Codes on Graphs and the Pseudocodewords

Wittawat Kositwattanarerk

Division of Mathematical Sciences
Nanyang Technological University, Singapore

September 7, 2012

Notation

$\mathbb{F}_2 = \{0, 1\}$, the binary field.
$\mathbb{N} = \{0, 1, 2, \ldots\}$, the set of nonnegative integers.

Notation

$\mathbb{F}_2 = \{0, 1\}$, the binary field.
$\mathbb{N} = \{0, 1, 2, \ldots\}$, the set of nonnegative integers.

A **binary linear code** C of length n and dimension k is a subspace of $\mathbb{F}_2^n$ of dimension k. Elements of C are called codewords.

Notation

$\mathbb{F}_2 = \{0, 1\}$, the binary field.
$\mathbb{N} = \{0, 1, 2, \ldots\}$, the set of nonnegative integers.

A **binary linear code** C of length n and dimension k is a subspace of $\mathbb{F}_2^n$ of dimension k. Elements of C are called codewords.

A **parity check matrix** of a code C is any matrix $H \in \mathbb{F}_2^{r \times n}$ such that C is the null space of H.

Notation

$\mathbb{F}_2 = \{0, 1\}$, the binary field.
$\mathbb{N} = \{0, 1, 2, \ldots\}$, the set of nonnegative integers.

A **binary linear code** C of length n and dimension k is a subspace of $\mathbb{F}_2^n$ of dimension k. Elements of C are called codewords.

A **parity check matrix** of a code C is any matrix $H \in \mathbb{F}_2^{r \times n}$ such that C is the null space of H.

Given a parity check matrix H of C and $\mathbf{y} \in \mathbb{F}_2^n$,

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

Parity check matrix of a code is not unique.

The code

$$C = \left\{ \begin{array}{ll} (0,0,0,0), & (0,1,1,1) \\ (1,0,1,0), & (1,1,0,1) \end{array} \right\} \subset \mathbb{F}_2^4$$

is a binary linear code of length 4 and dimension 2.

Parity check matrix of a code is not unique.

The code

$$C = \left\{ \begin{array}{ll} (0,0,0,0), & (0,1,1,1) \\ (1,0,1,0), & (1,1,0,1) \end{array} \right\} \subset \mathbb{F}_2^4$$

is a binary linear code of length 4 and dimension 2. This code has as a parity check matrix

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

Parity check matrix of a code is not unique.

The code

$$C = \left\{ \begin{array}{ll} (0,0,0,0), & (0,1,1,1) \\ (1,0,1,0), & (1,1,0,1) \end{array} \right\} \subset \mathbb{F}_2^4$$

is a binary linear code of length 4 and dimension 2. This code has as a parity check matrix

$$H = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \end{pmatrix}, \qquad H_1 = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix},$$

$$H_2 = \begin{pmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \end{pmatrix}, \qquad H_3 = \begin{pmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}.$$

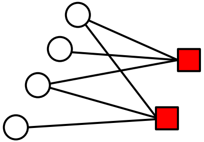
A code C given by a parity check matrix H is denoted $C(H)$.

A parity check matrix corresponds to a bipartite graph called the Tanner graph.

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

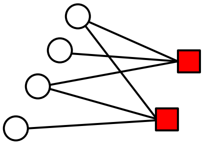
A parity check matrix corresponds to a bipartite graph called the Tanner graph.

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



A parity check matrix corresponds to a bipartite graph called the Tanner graph.

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

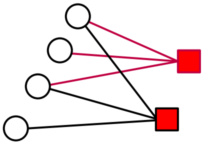


The **Tanner graph** of H is a bipartite graph with a vertex set $X \cup F$ such that:

- Each vertex in X corresponds to a column of H and is called a bit node.
- Each vertex in F corresponds to a row of H and is called a check node.
- $\{x_i, f_j\}$ is an edge if and only if $h_{ji} = 1$.

A parity check matrix corresponds to a bipartite graph called the Tanner graph.

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

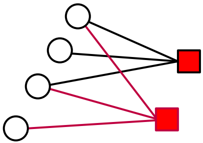


The **Tanner graph** of H is a bipartite graph with a vertex set $X \cup F$ such that:

- Each vertex in X corresponds to a column of H and is called a bit node.
- Each vertex in F corresponds to a row of H and is called a check node.
- $\{x_i, f_j\}$ is an edge if and only if $h_{ji} = 1$.

A parity check matrix corresponds to a bipartite graph called the Tanner graph.

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

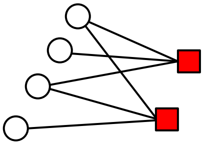


The **Tanner graph** of H is a bipartite graph with a vertex set $X \cup F$ such that:

- Each vertex in X corresponds to a column of H and is called a bit node.
- Each vertex in F corresponds to a row of H and is called a check node.
- $\{x_i, f_j\}$ is an edge if and only if $h_{ji} = 1$.

A parity check matrix corresponds to a bipartite graph called the Tanner graph.

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



The **Tanner graph** of H is a bipartite graph with a vertex set $X \cup F$ such that:

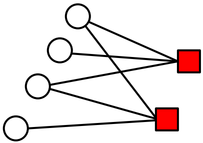
- Each vertex in X corresponds to a column of H and is called a bit node.
- Each vertex in F corresponds to a row of H and is called a check node.
- $\{x_i, f_j\}$ is an edge if and only if $h_{ji} = 1$.

A codeword corresponds to a valid configuration on the Tanner graph.

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

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

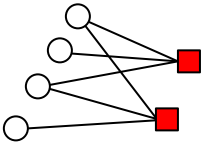
$\Longleftrightarrow$



A codeword corresponds to a valid configuration on the Tanner graph.

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

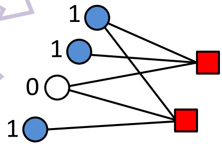
$$\begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 1 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \Longleftrightarrow$$



A codeword corresponds to a valid configuration on the Tanner graph.

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

$$\begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 1 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \Longleftrightarrow$$

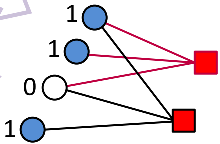


Since a check node represents a row, i.e. a parity condition, of H, $\mathbf{c} = (c_1, c_2, \ldots, c_n)$ is a codeword of $C(H)$ if and only if every check node is adjacent to an even number of 1's.

A codeword corresponds to a valid configuration on the Tanner graph.

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

$$\begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 1 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad \Longleftrightarrow$$

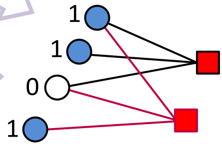


Since a check node represents a row, i.e. a parity condition, of H, $\mathbf{c} = (c_1, c_2, \ldots, c_n)$ is a codeword of $C(H)$ if and only if every check node is adjacent to an even number of 1's.

A codeword corresponds to a valid configuration on the Tanner graph.

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

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

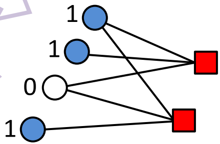


Since a check node represents a row, i.e. a parity condition, of H, $\mathbf{c} = (c_1, c_2, \ldots, c_n)$ is a codeword of $C(H)$ if and only if every check node is adjacent to an even number of 1's.

A codeword corresponds to a valid configuration on the Tanner graph.

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

$$\begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 1 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \Longleftrightarrow$$

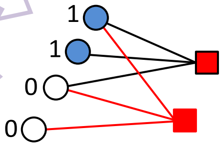


Since a check node represents a row, i.e. a parity condition, of H, $\mathbf{c} = (c_1, c_2, \ldots, c_n)$ is a codeword of $C(H)$ if and only if every check node is adjacent to an even number of 1's.

A codeword corresponds to a valid configuration on the Tanner graph.

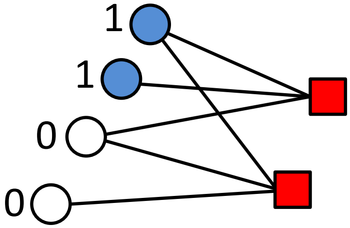
$(1, 1, 0, 0)$ is not a codeword

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



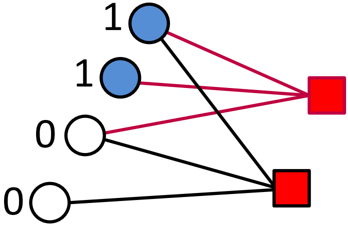
Since a check node represents a row, i.e. a parity condition, of H, $\mathbf{c} = (c_1, c_2, \ldots, c_n)$ is a codeword of $C(H)$ if and only if every check node is adjacent to an even number of 1's.

Thou shall vote!



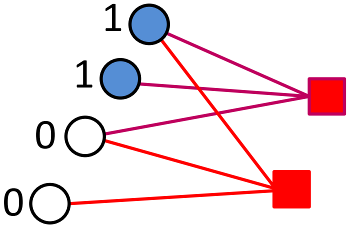
Let the check nodes "vote" for the bit nodes to be flipped. Reiterate as necessary.

Thou shall vote!



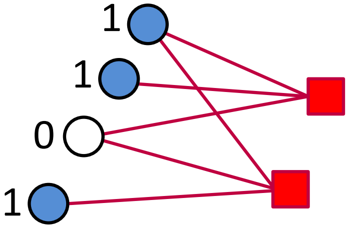
Let the check nodes "vote" for the bit nodes to be flipped. Reiterate as necessary.

Thou shall vote!



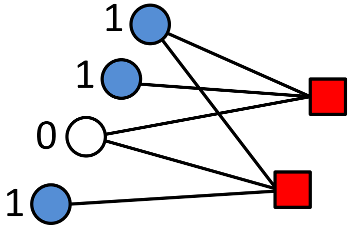
Let the check nodes "vote" for the bit nodes to be flipped. Reiterate as necessary.

Thou shall vote!



Let the check nodes "vote" for the bit nodes to be flipped. Reiterate as necessary.

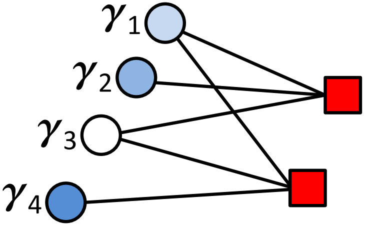
Thou shall vote!



Let the check nodes "vote" for the bit nodes to be flipped. Reiterate as necessary.

Gallager's algorithm A reassigns the value of the bit notes that are adjacent to a certain number of unsatisfied check nodes.

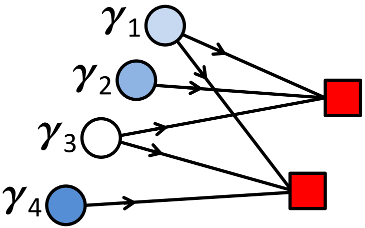
Soft decision decoding works well on graphs.



Upon receiving the message $\mathbf{w}$, initialize each bit with $\gamma_i = \log\left(\frac{P(w_i|c_i=0)}{P(w_i|c_i=1)}\right)$, the log-likelihood ratio at the i^{th} coordinate.

The maximum likelihood decoding is equivalent to finding a binary value assignment $(c_1, c_2, \ldots, c_n)$ to the bit nodes such that $\sum_{i=1}^{n} \gamma_i c_i$ is minimized and every check node is adjacent to an even number of 1's.

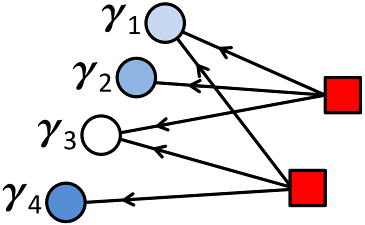
Soft decision decoding works well on graphs.



Upon receiving the message $\mathbf{w}$, initialize each bit with $\gamma_i = \log\left(\frac{P(w_i|c_i=0)}{P(w_i|c_i=1)}\right)$, the log-likelihood ratio at the i^{th} coordinate.

The maximum likelihood decoding is equivalent to finding a binary value assignment $(c_1, c_2, \ldots, c_n)$ to the bit nodes such that $\sum_{i=1}^{n} \gamma_i c_i$ is minimized and every check node is adjacent to an even number of 1's.

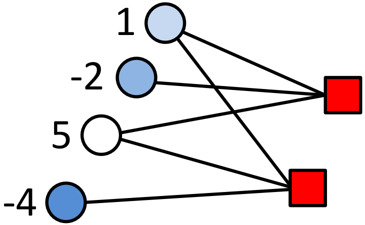
Soft decision decoding works well on graphs.



Upon receiving the message **w**, initialize each bit with $\gamma_i = \log\left(\frac{P(w_i|c_i=0)}{P(w_i|c_i=1)}\right)$, the log-likelihood ratio at the i^{th} coordinate.

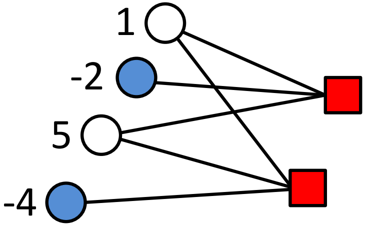
The maximum likelihood decoding is equivalent to finding a binary value assignment $(c_1, c_2, \ldots, c_n)$ to the bit nodes such that $\sum_{i=1}^{n} \gamma_i c_i$ is minimized and every check node is adjacent to an even number of 1's.

