every $j \in [n\rangle$,

$$\begin{aligned} \left[(H)_{\mathcal{J}}^{-1}\right]_0 \boldsymbol{h}_j &= \frac{1}{\sin(t\alpha)} \begin{pmatrix} \sin(t\alpha) & -\cos(t\alpha) \end{pmatrix} \begin{pmatrix} \cos(j\alpha) \\ \sin(j\alpha) \end{pmatrix} \\ &= \frac{\sin((t-j)\alpha)}{\sin(t\alpha)}. \end{aligned} \tag{28}$$

Now,

$$\begin{aligned} \sum_{j\in[1:n\rangle\setminus\{t\}} |\sin((t-j)\alpha)| &= \sum_{j\in[1:n\rangle\setminus\{t\}} \sin(j\alpha) \\ &= \Big(\sum_{j\in[n\rangle} \sin(j\alpha)\Big) - \sin(t\alpha) = \cot(\alpha/2) - \sin(t\alpha), \end{aligned}$$

where the last step follows from Lagrange's trigonometric identities [8, §2.4.1.6, No. 13]. Combining this with (28), we conclude that

$$\left\| \left[(H)_{\mathcal{J}}^{-1}\right]_0 (H)_{\overline{\mathcal{J}}} \right\|_1 = \frac{\cot(\alpha/2)}{\sin(t\alpha)} - 1. \tag{29}$$

For $m = 2$, we substitute $\mathcal{S} = \mathcal{J} = \{0, t\}$ in (21), which translates into maximizing (29) over $t \in [1 : n\rangle$. The maximum is attained at $t \in \{1, n-1\}$, which (with $\sin\alpha = 2\sin(\alpha/2)\cos(\alpha/2)$) yields (26). By Corollary 9 and Eq. (28), one of the 2-extremal codewords is all 1's except for its first two entries, which are given by:

$$c_0 = \mathsf{h}_2(\mathcal{C}(n)) \quad \text{and} \quad c_1 = -\mathsf{h}_2(\mathcal{C}(n)).$$

For $m = 1$, we substitute $\mathcal{S} = \{0\}$ and $\mathcal{J} = \{0, t\}$ in (21), which means that we need now to minimize (29) over $t \in [1 : n\rangle$. When n is even, the minimum is attained at $t = n/2$, with a 1-extremal codeword which is all 1's except for the following two entries:

$$c_0 = 0 \quad \text{and} \quad c_{n/2} = -\mathsf{h}_1(\mathcal{C}(n)).$$

When n is odd, the minimum is attained at $t = (n \pm 1)/2$ with a 1-extremal codeword[10] which is all 1's except for the middle entry, which is:

$$c_{(n-1)/2} = -\mathsf{h}_1(\mathcal{C}(n)).$$

These results, in turn, establish (27). □

Remark 8. It has been recently shown in [12] that the code $\mathcal{C}(n)$ in Example 1 has the smallest 2-height among all linear $[n, n-2]$ codes over $\mathbb{R}$. With respect to the 1-height, however, a better code is obtained by applying the simple construction in [14, §III.B] with $r = 2$: the $2 \times n$ parity-check matrix then consists of $\lfloor n/2 \rfloor$ copies of the column vector $(1\ 0)^\top$ and $\lceil n/2 \rceil$ copies of $(0\ 1)^\top$. The 1-height in this case is $\lceil n/2 \rceil - 1$, which, too, has been shown in [12] to be optimal (see also [10]); that is compared to (at least) $\cot(\pi/(2n)) - 1 \approx (2/\pi)n - 1$ for the code $\mathcal{C}(n)$. □

C. Height at the redundancy of the code

In this section, we look at the m-height of a linear $[n, k{=}n{-}r, d]$ code $\mathcal{C}$ when $m = r$; recall by (1) that the r-height is finite if and only if $r \le d - 1$, which happens (with equality) if and only if $\mathcal{C}$ is MDS.

The next corollary follows from Lemma 3.

Corollary 10. *Let $\mathcal{C}$ be a linear $[n, k{=}n{-}r]$ MDS code over $\mathbb{R}$. Then for any r-extremal codeword $\boldsymbol{c}$ in $\mathcal{C}$,*

$$\mathcal{P}_r(\boldsymbol{c}) \supseteq [r : n\rangle$$

and, in particular, $\mathsf{w}(\boldsymbol{c}) = n$.

In the Appendix, we present an alternate proof of Corollary 10 that uses Theorem 1 directly without relying on the theory of duality of linear programming.

Remark 9. It follows from Corollary 10 that any shortening of a linear $[n, k{=}n{-}r]$ MDS code strictly decreases the r-height of the code. □

The next corollary follows from Theorem 8.

Corollary 11. *Let $\mathcal{C}$ be a linear $[n, k{=}n{-}r]$ MDS code over $\mathbb{R}$. Then the following holds.*

(i) For any $k \times n$ generator matrix $G = (\boldsymbol{g}_0\ \boldsymbol{g}_1\ \cdots\ \boldsymbol{g}_{n-1})$ of $\mathcal{C}$ over $\mathbb{R}$,

$$\mathsf{h}_r(\mathcal{C}) = \max_{\mathcal{S}} \max_{i\in\mathcal{S}} \left\| (G)_{\overline{\mathcal{S}}}^{-1} \boldsymbol{g}_i \right\|_1, \tag{30}$$

where $\mathcal{S}$ ranges over all r-subsets of $[n\rangle$.

(ii) For any $r \times n$ parity-check matrix H of $\mathcal{C}$ over $\mathbb{R}$,

$$\mathsf{h}_r(\mathcal{C}) = \max_{\mathcal{S}} \max_{i\in\mathcal{S}} \left\| \left[(H)_{\mathcal{S}}^{-1}\right]_i (H)_{\overline{\mathcal{S}}} \right\|_1,$$

where $\mathcal{S}$ is as in part (i).

(iii) Denoting by $\mathcal{C}_{\min}^{\perp}$ the set of all minimum-weight codewords in $\mathcal{C}^{\perp}$ that have at least one entry equaling 1,

$$\mathsf{h}_r(\mathcal{C}) = -1 + \max_{\boldsymbol{c}\in\mathcal{C}_{\min}^{\perp}} \|\boldsymbol{c}\|_1.$$

[10] Strictly speaking, these particular 1-extremal codewords are negacyclic shifts of the codewords that we get from Corollary 9 and Eq. (28).

Proof. (i)+(ii) When $m = r$ (and $\mathcal{C}$ is MDS), the inner minima in (20) and (21) are over sets of size 1.

