Codeword polytopes and linear programming relaxations for error-control decoding

Martin Wainwright
Department of Electrical Engineering and Computer Science
Department of Statistics
UC Berkeley, CA
Email: wainwrig@{eecs,stat}.berkeley.edu

Collaborators

LP decoding: Jon Feldman (Columbia), David Karger (MIT)

Tree-reweighted max-product: Tommi Jaakkola (MIT), Alan Willsky (MIT)

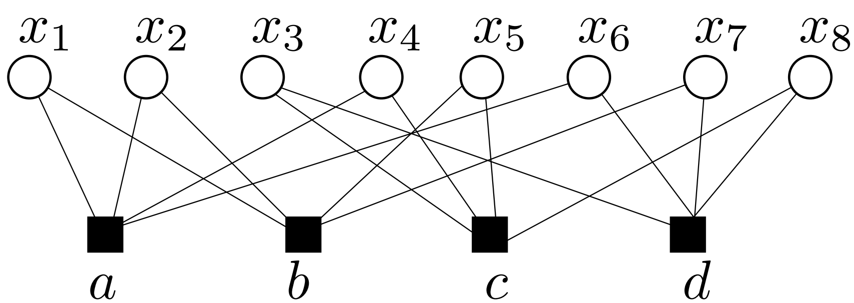
Expander-based bounds: Jon Feldman (Columbia), Tal Malkin (Columbia), Rocco Servedio (Columbia), Cliff Stein (Columbia)

Outline

1. Introduction
2. Linear programming relaxations for decoding
   (a) Basic LP relaxation
   (b) Link to standard iterative methods
   (c) Tree-reweighted max-product
3. Properties of LP decoding
   (a) Polytope vertices and pseudocodewords
   (b) BSC: fractional distance and expander-based bounds
4. Higher-order LP relaxations
   (a) Canonical full relaxation: metric polytope
   (b) Open issues
5. Summary and open questions

§1. Introduction

- many modern codes (e.g., turbo, LDPC) based on bipartite graph $G = (V, C, E)$:



V	$\equiv$	set of variable nodes
C	$\equiv$	set of check nodes
E	$\equiv$	variable–check edges

- $x_i \in \{0, 1\}$ is bit associated with node $i \in V = \{1, \ldots, n\}$
- check a connected to bit neighbors in $V(a)$ defines local parity check

$$f_a(x_{V(a)}) \quad = \quad \begin{cases} 1 & \text{if } \oplus_{i \in V(a)} x_i = 0 \\ 0 & \text{otherwise.} \end{cases}$$

- overall code $\mathbb{C}$ defined by product of checks

$$\mathbb{C} \quad := \quad \{x \in \{0,1\}^n \mid \prod_{a \in C} f_a(x_{V(a)}) = 1\}.$$

Decoding problem

- channel provides noisy observation vector $\mathbf{y} \in \mathcal{Y}^n$
- defines a probability distribution over codewords:

$$p(\mathbf{x}|\mathbf{y}) \;\propto\; \prod_{v \in V} f_v(x_v) \;\; \prod_{a \in C} f_a(x_{V(a)})$$

  where $f_v(x_v) = p(y_v \,|\, x_v)$.

- different types of decoding:
  - for minimal *bit error rate*, compute the marginal probability $p(x_v = 1 \,|\, \mathbf{y})$ and then set

$$\widehat{x}_v \quad = \quad \begin{cases} 1 & \text{if } p(x_v = 1 \,|\, \mathbf{y}) > 0.5 \\ 0 & \text{otherwise} \end{cases}$$

  - for minimal *word error rate*, decode to

$$\widehat{\mathbf{x}} \quad = \quad \left. \arg\min_{\mathbf{x} \in \mathbb{C}} p(\mathbf{x} \,|\, \mathbf{y}) \quad \right\} \text{maximum likelihood decoding}$$

Iterative decoding of graphical codes

- iterative "message-passing" techniques (sum-product or belief propagation; max-product or min-sum) have become the standard approach
- exact for trees, but approximate for graphs with cycles
- remarkably good practical performance
- behavior well-understood for random code ensembles in asymptotic regime as blocklength $n \to +\infty$ (e.g., Luby et al., 2001; Richardson & Urbanke, 2001)
- open issues: performance guarantees for intermediate length codes?