Soft decision decoding works well on graphs.



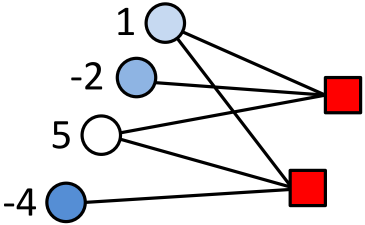
Each bit is initialized with the log-likelihood ratio $\gamma_i = \log\left(\frac{P(w_i|c_i=0)}{P(w_i|c_i=1)}\right)$.

Soft decision decoding works well on graphs.



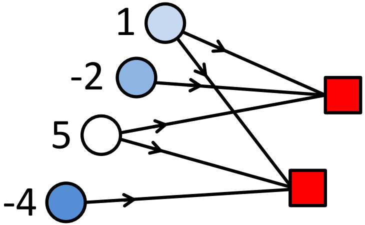
Each bit is initialized with the log-likelihood ratio $\gamma_i = \log\left(\frac{P(w_i|c_i=0)}{P(w_i|c_i=1)}\right)$.

Soft decision decoding works well on graphs.



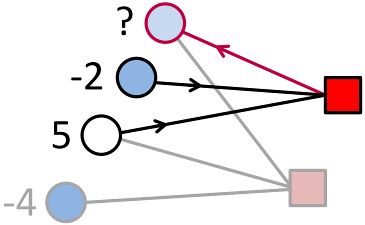
Each bit is initialized with the log-likelihood ratio $\gamma_i = \log\left(\frac{P(w_i|c_i=0)}{P(w_i|c_i=1)}\right)$.

Soft decision decoding works well on graphs.



Each bit is initialized with the log-likelihood ratio $\gamma_i = \log\left(\frac{P(w_i|c_i=0)}{P(w_i|c_i=1)}\right)$.

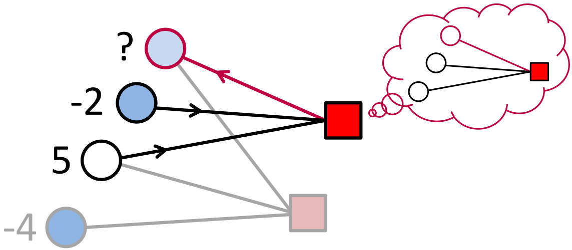
Soft decision decoding works well on graphs.



Each bit is initialized with the log-likelihood ratio $\gamma_i = \log\left(\frac{P(w_i|c_i=0)}{P(w_i|c_i=1)}\right)$.

To compute the message to the first bit, the first check takes into consideration only the message from the second the third bits.

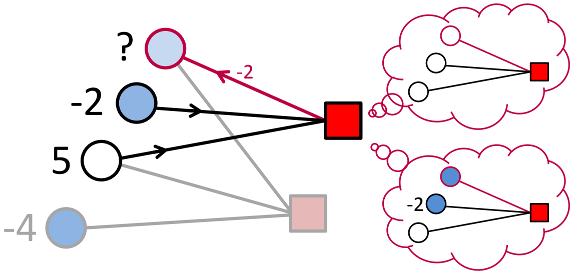
Soft decision decoding works well on graphs.



Each bit is initialized with the log-likelihood ratio $\gamma_i = \log\left(\frac{P(w_i|c_i=0)}{P(w_i|c_i=1)}\right)$.

To compute the message to the first bit, the first check takes into consideration only the message from the second the third bits.

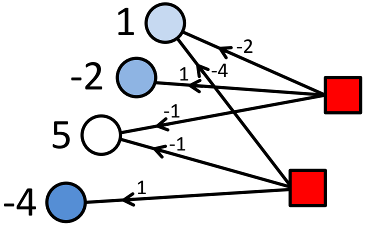
Soft decision decoding works well on graphs.



Each bit is initialized with the log-likelihood ratio $\gamma_i = \log\left(\frac{P(w_i|c_i=0)}{P(w_i|c_i=1)}\right)$.

To compute the message to the first bit, the first check takes into consideration only the message from the second the third bits.

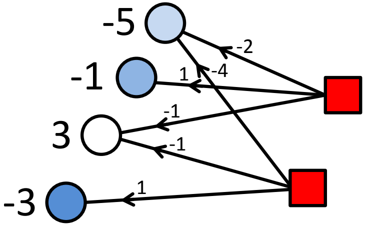
Soft decision decoding works well on graphs.



Each bit is initialized with the log-likelihood ratio $\gamma_i = \log\left(\frac{P(w_i|c_i=0)}{P(w_i|c_i=1)}\right)$.

To compute the message to the first bit, the first check takes into consideration only the message from the second the third bits.

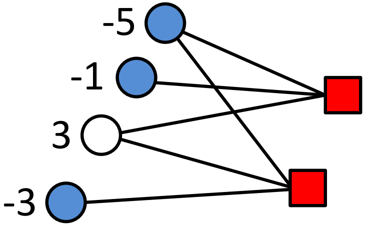
Soft decision decoding works well on graphs.



Each bit is initialized with the log-likelihood ratio $\gamma_i = \log\left(\frac{P(w_i|c_i=0)}{P(w_i|c_i=1)}\right)$.

To compute the message to the first bit, the first check takes into consideration only the message from the second the third bits.

Soft decision decoding works well on graphs.



Each bit is initialized with the log-likelihood ratio $\gamma_i = \log\left(\frac{P(w_i|c_i=0)}{P(w_i|c_i=1)}\right)$.

To compute the message to the first bit, the first check takes into consideration only the message from the second the third bits.

Min-sum algorithm marginalizes the cost function.

- Initialization: For each bit node i, initialize the local cost γ_i. For each check node j and for all $s \in Nbhd(j)$, initialize $\mu_{j,s}^{(0)} := 0$.
- Iteration: For $i = 1, \ldots, m$:
  1. For all bit nodes s and for all $j \in Nbhd(s)$, bit-to-check messages are given by
  $$\mu_{s,j}^{(i)} := \gamma_s + \sum_{j' \in Nbhd(s)-\{j\}} \mu_{j',s}^{(i-1)}.$$
  2. For all check nodes j and for all $s \in Nbhd(j)$, check-to-bit messages are given by
  $$\mu_{j,s}^{(i)} := \prod_{s' \in Nbhd(j)-\{s\}} \operatorname{sgn}\left(\mu_{s',j}^{(i)}\right) \cdot \min_{s' \in Nbhd(j)-\{s\}} \left|\mu_{s',j}^{(i)}\right|$$
- Final cost computation: The final cost at the bit node i after m iterations is
$$\mu_i := \gamma_i + \sum_{j \in Nbhd(i)} \mu_{j,i}^{(m)}.$$

Sum-product algorithm marginalizes the probability.

- Initialization: For each bit node i, initialize the local cost $\gamma_i = \log\left(\frac{P(c_i=0|w_i)}{P(c_i=1|w_i)}\right)$. For each check node j and for all $s \in Nbhd(j)$, initialize $\mu_{j,s}^{(0)} := 0$.
- Iteration: For $i = 1, \ldots, m$:
  1. For all bit nodes s and for all $j \in Nbhd(s)$, bit-to-check messages are given by
  $$\mu_{s,j}^{(i)} := \gamma_s + \sum_{j' \in Nbhd(s)-\{j\}} \mu_{j',s}^{(i-1)}.$$
  2. For all check nodes j and for all $s \in Nbhd(j)$, check-to-bit messages are given by
  $$\mu_{j,s}^{(i)} := \log\left(\frac{1+\prod_{s' \in Nbhd(j)-\{s\}} \tanh\left(\mu_{s',j}^{(i)}/2\right)}{1-\prod_{s' \in Nbhd(j)-\{s\}} \tanh\left(\mu_{s',j}^{(i)}/2\right)}\right)$$
- Final cost computation: The final cost at the bit node i after m iterations is
$$\mu_i := \gamma_i + \sum_{j \in Nbhd(i)} \mu_{j,i}^{(m)}.$$

Advantages and disadvantages of iterative decoders

- Easy to implement.
- Can be applied to any code.
- Low-complexity: exponential in the check node degree, linear in code length for low-density parity-check (LDPC) codes.
- Allow communication at rates near the channel capacity.

- Depends on the parity check matrix.

- Converge to a maximum-likelihood codeword in 1 iteration if the Tanner graph is cycle-free.
- May converge to a noncodeword output called a **pseudocodeword**.

Advantages and disadvantages of iterative decoders

- Easy to implement.
- Can be applied to any code.
- Low-complexity: exponential in the check node degree, linear in code length for low-density parity-check (LDPC) codes.
- Allow communication at rates near the channel capacity.

- Depends on the parity check matrix.

- Converge to a maximum-likelihood codeword in 1 iteration if the Tanner graph is cycle-free.
- May converge to a noncodeword output called a **pseudocodeword**.

Advantages and disadvantages of iterative decoders

- Easy to implement.
- Can be applied to any code.
- Low-complexity: exponential in the check node degree, linear in code length for low-density parity-check (LDPC) codes.
- Allow communication at rates near the channel capacity.
- Depends on the parity check matrix.
- Converge to a maximum-likelihood codeword in 1 iteration if the Tanner graph is cycle-free.
- May converge to a noncodeword output called a **pseudocodeword**.

Advantages and disadvantages of iterative decoders

- Easy to implement.
- Can be applied to any code.
- Low-complexity: exponential in the check node degree, linear in code length for low-density parity-check (LDPC) codes.
- Allow communication at rates near the channel capacity.

- Depends on the parity check matrix.

- Converge to a maximum-likelihood codeword in 1 iteration if the Tanner graph is cycle-free.
- May converge to a noncodeword output called a **pseudocodeword**.

Advantages and disadvantages of iterative decoders

- Easy to implement.
- Can be applied to any code.
- Low-complexity: exponential in the check node degree, linear in code length for low-density parity-check (LDPC) codes.
- Allow communication at rates near the channel capacity.

- Depends on the parity check matrix.

- Converge to a maximum-likelihood codeword in 1 iteration if the Tanner graph is cycle-free.
- May converge to a noncodeword output called a **pseudocodeword**.

Advantages and disadvantages of iterative decoders

- Easy to implement.
- Can be applied to any code.
- Low-complexity: exponential in the check node degree, linear in code length for low-density parity-check (LDPC) codes.
- Allow communication at rates near the channel capacity.

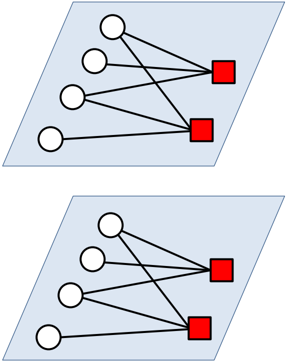
- Depends on the parity check matrix.

- Converge to a maximum-likelihood codeword in 1 iteration if the Tanner graph is cycle-free.
- May converge to a noncodeword output called a **pseudocodeword**.

Advantages and disadvantages of iterative decoders

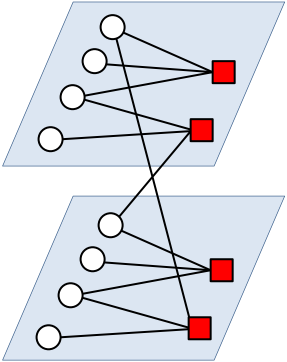
- Easy to implement.
- Can be applied to any code.
- Low-complexity: exponential in the check node degree, linear in code length for low-density parity-check (LDPC) codes.
- Allow communication at rates near the channel capacity.
- Depends on the parity check matrix.
- Converge to a maximum-likelihood codeword in 1 iteration if the Tanner graph is cycle-free.
- May converge to a noncodeword output called a **pseudocodeword**.

A graph cover is a multi-level copy of the graph.



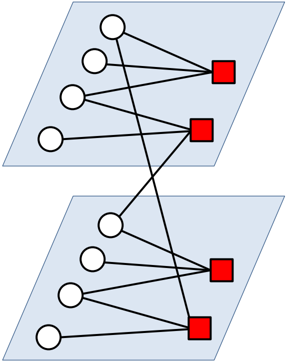
A **graph cover of degree** m of the Tanner graph of H is a bipartite graph $\widetilde{G}$ such that for each vertex v there is a set of vertices $\{v_1, \ldots, v_m\}$ of $\widetilde{G}$ with $\deg v_i = \deg v$ for all $1 \leq i \leq m$, and for every edge $\{u, v\} \in E$ there are m edges from $\{u_1, \ldots, u_m\}$ to $\{v_1, \ldots, v_m\}$ connected in a 1-1 manner.

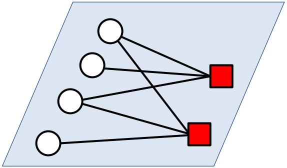
A graph cover is a multi-level copy of the graph.