(iii) We observe that as $\mathcal{S}$ ranges over all the r-subsets of $[n\rangle$, the rows of $(H)_{\mathcal{S}}^{-1}H$ range over all the minimum-weight codewords in $\mathcal{C}_{\min}^{\perp}$ (a codeword $\boldsymbol{c}$ may appear as a row for multiple r-subsets—one subset for each entry of $\boldsymbol{c}$ that equals 1). □

Remark 10. With the code $\mathcal{C}$ as in Corollary 11, every $(k+1)$-subset $\mathcal{X}$ of $[n\rangle$ defines a one-dimensional subspace of $\mathcal{C}^{\perp}$ whose nonzero elements are minimum-weight codewords of $\mathcal{C}^{\perp}$ with support $\mathcal{X}$ (and, as $\mathcal{X}$ ranges over all $(k+1)$-subsets of $[n\rangle$, we get this way all the minimum-weight codewords of $\mathcal{C}^{\perp}$). Hence,

$$\left|\mathcal{C}_{\min}^{\perp}\right| \le (n-r+1)\cdot\binom{n}{n-r+1} = r\cdot\binom{n}{r}.$$

The right-hand side expression, in turn, is what we get in (22) when $m = r$. □

In the Appendix, we present an alternate proof of Corollary 11(i) that does not rely on the theory of duality of linear programming.

IV. Spherical parity-check codes and their duals

In this section, we consider codes with parity-check matrices H whose columns are all unit vectors, namely $\mathrm{diag}(H^{\top}H) = \mathbf{1}$. We will say that a matrix over $\mathbb{R}$ is *spherical* if it satisfies this property, and a linear code over $\mathbb{R}$ that has such a parity-check matrix will be called a *spherical parity-check* (in short, SPC) code.

The code in Example 1 is an SPC code, and additional SPC codes have been proposed and analyzed in [14] and [22]. One of the constructions therein is a family of linear $[n, n-r]$ codes over $\mathbb{R}$ which are defined by means of a linear $[r, \kappa, \partial]$ code B over $\mathbb{F}_2$ that contains the codeword $\mathbf{1}$ and, in addition, its dual code, $B^{\perp}$, has minimum distance > 2. The induced linear $[n, n-r]$ code, $\mathcal{C}_B$, over $\mathbb{R}$ has length $n = 2^{\kappa-1}$ and an $r \times n$ parity-check matrix H_B whose columns are

$$(1/\sqrt{r})\cdot\left((-1)^{x_j}\right)_{j\in[r\rangle},$$

where $\boldsymbol{x} = (x_j)_{j\in[r\rangle}$ ranges over all the codewords in B that start with a 0. The set of columns of H_B, in turn, forms a spherical code in $\mathbb{R}^r$ whose minimum angle between its elements (i.e., its packing property) increases with ∂ and, consequently, for every $m \le \lceil r/(r-2\partial)\rceil$, the m-height of $\mathcal{C}_B$ decreases with ∂: see Theorem 14 in [14] and Theorems 11 and 13 in [22]. This allows us to speculate that, as a rule of thumb, good packing spherical codes in $\mathbb{R}^r$ yield SPC codes with good (i.e., small) m-heights.

The requirements from the code B imply that the r rows of H_B are orthogonal and have the same L_2 norm (which then has to be $\sqrt{n/r}$, since all the columns in H_B are unit vectors); equivalently, $H_B\,H_B^{\top} = (n/r)\cdot I_r$. We say that an $r \times n$ matrix H is *ortho-spherical* if

$$\mathrm{diag}(H^{\top}H) = \mathbf{1} \quad \text{and} \quad H\,H^{\top} = (n/r)\cdot I_r,$$

and a linear $[n, n-r]$ code over $\mathbb{R}$ that has such a parity-check matrix will be called an *ortho-spherical parity-check* (OSPC) code. The code in Example 1 is, too, an OSPC code. In frame theory and signal processing, (the set of columns of) an $r \times n$ ortho-spherical matrix H is equivalent to a *unit-norm tight frame*, and the scaled matrix $\sqrt{r/n}\cdot H$ is equivalent to an *equal-norm Parseval frame*; see [3],[4],[5],[21].

The next (known) lemma states that the OSPC property is preserved under duality. For completeness, we include a proof in the Appendix.

Lemma 12 ([3, Proposition 3.1(c)])**.** *A linear code is an OSPC code, if and only if its dual code is.*

Letting G be a $k \times n$ ortho-spherical generator matrix of an $[n, k]$ OSPC code $\mathcal{C}$, we have $G\,G^{\top} = (n/k)\cdot I_k$; hence, for every $\boldsymbol{u} \in \mathbb{R}^k$,

$$\|\boldsymbol{u}G\|_2^2 = \boldsymbol{u}G\,G^{\top}\boldsymbol{u}^{\top} = (n/k)\cdot\|\boldsymbol{u}\|_2^2,$$

namely, a (non-systematic) encoder $\boldsymbol{u} \mapsto \sqrt{k/n}\cdot\boldsymbol{u}G$ is norm reserving.

The following two examples present additional OSPC codes.

Example 2. Let $\mathcal{C}_{\text{ico}}$ be the $[6, 3]$ MDS code with the parity-check matrix:

$$H = \frac{1}{\sqrt{\varphi+2}}\begin{pmatrix} 0 & \varphi & 1 & 0 & 1 & -\varphi \\ 1 & 0 & \varphi & 1 & -\varphi & 0 \\ \varphi & 1 & 0 & -\varphi & 0 & 1 \end{pmatrix},$$

where φ stands for the golden ratio $(1+\sqrt{5})/2$. The columns of H range over six out of the 12 vertices—one from each antipodal pair—of a regular icosahedron in $\mathbb{R}^3$ that is inscribed in the unit sphere $\mathbb{S}^2$. Moreover, it is easy to check that $H\,H^{\top} = \sqrt{2}\cdot I_3$, namely, $\mathcal{C}_{\text{ico}}$ is a $[6, 3]$ OSPC code, and so is its dual $\mathcal{C}_{\text{ico}}^{\perp}$. In fact, $\mathcal{C}_{\text{ico}}$ is isometrically self-dual: a generator matrix can be obtained by negating the first three columns of H and then swapping them with the last three columns. Table I presents the m-heights of $\mathcal{C}_{\text{ico}}$ (and of $\mathcal{C}_{\text{ico}}^{\perp}$), as computed by Theorems 5 or 8 (see also Example 5 below). □