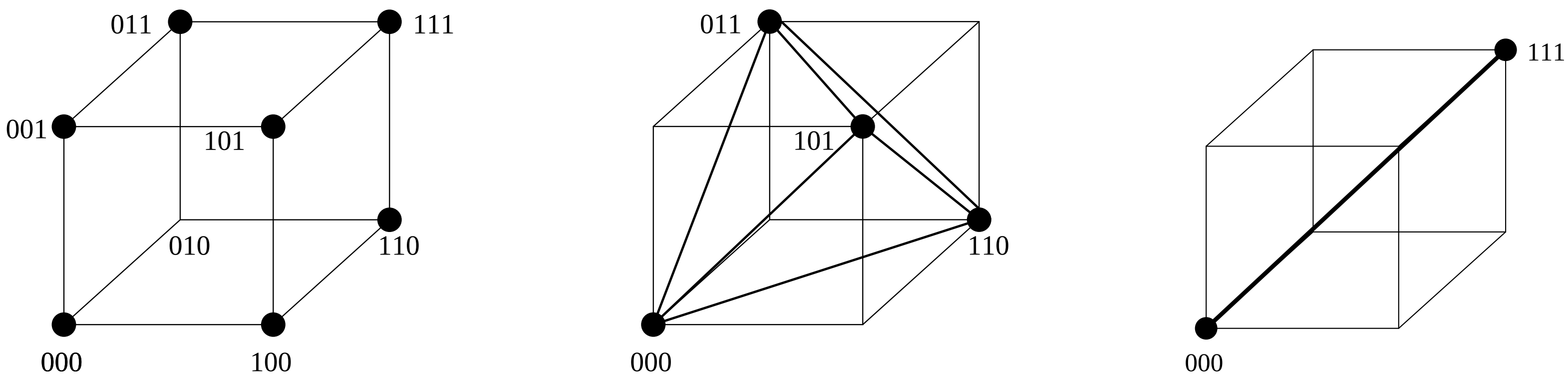
§2. Our approach: Linear program relaxation

- reformulate maximum-likelihood (ML) decoding as a linear program over the *codeword polytope*
- solve the LP over a relaxed polytope: linear programming (LP) decoder
- linear programs are graph-structured, and can be solved either by standard LP solvers, or variants of iterative message-passing
- error analysis reduces to study of linear program with random cost function
- amenable to some analysis in finite-length setting

Codeword polytope

Definition: The *codeword polytope* $\mathrm{CH}(\mathbb{C}) \subseteq [0,1]^n$ is the convex hull of all codewords

$$\mathrm{CH}(\mathbb{C}) = \left\{ \mu \in [0,1]^n \mid \mu_s = \sum_{\mathbf{x} \in \mathbb{C}} p(\mathbf{x})\, x_s \right\}$$



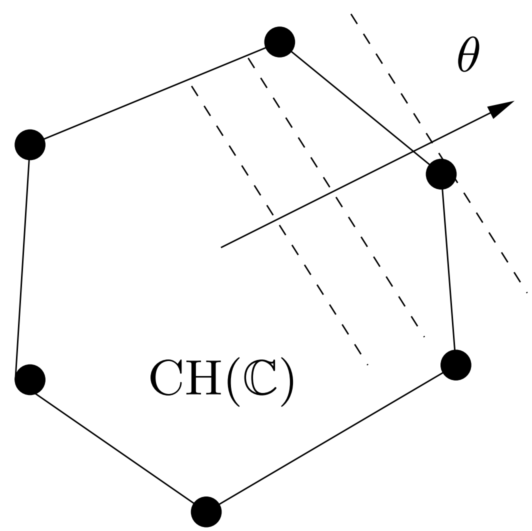
(a) Uncoded (b) One check (c) Two checks

- the codeword polytope is always contained within the unit hypercube $[0,1]^n$
- vertices correspond to codewords

From integer program to linear program

Given a noisy observation y, define cost vector $\theta = \theta(y)$.