A **graph cover of degree** m of the Tanner graph of H is a bipartite graph $\widetilde{G}$ such that for each vertex v there is a set of vertices $\{v_1, \ldots, v_m\}$ of $\widetilde{G}$ with $\deg v_i = \deg v$ for all $1 \leq i \leq m$, and for every edge $\{u, v\} \in E$ there are m edges from $\{u_1, \ldots, u_m\}$ to $\{v_1, \ldots, v_m\}$ connected in a 1-1 manner.

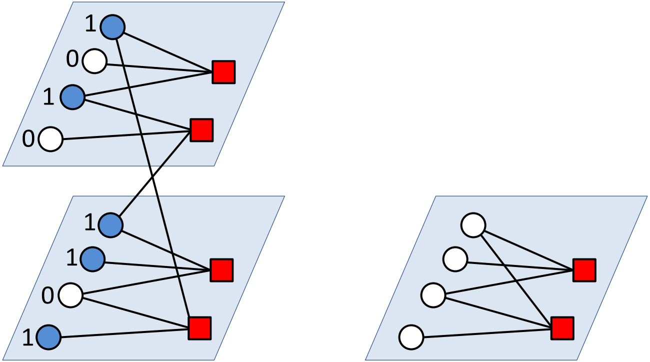
A pseudocodeword corresponds to a codeword of the graph cover.





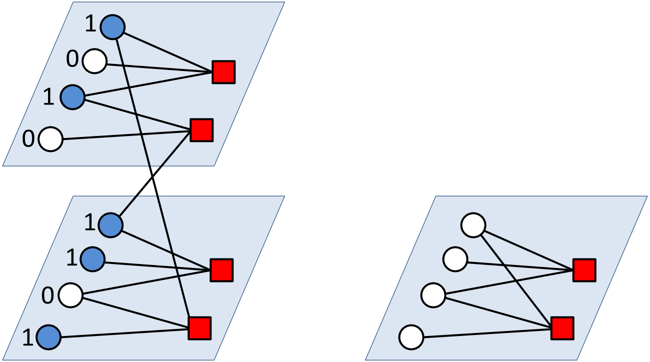
Iterative decoders operate locally on the Tanner graph, and so they cannot distinguish between the code defined by the graph cover of H and the code defined by H.

A pseudocodeword corresponds to a codeword of the graph cover.



Iterative decoders operate locally on the Tanner graph, and so they cannot distinguish between the code defined by the graph cover of H and the code defined by H.

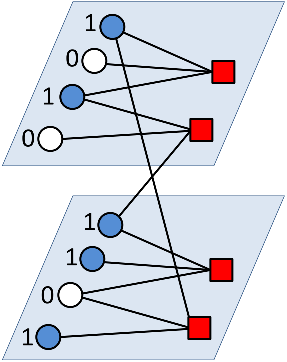
A pseudocodeword corresponds to a codeword of the graph cover.



Iterative decoders operate locally on the Tanner graph, and so they cannot distinguish between the code defined by the graph cover of H and the code defined by H.

Let $C(\widetilde{G}) \subset \mathbb{F}_2^{mn}$ be the code defined by a graph cover $\widetilde{G}$ of the Tanner graph of H.

A pseudocodeword corresponds to a codeword of the graph cover.

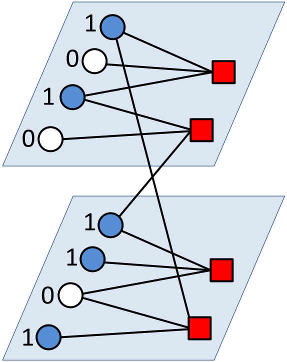


A **pseudocodeword** of $C(H)$ is $\mathbf{p} = (p_1, \ldots, p_n) \in \mathbb{N}^n$ such that there exists a graph cover $\widetilde{G}$ of degree m and a codeword

$$\left(c_{(1,1)}, \ldots, c_{(1,m)}; \ldots; c_{(n,1)}, \ldots, c_{(n,m)}\right) \in \mathbb{F}_2^{mn}$$

of $C(\widetilde{G})$ such that $p_i = |\{l \mid c_{(i,l)} = 1\}|$ for all i.

A pseudocodeword corresponds to a codeword of the graph cover.

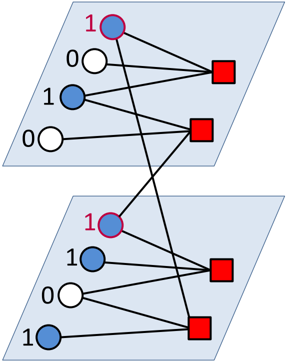


For example,

$$(2, 1, 1, 1)$$

is a pseudocodeword of $C(H)$.

A pseudocodeword corresponds to a codeword of the graph cover.

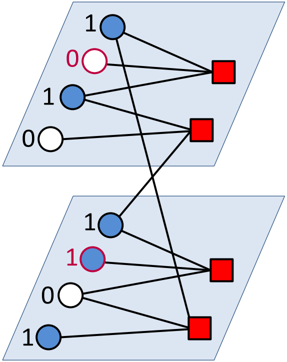


For example,

$$(2, 1, 1, 1)$$

is a pseudocodeword of $C(H)$.

A pseudocodeword corresponds to a codeword of the graph cover.

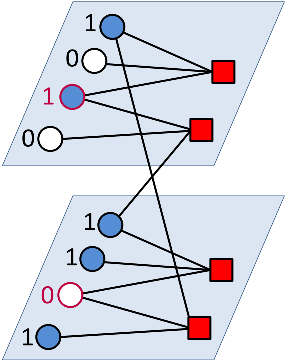


For example,

$$(2, 1, 1, 1)$$

is a pseudocodeword of $C(H)$.

A pseudocodeword corresponds to a codeword of the graph cover.

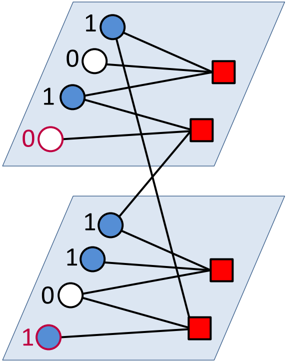


For example,

$$(2, 1, 1, 1)$$

is a pseudocodeword of $C(H)$.

A pseudocodeword corresponds to a codeword of the graph cover.

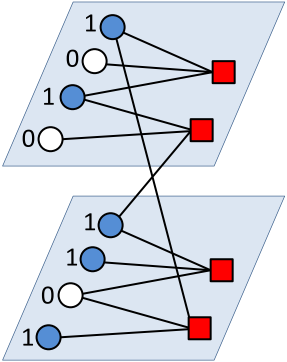


For example,

$$(2, 1, 1, 1)$$

is a pseudocodeword of $C(H)$.

A pseudocodeword corresponds to a codeword of the graph cover.

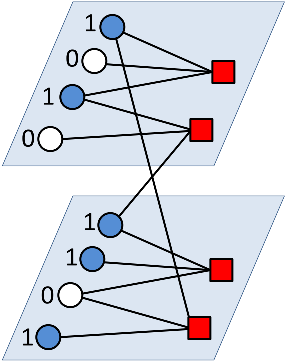


For example,

$$(2, 1, 1, 1)$$

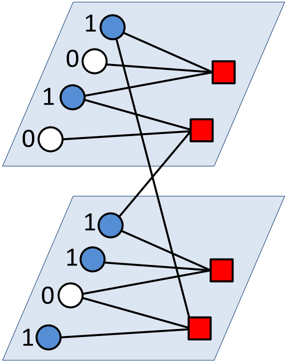
is a pseudocodeword of $C(H)$.

Some pseudocodewords are sums of codewords.



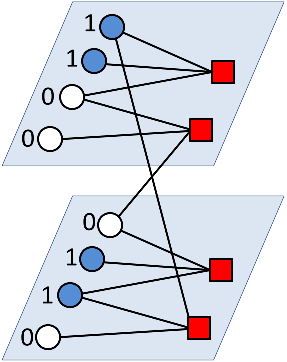
$$(2, 1, 1, 1)$$
$$= (1, 0, 1, 0) + (1, 1, 0, 1)$$

Some pseudocodewords are sums of codewords.



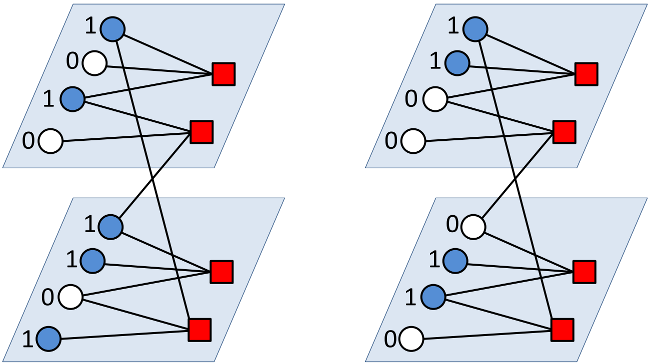
$(2, 1, 1, 1)$
$= (1, 0, 1, 0) + (1, 1, 0, 1)$

Some are not.



$(1, 2, 1, 0)$
$\neq$ any sum of codewords

Some pseudocodewords are sums of codewords. Some are not.



$(2,1,1,1)$
$= (1,0,1,0) + (1,1,0,1)$

$(1,2,1,0)$
$\neq$ any sum of codewords

Let

$$P(H) = \{\mathbf{p} \in \mathbb{N}^n \mid \mathbf{p} \text{ is a pseudocodeword of } C(H)\}.$$

Given H, the set of pseudocodewords can be completely characterized via the fundamental cone.

Definition

The **fundamental cone** of a parity check matrix $H \in \mathbb{F}_2^{r \times n}$ is

$$\mathcal{K}(H) = \left\{ \mathbf{v} = (v_1, \ldots, v_n) \in \mathbb{R}^n \mid v_i \geq 0 \text{ and } Row_j(H)\mathbf{v}^T \geq 2h_{ji}v_i \; \forall i, j \right\}.$$

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

Given $H \in \mathbb{F}_2^{r \times n}$, $\mathbf{p} \in \mathbb{N}^n$ is a pseudocodeword of $C(H)$ if and only if

- $\mathbf{p} \in \mathcal{K}(H)$, and
- $\mathbf{p}$ reduces mod 2 to a codeword.

The generating function for the pseudocodewords of a cycle code is an edge zeta function.

Consider the pseudocodeword enumerator

$$\sum_{\mathbf{v} \in P(H)} \mathbf{x}^{\mathbf{v}}$$

The generating function for the pseudocodewords of a cycle code is an edge zeta function.

Consider the generating function

$$\sum_{\mathbf{v} \in P(H)} \mathbf{x}^{\mathbf{v}}$$

where $\mathbf{x}^{\mathbf{v}} = x_1^{v_1} \dots x_n^{v_n}$ for $\mathbf{v} = (v_1, \dots, v_n) \in \mathbb{N}^n$.

The generating function for the pseudocodewords of a cycle code is an edge zeta function.

Consider the generating function

$$\sum_{\mathbf{v} \in P(H)} \mathbf{x}^{\mathbf{v}}$$

where $\mathbf{x}^{\mathbf{v}} = x_1^{v_1} \dots x_n^{v_n}$ for $\mathbf{v} = (v_1, \dots, v_n) \in \mathbb{N}^n$.

Theorem (Koetter et al. 2007)

If $C(H)$ is a cycle code, a code for which the parity check matrix H has exactly two 1's in each column, then the following are equivalent:

- $(p_1, p_2, \dots, p_n)$ is a pseudocodeword of $C(H)$.
- $x_1^{p_1} x_2^{p_2} \dots x_n^{p_n}$ has a nonzero coefficient in the power series expansion of the edge zeta function $\zeta(x_1, x_2, \dots, x_n)$ of the normal graph of H, which is a rational function.

The generating function for the pseudocodewords of a cycle code is an edge zeta function.

Consider the generating function

$$\sum_{\mathbf{v}\in P(H)} \mathbf{x}^{\mathbf{v}}$$

where $\mathbf{x}^{\mathbf{v}} = x_1^{v_1} \dots x_n^{v_n}$ for $\mathbf{v} = (v_1, \dots, v_n) \in \mathbb{N}^n$.

Theorem (Koetter et al. 2007)

If $C(H)$ is a cycle code, a code for which the parity check matrix H has exactly two 1's in each column, then the following are equivalent:

- $(p_1, p_2, \dots, p_n)$ is a pseudocodeword of $C(H)$.
- $x_1^{p_1} x_2^{p_2} \dots x_n^{p_n}$ has a nonzero coefficient in the power series expansion of the edge zeta function $\zeta(x_1, x_2, \dots, x_n)$ of the normal graph of H, which is a rational function.

Question: Can we generalize this?

The set of pseudocodewords itself does not form a cone.

Theorem

The generating function of integer points in a rational cone is a rational function.

The set of pseudocodewords itself does not form a cone.

Theorem

The generating function of integer points in a rational cone is a rational function.

Theorem (Koetter et al. 2007)

Given $H \in \mathbb{F}_2^{r\times n}$, $\mathbf{p} \in \mathbb{N}^n$ is a pseudocodeword of $C(H)$ if and only if

- $\mathbf{p} \in \mathcal{K}(H)$, and
- $\mathbf{p}$ reduces mod 2 to a codeword.

