Parity-check codes and their representations

Gretchen L. Matthews

July 21, 2015

CLEMSON MATHEMATICAL SCIENCES

This work is supported by NSF DMS-0901693, NSA MSP-111006, NSF DMS-1403062.

Linear codes can be represented by parity-check matrices.

Let $\mathbb{F}_q$ be the finite field with q elements, where q is a power of a prime.
A linear code C of length n over $\mathbb{F}_q$ is a subspace of $\mathbb{F}_q^n$.
Elements of C are called codewords.

Linear codes can be represented by parity-check matrices.

Let $\mathbb{F}_q$ be the finite field with q elements, where q is a power of a prime.
A linear code C of length n over $\mathbb{F}_q$ is a subspace of $\mathbb{F}_q^n$.
Elements of C are called codewords.

All codes considered here are linear codes.

Linear codes can be represented by parity-check matrices.

Let $\mathbb{F}_q$ be the finite field with q elements, where q is a power of a prime.
A linear code C of length n over $\mathbb{F}_q$ is a subspace of $\mathbb{F}_q^n$.
Elements of C are called codewords.

All codes considered here are linear codes.

Given a code C of length n, there exists a matrix $H \in \mathbb{F}_q^{r \times n}$ for some r whose nullspace is C, meaning

$$C = NS(H).$$

Linear codes can be represented by parity-check matrices.

Let $\mathbb{F}_q$ be the finite field with q elements, where q is a power of a prime.
A linear code C of length n over $\mathbb{F}_q$ is a subspace of $\mathbb{F}_q^n$.
Elements of C are called codewords.

All codes considered here are linear codes.

Given a code C of length n, there exists a matrix $H \in \mathbb{F}_q^{r \times n}$ for some r whose nullspace is C, meaning

$$C = NS(H).$$

The matrix H is called a parity-check matrix for C, because

$$y \in C \text{ if and only if } Hy^T = 0.$$

The code C is sometimes called a parity-check code. If H is sparse, then C is a low-density parity-check (LDPC) code.

A parity-check matrix for a linear code is not unique.

Example

Consider the binary code

$$C = \left\{ \begin{array}{l} (0,0,0,0,0,0,0),(0,0,1,0,1,1,1),(0,1,0,1,0,1,1),(0,1,1,1,1,0,0), \\ (1,0,0,1,1,0,1),(1,0,1,1,0,1,0),(1,1,0,0,1,1,0),(1,1,1,0,0,0,1) \end{array} \right\} \subseteq \mathbb{F}_2^7.$$

A parity-check matrix for a linear code is not unique.

Example

Consider the binary code

$$C = \left\{ \begin{array}{l} (0,0,0,0,0,0,0),(0,0,1,0,1,1,1),(0,1,0,1,0,1,1),(0,1,1,1,1,0,0), \\ (1,0,0,1,1,0,1),(1,0,1,1,0,1,0),(1,1,0,0,1,1,0),(1,1,1,0,0,0,1) \end{array} \right\} \subseteq \mathbb{F}_2^7.$$

Then C has

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 \end{pmatrix}$$

as a parity-check matrix.

A parity-check matrix for a linear code is not unique.

Example

Consider the binary code

$$C = \left\{ \begin{array}{l} (0,0,0,0,0,0,0),(0,0,1,0,1,1,1),(0,1,0,1,0,1,1),(0,1,1,1,1,0,0), \\ (1,0,0,1,1,0,1),(1,0,1,1,0,1,0),(1,1,0,0,1,1,0),(1,1,1,0,0,0,1) \end{array} \right\} \subseteq \mathbb{F}_2^7.$$

Then C has

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 \end{pmatrix}$$

as a parity-check matrix. Another parity-check matrix for C is

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{pmatrix}.$$

A code can be represented by a Tanner graph.

The Tanner graph of a binary matrix H is a bipartite graph $T(H) = (X \cup F, E)$ with biadjacency matrix H, meaning

-
-
-

Example

$$H = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}$$

A code can be represented by a Tanner graph.

The Tanner graph of a binary matrix H is a bipartite graph $T(H) = (X \cup F, E)$ with biadjacency matrix H, meaning

- vertices in X correspond to codeword coordinates, called bit nodes,
-
-

Example

bit node

$$H = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}$$

A code can be represented by a Tanner graph.

The Tanner graph of a binary matrix H is a bipartite graph $T(H) = (X \cup F, E)$ with biadjacency matrix H, meaning

- vertices in X correspond to codeword coordinates, called bit nodes,
- vertices in F correspond to parity checks, called check nodes, and
-

Example

check node bit node

$$H = \begin{array}{c} \\ \boxplus \\ \boxplus \\ \boxplus \\ \boxplus \\ \boxplus \\ \boxplus \end{array} \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}$$

A code can be represented by a Tanner graph.

The Tanner graph of a binary matrix H is a bipartite graph $T(H) = (X \cup F, E)$ with biadjacency matrix H, meaning

- vertices in X correspond to codeword coordinates, called bit nodes,
- vertices in F correspond to parity checks, called check nodes, and
- $\{x_i, f_j\} \in E$ if and only if $h_{ji} = 1$.

Example

check node bit node

$$H = \begin{array}{c} \\ \boxplus \\ \boxplus \\ \boxplus \\ \boxplus \\ \boxplus \\ \boxplus \end{array} \begin{array}{c} \begin{array}{ccccccc} \circ & \circ & \circ & \circ & \circ & \circ & \circ \end{array} \\ \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix} \end{array}$$

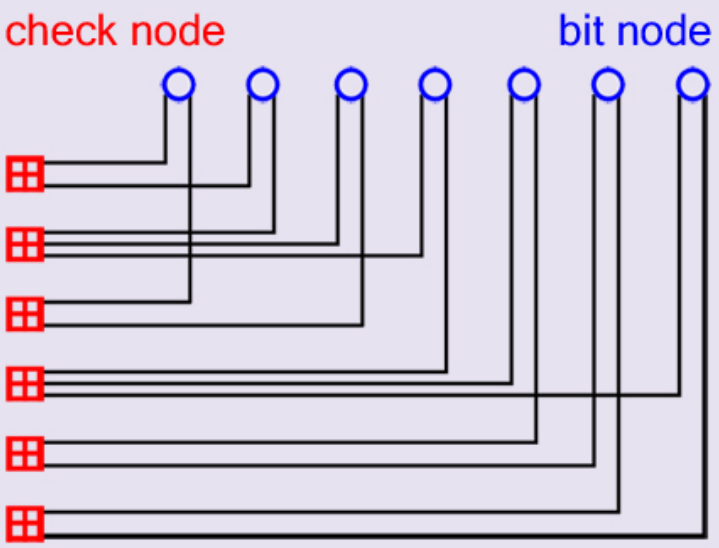
A code can be represented by a Tanner graph.

The Tanner graph of a binary matrix H is a bipartite graph $T(H) = (X \cup F, E)$ with biadjacency matrix H, meaning

- vertices in X correspond to codeword coordinates, called bit nodes,
- vertices in F correspond to parity checks, called check nodes, and
- $\{x_i, f_j\} \in E$ if and only if $h_{ji} = 1$.

Example

Tanner graph of H :



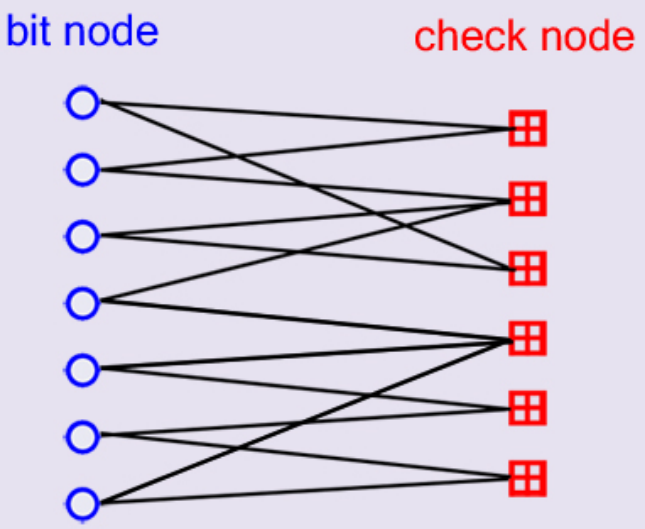
A code can be represented by a Tanner graph.