Example: For the BSC, set $\theta_s = 1$ if $y_s = 0$ and $\theta_s = -1$ if $y_s = 1$.



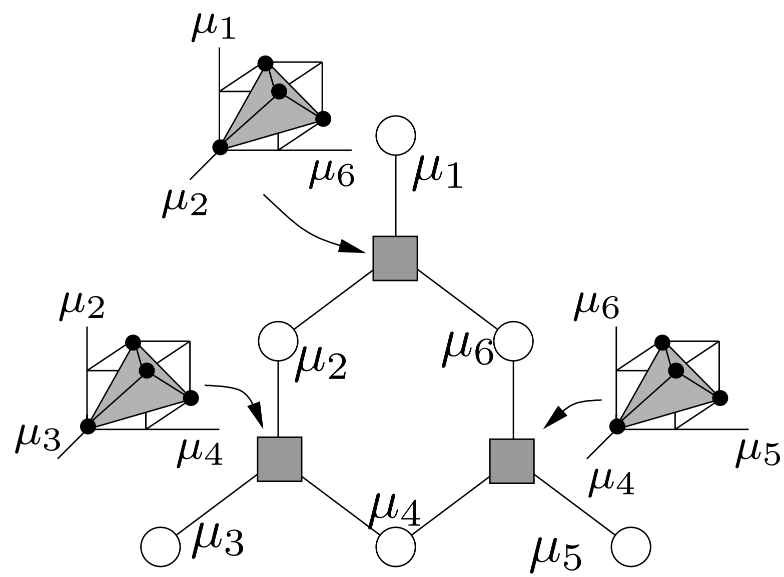
Key: Given received word $\mathbf{y}$, optimal maximum likelihood (ML) decoding can be re-formulated linear program (LP) over the codeword polytope:

$$\min_{\mathbf{x} \in \mathbb{C}} \sum_{s=1}^{n} \theta_s x_s \quad = \quad \min_{\mu \in \mathrm{CH}(\mathbb{C})} \sum_{s=1}^{n} \theta_s \mu_s.$$

LP relaxation for approximate decoding

- each parity check $a \in C$ defines a *local codeword polytope* $\mathrm{LOC}(a)$
- impose all local constraints:

$$\mathrm{LOC}(\mathbb{C}) \;:=\; \cap_{a \in C} \mathrm{LOC}(a).$$



Properties:

1. For trees, $\mathrm{LOC}(\mathbb{C}) = \mathrm{CH}(\mathbb{C})$.
2. In general, $\mathrm{LOC}(\mathbb{C})$ is a *relaxation* (i.e., $\mathrm{CH}(\mathbb{C}) \subset \mathrm{LOC}(\mathbb{C})$).

Strategy: Solve the relaxed LP $\min\limits_{\mu \in \mathrm{LOC}(\mathbb{C})} \sum_{s=1}^{n} \theta_s \mu_s$.

Solve with standard LP solver (e.g., simplex), or tree-reweighted max-product algorithm. (Feldman, Karger & Wainwright, IEEE Info. Theory (to appear))

Different representations of relaxed polytope

The polytope $\mathrm{LOC}(\mathbb{C})$ has distinct representations:

1. Lifted representation
   (a) polytope defined with variables

   $\mu_s \in [0,1]$ for each bit $s = 1, \ldots, n$

   $w_{a,J} \in [0,1]$ auxiliary var. for check a

   J even-sized subset of $V(a)$

   (b) interpret $w_{a,\cdot}$ defining the local codeword polytope associated with check a