The lifted fundamental cone has as a projection $\mathcal{K}(H)$.

Definition

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

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

The lifted fundamental cone has as a projection $\mathcal{K}(H)$.

Definition

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

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

The lifted fundamental cone has as a projection $\mathcal{K}(H)$.

Definition

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

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

The lifted fundamental cone has as a projection $\mathcal{K}(H)$.

Definition

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

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

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

The lifted fundamental cone has as a projection $\mathcal{K}(H)$.

Definition

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

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

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

Proposition

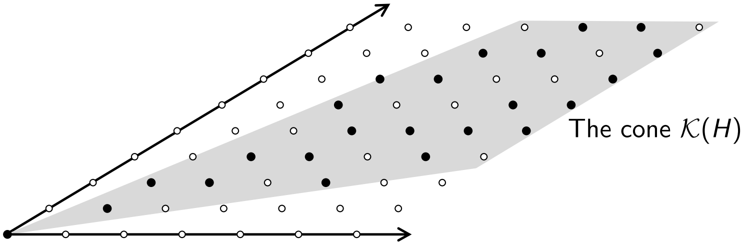
Let $H \in \mathbb{F}_2^{r\times n}$. Then

$$\pi\left(\hat{\mathcal{K}}(H)\right) = \mathcal{K}(H),$$

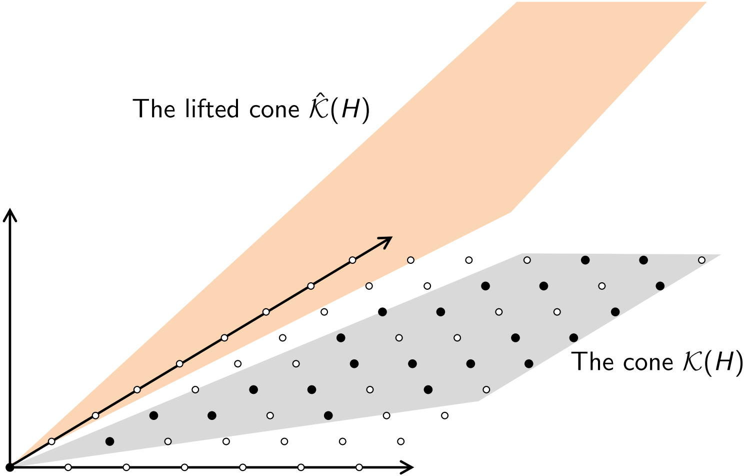
and the set of pseudocodewords of $C(H)$ is

$$\pi\left(\hat{\mathcal{K}}(H) \cap \mathbb{Z}^{n+r}\right) = P(H).$$

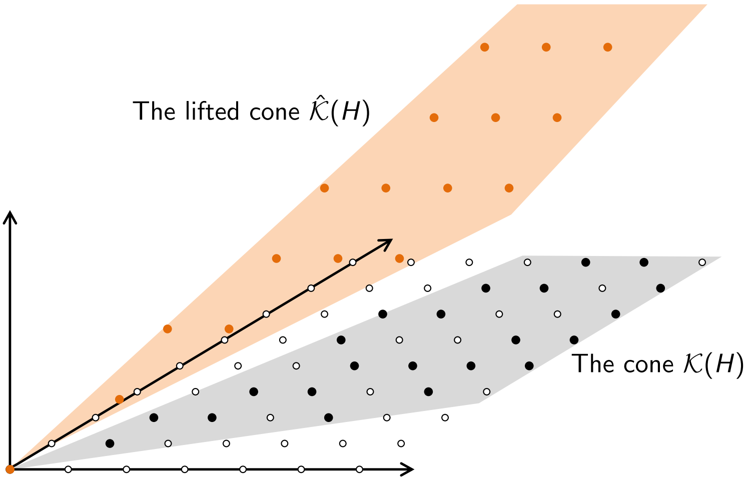
The lifted fundamental cone sieves out precisely the the lattice points in $\mathcal{K}(H)$ which are not pseudocodewords.



The lifted fundamental cone sieves out precisely the the lattice points in $\mathcal{K}(H)$ which are not pseudocodewords.



The lifted fundamental cone sieves out precisely the the lattice points in $\mathcal{K}(H)$ which are not pseudocodewords.



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

Theorem

The generating function of integer points in a rational cone is a rational function.

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

Theorem

The generating function of integer points in a rational cone is a rational function.

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

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

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

Theorem

The generating function of integer points in a rational cone is a rational function.

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

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

Specializing the above function at $(x_1, \ldots, x_n, 1, \ldots, 1)$ yields

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

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

Theorem

The generating function of integer points in a rational cone is a rational function.

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

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

Specializing the above function at $(x_1, \ldots, x_n, 1, \ldots, 1)$ yields

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

Theorem (K. and Matthews 2011)

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

Barvinok's algorithm produces the generating function of a cone as a rational function.

Theorem (Barvinok 1994)

Fix the dimension d. Given a rational cone $K \subset \mathbb{R}^d$, there exists a polynomial time algorithm which computes the generating function $\sum_{\mathbf{a} \in K \cap \mathbb{Z}^d} \mathbf{x}^{\mathbf{a}}$ of the form

$$\sum_{i \in I} \epsilon_i \frac{\mathbf{x}^{\mathbf{u}_i}}{(1 - \mathbf{x}^{\mathbf{u}_{i1}}) \cdots (1 - \mathbf{x}^{\mathbf{u}_{id}})}$$

where $\epsilon_i \in \{1, -1\}$, and $\mathbf{u}_i, \mathbf{u}_{ij}$ are integer vectors.

Example

Consider the code $C(H)$ where $H = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \end{pmatrix}$.

Example

Consider the code $C(H)$ where $H = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \end{pmatrix}$.

Barvinok 0.27 computes

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

Example

Consider the code $C(H)$ where $H = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \end{pmatrix}$.

Barvinok 0.27 computes

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

Example

Consider the code $C(H)$ where $H = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \end{pmatrix}$.

Barvinok 0.27 computes

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

The pseudocodewords of $C(H)$ are

$$(0,0,0,0), (1,0,1,0), (0,1,1,1), (1,1,0,1), \dots$$

Example

Consider the code $C(H)$ where $H = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \end{pmatrix}$.

Barvinok 0.27 computes

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

$$= 1 + x_1 x_3 + x_2 x_3 x_4 + x_1 x_2 x_4 + x_1 x_3 x_4^2 + x_1 x_2^2 x_3 + \dots$$

The pseudocodewords of $C(H)$ are

$$(0,0,0,0), (1,0,1,0), (0,1,1,1), (1,1,0,1), (1,0,1,2), (1,2,1,0), \dots$$

Example

Consider the code $C(H)$ where $H = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \end{pmatrix}$.

Barvinok 0.27 computes

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

$$= 1 + x_1 x_3 + x_2 x_3 x_4 + x_1 x_2 x_4 + x_1 x_3 x_4^2 + x_1 x_2^2 x_3 + x_1^2 x_3^2 + x_1^2 x_2 x_3 x_4 + x_1 x_2 x_3^2 x_4 + \dots$$

The pseudocodewords of $C(H)$ are

$$(0,0,0,0), (1,0,1,0), (0,1,1,1), (1,1,0,1), (1,0,1,2), (1,2,1,0),$$

$$(2,0,2,0), (2,1,1,1), (1,1,2,1), \dots$$

Example

Consider the code $C(H)$ of length 7 and dimension 2 given by a parity check matrix

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

Barvinok 0.27 computes

$$\sum_{\mathbf{p} \in P(H)} \mathbf{x}^{\mathbf{p}} = \frac{1}{(1 - x_1 x_2 x_3)(1 - x_1 x_2 x_3 x_4^2 x_5 x_6 x_7)(1 - x_5 x_6 x_7)}.$$

Example

Consider the code $C(H)$ of length 7 and dimension 2 given by a parity check matrix

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

Barvinok 0.27 computes

$$\sum_{\mathbf{p} \in P(H)} \mathbf{x}^{\mathbf{p}} = \frac{1}{(1 - x_1 x_2 x_3)(1 - x_1 x_2 x_3 x_4^2 x_5 x_6 x_7)(1 - x_5 x_6 x_7)}.$$

This gives a complete characterization of the pseudocodewords of $C(H)$; that is, $P(H)$ can be written as

$$\{a(1,1,1,0,0,0,0) + b(1,1,1,2,1,1,1) + c(0,0,0,0,1,1,1) \mid a, b, c \in \mathbb{Z}\}.$$

Irreducible pseudocodewords are those most likely to cause decoding error.

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

Given a parity check matrix $H \in \mathbb{F}_2^{r \times n}$, let $\mathrm{Irr}(H)$ denote the set of irreducible pseudocodewords of $C(H)$.

Irreducible pseudocodewords are those most likely to cause decoding error.

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

Given a parity check matrix $H \in \mathbb{F}_2^{r\times n}$, let $\mathrm{Irr}(H)$ denote the set of irreducible pseudocodewords of $C(H)$.

The **Hilbert basis** of an additive semigroup $(G,+)$ is the minimal set of elements $\{b_1,\ldots,b_t\}$ such that

$$G = \{\lambda_1 b_1, + \ldots + \lambda_t b_t \mid \lambda_1,\ldots,\lambda_t \in \mathbb{N}\}.$$

Irreducible pseudocodewords are those most likely to cause decoding error.

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

Given a parity check matrix $H \in \mathbb{F}_2^{r \times n}$, let $\mathrm{Irr}(H)$ denote the set of irreducible pseudocodewords of $C(H)$.

The **Hilbert basis** of an additive semigroup $(G, +)$ is the minimal set of elements $\{b_1, \ldots, b_t\}$ such that

$$G = \{\lambda_1 b_1, + \ldots + \lambda_t b_t \mid \lambda_1, \ldots, \lambda_t \in \mathbb{N}\}.$$

Theorem (K. and Matthews 2011)

Given $H \in \mathbb{F}_2^{r \times n}$, the set of integer points in the lifted fundamental cone $\hat{\mathcal{K}}(H)$ forms a semigroup under addition.

Furthermore, if $\mathfrak{B}$ is the Hilbert basis of $\hat{\mathcal{K}}(H) \cap \mathbb{Z}^{n+r}$, then

$$\mathrm{Irr}(H) = \pi(\mathfrak{B}).$$

Example

Consider the simplex code of length 7 and dimension 3 with two choices for parity check matrix

$$H_1 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & \color{red}{1} & \color{red}{0} & \color{red}{0} & 1 \end{pmatrix} \text{ and } H_2 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & \color{red}{0} & \color{red}{1} & \color{red}{1} & 1 \end{pmatrix}.$$

Example

Consider the simplex code of length 7 and dimension 3 with two choices for parity check matrix

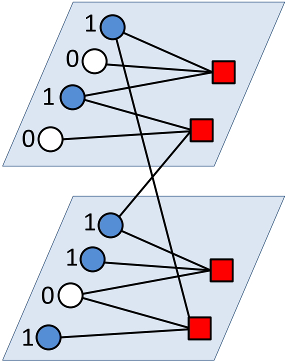
$$H_1 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & \color{red}{1} & \color{red}{0} & \color{red}{0} & 1 \end{pmatrix} \text{ and } H_2 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & \color{red}{0} & \color{red}{1} & \color{red}{1} & 1 \end{pmatrix}.$$

The noncodeword irreducible pseudocodewords, found using 4ti2, are

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

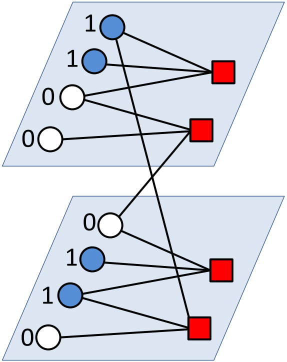
Question: Given a code C, what is a good choice of parity check matrix for C?

Some pseudocodewords are sums of codewords.



$(2, 1, 1, 1)$
$= (1, 0, 1, 0) + (1, 1, 0, 1)$

Some are not.



$(1, 2, 1, 0)$
$\neq$ any sum of codewords

A geometrically perfect code has no "bad" pseudocodewords.

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

$$\left\{ \sum_{\mathbf{c} \in C(H)} a_{\mathbf{c}} \mathbf{c} \;\middle|\; a_{\mathbf{c}} \in \mathbb{N} \right\} \subseteq P(H),$$

where $\sum_{\mathbf{c} \in C(H)} a_{\mathbf{c}} \mathbf{c} \in \mathbb{N}^n$.

If the equality holds, $C(H)$ is called **geometrically perfect**.

A geometrically perfect code has no “bad” pseudocodewords.

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

$$\left\{ \sum_{\mathbf{c}\in C(H)} a_{\mathbf{c}}\mathbf{c} \;\middle|\; a_{\mathbf{c}} \in \mathbb{N} \right\} \subseteq P(H),$$

where $\sum_{\mathbf{c}\in C(H)} a_{\mathbf{c}}\mathbf{c} \in \mathbb{N}^n$.

If the equality holds, $C(H)$ is called **geometrically perfect**.