The Tanner graph of a binary matrix H is a bipartite graph $T(H) = (X \cup F, E)$ with biadjacency matrix H, meaning

- vertices in X correspond to codeword coordinates, called bit nodes,
- vertices in F correspond to parity checks, called check nodes, and
- $\{x_i, f_j\} \in E$ if and only if $h_{ji} = 1$.

Example

Tanner graph of H :



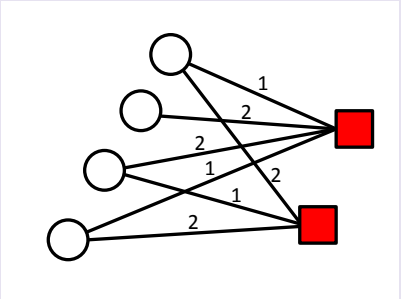
Tanner graphs of q-ary codes are weighted bipartite graphs.

The Tanner graph $T(H)$ of $H \in \mathbb{F}_q^{r \times n}$ is a weighted bipartite graph with biadjacency matrix H; that is,

- the vertex set of $T(H)$ is $\{x_1, \ldots, x_n\} \cup \{f_1, \ldots, f_r\}$,
- the edge set of $T(H)$ is $\{\{x_i, f_j\} : h_{ji} \neq 0\}$, and
- edge weights are given by $wt\{x_i, f_j\} = h_{ji}$.

Example

The Tanner graph of $H = \begin{bmatrix} 1 & 2 & 2 & 1 \\ 2 & 0 & 1 & 2 \end{bmatrix}$ over $\mathbb{F}_3$ is



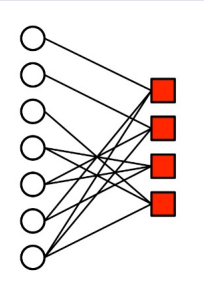
A code may have more than one Tanner graph.

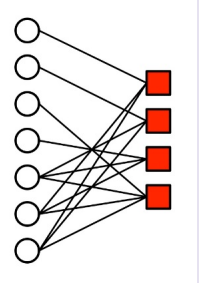
The Tanner graph of a code depends on the choice of parity-check matrix.

Example

Tanner graphs $T(H_1)$ and $T(H_2)$ for the same binary code are shown below.

$$H_1 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 \end{pmatrix} \quad H_2 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{pmatrix}$$





To emphasize that a code C is being considered with a fixed parity-check matrix H, we write

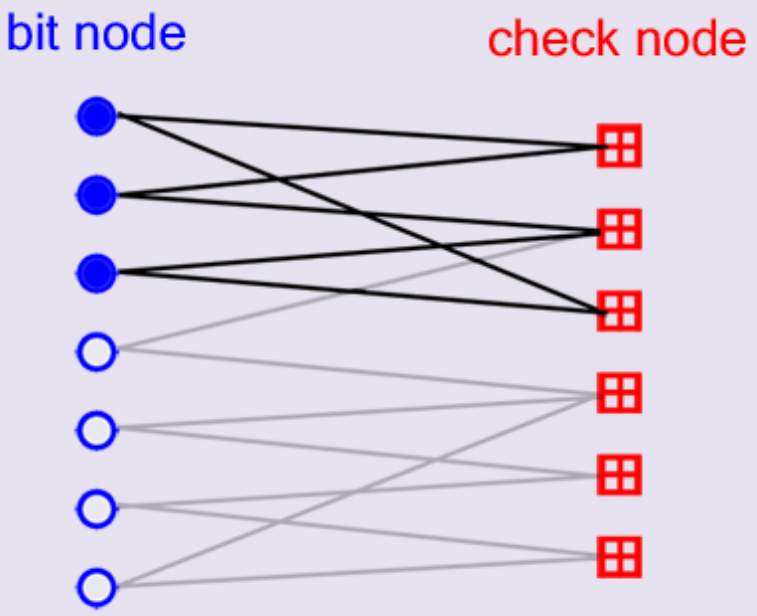
$$C = C(H).$$

Codewords may be represented graphically.

An element $c = (c_1, c_2, \ldots, c_n)$ is a codeword of a binary code $C = C(H)$ if and only if the binary assignment $(c_1, c_2, \ldots, c_n)$ to the bit nodes of $T(H)$ make the binary sum at every check node 0.

Example

$(1, 1, 1, 0, 0, 0, 0)$
corresponds to

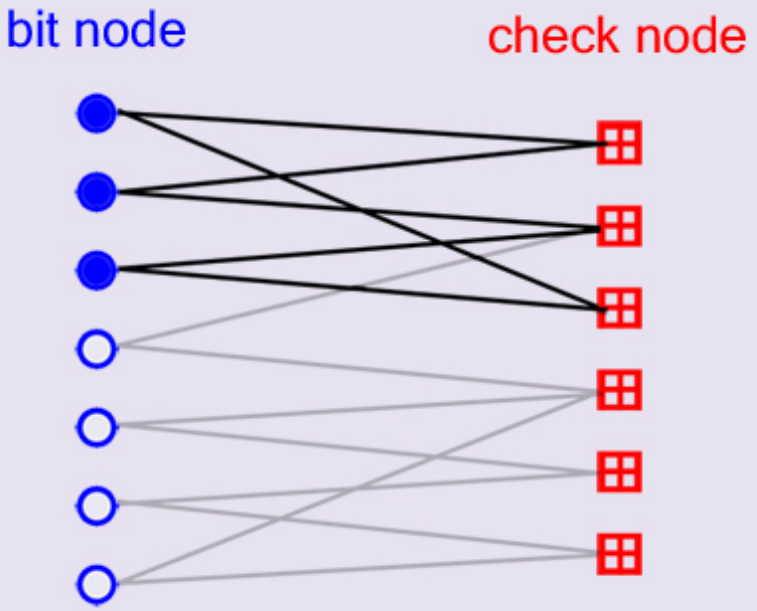


Codewords may be represented graphically.

An element $c = (c_1, c_2, \ldots, c_n)$ is a codeword of a binary code $C = C(H)$ if and only if the binary assignment $(c_1, c_2, \ldots, c_n)$ to the bit nodes of $T(H)$ make the binary sum at every check node 0.

Example

$(1, 1, 1, 0, 0, 0, 0)$
corresponds to



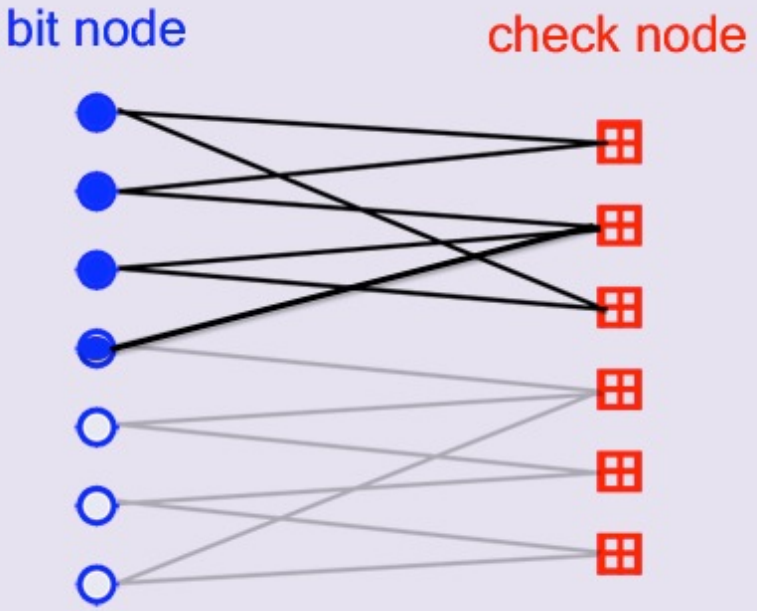
Thus, $(1, 1, 1, 0, 0, 0, 0)$ is a codeword.