TABLE I
m-HEIGHTS OF THE ICOSAHEDRAL CODE $\mathcal{C}_{\text{ico}}$ IN EXAMPLE 2.

m	$\mathsf{h}_m(\mathcal{C}_{\text{ico}})$	approx.
$1, 2$	$\sqrt{5}$	2.2361
3	$2+\sqrt{5}$	4.2361

Example 3. Let $\mathcal{C}_{\text{dod}}$ be the $[10, 7]$ MDS code with the parity-check matrix:

$$H = \frac{1}{\sqrt{3}}\begin{pmatrix} 1 & 1 & 1 & 1 & 0 & \psi & \varphi & 0 & -\psi & \varphi \\ 1 & 1 & -1 & -1 & \psi & \varphi & 0 & -\psi & \varphi & 0 \\ 1 & -1 & 1 & -1 & \varphi & 0 & \psi & \varphi & 0 & -\psi \end{pmatrix},$$

where $\varphi = (1+\sqrt{5})/2$ and $\psi = -1/\varphi = (1-\sqrt{5})/2$. The columns of H range over ten out of the 20 vertices (one from each antipodal pair) of a regular dodecahedron in $\mathbb{R}^3$ that is inscribed in $\mathbb{S}^2$; also, $H\,H^{\top} = \sqrt{10/3}\cdot I_3$ and, so, $\mathcal{C}_{\text{dod}}$ is an OSPC code. Table II presents the m-heights of $\mathcal{C}_{\text{dod}}$ and Table III presents the m-heights of its $[10, 3]$ dual code (see also Example 6 below). □

TABLE II
m-HEIGHTS OF THE DODECAHEDRAL CODE $\mathcal{C}_{\text{dod}}$ IN EXAMPLE 3.

m	$\mathsf{h}_m(\mathcal{C}_{\text{dod}})$	approx.
1	$2+\sqrt{5}$	4.2361
2	$4+\sqrt{5}$	6.2361
3	$9+4\sqrt{5}$	17.9443

TABLE III
m-HEIGHTS OF THE DUAL DODECAHEDRAL CODE $\mathcal{C}_{\text{dod}}^{\perp}$ IN EXAMPLE 3.

m	$\mathsf{h}_m(\mathcal{C}_{\text{dod}}^{\perp})$	approx.
1	$3/\sqrt{5}$	1.3416
2	$(1+\sqrt{5})/2$	1.6180
3	$4-\sqrt{5}$	1.7639
4	3	3.0000
5, 6	$2+\sqrt{5}$	4.2361
7	$5+2\sqrt{5}$	9.4721

Remark 11. In addition to the dodecahedron and the icosahedron, there are three more platonic solids, yet they do not yield interesting constructions: both the tetrahedron and the cube yield the $[4,1]$ repetition code, and the octahedron yields the trivial $[3,0]$ code. □

Remark 12. A construction due to Ruthishauser [16] yields a 20-vertex polyhedron with a minimum angle (between vertices) that is larger than that in the regular dodecahedron. This polyhedron has several symmetries, including closure under antipodality. Let $\mathcal{C}_{\text{Rut}}$ be the linear $[10,7]$ code with a 3×10 parity-check matrix whose columns are obtained by selecting one vertex from each antipodal pair in Ruthishauser's construction. The code $\mathcal{C}_{\text{Rut}}$ turns out to be an OSPC code, yet it is not MDS (neither is its $[10,3]$ dual $\mathcal{C}_{\text{Rut}}^{\perp}$) and, therefore, $\mathsf{h}_3(\mathcal{C}_{\text{Rut}})=\mathsf{h}_7(\mathcal{C}_{\text{Rut}}^{\perp})=\infty$. Still, $\mathsf{h}_1(\mathcal{C}_{\text{Rut}})=1+2\sqrt{2}\approx 3.8284$ and $\mathsf{h}_2(\mathcal{C}_{\text{Rut}})=3+2\sqrt{2}\approx 5.8284$, which are smaller than the respective heights of $\mathcal{C}_{\text{dod}}$, and $\mathsf{h}_4(\mathcal{C}_{\text{Rut}}^{\perp})=\sqrt{2/3}+\sqrt{3}\approx 2.5485$ and $\mathsf{h}_5(\mathcal{C}_{\text{Rut}}^{\perp})=\sqrt{3/2}+\sqrt{3}\approx 2.9567$, which are smaller than the respective heights of $\mathcal{C}_{\text{dod}}^{\perp}$. An improvement to Ruthishauser's construction due to van der Waerden [20] breaks its symmetries, including the closure under antipodality. □

We now turn to consider the duals of SPC codes (including all OSPC codes). Let $\mathcal{C}$ be such a linear $[n,k,d]$ code over $\mathbb{R}$, namely, $\mathcal{C}$ has a spherical $k\times n$ generator matrix $G=(\boldsymbol{g}_j)_{j\in[n\rangle}$. We associate with G the multiset

$$\mathcal{G}=\{\boldsymbol{g}_j\}_{j\in[n\rangle}\cup\{-\boldsymbol{g}_j\}_{j\in[n\rangle}\,, \tag{31}$$

which contains $2n$ vectors (or points) in $\mathbb{S}^{k-1}$ (which are not necessarily distinct). Also, let $m\in[1:d\rangle$.

Consider any codeword $\boldsymbol{c}=(c_j)_{j\in[n\rangle}=\boldsymbol{u}G$ where (without loss of generality) $\boldsymbol{u}\in\mathbb{S}^{k-1}$. Then,

$$c_j=\boldsymbol{u}\cdot\boldsymbol{g}_j,\quad j\in[n\rangle.$$

Letting $\pi(\cdot)$ be a sorting permutation of $\boldsymbol{c}$, we have

$$\mathsf{h}_m(\boldsymbol{c})=\left|\frac{\boldsymbol{u}\cdot\boldsymbol{g}_{\pi(0)}}{\boldsymbol{u}\cdot\boldsymbol{g}_{\pi(m)}}\right|.$$