Lemma (Wiberg 1996)

If H is cycle-free, then $C(H)$ is geometrically perfect.

A geometrically perfect code has no "bad" pseudocodewords.

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

$$\left\{ \sum_{\mathbf{c} \in C(H)} a_{\mathbf{c}} \mathbf{c} \;\middle|\; a_{\mathbf{c}} \in \mathbb{N} \right\} \subseteq P(H),$$

where $\sum_{\mathbf{c} \in C(H)} a_{\mathbf{c}} \mathbf{c} \in \mathbb{N}^n$.

If the equality holds, $C(H)$ is called **geometrically perfect**.

Lemma (Wiberg 1996)

If H is cycle-free, then $C(H)$ is geometrically perfect.

Sketch of Proof: A graph cover of H is disconnected copies of H.

Theorem (K. 2012)

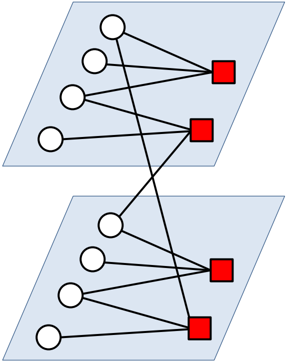
If C is a code which can be represented by a cycle-free parity check matrix, then the following are equivalent:

1. $C(H)$ is geometrically perfect.
2. There exist rows $s_1, s_2, \ldots, s_t$ of H such that

$$T = H - \{s_1, s_2, \ldots, s_t\}$$

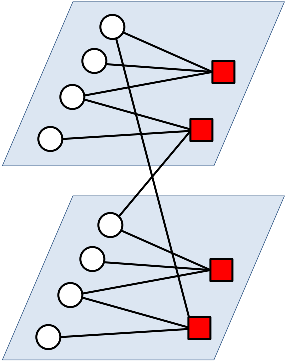
   is cycle-free and $C(T) = C(H)$.

To study pseudocodewords, we define pseudocheck.

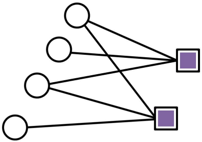


We would like to “collapse” the graph cover.

To study pseudocodewords, we define pseudocheck.



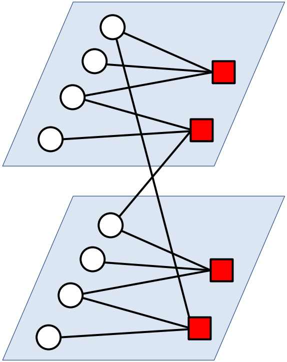
We would like to "collapse" the graph cover.



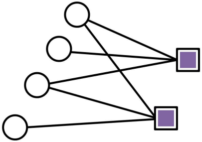
A **pseudocheck** node is satisfied if and only if the integer value assignment $(a_1, \ldots, a_t)$ to its neighbors satisfies the following conditions:

- $a_i \geq 0$ for all i,
- $\sum_{j=1}^{t} a_j = 0 \mod 2$, and
- $\sum_{j=1, j\neq i}^{t} a_j \geq a_i$ for all i.

To study pseudocodewords, we define pseudocheck.



We would like to "collapse" the graph cover.



An integer vector **p** is a pseudocodeword if and only if every pseudocheck is satisfied.

Theorem

If C is a code which can be represented by a cycle-free parity check matrix, then the following are equivalent:

1. $C(H)$ is geometrically perfect.
2. There exist rows $s_1, s_2, \ldots, s_t$ of H such that
   $T = H - \{s_1, s_2, \ldots, s_t\}$ is cycle-free and $C(T) = C(H)$.

Proof (2. $\Rightarrow$ 1.):

Theorem

If C is a code which can be represented by a cycle-free parity check matrix, then the following are equivalent:

1. $C(H)$ is geometrically perfect.
2. There exist rows $s_1, s_2, \ldots, s_t$ of H such that $T = H - \{s_1, s_2, \ldots, s_t\}$ is cycle-free and $C(T) = C(H)$.

Lemma (Kelley and Sridhara 2007)

If $T = H - \{s_1, s_2, \ldots, s_t\}$, then $P(H) \subseteq P(T)$.

Proof (2. $\Rightarrow$ 1.):

Theorem

If C is a code which can be represented by a cycle-free parity check matrix, then the following are equivalent:

1. $C(H)$ is geometrically perfect.
2. There exist rows $s_1, s_2, \ldots, s_t$ of H such that $T = H - \{s_1, s_2, \ldots, s_t\}$ is cycle-free and $C(T) = C(H)$.

Lemma (Kelley and Sridhara 2007)

If $T = H - \{s_1, s_2, \ldots, s_t\}$, then $P(H) \subseteq P(T)$.

Proof (2. $\Rightarrow$ 1.): If T is cycle free and $C(T) = C(H)$, then

$$P(H) \subseteq P(T) = \left\{ \sum_{\mathbf{c} \in C(T)} a_{\mathbf{c}} \mathbf{c} \;\middle|\; a_{\mathbf{c}} \in \mathbb{N} \right\} = \left\{ \sum_{\mathbf{c} \in C(H)} a_{\mathbf{c}} \mathbf{c} \;\middle|\; a_{\mathbf{c}} \in \mathbb{N} \right\} \subseteq P(H).$$

Therefore, $C(H)$ is geometrically perfect.

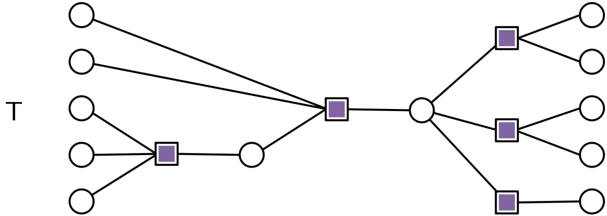
Theorem

If C is a code which can be represented by a cycle-free parity check matrix, then the following are equivalent:

1. $C(H)$ is geometrically perfect.
2. There exist rows $s_1, s_2, \ldots, s_t$ of H such that $T = H - \{s_1, s_2, \ldots, s_t\}$ is cycle-free and $C(T) = C(H)$.

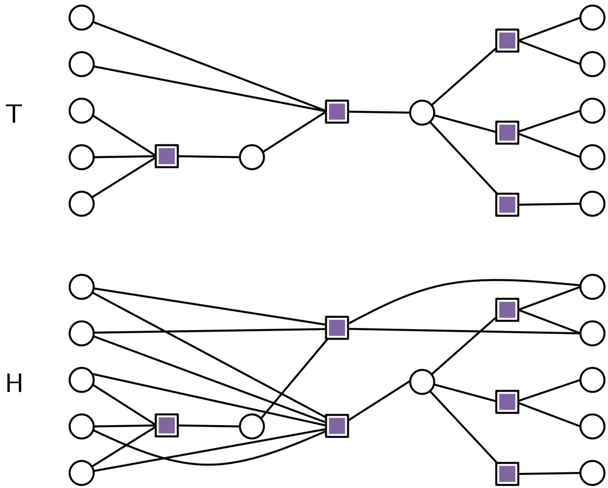
Proof (1. $\Rightarrow$ 2.): Fix a cycle-free parity check matrix T with the smallest number of edges such that $C(T) = C(H)$.

The code C can be represented by a cycle-free parity check matrix T.

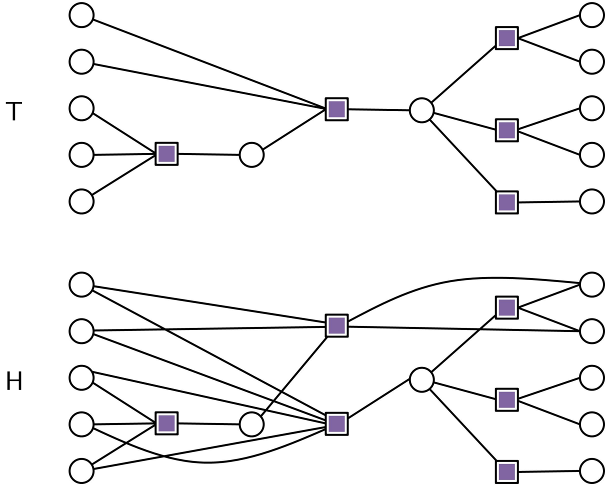


Proof (1. $\Rightarrow$ 2.): Fix a cycle-free parity check matrix T with the smallest number of edges such that $C(T) = C(H)$.

The code C can be represented by a cycle-free parity check matrix T.

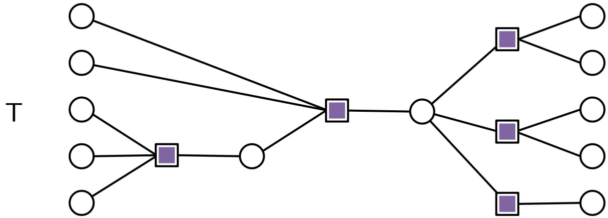


The code C can be represented by a cycle-free parity check matrix T.



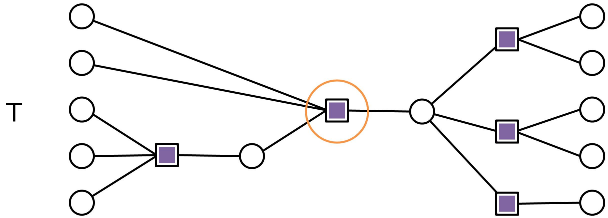
There exists a pseudocheck in T which is not in H.

A pivotal pseudocheck in T cannot be replaced by a pseudocheck from H.



A pseudocheck u of T is **pivotal** if there does not exist a pseudocheck h of H such that $(T - \{u\}) \cup \{h\}$ is cycle-free and $C((T - \{u\}) \cup \{h\}) = C(T)$.

A pivotal pseudocheck in T cannot be replaced by a pseudocheck from H.

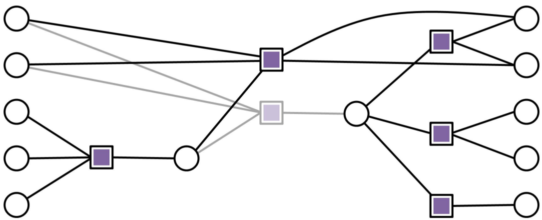


A pseudocheck u of T is **pivotal** if there does not exist a pseudocheck h of H such that $(T - \{u\}) \cup \{h\}$ is cycle-free and $C((T - \{u\}) \cup \{h\}) = C(T)$.

Fact

There exists a pivotal pseudocheck.

A pivotal pseudocheck in T cannot be replaced by a pseudocheck from H.

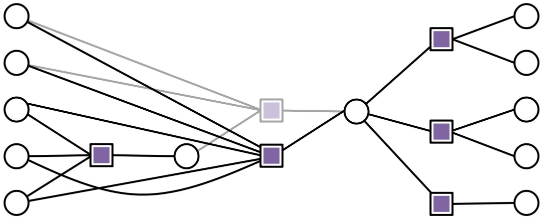


A pseudocheck u of T is **pivotal** if there does not exist a pseudocheck h of H such that $(T - \{u\}) \cup \{h\}$ is cycle-free and $C((T - \{u\}) \cup \{h\}) = C(T)$.

Fact

There exists a pivotal pseudocheck.

A pivotal pseudocheck in T cannot be replaced by a pseudocheck from H.

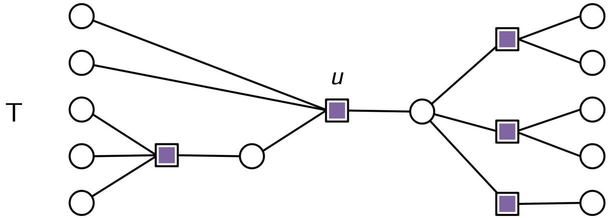


A pseudocheck u of T is **pivotal** if there does not exist a pseudocheck h of H such that $(T - \{u\}) \cup \{h\}$ is cycle-free and $C((T - \{u\}) \cup \{h\}) = C(T)$.

Fact

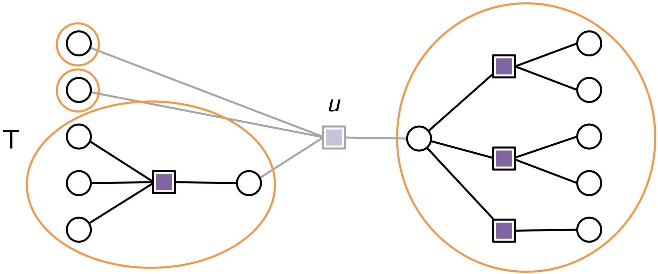
There exists a pivotal pseudocheck.

Construct a pseudocodeword from a pivotal pseudocheck.



Fix a pivotal pseudocheck u.

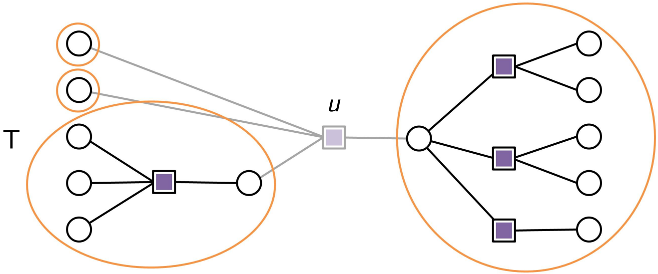
Construct a pseudocodeword from a pivotal pseudocheck.