Codewords may be represented graphically.

An element $c = (c_1, c_2, \ldots, c_n)$ is a codeword of a binary code $C = C(H)$ if and only if the binary assignment $(c_1, c_2, \ldots, c_n)$ to the bit nodes of $T(H)$ make the binary sum at every check node 0.

Example

$(1, 1, 1, 1, 0, 0, 0)$
corresponds to

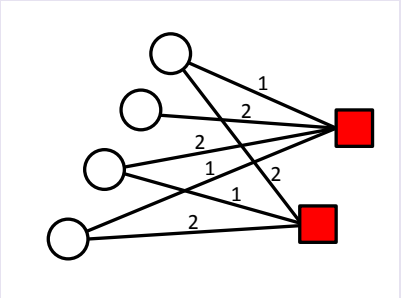


Thus, $(1, 1, 1, 0, 0, 0, 0)$ is a codeword.
However, $(1, 1, 1, 1, 0, 0, 0)$ is not a codeword.

Codewords may be represented graphically.

Example

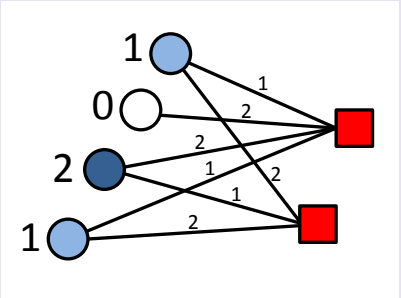
The Tanner graph of $H = \begin{bmatrix} 1 & 2 & 2 & 1 \\ 2 & 0 & 1 & 2 \end{bmatrix}$ over $\mathbb{F}_3$ is



Codewords may be represented graphically.

Example

The Tanner graph of $H = \begin{bmatrix} 1 & 2 & 2 & 1 \\ 2 & 0 & 1 & 2 \end{bmatrix}$ over $\mathbb{F}_3$ is



Note that $(1, 0, 2, 1)$ is a codeword of the code defined by H.

Tanner graphs have played various roles in coding theory.

The Tanner graph was introduced in 1981 as a tool for

- the recursive construction of codes and
- providing bounds on code parameters.

Tanner graphs have played various roles in coding theory.

The Tanner graph was introduced in 1981 as a tool for

- the recursive construction of codes and
- providing bounds on code parameters.

The Hamming distance between two words $w, y \in \mathbb{F}_q^n$ is

$$d(w, y) := \#\{i : w_i \neq y_i\}.$$

Code parameters

Classically, the quality of a code C is measured by parameters

- n, the length of the code C
- $k := \dim_{\mathbb{F}_q} C$, the dimension of the code C
- $d := \min\{d(c, c') : c, c' \in C, c \neq c'\}$, the minimum distance of C

A code C with these parameters is called an $[n, k, d]$ code.

Given a received word, a decoder seeks to find the most likely codeword sent.

Minimum distance decoding problem

Given a received word $w \in \mathbb{F}_q^n$, find $y \in C$ such that

$$d(w, y) \leq d(w, c) \;\; \forall c \in C.$$

Given a received word, a decoder seeks to find the most likely codeword sent.

Minimum distance decoding problem

Given a received word $w \in \mathbb{F}_q^n$, find $y \in C$ such that

$$d(w, y) \leq d(w, c) \;\; \forall c \in C.$$

Maximum likelihood (ML) decoding problem

Given a received word $w \in \mathbb{F}_q^n$, find $y \in C$ that maximizes

$$P\left(y \textit{ sent} | w \textit{ received}\right),$$

the probability that the codeword y was sent given that w is received.

Given a received word, a decoder seeks to find the most likely codeword sent.

Minimum distance decoding problem

Given a received word $w \in \mathbb{F}_q^n$, find $y \in C$ such that

$$d(w, y) \leq d(w, c) \;\; \forall c \in C.$$

Maximum likelihood (ML) decoding problem

Given a received word $w \in \mathbb{F}_q^n$, find $y \in C$ that maximizes

$$P\left(y \textit{ sent} | w \textit{ received}\right),$$

the probability that the codeword y was sent given that w is received.

Maximum likelihood decoding is equivalent to minimum distance decoding (provided the channel has error probability $< .5$).

The ML decoding problem can be stated as an LP.

Definition

Given a binary code C of length n,

$$poly(C) := \left\{ \sum_{\mathbf{y} \in C} \lambda_{\mathbf{y}} \mathbf{y} : \lambda_{\mathbf{y}} \geq 0, \sum_{y \in C} \lambda_y = 1 \right\} \subseteq [0,1]^n$$

denotes the codeword polytope of C.

The ML decoding problem can be stated as an LP.

Definition

Given a binary code C of length n,

$$poly(C) := \left\{ \sum_{\mathbf{y} \in C} \lambda_{\mathbf{y}} \mathbf{y} : \lambda_{\mathbf{y}} \geq 0, \sum_{y \in C} \lambda_y = 1 \right\} \subseteq [0,1]^n$$

denotes the codeword polytope of C.

Let

$$\gamma_i := log \left(\frac{P\left(w_i | y_i = 0\right)}{P\left(w_i | y_i = 1\right)} \right)$$

denote the negative log-likelihood ratio at the ith coordinate.

The ML decoding problem is equivalent to the following linear program (LP):

$$\text{minimize} \sum_{i=1}^{n} \gamma_i f_i \text{ subject to } f \in poly(C).$$

The ML decoding problem can be stated as an LP.

Write $\mathbb{F}_q = \{0, \alpha, \ldots, \alpha^{q-1}\}$, and define $\begin{array}{rccc} \phi: & \mathbb{F}_q^n & \to & \{0,1\}^{(q-1)n} \\ & x & \mapsto & (\mathbf{e}_{x_1}, \mathbf{e}_{x_2}, \ldots, \mathbf{e}_{x_n}). \end{array}$

Definition

Given a code C of length n over $\mathbb{F}_q$, the codeword polytope of C is

$$poly(C) := \left\{ \sum_{c \in C} \lambda_c \phi(c) : \lambda_c \geq 0, \sum_{c \in C} \lambda_c = 1 \right\} \subseteq [0,1]^{(q-1)n}.$$

The ML decoding problem can be stated as an LP.

Write $\mathbb{F}_q = \{0, \alpha, \ldots, \alpha^{q-1}\}$, and define $\begin{array}{rccc} \phi: & \mathbb{F}_q^n & \to & \{0,1\}^{(q-1)n} \\ & x & \mapsto & (\mathbf{e}_{x_1}, \mathbf{e}_{x_2}, \ldots, \mathbf{e}_{x_n}). \end{array}$

Definition

Given a code C of length n over $\mathbb{F}_q$, the codeword polytope of C is

$$poly(C) := \left\{ \sum_{c \in C} \lambda_c \phi(c) : \lambda_c \geq 0, \sum_{c \in C} \lambda_c = 1 \right\} \subseteq [0,1]^{(q-1)n}.$$

The ML decoding problem for C is equivalent to the following LP:

$$\text{minimize} \sum_{k=1}^{(q-1)n} \gamma_k f_k \text{ subject to } f \in poly(C),$$

where $\gamma_{(q-1)(i-1)+j} := \log\left(\frac{P(y_i|0)}{P(y_i|\alpha^j)}\right)$ is a log-likelihood ratio at the ith coordinate and $j = 1, \ldots, q-1$.

LP relaxation of ML decoding is more managable.

Definition

Suppose $C = C(H)$ be a code over $\mathbb{F}_q$ of length n and $Row_j(H)$ denote the jth row of H. The fundamental polytope of H is

$$Q(H) = \cap_{j=1}^{r} poly\left(C\left(Row_j(H)\right)\right) \subseteq [0,1]^{(q-1)n}.$$

LP relaxation of ML decoding is more managable.

Definition

Suppose $C = C(H)$ be a code over $\mathbb{F}_q$ of length n and $Row_j(H)$ denote the jth row of H. The fundamental polytope of H is

$$Q(H) = \cap_{j=1}^{r} poly\left(C\left(Row_j(H)\right)\right) \subseteq [0,1]^{(q-1)n}.$$

Example

Consider again

$$H_1 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 \end{pmatrix}.$$

LP relaxation of ML decoding is more managable.