Thus, to get a small m-height of $\mathcal{C}$, we would like $|\boldsymbol{u}\cdot\boldsymbol{g}_{\pi(m)}|$ to be sufficiently close to $|\boldsymbol{u}\cdot\boldsymbol{g}_{\pi(0)}|$ for every vector $\boldsymbol{u}\in\mathbb{S}^{k-1}$. Yet $|\boldsymbol{u}\cdot\boldsymbol{g}_j|$ is the cosine of the angle between $\boldsymbol{u}$ and either $\boldsymbol{g}_j$ or $-\boldsymbol{g}_j$. This means that for each vector $\boldsymbol{u}\in\mathbb{S}^{k-1}$, we would like to see small disparity among the (Euclidean) distances between $\boldsymbol{u}$ and the $m+1$ points in $\mathcal{G}$ that are the closest to $\boldsymbol{u}$. One possible strategy to achieve this goal is requiring that $\mathcal{G}$ forms a good $(m+1)$-fold θ-covering spherical code[11] in $\mathbb{S}^{k-1}$. Under these conditions, every vector $\boldsymbol{u}\in\mathbb{S}^{k-1}$ is at angular distance $\le\theta$ from at least $m+1$ points in $\mathcal{G}$ and, so,

$$\mathsf{h}_m(\mathcal{C})\le\frac{1}{\cos\theta}.$$

For a linear $[n,k,d]$ code $\mathcal{C}$ whose dual code is an $[n,n-k,d^{\perp}]$ SPC code and for $m\in[1:d\rangle$, Lemma 3 implies[12] the following necessary geometric condition on the m-extremal codewords of $\mathcal{C}$. If $\boldsymbol{c}=\boldsymbol{u}G$ is such a codeword, then there must be a $(d^{\perp}-1)$-subset $\mathcal{G}_{\boldsymbol{u}}\subseteq\mathcal{G}$ such that the dot products $\boldsymbol{u}\cdot\boldsymbol{g}$ are equal for all $\boldsymbol{g}\in\mathcal{G}_{\boldsymbol{u}}$; moreover, the elements in $\mathcal{G}_{\boldsymbol{u}}$ are linearly independent, thereby defining a $(d^{\perp}-2)$-dimensional affine hyperplane $\mathcal{A}_{\boldsymbol{u}}$ in $\mathbb{R}^k$. This means that the point $\boldsymbol{u}$ is equidistant—in Euclidean as well as angular terms—from all the points in $\mathcal{G}_{\boldsymbol{u}}$. Equivalently, $\boldsymbol{u}$ points toward the center of the (unique) hypersphere in $\mathcal{A}_{\boldsymbol{u}}$ that passes through all the points of $\mathcal{G}_{\boldsymbol{u}}$.

Similarly, by Lemma 4, there exists an m-extremal codeword $\boldsymbol{c}=\boldsymbol{u}G$ where $\boldsymbol{u}$ is equidistant from k linearly independent points in $\mathcal{G}$. Here, $\boldsymbol{u}$ points toward the center of the hypersphere in the $(k-1)$-dimensional affine hyperplane that passes through these k points. (Recall that when $\mathcal{C}$ is MDS then $d^{\perp}=k+1$ and so, by Lemma 3, this condition on $\boldsymbol{u}$ must hold for every m-extremal codeword $\boldsymbol{u}G$.)

The next theorem summarizes our previous discussion.

Theorem 13. *Let $\mathcal{C}$ be a linear $[n,k,d]$ code over $\mathbb{R}$ whose dual is an $[n,n-k,d^{\perp}]$ SPC code, let G be a $k\times n$ spherical generator matrix of $\mathcal{C}$, and let $\mathcal{G}$ be defined as in (31). Then the following holds for every $m\in[1:d\rangle$.*

(i) For any m-extremal codeword $\boldsymbol{c}=\boldsymbol{u}G$ in $\mathcal{C}$, the point $\boldsymbol{u}$ (in $\mathbb{R}^k$) is equidistant from at least $d^{\perp}-1$ linearly independent points in $\mathcal{G}$.

(ii) There exists an m-extremal codeword $\boldsymbol{c}=\boldsymbol{u}G$ in $\mathcal{C}$ such that the point $\boldsymbol{u}$ is equidistant from at least k linearly independent points in $\mathcal{G}$.

Example 4. Let $\mathcal{C}^{\perp}(n)$ be the linear $[n,2,n-1]$ code which is the dual of the negacyclic code $\mathcal{C}(n)$ in Example 1, let $G=(\boldsymbol{g}_j)_{j\in[n\rangle}$ be its $2\times n$ generator matrix whose columns are given by (25), and let $\mathcal{G}$ be defined as in (31). The $2n$ points in $\mathcal{G}$ form the vertices of a regular $(2n)$-gon in $\mathbb{R}^2$. By Theorem 13, for any $m\in[1:n-1\rangle$, the point $\boldsymbol{u}\in\mathbb{R}^2$ that corresponds to an m-extremal codeword $\boldsymbol{u}G$ must be

[11] Earlier, we mentioned the relation between the height profile of an SPC code and the *packing* properties of the spherical code that is formed by the columns of its spherical parity-check matrix, which is also the generator matrix of its dual code. Here, when considering the dual code, we refer instead to the *covering* properties of these columns.

[12] Since $\mathcal{C}^{\perp}$ is an SPC code, none of its parity-check matrices has an all-zero column and, therefore, the condition $d^{\perp}\ge 2$ in the lemma is met. For the case $d^{\perp}=2$, the upcoming discussion becomes trivial.

equidistant from two vertices of this polygon. Due to the $(2n)$-fold rotational symmetry of $\mathcal{G}$, the vector $\boldsymbol{u}$ either points toward one of the vertices of $\mathcal{G}$ (say, $\boldsymbol{g}_0$) or toward the midpoint of an edge of the polygon (say, the edge with endpoints $\boldsymbol{g}_0$ and $\boldsymbol{g}_1$).

Letting $\beta = \alpha/2 = \pi/(2n)$, in the first case we get

$$\boldsymbol{c}_1 = \begin{pmatrix} 1 & 0 \end{pmatrix} G = \big(\cos(2j\beta)\big)_{j\in[n\rangle}$$

and, respectively,

$$\mathsf{h}_m(\boldsymbol{c}_1) = \frac{1}{\cos(2\lceil m/2\rceil\beta)}, \quad m \in [n{-}1\rangle. \tag{32}$$

In the second case,

$$\boldsymbol{c}_2 = \begin{pmatrix} \cos\beta & \sin\beta \end{pmatrix} G = \big(\cos((2j{-}1)\beta)\big)_{j\in[n\rangle}$$

and, respectively,

$$\mathsf{h}_m(\boldsymbol{c}_2) = \frac{\cos\beta}{\cos((2\lfloor m/2\rfloor{+}1)\beta)}, \quad m \in [n{-}1\rangle. \tag{33}$$

Hence, when m is odd,

$$\mathsf{h}_m(\boldsymbol{c}_1) \overset{(32)}{=} \frac{1}{\cos((m{+}1)\beta)} > \frac{\cos\beta}{\cos(m\beta)} \overset{(33)}{=} \mathsf{h}_m(\boldsymbol{c}_2)$$

(since $1 > \cos\beta$ and $\cos((m{+}1)\beta) < \cos(m\beta)$), and when m is even,

$$\mathsf{h}_m(\boldsymbol{c}_1) \overset{(32)}{=} \frac{1}{\cos(m\beta)} < \frac{\cos\beta}{\cos((m{+}1)\beta)} \overset{(33)}{=} \mathsf{h}_m(\boldsymbol{c}_2)$$

(since $\cos((m{+}1)\beta) = (\cos\beta)\cos(m\beta) - (\sin\beta)\sin(m\beta) < (\cos\beta)\cos(m\beta)$).