Fix a pivotal pseudocheck u.

The graph of $T - \{u\}$ has several connected components.

Construct a pseudocodeword from a pivotal pseudocheck.



Fix a pivotal pseudocheck u.
The graph of $T - \{u\}$ has several connected components.
Assign the value $2\deg(u)$ to the bit nodes in one component, 0 to the bit nodes adjacent to a pseudocheck of degree 1, and 2 to all other bit nodes.

Claim

The assignment is valid for H, but not for T.

Proof of Claim: The assignment is invalid for T because it violates pseudocheck u.

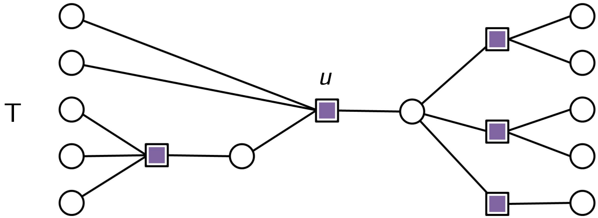
Fact

Suppose that $\tau_1, \ldots, \tau_{\deg(u)}$ are connected components in $T - \{u\}$ where u is adjacent to a bit node in $\tau_1, \ldots, \tau_{\deg(u)}$ in T.
For any pseudocheck h in H, if $\deg\left(h\mid_{\tau_i}\right) \geq 1$ for some i, then either

- $\deg\left(h\mid_{\tau_i}\right) \geq 2$, or
- $\deg(h) \geq \deg(u) + 1$.

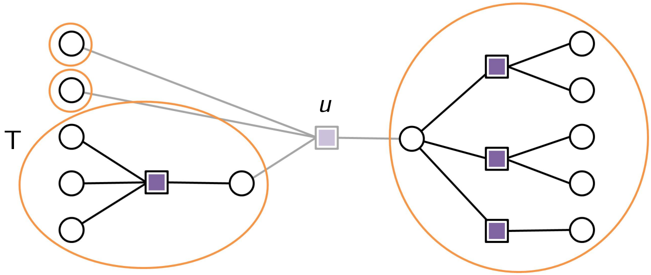
So, the assignment is valid for H.

Example



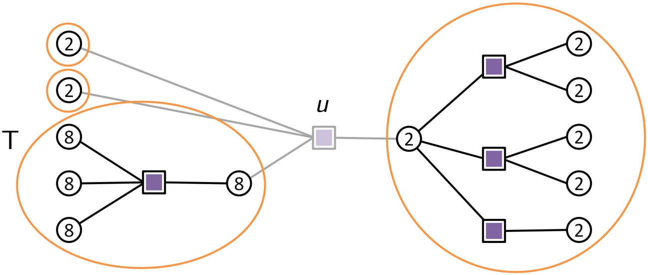
The pivotal pseudocheck u has degree 4.

Example



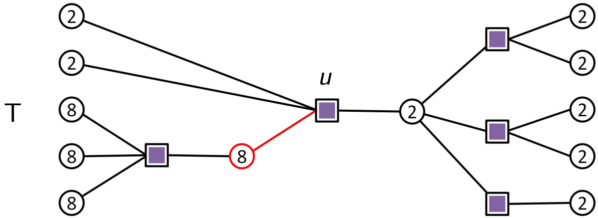
The pivotal pseudocheck u has degree 4.
There are 4 connected components in $T - \{u\}$.

Example



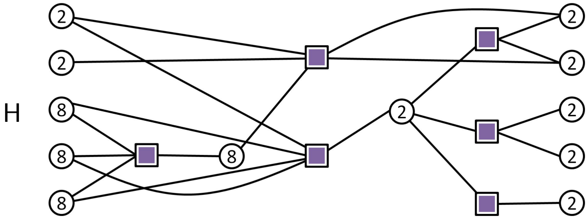
The pivotal pseudocheck u has degree 4.
There are 4 connected components in $T - \{u\}$.
Assign the value $8 = 2\deg(u)$ to the bit nodes in one component, and 2 to all other bit nodes.

Example



The pivotal pseudocheck u has degree 4.
There are 4 connected components in $T - \{u\}$.
Assign the value $8 = 2\deg(u)$ to the bit nodes in one component, and 2 to all other bit nodes.
The assignment violates pseudocheck u.

Example



The pivotal pseudocheck u has degree 4.
There are 4 connected components in $T - \{u\}$.
Assign the value $8 = 2\deg(u)$ to the bit nodes in one component, and 2 to all other bit nodes.
The assignment violates pseudocheck u.
The assignment is valid for H.

Theorem (K. 2012)

If C is a code which can be represented by a cycle-free parity check matrix, then the following are equivalent:

1. $C(H)$ is geometrically perfect.
2. There exist rows $s_1, s_2, \ldots, s_t$ of H such that

$$T = H - \{s_1, s_2, \ldots, s_t\}$$

   is cycle-free and $C(T) = C(H)$.

Theorem (K. 2012)

If C is a code which can be represented by a cycle-free parity check matrix, then the following are equivalent:

1. $C(H)$ is geometrically perfect.
2. There exist rows $s_1, s_2, \dots, s_t$ of H such that

$$T = H - \{s_1, s_2, \dots, s_t\}$$

is cycle-free and $C(T) = C(H)$.

- There exists a class of parity check matrices with many small cycles that perform well under iterative decoders.

Theorem (K. 2012)

If C is a code which can be represented by a cycle-free parity check matrix, then the following are equivalent:

1. $C(H)$ is geometrically perfect.
2. There exist rows $s_1, s_2, \ldots, s_t$ of H such that

$$T = H - \{s_1, s_2, \ldots, s_t\}$$

is cycle-free and $C(T) = C(H)$.

- There exists a class of parity check matrices with many small cycles that perform well under iterative decoders.
- A code C is capable of being geometrically perfect if and only if $\mathcal{H}_7^{\perp}$, $\mathcal{R}_{10}$, or $C(\mathcal{K}_5)^{\perp}$ cannot be obtained from C via a sequence of shortening and puncturing operations (Kashyap 2008).

Table of contents

Notation

$\mathbb{F}_p = \{0, 1, \ldots, p-1\}$ is the finite field with p elements where p is prime.
$\oplus$ and $\odot$ denote finite field addition and multiplication.

Notation

$\mathbb{F}_p = \{0, 1, \ldots, p-1\}$ is the finite field with p elements where p is prime. $\oplus$ and $\odot$ denote finite field addition and multiplication.

A **linear code** C of length n and dimension k over $\mathbb{F}_p$ is a subspace of $\mathbb{F}_p^n$ of dimension k.

A **parity check matrix** of a code C is any matrix $H \in \mathbb{F}_p^{r \times n}$ such that C is the null space of H.

Given a parity check matrix H of C and $\mathbf{y} \in \mathbb{F}_p^n$,

$$\mathbf{y} \in C \text{ if and only if } H \odot \mathbf{y}^T = \mathbf{0} \in \mathbb{F}_p^{r \times 1}.$$

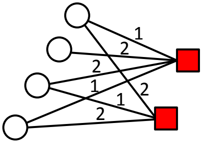
Denote $C(H)$ the code given by a parity-check matrix H.

The Tanner graph of a nonbinary parity-check matrix is a graph with weighted edges.

$$H = \begin{pmatrix} 1 & 2 & 2 & 1 \\ 2 & 0 & 1 & 2 \end{pmatrix} \in \mathbb{F}_3^{2\times 4}$$

The Tanner graph of a nonbinary parity-check matrix is a graph with weighted edges.

$$H = \begin{pmatrix} 1 & 2 & 2 & 1 \\ 2 & 0 & 1 & 2 \end{pmatrix} \in \mathbb{F}_3^{2\times 4} \quad \Longleftrightarrow$$

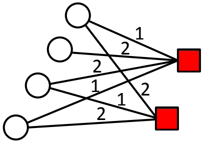


The bit nodes $X = \{x_1, \ldots, x_n\}$ correspond to a column of H, the check nodes $F = \{f_1, \ldots, f_r\}$ correspond to a row of H, and if $h_{ji} \neq 0$ then $\{x_i, f_j\}$ is an edge with weight

$$w(x_i, f_j) = h_{ji}.$$

The Tanner graph of a nonbinary parity-check matrix is a graph with weighted edges.

$$H = \begin{pmatrix} 1 & 2 & 2 & 1 \\ 2 & 0 & 1 & 2 \end{pmatrix} \in \mathbb{F}_3^{2\times 4} \quad \Longleftrightarrow$$



The bit nodes $X = \{x_1, \ldots, x_n\}$ correspond to a column of H, the check nodes $F = \{f_1, \ldots, f_r\}$ correspond to a row of H, and if $h_{ji} \neq 0$ then $\{x_i, f_j\}$ is an edge with weight

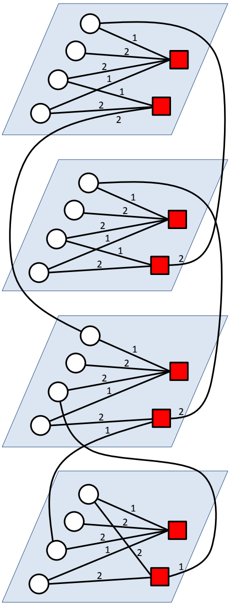
$$w(x_i, f_j) = h_{ji}.$$

A vector $\mathbf{c} = (c_1, c_2, \ldots, c_n) \in \mathbb{F}_p^n$ is a codeword of $C(H)$ if and only if

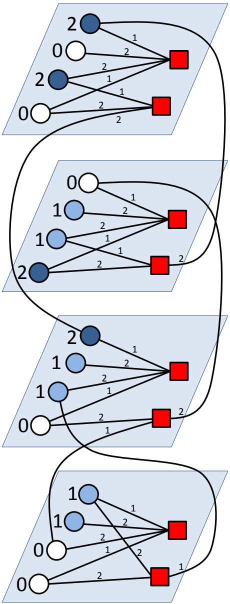
$$\sum_{i \in Nbhd(f_j)} w(x_i, f_j) \odot c_i = 0$$

for all j where the summation is taken over $\mathbb{F}_p$.

The Tanner graph of a nonbinary code permits graph covers similar to the Tanner graph of a binary code.



The Tanner graph of a nonbinary code permits graph covers similar to the Tanner graph of a binary code.

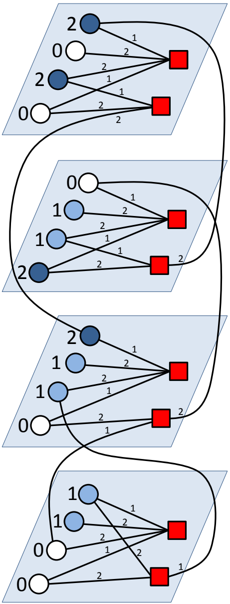


A pseudocodeword of $C(H)$ is a matrix $M \in \mathbb{Z}^{p-1 \times n}$ such that there exists a graph cover $\widetilde{G}$ and a codeword $\tilde{\mathbf{c}}$ of $C(\widetilde{G})$ where

$$m_{bi} := |\{1 \leq l \leq m \mid c_{(i,l)} = b\}|$$

for all b, i. We will also denote $m_i(b) := m_{bi}$.

The Tanner graph of a nonbinary code permits graph covers similar to the Tanner graph of a binary code.



A pseudocodeword of $C(H)$ is a matrix $M \in \mathbb{Z}^{p-1\times n}$ such that there exists a graph cover $\widetilde{G}$ and a codeword $\tilde{\mathbf{c}}$ of $C(\widetilde{G})$ where

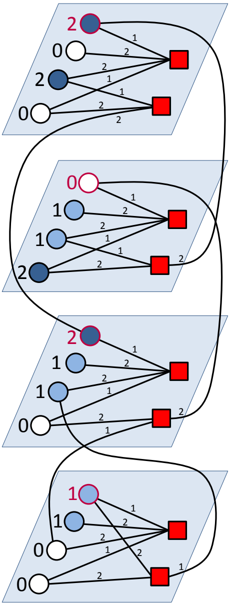
$$m_{bi} := \left|\{1 \leq l \leq m \mid c_{(i,l)} = b\}\right|$$

for all b, i. We will also denote $m_i(b) := m_{bi}$.

The pseudocodeword on the left can be represented by

$$M = \begin{pmatrix} 1 & 3 & 2 & 0 \\ 2 & 0 & 1 & 1 \end{pmatrix}.$$

The Tanner graph of a nonbinary code permits graph covers similar to the Tanner graph of a binary code.