   (c) most closely related to belief propagation and Bethe formulation

2. Projected representation:
   (a) auxiliary variables $w_{a,\cdot}$ can be eliminated by projection

   (b) leads to a reduced representation over $\mu = \{\mu_1, \ldots, \mu_n\}$

Lifted representation and local codeword polytopes

- for each check a, let $\mathbb{C}(a)$ denote set of local codewords
- for example, for a 3-check of the form $a = \{1, 2, 3\}$, then

$$\mathbb{C}(a) = \{000,\ 110,\ 101,\ 011\}$$

- define prob. distribution $w = \{w_{a,J} \mid J \in \mathbb{C}(a)\}$ over local codewords and impose constraints

$$\begin{array}{rccl}
\underline{\text{Non-negativity:}} & w_{a,J} & \geq & 0 \\
\underline{\text{Normalization:}} & \sum_{J \in \mathbb{C}(a)} w_{a,J} & = & 1 \\
\underline{\text{Marginalization:}} & \sum_{J \in \mathbb{C}(a), J_s = 1} w_{a,J} & = & \mu_s \quad \text{for any bit node } s
\end{array}$$

Projected form of relaxed codeword polytope

- involves imposing constraints only on vector $\mu = \{\mu_1, \ldots, \mu_n\}$
- Probability constraints: Require that μ_v are marginal probabilities $\quad 0 \leq \mu_v \leq 1$
- Check constraints: for each check, let $V(a)$ be the set of bit neighbors
  - let S be an *odd-sized* subset of the check neighborhood $V(a)$, indexing an odd-parity subvector $\mathbb{I}_S$ over $V(a)$
  - require that $\mu_{V(a)}$ is separated from $\mathbb{I}_S$ by Hamming distance at least 1:

$$\sum_{v \in S}(1-\mu_v) + \sum_{v \in V(a)\setminus S} \mu_v \quad \geq \quad 1.$$

  - leads to a total of $2^{|V(a)|-1}$ constraints per check a

Pseudocodewords as fractional vertices in the relaxed polytope

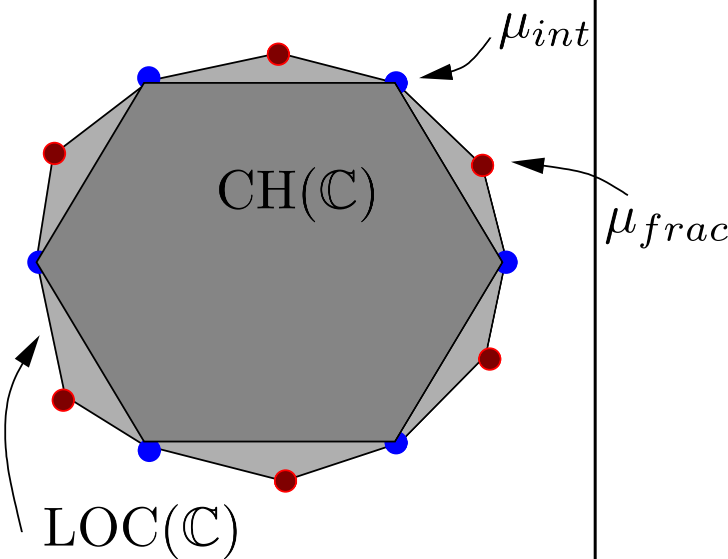
Two vertex types in relaxed polytope:

integral: codewords

(e.g., $\begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}$)

fractional: *pseudo*codewords

(e.g., $\begin{bmatrix} 1 & \frac{1}{2} & 1 & \frac{1}{2} & 1 & \frac{1}{2} \end{bmatrix}$)



Possible outputs of LP decoder

1. codeword with guarantee of ML correctness
2. pseudocodeword

Link to standard iterative methods

The relaxed polytope $\mathrm{LOC}(\mathbb{C})$ is closely related to the standard sum-product and max-product algorithms:

1. Relation to sum-product:
   (a) polytope $\mathrm{LOC}(\mathbb{C})$ imposes constraints equivalent to the Bethe formulation of belief propagation (Yedidia et al., 2001)
   (b) this equivalence guarantees exactness for trees
   (c) optimum of BP not necessarily attained at polytope vertex

2. Relation to max-product:
   (a) link to graph cover and ordinary max-product algorithm (Koetter & Vontobel, 2003)
   (b) max-product is an algorithm for solving dual of LP relaxation on trees, but not in general (Wainwright et al., 2003)