We conclude that $\boldsymbol{c}_1$ (respectively, $\boldsymbol{c}_2$), as well as each of its negacyclic shifts, is m-extremal for any odd (respectively, even) $m \in [n{-}1\rangle$, namely,

$$\mathsf{h}_m(\mathcal{C}^\perp(n)) = \begin{cases} \dfrac{1}{\cos((m{+}1)\beta)}, & \text{when } m \text{ is odd} \\ \dfrac{\cos\beta}{\cos((m{+}1)\beta)}, & \text{when } m \text{ is even.} \end{cases} \tag{34}$$

□

Interestingly, the height profile of the code $\mathcal{C}^\perp(n)$ in the example is related to its primal counterpart in Example 1 by:

$$\mathsf{h}_{n-2}(\mathcal{C}^\perp(n)) = \mathsf{h}_1(\mathcal{C}(n)) + 1.$$

Remark 13. For many values of $m \in [n{-}1\rangle$, the code $\mathcal{C}^\perp(n)$ in Example 4 does not have the smallest m-height among all linear $[n, 2]$ codes. For example, by (possibly puncturing) the construction in [14, §III.A], for any $k \in [2 : n\rangle$ we get linear $[n, k]$ codes whose m-height is 1 for any $m \le n/k - 1$; in particular, their $(\lfloor n/2\rfloor{-}1)$-height is 1 when $k = 2$.

Better codes exist also for larger m. For example, suppose that n is divisible by 4 and let $\tilde{\mathcal{C}}$ be the linear $[n, 2]$ code that is obtained from the construction $\mathcal{C}^\perp(n/2)$ by duplicating every coordinate in the codewords of the latter. For $m = n{-}3$ (which is odd) we have

$$\mathsf{h}_m(\mathcal{C}^\perp(n)) \overset{(34)}{=} \frac{1}{\cos((n{-}2)\pi/(2n))} = \frac{1}{\sin(\pi/n)}. \tag{35}$$

On the other hand, $\lfloor m/2\rfloor = n/2 - 2$ is even and, so,

$$\begin{aligned} \mathsf{h}_m(\tilde{\mathcal{C}}) &= \mathsf{h}_{\lfloor m/2\rfloor}(\mathcal{C}^\perp(n/2)) \\ &\overset{(34)}{=} \frac{\cos(\pi/n)}{\cos((n/2-1)\pi/n)} \\ &= \cot(\pi/n), \end{aligned}$$

which is smaller than (35). □

Example 5. Let $\mathcal{C}^\perp_{\text{ico}}$ be the $[6, 3]$ dual of the MDS code $\mathcal{C}_{\text{ico}}$ in Example 2 ($\mathcal{C}^\perp_{\text{ico}}$ is identical to $\mathcal{C}_{\text{ico}}$ up to permutation and negation of coordinates). Let $G = (\boldsymbol{g}_j)_{j\in[6\rangle}$ be the 3×6 generator matrix of $\mathcal{C}^\perp_{\text{ico}}$ which is given by the ortho-spherical matrix H in that example and let $\mathcal{G}$ be defined as in (31), namely, $\mathcal{G}$ consists of the 12 vertices of a regular icosahedron.

By Theorem 13, any point $\boldsymbol{u} \in \mathbb{R}^3$ that corresponds to an m-extremal codeword $\boldsymbol{u}G$ of $\mathcal{C}^\perp_{\text{ico}}$ must be equidistant from three linearly independent vertices in $\mathcal{G}$; equivalently, the vector $\boldsymbol{u}$ must point toward the circumcenter of a triangle formed by three vertices of $\mathcal{G}$, no two of which are antipodes. There are $\binom{6}{3} \cdot 2^3 = 160$ ways to select such triples, yet we will get the same codeword multiple times. For example, consider some vertex $\pm\boldsymbol{g}_\ell \in \mathcal{G}$; its five neighbors in the icosahedron form a regular pentagon. There are 10 ways to form a triangle when selecting any three of these five neighbors, yet these triangles will all have the same circumcenter (and the same circumscribed circle) and will therefore result in the same codeword (up to scaling). In fact, $\boldsymbol{u}$ in this case will point toward the vertex $\pm\boldsymbol{g}_\ell$ (i.e., it will be a scalar multiple of $\boldsymbol{g}_\ell$). Moreover, we will get the same set of codewords if we consider instead the five neighbors of the antipodal vertex $\mp\boldsymbol{g}_\ell$. After eliminating such replications, we are left with 16 codewords, which are all extremal, and Table IV shows their breakdown according to their m-extremality. When describing the properties of each m-extremal codeword $\boldsymbol{c} = \boldsymbol{u}G$ (last column in the table), we assume that $\boldsymbol{c}$ (or $\boldsymbol{u}$) is scaled so that $|c_{\pi(m)}| = 1$. The respective m-heights are found in Table I. □

Example 6. Let $\mathcal{C}^\perp_{\text{dod}}$ be the $[10, 3]$ dual code of the MDS code $\mathcal{C}_{\text{dod}}$ in Example 3 and let $G = (\boldsymbol{g}_j)_{j\in[10\rangle}$ be the 3×10 generator matrix that is given by the ortho-spherical matrix H in that example. Also, let $\mathcal{G}$ be defined as in (31), namely, $\mathcal{G}$ consists of the 20 vertices of a regular dodecahderon. Here, there are $\binom{10}{3} \cdot 2^3 = 960$ ways to select triples from $\mathcal{G}$ that form nondegenerate triangles, and after pruning replications, we are left with 121 codewords, out of which 30 codewords are not extremal. The breakdown of the remaining 91 codewords, according to their m-extremality, is shown in[13] Table V, and the respective m-heights are found in Table III. □

For the codes $\mathcal{C}^\perp(n)$, $\mathcal{C}^\perp_{\text{ico}}$, and $\mathcal{C}^\perp_{\text{dod}}$ in Examples 4–6, the characterization of the extremal codewords, $\boldsymbol{u}G$, and, in particular, of the height profile of the codes, can also be obtained directly as follows: (a) exploit the symmetry of the geometric objects to limit the search space for the vectors $\boldsymbol{u}$ to a small domain, (b) characterize the order statistics of the

[13] The description in the last column in the table assumes that $|c_{\pi(m)}| = 1$, except when $m = 1$, in which case $|c_{\pi(1)}| = \sqrt{5}$.