A pseudocodeword of $C(H)$ is a matrix $M \in \mathbb{Z}^{p-1 \times n}$ such that there exists a graph cover $\widetilde{G}$ and a codeword $\tilde{\mathbf{c}}$ of $C(\widetilde{G})$ where

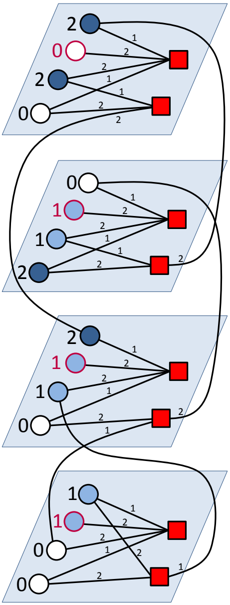
$$m_{bi} := \left|\{1 \leq l \leq m \mid c_{(i,l)} = b\}\right|$$

for all b, i. We will also denote $m_i(b) := m_{bi}$.

The pseudocodeword on the left can be represented by

$$M = \begin{pmatrix} 1 & 3 & 2 & 0 \\ 2 & 0 & 1 & 1 \end{pmatrix}.$$

The Tanner graph of a nonbinary code permits graph covers similar to the Tanner graph of a binary code.



A pseudocodeword of $C(H)$ is a matrix $M \in \mathbb{Z}^{p-1 \times n}$ such that there exists a graph cover $\widetilde{G}$ and a codeword $\tilde{\mathbf{c}}$ of $C(\widetilde{G})$ where

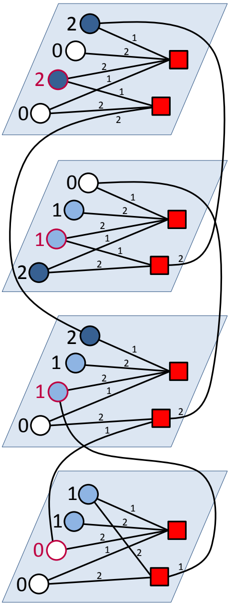
$$m_{bi} := \left|\{1 \leq l \leq m \mid c_{(i,l)} = b\}\right|$$

for all b, i. We will also denote $m_i(b) := m_{bi}$.

The pseudocodeword on the left can be represented by

$$M = \begin{pmatrix} 1 & 3 & 2 & 0 \\ 2 & 0 & 1 & 1 \end{pmatrix}.$$

The Tanner graph of a nonbinary code permits graph covers similar to the Tanner graph of a binary code.



A pseudocodeword of $C(H)$ is a matrix $M \in \mathbb{Z}^{p-1 \times n}$ such that there exists a graph cover $\widetilde{G}$ and a codeword $\tilde{\mathbf{c}}$ of $C(\widetilde{G})$ where

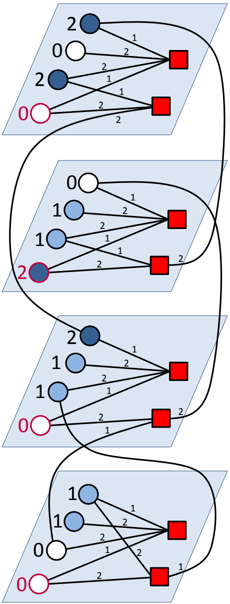
$$m_{bi} := |\{1 \leq l \leq m \mid c_{(i,l)} = b\}|$$

for all b, i. We will also denote $m_i(b) := m_{bi}$.

The pseudocodeword on the left can be represented by

$$M = \begin{pmatrix} 1 & 3 & 2 & 0 \\ 2 & 0 & 1 & 1 \end{pmatrix}.$$

The Tanner graph of a nonbinary code permits graph covers similar to the Tanner graph of a binary code.



A pseudocodeword of $C(H)$ is a matrix $M \in \mathbb{Z}^{p-1\times n}$ such that there exists a graph cover $\widetilde{G}$ and a codeword $\tilde{\mathbf{c}}$ of $C(\widetilde{G})$ where

$$m_{bi} := |\{1 \le l \le m \mid c_{(i,l)} = b\}|$$

for all b, i. We will also denote $m_i(b) := m_{bi}$.

The pseudocodeword on the left can be represented by

$$M = \begin{pmatrix} 1 & 3 & 2 & 0 \\ 2 & 0 & 1 & 1 \end{pmatrix}.$$

We aim to generalize the binary fundamental cone.

Definition of Binary Fundamental Cone

The fundamental cone of a parity check matrix $H \in \mathbb{F}_2^{r\times n}$ is

$$\mathcal{K}(H) = \left\{ \mathbf{v} = (v_1, \ldots, v_n) \in \mathbb{R}^n \mid v_i \geq 0 \text{ and } Row_j(H)\mathbf{v}^T \geq 2h_{ji}v_i \;\forall i, j \right\}.$$

We aim to generalize the binary fundamental cone.

Definition of Binary Fundamental Cone

The fundamental cone of a parity check matrix $H \in \mathbb{F}_2^{r \times n}$ is

$$\mathcal{K}(H) = \left\{ \mathbf{v} = (v_1, \ldots, v_n) \in \mathbb{R}^n \mid v_i \geq 0 \text{ and } Row_j(H)\mathbf{v}^T \geq 2h_{ji}v_i \; \forall i, j \right\}.$$

We aim to generalize the binary fundamental cone.

Definition of Binary Fundamental Cone

The fundamental cone of a parity check matrix $H \in \mathbb{F}_2^{r \times n}$ is

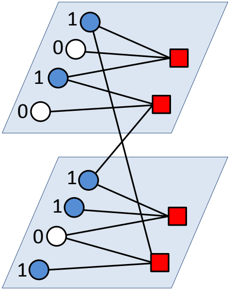
$$\mathcal{K}(H) = \left\{ \mathbf{v} = (v_1, \ldots, v_n) \in \mathbb{R}^n \mid v_i \geq 0 \text{ and } Row_j(H)\mathbf{v}^T \geq 2h_{ji}v_i \; \forall i, j \right\}.$$

We aim to generalize the binary fundamental cone.

Definition of Binary Fundamental Cone

The fundamental cone of a parity check matrix $H \in \mathbb{F}_2^{r \times n}$ is

$$\mathcal{K}(H) = \left\{ \mathbf{v} = (v_1, \ldots, v_n) \in \mathbb{R}^n \mid v_i \geq 0 \text{ and } Row_j(H)\mathbf{v}^T \geq 2h_{ji}v_i \; \forall i, j \right\}.$$



If $\mathbf{v}$ is a pseudocodeword,

$$Row_j(H)\mathbf{v}^T = \sum_{i=1}^{n} h_{ji} v_i = \sum_{i=1}^{n} \sum_{l=1}^{m} h_{ji} c_{(i,l)}.$$

Consider

$$\Theta_j = \sum_{i=1}^{n} \sum_{l=1}^{m} h_{ji} \odot c_{(i,l)}$$

where $1 \leq j \leq r$.

Consider

$$\Theta_j = \sum_{i=1}^{n} \sum_{l=1}^{m} h_{ji} \odot c_{(i,l)}$$

where $1 \leq j \leq r$.

Consider

$$\Theta_j = \sum_{i=1}^{n} \sum_{l=1}^{m} h_{ji} \odot c_{(i,l)}$$

where $1 \leq j \leq r$. We can compute Θ_j as

$$\begin{aligned}
\Theta_j &= \sum_{i=1}^{n} \sum_{l=1}^{m} h_{ji} \odot c_{(i,l)} \\
&= \sum_{i=1}^{n} \sum_{b=1}^{p-1} \sum_{\{l \mid c_{(i,l)}=b\}} b \odot h_{ji} \\
&= \sum_{b=1}^{p-1} \sum_{i=1}^{n} \left(\left(b \odot h_{ji} \right) \cdot \sum_{\{l \mid c_{(i,l)}=b\}} 1 \right) \\
&= \sum_{b=1}^{p-1} \sum_{i=1}^{n} \left(b \odot h_{ji} \right) m_i(b) \\
&= \sum_{b=1}^{p-1} \left(b \odot Row_j(H) \right) Row_b(M)^T.
\end{aligned}$$

Multiple of a codeword is a codeword.

If $\mathbf{c}$ is a codeword of a p-ary code C, so is $a \odot \mathbf{c}$ where $a \in \mathbb{F}_p^*$.

Multiple of a codeword is a codeword.

If **c** is a codeword of a p-ary code C, so is $a \odot \mathbf{c}$ where $a \in \mathbb{F}_p^*$.

Definition

Define

$$\Theta_j(a, M) = \sum_{b=1}^{p-1} \Big(a \odot b \odot Row_j(H) \Big) Row_b(M)^T$$

where $a \in \mathbb{F}_p^*$, $1 \leq j \leq r$, and $M \in \mathbb{Z}^{(p-1)\times n}$.

Theta function gives bound similar to the binary case.

Recall the inequality $\frac{1}{2} Row_j(H)\mathbf{v}^T \geq h_{ji} v_i$

Proposition

Consider a parity-check matrix $H \in \mathbb{F}_p^{r \times n}$ where p is prime. If M is a pseudocodeword of $C(H)$, then

$$\frac{1}{p}\Theta_j(a, M) \geq m(1) + m(2) + \cdots + m(p-1)$$

where $1 \leq j \leq r$, $a \in \mathbb{F}_p$ and $i \in supp(Row_j(H))$.

Theta function gives bound similar to the binary case.

Recall the inequality $\frac{1}{2}Row_j(H)\mathbf{v}^T \geq h_{ji}v_i$

Proposition

Consider a parity-check matrix $H \in \mathbb{F}_p^{r \times n}$ where p is prime. If M is a pseudocodeword of $C(H)$, then

$$\frac{1}{p}\Theta_j(a, M) \geq m_i(1) + m_i(2) + \cdots + m_i(p-1)$$

where $1 \leq j \leq r$, $a \in \mathbb{F}_p^*$, and $i \in supp(Row_j(H))$.

Theta function gives bound similar to the binary case.

Recall the inequality $\frac{1}{2} Row_j(H)\mathbf{v}^T \geq h_{ji} v_i = v_i$ if $i \in supp(Row_j(H))$.

Proposition

Consider a parity-check matrix $H \in \mathbb{F}_p^{m \times n}$ where p is prime. If M is a pseudocodeword of $C(H)$, then

$$\frac{1}{p}\Theta_j(a, M) \geq m_i(1) + m_i(2) + \cdots + m_i(p-1)$$

where $1 \leq j \leq m$, $a \in \mathbb{F}_p$ and $i \in supp(Row_j(H))$.

Theta function gives bound similar to the binary case.

Recall the inequality $\frac{1}{2} Row_j(H)\mathbf{v}^T \geq h_{ji} v_i = v_i$ if $i \in supp(Row_j(H))$.

Proposition

Consider a parity-check matrix $H \in \mathbb{F}_p^{r \times n}$ where p is prime. If M is a pseudocodeword of $C(H)$, then

$$\frac{1}{p}\Theta_j(a, M) \geq m_i(1) + m_i(2) + \cdots + m_i(p-1)$$

where $1 \leq j \leq r$, $a \in \mathbb{F}_p$, and $i \in supp(Row_j(H))$.

Theta function gives bound similar to the binary case.

Recall the inequality $\frac{1}{2}Row_j(H)\mathbf{v}^T \geq h_{ji}v_i = v_i$ if $i \in supp(Row_j(H))$.

Proposition

Consider a parity-check matrix $H \in \mathbb{F}_p^{r \times n}$ where p is prime. If M is a pseudocodeword of $C(H)$, then

$$\frac{1}{p}\Theta_j(a, M) \geq m_i(1) + m_i(2) + \ldots + m_i(p-1)$$

where $1 \leq j \leq r$, $a \in \mathbb{F}_p$, and $i \in supp(Row_j(H))$.

Theta function gives bound similar to the binary case.

Recall the inequality $\frac{1}{2} Row_j(H)\mathbf{v}^T \geq h_{ji} v_i = v_i$ if $i \in supp(Row_j(H))$.

Proposition

Consider a parity-check matrix $H \in \mathbb{F}_p^{r \times n}$ where p is prime. If M is a pseudocodeword of $C(H)$, then

$$\frac{1}{p}\Theta_j(a, M) \geq m_i(1) + m_i(2) + \ldots + m_i(p-1)$$

where $1 \leq j \leq r$, $a \in \mathbb{F}_p^*$, and $i \in supp(Row_j(H))$.

Theta function gives bound similar to the binary case.

Recall the inequality $\frac{1}{2}Row_j(H)\mathbf{v}^T \geq h_{ji}v_i = v_i$ if $i \in supp(Row_j(H))$.

Proposition

Consider a parity-check matrix $H \in \mathbb{F}_p^{r\times n}$ where p is prime. If M is a pseudocodeword of $C(H)$, then

$$\frac{1}{p}\Theta_j(a, M) \geq m_i(1) + m_i(2) + \ldots + m_i(p-1)$$

where $1 \leq j \leq r$, $a \in \mathbb{F}_p^*$, and $i \in supp(Row_j(H))$.

Sketch of Proof:

$$\begin{aligned}\frac{1}{p}\Theta_j(a, M) &\geq \text{the minimum number of covers needed to realize } M\\ &\geq m_i(1) + m_i(2) + \ldots + m_i(p-1).\end{aligned}$$

Theta function gives bound similar to the binary case.

Recall the inequality $\frac{1}{2} Row_j(H)\mathbf{v}^T \geq h_{ji}v_i = v_i$ if $i \in supp(Row_j(H))$.