Definition

Suppose $C = C(H)$ be a code over $\mathbb{F}_q$ of length n and $Row_j(H)$ denote the jth row of H. The fundamental polytope of H is

$$Q(H) = \cap_{j=1}^{r} poly\left(C\left(Row_j(H)\right)\right) \subseteq [0,1]^{(q-1)n}.$$

LP relaxation of ML decoding is more managable.

Definition

Suppose $C = C(H)$ be a code over $\mathbb{F}_q$ of length n and $Row_j(H)$ denote the jth row of H. The fundamental polytope of H is

$$Q(H) = \cap_{j=1}^{r} poly\left(C\left(Row_j(H)\right)\right) \subseteq [0,1]^{(q-1)n}.$$

Note: $poly(C) \subseteq Q(H)$.

LP relaxation of ML decoding is more managable.

Definition

Suppose $C = C(H)$ be a code over $\mathbb{F}_q$ of length n and $Row_j(H)$ denote the jth row of H. The fundamental polytope of H is

$$Q(H) = \cap_{j=1}^{r} poly\left(C\left(Row_j(H)\right)\right) \subseteq [0,1]^{(q-1)n}.$$

Note: $poly(C) \subseteq Q(H)$.

Linear Code Linear Program [Feldman, Wainwright, & Karger, 2005; Flanagan, Skachek, Byrne, & Greferath, 2009]

The Linear Code LP is

$$\text{minimize} \sum_{i=1}^{(q-1)n} \gamma_i f_i \text{ subject to } f \in Q(H).$$

LP relaxation of ML decoding is more managable.

Definition

Suppose $C = C(H)$ be a code over $\mathbb{F}_q$ of length n and $Row_j(H)$ denote the jth row of H. The fundamental polytope of H is

$$Q(H) = \cap_{j=1}^{r} poly\left(C\left(Row_j(H)\right)\right) \subseteq [0,1]^{(q-1)n}.$$

Note: $poly(C) \subseteq Q(H)$.

Linear Code Linear Program [Feldman, Wainwright, & Karger, 2005; Flanagan, Skachek, Byrne, & Greferath, 2009]

The Linear Code LP is

$$\text{minimize} \sum_{i=1}^{(q-1)n} \gamma_i f_i \text{ subject to } f \in Q(H).$$

Definition

The vertices of $Q(H)$ are called pseudocodewords.

LP relaxation may lead to a noncodeword output, called a pseudocodeword.

Some, but not necessarily all, pseudocodewords are codewords.

LP relaxation may lead to a noncodeword output, called a pseudocodeword.

Some, but not necessarily all, pseudocodewords are codewords.

If $T(H)$ is acyclic, then the Linear Code LP and ML decoding for $C(H)$ are equivalent.

LP relaxation may lead to a noncodeword output, called a pseudocodeword.

Some, but not necessarily all, pseudocodewords are codewords.

If $T(H)$ is acyclic, then the Linear Code LP and ML decoding for $C(H)$ are equivalent.

Theorem [Etzion, Trachtenberg, & Vardy, 1999]

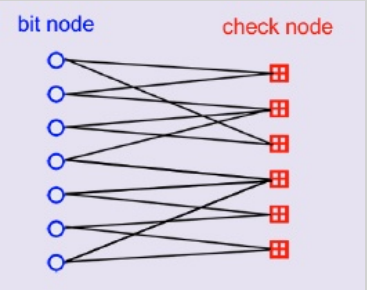
Given an $[n, k, d]$ code with acyclic Tanner graph, either $\frac{k}{n} \leq 0.5$ or

$$\frac{k}{n} > 0.5 \text{ and } d = 2.$$

Pseudocodewords can also be defined via graph covers.

Example

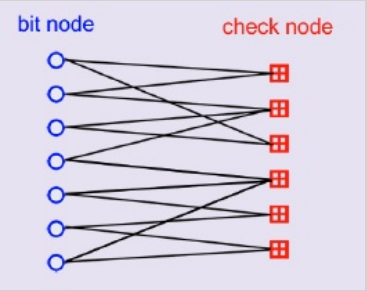
Consider the Tanner graph $T(H)$:



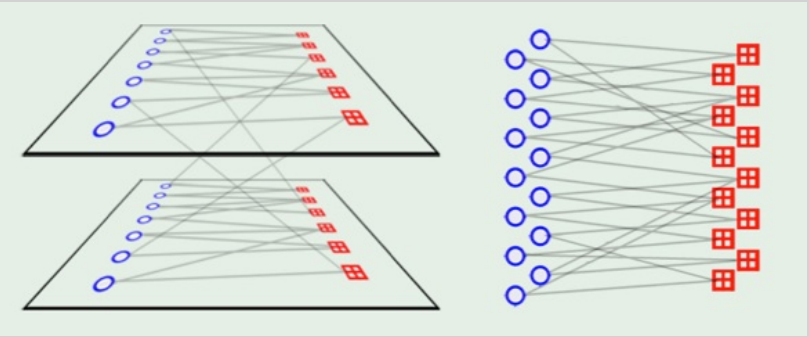
Pseudocodewords can also be defined via graph covers.

Example

Consider the Tanner graph $T(H)$:



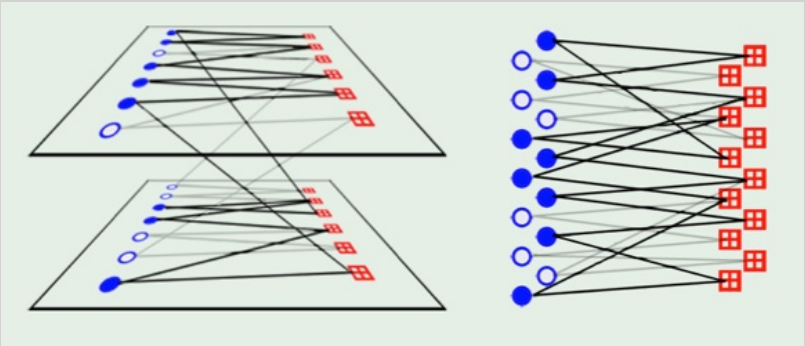
A degree 2 cover of $T(H)$ is



Pseudocodewords can also be defined via graph covers.

Example

The degree 2 cover gives rise to codewords:

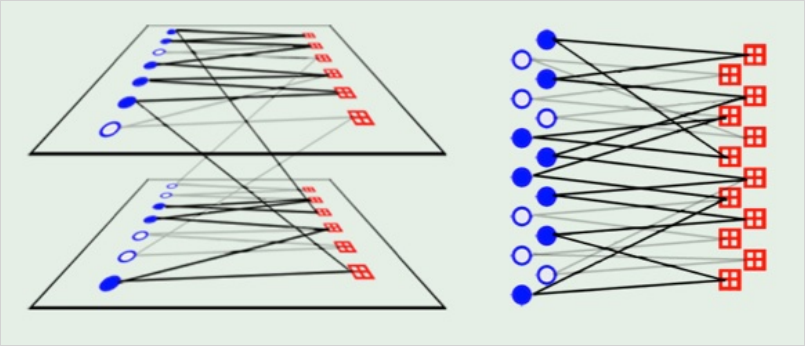


In this case, we obtain $\frac{1}{2}(1,1,1,2,1,1,1)$.

Pseudocodewords can also be defined via graph covers.

Example

The degree 2 cover gives rise to codewords:



Consider

$$\frac{1}{M}(m_1(1), \ldots, m_n(1)) \in [0,1]^n$$

where $m_i(b)$ is the number of copies of the ith symbol that take the value $b \in \mathbb{F}_2$ in a codeword c from $C(\tilde{T}(H))$
In this case, we obtain $\frac{1}{2}(1,1,1,2,1,1,1)$.

Pseudocodewords can also be defined via graph covers.