TABLE IV
PROPERTIES OF THE m-EXTREMAL CODEWORDS OF THE (DUAL) ICOSAHEDRAL CODE $\mathcal{C}_{\text{ico}}^{\perp}$ IN EXAMPLE 5.

m	# codewords	The vector $\boldsymbol{u}$ points toward —	Properties of $\boldsymbol{c} = \boldsymbol{u}G$
$1, 2$	6	a vertex $\pm\boldsymbol{g}_\ell$ among the 12 vertices of the icosahedron (i.e., $\boldsymbol{u}$ is a scalar multiple of $\boldsymbol{g}_\ell$).	• The entry c_ℓ has magnitude $\sqrt{5}$. • Five entries have magnitude 1 (corresponding to the five neighbors of $\pm\boldsymbol{g}_\ell$ in the icosahedron).
3	10	the circumcenter of a face F among the 20 faces of the icosahedron. It also points toward the circumcenter of the triangle T formed by the three vertices outside F that are on the three faces which are adjacent to F.	• Three entries have magnitude $2 + \sqrt{5}$ (corresponding to the three vertices of the face F). • Three entries have magnitude 1 (corresponding to the three vertices of the triangle T).

TABLE V
PROPERTIES OF THE m-EXTREMAL CODEWORDS OF THE DUAL DODECAHEDRAL CODE $\mathcal{C}_{\text{dod}}^{\perp}$ IN EXAMPLE 6.

m	# codewords	The vector $\boldsymbol{u}$ points toward —	Properties of $\boldsymbol{c} = \boldsymbol{u}G$
$1, 4$	10	a vertex $\pm\boldsymbol{g}_\ell$ among the 20 vertices of the dodecahedron.	• The entry c_ℓ has magnitude 3. • Three entries have magnitude $\sqrt{5}$ (corresponding to the three neighbors of $\pm\boldsymbol{g}_\ell$ in the dodecahedron). • Six entries have magnitude 1 (corresponding to the neighbors of these three neighbors).
$5, 6$	6	the circumcenter of a face F among the 12 faces of the dodecahedron.	• Five entries have magnitude $2+\sqrt{5}$ (corresponding to the five vertices of the face F). • Five entries have magnitude 1 (corresponding to the five neighbors of the vertices of F that are not on F).
2	15	the midpoint of an edge e among the 30 edges of the dodecahedron. As such, $\boldsymbol{u}$ points toward the circumcenter of the square Q formed by the four neighbors of the two endpoints of e, and is also perpendicular to a plane that passes through two pairs of antipodal vertices.	• Two entries have magnitude $\varphi = (1 + \sqrt{5})/2$ (corresponding to the two endpoints of the edge e). • Four entries have magnitude 1 (corresponding to the vertices of the square Q). • Two entries have magnitude $1/\varphi$. • Two entries are zero (i.e., $\boldsymbol{c}$ is a minimum-weight codeword).
3	30	the circumcenter of an isosceles ΔABC with a base BC which is constructed as follows. Select a face F and let $-$A be a vertex of F (this vertex will be the antipode of A). Then B and C are the two (adjacent) vertices of F that are not neighbors of $-$A.	• One entry has magnitude $4 - \sqrt{5}$ (corresponding to the unique vertex that is not a neighbor of B or C on the face that is adjacent to F along BC). • Three entries have magnitude 1. • The magnitudes $-3 + 2\sqrt{5}$, $5 - 2\sqrt{5}$, and $-2 + \sqrt{5}$ appear twice each.
7	30	the circumcenter of an isosceles ΔABC with a base BC which is constructed as follows. Select a face F, let $-$A be a vertex of F, and let D and E be the two neighbors of $-$A on F. Then B (respectively, C) is the unique neighbor of D (respectively, E) that is not on F.	• Two entries have magnitude $5+2\sqrt{5}$ (corresponding to the antipodes of the two remaining vertices of F). • One entry has magnitude $4 + \sqrt{5}$. • Three entries have magnitude 1. • The magnitudes $3 + 2\sqrt{5}$ and $2 + \sqrt{5}$ appear twice each.

entries of the respective codewords $\boldsymbol{u}G$, and (c) determine via critical point analysis the specific vectors $\boldsymbol{u}$ that correspond to the m-extremal codewords. See [24] for more details.

APPENDIX
ADDED PROOFS

Proof of Lemma 6. Write $w = \mathsf{w}(\boldsymbol{b})$. Since the function f remains unchanged under (the same) permutation on the coordinates of $\boldsymbol{a}$ and $\boldsymbol{b}$, we can permute this vectors and assume that $\mathsf{Supp}(\boldsymbol{b}) = [w\rangle$ and that

$$\frac{a_0}{b_0} \le \frac{a_1}{b_1} \le \cdots \le \frac{a_{w-1}}{b_{w-1}}.$$

Writing $\rho_j = a_j/b_j$ when $j \in [w\rangle$, the function f can be written as

$$f(v) = \sum_{j\in[w\rangle} |b_j| \cdot |v - \rho_j| + \sum_{j\in[w:\ell\rangle} |a_j|.$$

The derivative of $f(v)$ at its differentiable points is

$$f'(v) = \sum_{j\in[w\rangle} |b_j| \cdot \operatorname{sgn}(v - \rho_j).$$

Hence, a minimizing v is given by ρ_h, where h is the largest integer in $[w\rangle$ such that

$$\sum_{j\in[h\rangle} |b_j| \le \frac{1}{2} \sum_{j\in[\ell\rangle} |b_j| = \frac{\|\boldsymbol{b}\|_1}{2}.$$