Proposition

Consider a parity-check matrix $H \in \mathbb{F}_p^{r\times n}$ where p is prime. If M is a pseudocodeword of $C(H)$, then

$$\frac{1}{p}\Theta_j(a, M) \geq m_i(1) + m_i(2) + \ldots + m_i(p-1)$$

where $1 \leq j \leq r$, $a \in \mathbb{F}_p^*$, and $i \in supp(Row_j(H))$.

Sketch of Proof:

$$\begin{aligned}\frac{1}{p}\Theta_j(a, M) &\geq \text{the minimum number of covers needed to realize } M \\ &\geq m_i(1) + m_i(2) + \ldots + m_i(p-1).\end{aligned}$$

This inequality, however, is insufficient.

Critical multiset is the "forbidden" configurations.

Definition

A multiset $\Gamma = \{\gamma_1, \ldots, \gamma_t\} \subseteq \mathbb{F}_p$ is critical if and only if $t > 1$ and

$$\sum_{\gamma_i \in \Gamma} \gamma_i > (t-1)p.$$

Critical multiset is the "forbidden" configurations.

Definition

A multiset $\Gamma = \{\gamma_1, \ldots, \gamma_t\} \subseteq \mathbb{F}_p$ is critical if and only if $t > 1$ and

$$\sum_{\gamma_i \in \Gamma} \gamma_i > (t-1)p.$$

There is no critical multiset over $\mathbb{F}_2$. The only critical multiset over $\mathbb{F}_3$ is $\{2, 2\}$. Critical multisets over $\mathbb{F}_5$ are:

$$\{2,4\}, \{3,3\}, \{3,4\}, \{4,4\}, \{3,4,4\}, \text{and } \{4,4,4\}.$$

Critical multiset is the "forbidden" configurations.

Definition

A multiset $\Gamma = \{\gamma_1, \ldots, \gamma_t\} \subseteq \mathbb{F}_p$ is critical if and only if $t > 1$ and

$$\sum_{\gamma_i \in \Gamma} \gamma_i > (t-1)p.$$

There is no critical multiset over $\mathbb{F}_2$. The only critical multiset over $\mathbb{F}_3$ is $\{2, 2\}$. Critical multisets over $\mathbb{F}_5$ are:

$$\{2,4\}, \{3,3\}, \{3,4\}, \{4,4\}, \{3,4,4\}, \text{and } \{4,4,4\}.$$

If a multiset $\Gamma = \{\gamma_1, \ldots, \gamma_t\} \subseteq \mathbb{F}_p$ is critical, then

$$\sum_{\gamma_i \in \Gamma} \gamma_i \neq 0 \mod p,$$

and any multisubset of Γ is critical.

Toward characterizing nonbinary pseudocodewords

Theorem (K. and Matthews 2012)

Let $H \in \mathbb{F}_p^{r\times n}$ where p is prime. If M is a pseudocodeword of $C(H)$, then

$$\frac{1}{p}\Theta_j(a,M) \geq m_i(1) + m_i(2) + \ldots + m_i(p-1),$$
$$\frac{1}{p}\Theta_j(a,M) \geq m_{i_1}\left(a \odot \gamma_1 \odot h_{ji_1}^{-1}\right) + \ldots + m_{i_t}\left(a \odot \gamma_t \odot h_{ji_t}^{-1}\right),$$
$$m_i(b) \geq 0,$$

and

$$H \odot M^T \odot \begin{pmatrix} 1 & 2 & \ldots & p-1 \end{pmatrix}^T = \mathbf{0}$$

for all $a, b \in \mathbb{F}_p^*$, $1 \leq j \leq r$, $i, i_1, \ldots, i_t \in supp(Row_j(H))$, and critical multiset $\{\gamma_1, \ldots, \gamma_t\}$.

Toward characterizing nonbinary pseudocodewords

Theorem (K. and Matthews 2012)

Let $H \in \mathbb{F}_p^{r \times n}$ where p is prime. If M is a pseudocodeword of $C(H)$, then

$$\frac{1}{p}\Theta_j(a, M) \geq m_i(1) + m_i(2) + \ldots + m_i(p-1),$$
$$\frac{1}{p}\Theta_j(a, M) \geq m_{i_1}\left(a \odot \gamma_1 \odot h_{ji_1}^{-1}\right) + \ldots + m_{i_t}\left(a \odot \gamma_t \odot h_{ji_t}^{-1}\right),$$
$$m_i(b) \geq 0,$$

and

$$H \odot M^T \odot \begin{pmatrix} 1 & 2 & \ldots & p-1 \end{pmatrix}^T = \mathbf{0}$$

for all $a, b \in \mathbb{F}_p^*$, $1 \leq j \leq r$, $i, i_1, \ldots, i_t \in supp(Row_j(H))$, and critical multiset $\{\gamma_1, \ldots, \gamma_t\}$.

- The converse is true for binary codes (Koetter et al. 2007) and ternary codes (Skachek 2010).

Toward characterizing nonbinary pseudocodewords

Theorem (K. and Matthews 2012)

Let $H \in \mathbb{F}_p^{r\times n}$ where p is prime. If M is a pseudocodeword of $C(H)$, then

$$\frac{1}{p}\Theta_j(a,M) \geq m_i(1) + m_i(2) + \ldots + m_i(p-1),$$
$$\frac{1}{p}\Theta_j(a,M) \geq m_{i_1}\left(a \odot \gamma_1 \odot h_{ji_1}^{-1}\right) + \ldots + m_{i_t}\left(a \odot \gamma_t \odot h_{ji_t}^{-1}\right),$$
$$m_i(b) \geq 0,$$

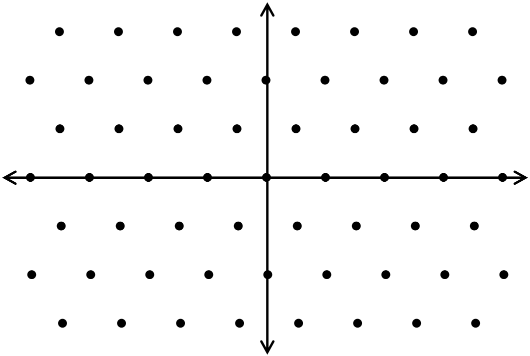
and

$$H \odot M^T \odot \begin{pmatrix} 1 & 2 & \ldots & p-1 \end{pmatrix}^T = \mathbf{0}$$

for all $a, b \in \mathbb{F}_p^*$, $1 \leq j \leq r$, $i, i_1, \ldots, i_t \in supp(Row_j(H))$, and critical multiset $\{\gamma_1, \ldots, \gamma_t\}$.

- The converse is true for binary codes (Koetter et al. 2007) and ternary codes (Skachek 2010).
- The number of inequalities is exponential in n.

Lattice codes are linear codes analog for continuous-valued AWGN channel.



A lattice is a discrete additive subgroup of $\mathbb{R}^n$.

We may apply iterative decoders to lattices constructed from codes using Construction A.

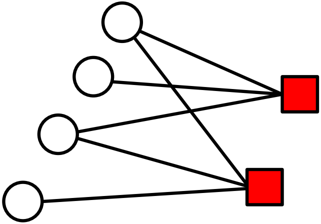
A binary code $C(H) \subseteq \mathbb{F}_2^n$ yields a lattice

$$\Lambda_A = \{\mathbf{v} \in \mathbb{Z}^n \mid \mathbf{v} \text{ reduces mod 2 to a codeword of } C(H)\}.$$

We may apply iterative decoders to lattices constructed from codes using Construction A.

A binary code $C(H) \subseteq \mathbb{F}_2^n$ yields a lattice

$$\Lambda_A = \{\mathbf{v} \in \mathbb{Z}^n \mid \mathbf{v} \text{ reduces mod 2 to a codeword of } C(H)\}.$$

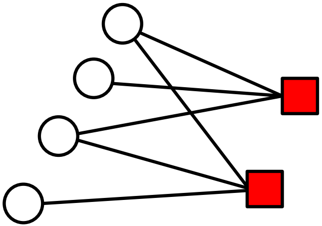


A check node is satisfied if the sum of its neighbors is an even integer.

We may apply iterative decoders to lattices constructed from codes using Construction A.

A binary code $C(H) \subseteq \mathbb{F}_2^n$ yields a lattice

$$\Lambda_A = \{\mathbf{v} \in \mathbb{Z}^n \mid \mathbf{v} \text{ reduces mod 2 to a codeword of } C(H)\}.$$

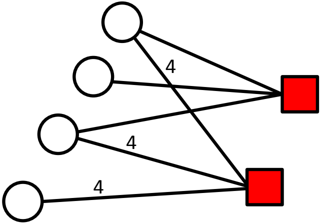


A check node is satisfied if the sum of its neighbors is an even integer. Apply iterative decoders for binary codes to $\mathbf{v} - \mathbf{a} \in [0,1]^n$ for an appropriately chosen $\mathbf{a}$ (Conway and Sloane 1999).

Sadeghi et al. (2006) apply iterative decoders to lattices constructed from codes using Construction D'.

Nested binary codes $C(H_a) \subseteq C(H_{a-1}) \subseteq \ldots \subseteq C(H_1)$ yield a lattice

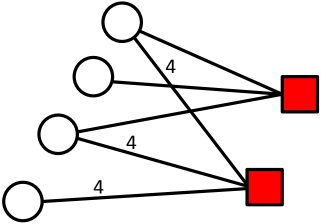
$$\Lambda_{D'} = \{\mathbf{v} \in \mathbb{Z}^n \mid H \cdot \mathbf{v}^T \equiv 0 \mod (2^{a+1}) \text{ for some } H\}.$$



Sadeghi et al. (2006) apply iterative decoders to lattices constructed from codes using Construction D'.

Nested binary codes $C(H_a) \subseteq C(H_{a-1}) \subseteq \ldots \subseteq C(H_1)$ yield a lattice

$$\Lambda_{D'} = \{\mathbf{v} \in \mathbb{Z}^n \mid H \cdot \mathbf{v}^T \equiv 0 \mod (2^{a+1}) \text{ for some } H\}.$$

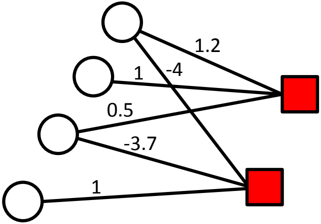


A check node is satisfied if the weighted sum of its neighbors is an integer divisible by 2^{a+1}.

Sommer et al. (2008) apply iterative decoders to lattices using the dual basis.

A lattice can be defined by its dual basis $\{\mathbf{b}_1, \ldots, \mathbf{b}_n\}$; that is,

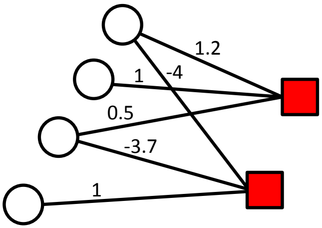
$$\Lambda = \{\mathbf{v} \mid \mathbf{v} \cdot \mathbf{b}_i \text{ is an integer vector for all } i\}.$$



Sommer et al. (2008) apply iterative decoders to lattices using the dual basis.

A lattice can be defined by its dual basis $\{\mathbf{b}_1, \ldots, \mathbf{b}_n\}$; that is,

$$\Lambda = \{\mathbf{v} \mid \mathbf{v} \cdot \mathbf{b}_i \text{ is an integer vector for all } i\}.$$



A check node is satisfied if the weighted sum of its neighbors is an integer. In this case, the message is a probability density function.

References

- A. Barvinok, A polynomial time algorithm for counting integral points in polyhedra when the dimension is fixed, Math. Oper. Res. **19** (1994), no. 4, 769–779.
- J. H. Conway and N. J. Sloane, Sphere packings, lattices and groups, New York: Springer, 1999.
- R. Gallager, Low-density parity-check codes, IRE Trans. **IT-8** (1962), 21–28.
- N. Kashyap, A decomposition theory for binary linear codes, IEEE Trans. Inform. Theory **54** (2008), no. 7, 3035–3058.
- C. Kelley and D. Sridhara, Pseudocodewords of Tanner graphs, IEEE Trans. Inform. Theory **53** (2007), no. 11, 4013–4038.
- R. Koetter, W.-C. W. Li, P. O. 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, Nonbinary pseudocodewords, to be submitted.
- 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.
- M.-R. Sadeghi, A. H. Banihashemi, and D. Panario, Low-density parity-check lattices: construction and decoding analysis, IEEE Trans. Inform. Theory **52** (2006), no. 11, 4481–4495.
- V. Skachek, Characterization of graph-cover pseudocodewords of codes over $\mathbb{F}_3$, Proc. IEEE Inform. Theory Workshop, Dublin, Ireland, Aug. 30 - Sep. 3, 2010.
- N. Sommer, M. Feder, and O. Shalvi, Low-density lattice codes, IEEE Trans. Inform. Theory **54** (2008), no. 4, 1561–1585.
- N. Wiberg, Codes and decoding on general graphs, Ph.D. thesis, Linköping University, Linköping, Sweden, 1996.