Let $C = C(H)$ be a code over $\mathbb{F}_q$ of length n.
Pseudocodewords from a degree M graph cover $\tilde{T}(H)$ of $T(H)$ are of the form

$$\frac{1}{M}(m_1(\alpha), \ldots, m_n(\alpha), \ldots, m_1(\alpha^{q-1}), \ldots, m_n(\alpha^{q-1})) \in [0,1]^{(q-1)n}$$

where $m_i(b)$ is the number of copies of the ith symbol that take the value $b \in \mathbb{F}_q$ in a codeword c from $C(\tilde{T}(H))$.

Pseudocodewords can also be defined via graph covers.

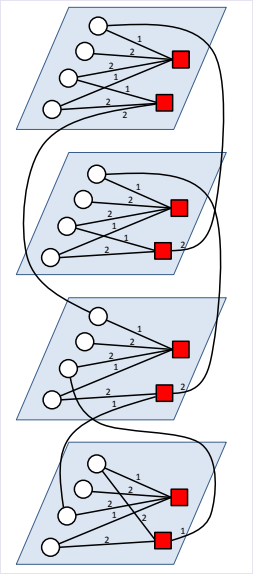
Example

The ternary code $C(H)$ with $H = \begin{bmatrix} 1 & 2 & 2 & 1 \\ 2 & 0 & 1 & 2 \end{bmatrix} \in \mathbb{F}_3^{2\times 4}$

Pseudocodewords can also be defined via graph covers.

Example

The ternary code $C(H)$ with $H = \begin{bmatrix} 1 & 2 & 2 & 1 \\ 2 & 0 & 1 & 2 \end{bmatrix} \in \mathbb{F}_3^{2\times 4}$ has a degree 4 graph cover of $T(H)$

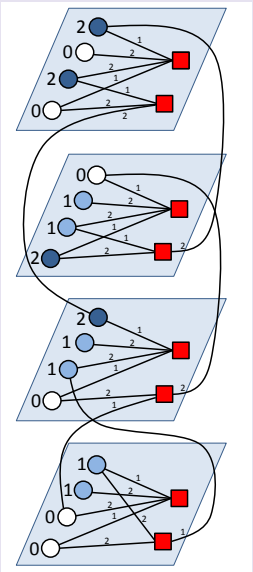


Pseudocodewords can also be defined via graph covers.

Example

The ternary code $C(H)$ with $H = \begin{bmatrix} 1 & 2 & 2 & 1 \\ 2 & 0 & 1 & 2 \end{bmatrix} \in \mathbb{F}_3^{2\times 4}$ has a degree 4 graph cover of $T(H)$ which has a codeword

$$\tilde{c} = (2, 0, 2, 1, 0, 1, 1, 1, 2, 1, 1, 0, 0, 2, 0, 0)$$

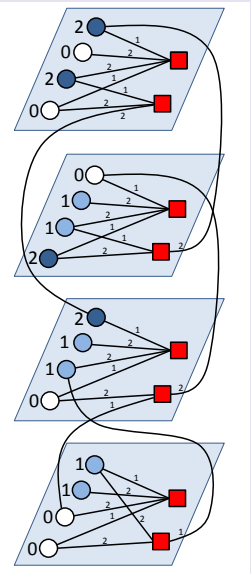


Pseudocodewords can also be defined via graph covers.

Example

The ternary code $C(H)$ with $H = \begin{bmatrix} 1 & 2 & 2 & 1 \\ 2 & 0 & 1 & 2 \end{bmatrix} \in \mathbb{F}_3^{2\times 4}$ has a degree 4 graph cover of $T(H)$ which has a codeword

$$\tilde{c} = (2, 0, 2, 1, 0, 1, 1, 1, 2, 1, 1, 0, 0, 2, 0, 0)$$



that gives rise to the pseudocodeword $\left(\frac{1}{4}, \frac{3}{4}, \frac{2}{2}, 0, \frac{2}{2}, 0, \frac{1}{4}, \frac{1}{4}\right)$.

LP decoding gives an approximation to iterative decoding algorithms.

Iterative decoders make use of local computations, while ML decoders search for a configuration that satisfies all check nodes.

Advantages of an iterative decoder:

- Can correct more errors than guaranteed by ML decoding
- Does not require any particular structure
- Fast (linear in code length for sparse matrices) and easy to implement

LP decoding gives an approximation to iterative decoding algorithms.

Iterative decoders make use of local computations, while ML decoders search for a configuration that satisfies all check nodes.

Advantages of an iterative decoder:

- Can correct more errors than guaranteed by ML decoding
- Does not require any particular structure
- Fast (linear in code length for sparse matrices) and easy to implement

Disadvantages of an iterative decoder:

- Decoding performance depends on parity-check matrix
- Algorithm may converge to a noncodeword output (pseudocodeword)

LP decoding gives an approximation to iterative decoding algorithms.

Iterative decoders make use of local computations, while ML decoders search for a configuration that satisfies all check nodes.

Advantages of an iterative decoder:

- Can correct more errors than guaranteed by ML decoding
- Does not require any particular structure
- Fast (linear in code length for sparse matrices) and easy to implement

Disadvantages of an iterative decoder:

- Decoding performance depends on parity-check matrix
- Algorithm may converge to a noncodeword output (pseudocodeword)

Aim

To provide tools for determining the expected performance of a parity-check code and for selecting parity-check matrices which lessen the impact of noncodeword pseudocodewords.

Two tools for studying pseudocodewords: generating functions and the fundamental cone.

Notation

Given a matrix H, let $\mathcal{P}(H)$ denote the set of pseudocodewords of $C(H)$

Generating function of the pseudocodewords

Given a code $C = C(H)$, we consider

$$\sum_{p \in \mathcal{P}(H)} x^p.$$

Two tools for studying pseudocodewords: generating functions and the fundamental cone.

Notation

Given a matrix H, let $\mathcal{P}(H)$ denote the set of pseudocodewords of $C(H)$

Generating function of the pseudocodewords

Given a code $C = C(H)$, we consider

$$\sum_{p \in \mathcal{P}(H)} x^p.$$

Fundamental cone

The fundamental cone $\mathcal{K}(H)$ of a code $C(H)$ of length n over $\mathbb{F}_q$ is the smallest cone in $\mathbb{R}^{(q-1)n}$ that contains all pseudocodewords of $C(H)$.

Two tools for studying pseudocodewords: generating functions and the fundamental cone.

Theorem (Koetter, Li, Vontobel, & Walker 2007)

Let $H \in \mathbb{F}_2^{r \times n}$. Given $\mathbf{p} \in \mathbb{Z}^n$, the following are equivalent:

1. $\mathbf{p}$ is a psuedocodeword of $C(H)$ and
2. $\mathbf{p} \in \mathcal{K}(H)$ and $H\mathbf{p}^T = \mathbf{0} \in \mathbb{F}_2^r$.

Theorem (Koetter, Li, Vontobel, & Walker 2007)

Suppose $H \in \mathbb{F}_2^{r \times n}$ is a matrix with exactly two 1's in each column. Then the generating function of the pseudocodewords of $C(H)$ is

$$\sum_{\mathbf{p} \in \mathcal{P}(H)} \mathbf{x}^{\mathbf{p}} = \zeta_{N(H)}(\mathbf{x}),$$

the edge zeta function of the normal graph of $T(H)$.

The generating function for the pseudocodewords of a code is a rational function.

Theorem (K-M)

Given a binary matrix $H \in \mathbb{F}_2^{r \times n}$,

- *the generating function for the pseudocodewords of $C(H)$, $\sum_{p \in \mathcal{P}(H)} x^p$, is a rational function; and*

The generating function for the pseudocodewords of a code is a rational function.

Theorem (K-M)

Given a binary matrix $H \in \mathbb{F}_2^{r \times n}$,

- *the generating function for the pseudocodewords of $C(H)$, $\sum_{p \in \mathcal{P}(H)} x^p$, is a rational function; and*
- *there exists a polynomial time algorithm which computes the generating function of the pseudocodewords, $\sum_{\mathbf{p} \in \mathcal{P}(H)} \mathbf{x}^{\mathbf{p}}$, as a finite sum*

$$\sum_i \alpha_i \frac{\mathbf{x}^{\mathbf{v}_i}}{(1 - \mathbf{x}^{\mathbf{w}_{i1}}) \dots (1 - \mathbf{x}^{\mathbf{w}_{id}})}$$

where $\alpha_i \in \{0, 1\}$ and $\mathbf{u}_i$, $\mathbf{w}_{ij}$ are integer vectors for all i, j.