At that minimum,

$$f(\rho_h) = \|\boldsymbol{a} - \rho_h \cdot \boldsymbol{b}\|_1,$$

and $\boldsymbol{a} - \rho_h \cdot \boldsymbol{b}$ is 0 at position h. □

Alternate proof of Corollary 10. Let $\boldsymbol{c}$ be an r-extremal codeword in $\mathcal{C}$ and let $\pi(\cdot)$ be a sorting permutation of $\boldsymbol{c}$; without loss of generality we assume that $|c_{\pi(r)}| = 1$. Denoting

$$\mathcal{S} = \{\pi(t) \,:\, t \in [r\rangle\},$$

we then have (see (2)):

$$\boldsymbol{c} \in \mathcal{C}|_{\overline{\mathcal{S}}}. \tag{36}$$

Since $\mathcal{C}$ is MDS, there is a $k \times n$ generator matrix G of $\mathcal{C}$ such that $(G)_{\overline{\mathcal{S}}}$ is the $k \times k$ identity matrix. Moreover, the remaining columns of G, denoted by $\boldsymbol{g}_j$, $j \in \mathcal{S}$, have no zero entries. From $\boldsymbol{c} = (\boldsymbol{c})_{\overline{\mathcal{S}}} G$ it follows that

$$c_j = (\boldsymbol{c})_{\overline{\mathcal{S}}} \cdot \boldsymbol{g}_j, \quad j \in \mathcal{S}. \tag{37}$$

Suppose now to the contrary that

$$|c_{\pi(n-1)}| < 1 \tag{38}$$

and consider the unique codeword $\boldsymbol{x} = (x_j)_{j \in [n\rangle} \in \mathcal{C}$ defined by

$$(\boldsymbol{x})_{\overline{\mathcal{S}}} = \operatorname{sgn}(\boldsymbol{g}_{\pi(0)}^{\top}) \tag{39}$$

(with $\operatorname{sgn}(\cdot)$ applied componentwise); the remaining entries of $\boldsymbol{x}$ are given by

$$x_j = (\boldsymbol{x})_{\overline{\mathcal{S}}} \boldsymbol{g}_j, \quad j \in \mathcal{S}. \tag{40}$$

Letting $\theta(\cdot)$ be a sorting permutation of $\boldsymbol{x}$, we have:

$$\begin{array}{rcl} |x_{\theta(0)}| & \geq & |x_{\pi(0)}| \\ & \overset{(40)}{=} & \left|(\boldsymbol{x})_{\overline{\mathcal{S}}} \cdot \boldsymbol{g}_{\pi(0)}\right| \\ & \overset{(39)}{=} & \left\|\boldsymbol{g}_{\pi(0)}\right\|_1 \\ & \overset{(36)+(38)}{>} & \left|(\boldsymbol{c})_{\overline{\mathcal{S}}} \cdot \boldsymbol{g}_{\pi(0)}\right| \\ & \overset{(37)}{=} & |c_{\pi(0)}|. \end{array}$$

On the other hand,

$$|x_{\theta(r)}| \leq 1$$

(since there are no more than r entries in $\boldsymbol{x}$ of magnitude greater than 1). We conclude that

$$\mathsf{h}_r(\boldsymbol{x}) = \left|\frac{x_{\theta(0)}}{x_{\theta(r)}}\right| > |c_{\pi(0)}| = \mathsf{h}_r(\boldsymbol{c}),$$

thereby contradicting the fact that $\boldsymbol{c}$ is r-extremal. □

Alternate proof of Corollary 11(i). Let $\mathcal{S}$ and i be maximizer in (30) and let $\boldsymbol{c} = (c_j)_{j \in [n\rangle}$ be the unique codeword in $\mathcal{C}$ (and also in $\mathcal{C}|_{\overline{\mathcal{S}}}$) that is defined by

$$\boldsymbol{c} = (\boldsymbol{c})_{\overline{\mathcal{S}}} (G)_{\overline{\mathcal{S}}}^{-1} G, \tag{41}$$

where

$$(\boldsymbol{c})_{\overline{\mathcal{S}}} = \operatorname{sgn}\left((G)_{\overline{\mathcal{S}}}^{-1} \boldsymbol{g}_i\right)^{\top} \tag{42}$$

(since $\mathcal{C}$ is MDS, the entries of the matrix $\left((G)_{\overline{\mathcal{S}}}^{-1}(G)_{\mathcal{S}}\right)^{\top}$ are all nonzero and, therefore, the entries in $(\boldsymbol{c})_{\overline{\mathcal{S}}}$ are all ± 1). Letting $\pi(\cdot)$ be a sorting permutation of $\boldsymbol{c}$, we have $|c_{\pi(r)}| \leq 1$ (since there are no more than r entries in $\boldsymbol{c}$ of magnitude greater than 1); so,

$$\mathsf{h}_r(\boldsymbol{c}) = \left|\frac{c_{\pi(0)}}{c_{\pi(r)}}\right| \geq |c_{\pi(0)}| \geq c_i \overset{(41)+(42)}{=} \left\|(G)_{\overline{\mathcal{S}}}^{-1} \boldsymbol{g}_i\right\|_1.$$

Hence, the right-hand side of (30) is a lower bound on $\mathsf{h}_r(\mathcal{C})$.

Conversely, let $\boldsymbol{c} = (c_j)_{j \in [n\rangle}$ be r-extremal in $\mathcal{C}$ and let $\pi(\cdot)$ be a sorting permutation of $\boldsymbol{c}$. Denoting

$$\mathcal{S} = \{\pi(t) \,:\, t \in [r\rangle\}$$

and assuming (without loss of generality) that $|c_{\pi(r)}| = 1$, it follows that for $i = \pi(0)$,

$$\mathsf{h}_r(\mathcal{C}) = \mathsf{h}_r(\boldsymbol{c}) = |c_i| \overset{(41)}{=} \left|(\boldsymbol{c})_{\overline{\mathcal{S}}} (G)_{\overline{\mathcal{S}}}^{-1} \boldsymbol{g}_i\right| \leq \left\|(G)_{\overline{\mathcal{S}}}^{-1} \boldsymbol{g}_i\right\|_1,$$

which, in turn, is bounded from above by the right-hand side of (30). □

Proof of Lemma 12. Let $\mathcal{C}$ be an $[n, n-r]$ OSPC code over $\mathbb{R}$ with an ortho-spherical $r \times n$ parity-check H. The r rows of $\sqrt{r/n} \cdot H$ form an orthonormal set, which can be extended by $k = n - r$ vectors into an orthonormal basis of $\mathbb{R}^n$. Let G be the $k \times n$ matrix whose rows are these k vectors, multiplied by $\sqrt{n/k}$. Then $G\,G^{\top} = (n/k) \cdot I_k$ and the $n \times n$ matrix

$$M = \begin{pmatrix} \sqrt{r/n} \cdot H \\ \hline \sqrt{k/n} \cdot G \end{pmatrix}$$

is orthonormal, namely,

$$I_n = M^{\top} M = \frac{r}{n} \cdot H^{\top} H + \frac{k}{n} \cdot G^{\top} G. \tag{43}$$