Barvinok's algorithm enumerates integer points of a rational cone.

Barvinok's algorithm takes as input a pointed rational cone described in terms of rational inequalities and produces a generating function that enumerates its integer points.

Barvinok's algorithm enumerates integer points of a rational cone.

Barvinok's algorithm takes as input a pointed rational cone described in terms of rational inequalities and produces a generating function that enumerates its integer points.

Recall that the fundamental cone of a binary code $C(H)$ is

$$\mathcal{K}(H) := \left\{ v \in \mathbb{R}^n : Row_j H v^T - 2h_{ji} v_i \geq 0, v_i \geq 0 \ \forall 1 \leq i \leq n, 1 \leq j \leq r \right\}.$$

Barvinok's algorithm enumerates integer points of a rational cone.

Barvinok's algorithm takes as input a pointed rational cone described in terms of rational inequalities and produces a generating function that enumerates its integer points.

Recall that the fundamental cone of a binary code $C(H)$ is

$$\mathcal{K}(H) := \left\{ v \in \mathbb{R}^n : Row_j H v^T - 2h_{ji} v_i \geq 0, v_i \geq 0 \ \forall 1 \leq i \leq n, 1 \leq j \leq r \right\}.$$

However, the pseudocodewords of $C(H)$ are those integer points of $\mathcal{K}(H)$ which satisfy $H\mathbf{p}^T = \mathbf{0} \in \mathbb{F}_2^r$.

Lifting the fundamental cone allows us to apply Barvinok's algorithm.

Given $H \in F_2^{r \times n}$, the fundamental cone of $C(H)$ is

$$\mathcal{K}(H) := \left\{ v \in \mathbb{R}^n : Row_j H v^T - 2h_{ji} v_i \geq 0, v_i \geq 0 \ \forall 1 \leq i \leq n, 1 \leq j \leq r \right\}.$$

Definition

Given $H \in \mathbb{F}_2^{r \times n}$, the lifted fundamental cone of $C(H)$ is

$$\hat{\mathcal{K}}(H) = \left\{ (\mathbf{v}, \mathbf{a}) \in \mathbb{R}^{n+r} \mid \begin{array}{l} v_i \geq 0, H\mathbf{v}^T = 2\mathbf{a}^T, \text{ and} \\ Row_j H v^T - 2h_{ji} v_i \geq 0 \\ \text{for all } 1 \leq i \leq n \text{ and } 1 \leq j \leq r \end{array} \right\}.$$

Lifting the fundamental cone allows us to apply Barvinok's algorithm.

Given $H \in F_2^{r \times n}$, the fundamental cone of $C(H)$ is

$$\mathcal{K}(H) := \left\{ v \in \mathbb{R}^n : Row_j H v^T - 2h_{ji} v_i \geq 0, v_i \geq 0 \ \forall 1 \leq i \leq n, 1 \leq j \leq r \right\}.$$

Definition

Given $H \in \mathbb{F}_2^{r \times n}$, the lifted fundamental cone of $C(H)$ is

$$\hat{\mathcal{K}}(H) = \left\{ (\mathbf{v}, \mathbf{a}) \in \mathbb{R}^{n+r} \mid \begin{array}{l} v_i \geq 0, H\mathbf{v}^T = 2\mathbf{a}^T, \text{ and} \\ Row_j H v^T - 2h_{ji} v_i \geq 0 \\ \text{for all } 1 \leq i \leq n \text{ and } 1 \leq j \leq r \end{array} \right\}.$$

Consider the projection

$$\begin{array}{rccc} \pi : & \mathbb{R}^{n+r} & \rightarrow & \mathbb{R}^n \\ & (\mathbf{v}, \mathbf{a}) & \mapsto & \mathbf{v}. \end{array}$$

Lifting the fundamental cone allows us to apply Barvinok's algorithm.

Given $H \in F_2^{r \times n}$, the fundamental cone of $C(H)$ is

$$\mathcal{K}(H) := \left\{ v \in \mathbb{R}^n : Row_j H v^T - 2h_{ji} v_i \geq 0, v_i \geq 0 \ \forall 1 \leq i \leq n, 1 \leq j \leq r \right\}.$$

Definition

Given $H \in \mathbb{F}_2^{r \times n}$, the lifted fundamental cone of $C(H)$ is

$$\hat{\mathcal{K}}(H) = \left\{ (\mathbf{v}, \mathbf{a}) \in \mathbb{R}^{n+r} \mid \begin{array}{l} v_i \geq 0, H\mathbf{v}^T = 2\mathbf{a}^T, \text{ and} \\ Row_j H v^T - 2h_{ji} v_i \geq 0 \\ \text{for all } 1 \leq i \leq n \text{ and } 1 \leq j \leq r \end{array} \right\}.$$

Consider the projection

$$\begin{array}{rccc} \pi : & \mathbb{R}^{n+r} & \rightarrow & \mathbb{R}^n \\ & (\mathbf{v}, \mathbf{a}) & \mapsto & \mathbf{v}. \end{array}$$

Proposition

Let $H \in \mathbb{F}_2^{r \times n}$. Then $\pi\left(\hat{\mathcal{K}}(H)\right) = \mathcal{K}(H)$ and $\mathcal{P}(H) = \pi\left(\hat{\mathcal{K}}(H) \cap \mathbb{Z}^{n+r}\right)$.

The generating function for the pseudocodewords of a code is a rational function.

Denote the generating function for integer points in the lifted fundamental cone by

$$f(x_1, x_2, \ldots, x_{n+r}) := \sum_{(\mathbf{v},\mathbf{a}) \in \hat{\mathcal{K}}(H) \cap \mathbb{Z}^{n+r}} \mathbf{x}^{(\mathbf{v},\mathbf{a})}.$$

The generating function for the pseudocodewords of a code is a rational function.

Denote the generating function for integer points in the lifted fundamental cone by

$$f(x_1, x_2, \ldots, x_{n+r}) := \sum_{(\mathbf{v},\mathbf{a}) \in \hat{\mathcal{K}}(H) \cap \mathbb{Z}^{n+r}} \mathbf{x}^{(\mathbf{v},\mathbf{a})}.$$

Then $f(x_1, x_2, \ldots, x_{n+r})$ can be expressed as a rational function.

The generating function for the pseudocodewords of a code is a rational function.

Denote the generating function for integer points in the lifted fundamental cone by

$$f(x_1, x_2, \ldots, x_{n+r}) := \sum_{(\mathbf{v},\mathbf{a}) \in \hat{\mathcal{K}}(H) \cap \mathbb{Z}^{n+r}} \mathbf{x}^{(\mathbf{v},\mathbf{a})}.$$

Then $f(x_1, x_2, \ldots, x_{n+r})$ can be expressed as a rational function. Applying the Barvinok-Woods approach to specialization gives that

$$\sum_{\mathbf{v} \in \mathcal{P}(H)} \mathbf{x}^{\mathbf{v}} = \sum_{\mathbf{v} \in \pi(\hat{\mathcal{K}}(H) \cap \mathbb{Z}^{n+r})} \mathbf{x}^{\mathbf{v}} = \sum_{(\mathbf{v},\mathbf{a}) \in \hat{\mathcal{K}}(H) \cap \mathbb{Z}^{n+r}} \mathbf{x}^{\mathbf{v}} = f(x_1, x_2, \ldots, x_n, 1, 1, \ldots, 1)$$

is rational.

The generating function for the pseudocodewords of a code is a rational function.

Denote the generating function for integer points in the lifted fundamental cone by

$$f(x_1, x_2, \ldots, x_{n+r}) := \sum_{(\mathbf{v},\mathbf{a}) \in \hat{\mathcal{K}}(H) \cap \mathbb{Z}^{n+r}} \mathbf{x}^{(\mathbf{v},\mathbf{a})}.$$

Then $f(x_1, x_2, \ldots, x_{n+r})$ can be expressed as a rational function. Applying the Barvinok-Woods approach to specialization gives that

$$\sum_{\mathbf{v} \in \mathcal{P}(H)} \mathbf{x}^{\mathbf{v}} = \sum_{\mathbf{v} \in \pi(\hat{\mathcal{K}}(H) \cap \mathbb{Z}^{n+r})} \mathbf{x}^{\mathbf{v}} = \sum_{(\mathbf{v},\mathbf{a}) \in \hat{\mathcal{K}}(H) \cap \mathbb{Z}^{n+r}} \mathbf{x}^{\mathbf{v}} = f(x_1, x_2, \ldots, x_n, 1, 1, \ldots, 1)$$

is rational.

Theorem (K-M)

Given $H \in \mathbb{F}_2^{r \times n}$, the generating function for the pseudocodewords of $C(H)$ $\sum_{p \in \mathcal{P}(H)} x^p$ is a rational function.

Critical multisets play a key role in understanding pseudocodewords.

Definition

Given a prime p, a critical multiset of $\mathbb{F}_p$ is a multiset $\{\gamma_1, \ldots, \gamma_t\} \subseteq \{0, 1, \ldots, p-1\}$ with $t \geq 2$ that is maximal with respect to the property

$$\sum_{i=1}^{t} \gamma_i > (t-1)p.$$

Let Γ_p be the set of critical multisets of $\mathbb{F}_p$.

Example

- $\Gamma_2 = \emptyset$,
- $\Gamma_3 = \{\{2,2\}\}$,
- $\Gamma_5 = \{\{2,4\},\{3,3\},\{3,4,4\},\{4,4,4,4\}\}$, and
- $\Gamma_7 = \{\{2,6\},\{3,5\},\{4,4\},\{3,6,6\},\{4,5,6\},\{5,5,5\},\{4,6,6,6\},$ $\{5,5,6,6\},\{5,6,6,6,6\},\{6,6,6,6,6,6\}\}$.

Critical multisets play a role in describing a cone containing the pseudocodewords of a nonbinary parity-check code.

Given a prime p and $H \in \mathbb{F}_p^{r \times n}$, consider the cone

$$K_p(H) = \left\{ v \in \mathbb{R}_{\geq 0}^{(p-1)n} : \begin{array}{l} \Theta_j(v) \geq m_i(1) + m_i(2) + \cdots + m_i(p-1), \text{ and} \\ \Theta_j(v) \geq \sum_{l=1}^{t} m_{\left(\gamma_l \odot a^{-1} \odot h_{ji_l}^{-1} - 1\right) n + i_l} \\ \forall 1 \leq j \leq r, i, i_1, \ldots, i_t \in \text{supp}\left(\text{Row}_j(H)\right), a \in \mathbb{F}_p^*, \\ \{\gamma_1, \ldots, \gamma_t\} \in \Gamma_p \end{array} \right\}$$

where $\Theta_j(v) = \frac{1}{p} \sum_{b=1}^{p-1} \left(b \, \text{Row}_j(H) \right) \left[m_1(b) \; m_2(b) \; \cdots \; m_n(b) \right]^T$.

The fundamental cone is the smallest cone containing the pseudocodewords.

Lemma (K-M)

For a prime p and a p-ary matrix H,

$$\mathcal{K}_p(H) \subseteq K_p(H).$$

If $p = 2, 3$, then equality holds.

Theorem (K-M)

Given a p-ary parity-check matrix H, where p is prime, then

$$\mathcal{P}(H) \subseteq \left\{ v \in K_p(H) : H\mathcal{M}(v)^T \begin{bmatrix} 1 \\ \vdots \\ p-1 \end{bmatrix} \mod p = \mathbf{0} \right\}.$$

If $p = 2, 3$, then equality holds.

The integer points of the lifted fundamental cone are precisely the pseudocodewords.

Definition

Given $H \in \mathbb{F}_p^{r \times n}$ where p is prime, the lifted fundamental cone of $C(H)$ is

$$\hat{\mathcal{K}}_p(H) = \left\{ (\mathbf{m}, \mathbf{a}) \in \mathbb{R}^{(p-1)n+r} \,\middle|\, v \in \mathcal{K}_p(H), H\mathcal{M}(v)^T \begin{bmatrix} 1 \\ \vdots \\ p-1 \end{bmatrix} = p\mathbf{a}^T \right\}.$$

The integer points of the lifted fundamental cone are precisely the pseudocodewords.

Definition

Given $H \in \mathbb{F}_p^{r \times n}$ where p is prime, the lifted fundamental cone of $C(H)$ is

$$\hat{\mathcal{K}}_p(H) = \left\{ (\mathbf{m}, \mathbf{a}) \in \mathbb{R}^{(p-1)n+r} \,\middle|\, v \in \mathcal{K}_p(H), H\mathcal{M}(v)^T \begin{bmatrix} 1 \\ \vdots \\ p-1 \end{bmatrix} = p\mathbf{a}^T \right\}.$$

Theorem (K-M)

Given a binary or ternary matrix H, there exists a polynomial time algorithm which computes the generating function of the pseudocodewords of $C(H)$ as a finite sum

$$\sum_{\mathbf{p} \in \mathcal{P}(H)} \mathbf{x}^{\mathbf{p}} = \sum_i \alpha_i \frac{\mathbf{x}^{\mathbf{v}_i}}{(1 - \mathbf{x}^{\mathbf{w}_{i1}}) \ldots (1 - \mathbf{x}^{\mathbf{w}_{id}})}$$

where $\alpha_i \in \{0, 1\}$ and $\mathbf{u}_i$, $\mathbf{w}_{ij}$ are integer vectors for all i, j.

Generating functions obtained via Barvinok's algorithm enumerate the pseudocodewords.

Example

Consider the binary code $C(H)$ given by a parity check matrix $H = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \end{pmatrix}$. Then Barvinok 0.27 computes

$$\sum_{\mathbf{p} \in \mathcal{P}(H)} \mathbf{x}^{\mathbf{p}} = \frac{1 - x_1^2 x_2^2 x_3^2 x_4^2}{(1 - x_1 x_3 x_4^2)(1 - x_1 x_2^2 x_3)(1 - x_2 x_3 x_4)(1 - x_1 x_2 x_4)(1 - x_1 x_3)}$$

Generating functions obtained via Barvinok's algorithm enumerate the pseudocodewords.

Example

Consider the binary code $C(H)$ given by a parity check matrix $H = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \end{pmatrix}$. Then Barvinok 0.27 computes

$$\begin{aligned} \sum_{\mathbf{p} \in \mathcal{P}(H)} \mathbf{x}^{\mathbf{p}} &= \frac{1 - x_1^2 x_2^2 x_3^2 x_4^2}{(1 - x_1 x_3 x_4^2)(1 - x_1 x_2^2 x_3)(1 - x_2 x_3 x_4)(1 - x_1 x_2 x_4)(1 - x_1 x_3)} \\ &= 1 + x_1 x_3 + x_2 x_3 x_4 + x_1 x_2 x_4 + x_1 x_3 x_4^2 + x_1 x_2^2 x_3 + \\ &\quad x_1^2 x_3^2 + x_1^2 x_2 x_3 x_4 + x_1 x_2 x_3^2 x_4 + \cdots . \end{aligned}$$

Generating functions obtained via Barvinok's algorithm enumerate the pseudocodewords.

Example

Consider the binary code $C(H)$ given by a parity check matrix $H = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \end{pmatrix}$. Then Barvinok 0.27 computes

$$\begin{aligned} \sum_{\mathbf{p}\in\mathcal{P}(H)} \mathbf{x}^{\mathbf{p}} &= \frac{1-x_1^2x_2^2x_3^2x_4^2}{(1-x_1x_3x_4^2)(1-x_1x_2^2x_3)(1-x_2x_3x_4)(1-x_1x_2x_4)(1-x_1x_3)} \\ &= 1 + x_1x_3 + x_2x_3x_4 + x_1x_2x_4 + x_1x_3x_4^2 + x_1x_2^2x_3 + \\ &\quad x_1^2x_3^2 + x_1^2x_2x_3x_4 + x_1x_2x_3^2x_4 + \cdots . \end{aligned}$$

Hence, $(0,0,0,0), (1,0,1,0), (0,1,1,1), (1,1,0,1), (1,0,1,2), (1,2,1,0), (2,0,2,0), (2,1,1,1), (1,1,2,1), \ldots$ are among the (unscaled) pseudocodewords of $C(H)$.