Looking at the main diagonals of the matrices in (43), we obtain

$$\begin{array}{rcl} \mathbf{1} \;=\; \operatorname{diag}(I_n) & = & \dfrac{r}{n} \cdot \operatorname{diag}(H^{\top} H) + \dfrac{k}{n} \cdot \operatorname{diag}(G^{\top} G) \\ & = & \dfrac{r}{n} \cdot \mathbf{1} + \dfrac{k}{n} \cdot \operatorname{diag}(G^{\top} G), \end{array}$$

namely,

$$\operatorname{diag}(G^{\top} G) = \frac{1}{k}\,(n \cdot \mathbf{1} - r \cdot \mathbf{1}) = \mathbf{1}.$$

□

REFERENCES

[1] A. Ankit, I. El Hajj, S.R. Chalamalasetti, G. Ndu, M. Foltin, R.S. Williams, P. Faraboschi, W.-M. Hwu, J.P. Strachan, K. Roy, D.S. Milojicic, "PUMA: A programmable ultra-efficient memristor-based accelerator for machine learning inference," *Proc. 24th Int'l Conf. Architectural Support for Programming Languages and Operating Systems (ASPLOS 2019),* Providence, RI (2019), 715–731.

[2] P. Bloomfield, W.L. Steiger, *Least Absolute Deviations: Theory, Applications, and Algorithms,* Birkhäuser, 1983.

[3] P.G. Casazza, M. Fickus, D.G. Mixon, J. Peterson, I. Smalyanau, "Every Hilbert space frame has a Naimark complement," *J. Math. Anal. Appl.,* 406 (2013), 111–119.

[4] P.G. Casazza, G. Kutyniok, F. Philipp, "Introduction to finite frame theory," in *Finite Frames: Theory and Applications,* P.G. Casazza, G. Kutyniok (eds.), 1–53, Birkhäuser, 2013.

[5] I. Daubechies, "The wavelet transform, time-frequency localization and signal analysis," *IEEE Trans. Inf. Theory,* 36 (1990), 961–1005.

[6] M. Hu, J.P. Strachan, Z. Li, E.M. Grafals, N. Davila, C. Graves, S. Lam, N. Ge, J. Yang, R.S. Williams, "Dot-product engine for neuromorphic computing: Programming 1T1M crossbar to accelerate matrix-vector multiplication," *Proc. 53rd Annual Design Automation Conference (DAC 2016),* Austin, TX (2016), #19.

[7] M. Hu, C.E. Graves, C. Li, Y. Li, N. Ge, E. Montgomery, N. Davila, H. Jiang, R.S. Williams, J.J. Yang, Q. Xia, J.P. Strachan, "Memristor-based analog computation and neural network classification with a dot product engine," *Adv. Mater.,* 30 (2018), #1705914.

[8] A. Jeffrey, H.-H. Dai, *Handbook of Mathematical Formulas and Integrals,* Fourth Edition, Academic Press, 2008.

[9] A. Jiang, "Analog error-correcting codes: Designs and analysis," *IEEE Trans. Inf. Theory,* 70 (2024), 7740–7756.

[10] Z. Jiang, W. Liu, Z. Huang, B. Bai, G. Zhang, H. Hou, "Tight lower bounds on the single-error detection threshold for analog error-correcting codes," available online at https://arxiv.org/pdf/2605.08973

[11] Z. Jiang, H. Shi, Z. Huang, B. Bai, G. Zhang, H. Hou, "Constructions of analog error-correcting codes for single-error detection and correction with efficient decoding algorithm," *Proc. 2025 IEEE In'l Symp. on Information Theory (ISIT 2025),* Ann Arbor, Michigan (June 2025).

[12] H. Li, Z. Jian, X. Wang, H. Wei, "Sharp bounds and new constructions for single-error detection and correction in analog codes," available online at https://arxiv.org/pdf/2606.03011v2

[13] D.H. Luenberger, *Linear and Nonlinear Programming,* Second Edition, Addison-Wesley, 1984.

[14] R.M. Roth, "Analog error-correcting codes," *IEEE Trans. Inf. Theory,* 66 (2020), 4075–4088 (see correction in 69 (2023), 3793–3794).

[15] R.M. Roth, "Fault-tolerant neuromorphic computing on nanoscale crossbar architectures," *Proc. IEEE Information Theory Workshop,* Mumbai, India (November 2022), 202–207.

[16] H. Ruthishauser, "Über Punktverteilungen auf der Kugelfläche," *Comment. Math. Helv.,* 17 (1945), 327–331.

[17] A. Shafiee, A. Nag, N. Muralimanohar, R. Balasubramonian, J.P. Strachan, M. Hu, R.S. Williams, V. Srikumar, "ISAAC: A convolutional neural network accelerator with in-situ analog arithmetic in crossbars," *Proc. 43rd ACM/IEEE Int'l Symp. on Computer Architecture (ISCA 2016),* Seoul, Korea (2016), 14–26.

[18] W. Song, K. Cai, "Analog error correcting codes with constant redundancy," available online at https://arxiv.org/abs/2603.07117

[19] H. Tsai, S. Ambrogio, P. Narayanan, R.M. Shelby, G.W. Burr, "Recent progress in analog memory-based accelerators for deep learning," *J. Phys. D: Appl. Phys.,* 51 (2018), #283001.

[20] B.L. van der Waerden, "Punkte auf der Kugel. Drei Zusätze," *Math. Ann.,* 125 (1952), 213–222.

[21] S.F.D. Waldron, *An Introduction to Finite Tight Frames,* Birkhäuser, 2018.

[22] H. Wei, R.M. Roth, "Multiple-error-correcting codes for analog computing on resistive crossbars," *IEEE Trans. Inf. Theory,* 70 (2024), 8647–8658.

[23] C. Yuan, A. Jiang, R.M. Roth, Z. Zhu, P.H. Siegel, "New results on analog error-correcting codes," available online at https://www.anxiao-jiang.com/publications/on-ai-and-deep-learning (paper #47).

[24] Z. Zhu, C. Yuan, R.M. Roth, P.H. Siegel, A. Jiang, "A new class of geometric analog error correction codes for crossbar based in-memory computing," available online at https://arxiv.org/abs/2603.03723