Irreducible pseudocodewords are those pseudocodewords most likely to cause decoder failure.

A nonzero pseudocodeword is irreducible provided it cannot be written as a sum of two or more nonzero pseudocodewords.

Theorem (K-M, K-M)

Let $H \in \mathbb{F}_p^{r \times n}$. Then the set of irreducible pseudocodewords of $C(H)$ is a projection of the Hilbert basis of the lifted fundamental cone of $C(H)$.

The choice of Tanner graph can impact the number of irreducible pseudocodewords.

Example

Recall that $C(H_1) = C(H_2)$ for

$$H_1 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 \end{pmatrix} \quad H_2 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{pmatrix}$$

The choice of Tanner graph can impact the number of irreducible pseudocodewords.

Example

Recall that $C(H_1) = C(H_2)$ for

$$H_1 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 \end{pmatrix} \quad H_2 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{pmatrix}$$

However, the sets of
noncodeword irreducible pseudocodewords for H_1 and H_2 differ, as shown below.

$$\underbrace{\underbrace{\begin{matrix} (0,0,0,2,2,2,2), & (0,3,0,1,2,1,1), \\ (0,0,3,2,1,1,1), & (0,0,1,2,1,1,1), \\ (3,0,0,1,1,2,1), & (0,1,0,1,2,1,1), \\ (2,1,0,1,0,1,1), & (1,2,0,1,1,0,1), \\ (2,0,1,0,1,1,1), & (0,2,1,0,1,1,1), \\ (1,0,0,1,1,2,1), & (0,1,2,1,0,1,1), \\ (1,0,2,1,1,0,1), & \end{matrix}}_{\text{Irreducible pseudocodeword of } C(H_1)} \quad \begin{matrix} (2,0,3,0,1,1,1), & (4,0,0,0,2,2,2), & (0,0,0,4,2,2,2), \\ (0,4,0,0,2,2,2), & (2,0,0,0,2,2,2), & (0,2,0,0,2,2,2), \\ (3,0,0,3,1,2,1), & (1,1,0,2,1,1,0), & (3,3,0,0,1,1,2), \\ (1,1,0,0,1,1,2), & (0,1,0,3,2,1,1), & (1,3,0,0,1,1,2), \\ (3,1,0,0,1,1,2), & (1,0,0,3,1,2,1), & (0,0,0,0,2,2,2), \\ (0,3,0,3,2,1,1), & (1,1,2,0,1,1,0), & (0,0,3,0,1,1,1), \\ (0,2,3,0,1,1,1). & & \end{matrix}}_{\text{Irreducible pseudocodeword of } C(H_2)}$$

The choice of Tanner graph can impact the number of irreducible pseudocodewords.

Example

If $H_1 = \begin{bmatrix} 1 & 2 & 2 & 1 \\ 2 & 0 & 1 & 2 \end{bmatrix}$ and $H_2 = \begin{bmatrix} 0 & 2 & 0 & 0 \\ 2 & 0 & 1 & 2 \end{bmatrix}$, then $C(H_1) = C(H_2)$.
However, the irreducible pseudocodewords of $C(H_1)$ and $C(H_2)$ are

$$Irr\,(H_1) = \left\{ \begin{array}{llll}
(0,0,0,0,0,0,1,1), & (0,0,0,0,1,0,1,0), & (0,0,0,1,1,0,0,0), & (0,0,1,0,1,0,0,1), \\
(0,0,1,1,0,0,0,0), & (1,0,0,0,0,0,0,1), & (1,0,0,1,0,0,1,0), & (1,0,1,0,0,0,0,0), \\
(0,0,0,1,1,3,1,1), & (0,0,0,1,2,3,1,0), & (0,0,1,0,1,3,0,1), & (0,0,1,1,0,3,1,1), \\
(0,0,1,1,1,3,1,0), & (0,1,0,0,0,1,1,1), & (0,1,0,0,1,1,1,0), & (0,1,0,1,1,1,0,0), \\
(0,1,1,0,1,1,0,1), & (0,1,1,1,0,1,0,0), & (0,3,0,0,1,0,2,1), & (0,3,0,1,1,0,1,1), \\
(0,3,1,0,1,0,1,2), & (0,3,1,0,2,0,1,1), & (0,3,1,1,0,0,1,1), & (0,3,1,1,1,0,1,0), \\
(0,3,1,1,2,0,0,1), & (0,3,1,2,1,0,0,0), & (0,3,2,0,2,0,0,2), & (0,3,2,1,1,0,0,1), \\
(1,0,0,0,0,3,1,2), & (1,0,0,0,1,3,1,1), & (1,0,0,1,0,3,2,1), & (1,0,0,1,1,3,0,1), \\
(1,0,0,1,1,3,2,0), & (1,0,0,2,1,3,1,0), & (1,0,1,0,0,3,1,1), & (1,0,1,0,1,3,1,0), \\
(1,0,1,1,0,3,0,1), & (1,0,1,1,1,3,0,0), & (1,0,1,2,0,3,1,0), & (1,0,2,1,0,3,0,0), \\
(1,1,0,0,0,1,0,1), & (1,1,0,1,0,1,1,0), & (1,1,1,0,0,1,0,0), & (1,3,0,0,1,0,1,1), \\
(1,3,0,1,0,0,1,0), & (1,3,0,1,1,0,0,1), & (1,3,1,0,0,0,1,1), & (1,3,1,0,1,0,0,2), \\
(1,3,1,0,1,0,1,0), & (1,3,1,1,0,0,0,1), & (1,3,1,1,1,0,0,0), & (1,3,2,0,1,0,0,1), \\
(2,0,0,1,0,3,1,1), & (2,0,0,2,0,3,2,0), & (2,0,1,1,0,3,1,0), & (2,3,1,0,0,0,0,1).
\end{array} \right\}$$

and

$$Irr\,(H_2) = \left\{ \begin{array}{llll}
(0,0,0,0,0,0,1,1), & (0,0,0,0,1,0,1,0), & (0,0,0,1,1,0,0,0), & (0,0,1,0,1,0,0,1), \\
(0,0,1,1,0,0,0,0), & (1,0,0,0,0,0,0,1), & (1,0,0,1,0,0,1,0), & (1,0,1,0,0,0,0,0)
\end{array} \right\}.$$

References

- T. Etzion, A. Trachtenberg, and A. Vardy, Which codes have cycle-free Tanner graphs? IEEE Trans. Inform. Theory **45** (1999), no. 6, 2173–2181.
- J. Feldman, M. J. Wainwright, and D. R. Karger, Using linear programming to decode binary linear codes, IEEE Trans. Inform. Theory **51** (2005), no. 3, 954–972.
- M. Flanagan, V. Skachek, E. Byrne, and M. Greferath, Linear-programming decoding of nonbinary linear codes, IEEE Trans. Inform. Theory **55** (2009), no. 9, 4134–4154.
- R. Koetter, W.-C. W. Li, P. Vontobel, and J. Walker, Characterizations of pseudo-codewords of (low-density) parity-check codes. Adv. Math. **213** (2007), no. 1, 205–229.
- W. Kositwattanarerk and G. L. Matthews, Lifting the fundamental cone and enumerating the pseudocodewords of a parity-check code, IEEE Trans. Inform. Theory **57** (2011), no. 2, 898-909.
- W. Kositwattanarerk and G. L. Matthews, Pseudocodewords of parity-check codes over fields of prime cardinality, IEEE Trans. Inform. Theory **60** (2014), no. 9, 5215 5227.
- R. M. Tanner, A recursive approach to low-complexity codes, IEEE Trans. Inform. Theory **27** (1981), no. 5, 533–547.

Thank you!