Tree-reweighted max-product algorithm

Message update from node t to node s:

$$M_{ts}(x_s) \quad \leftarrow \quad \kappa \max_{x'_t \in \mathcal{X}_t} \left\{ \underbrace{\left[\psi_{st}(x_s, x_t)\right]^{\frac{1}{\rho_{st}}}}_{\text{reweighted potential}} \; \psi_t(x'_t) \; \frac{\prod_{v \in \mathcal{N}(t)\backslash s} \overbrace{\left[M_{vt}(x_t)\right]^{\rho_{vt}}}^{\text{reweighted messages}}}{\underbrace{\left[M_{st}(x_t)\right]^{(1-\rho_{ts})}}_{\text{opposite message}}} \right\}.$$

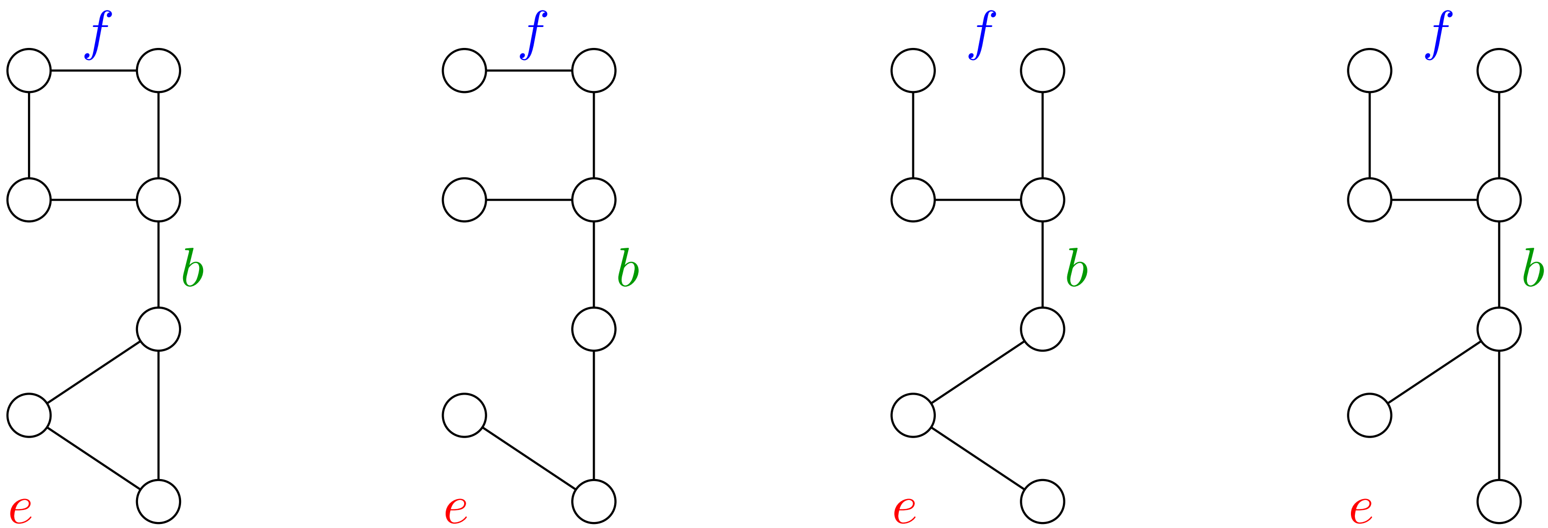
Properties:

1. Modified updates have same complexity as standard updates.
2. Key differences:
   - Messages are reweighted with $\rho_{st} \in [0,1]$.
   - Potential on edge (s,t) is rescaled by $\rho_{st} \in [0,1]$.
   - Update involves the reverse direction edge.
3. The choice $\rho_{st} = 1$ for all edges (s,t) recovers standard update.

(Wainwright, Jaakkola & Willsky, 2003)

Edge appearance probabilities

Experiment: What is the probability ρ_e that a given edge $e \in E$ belongs to a tree T drawn randomly under $\boldsymbol{\rho}$?



(a) Original (b) $\rho(T^1) = \frac{1}{3}$ (c) $\rho(T^2) = \frac{1}{3}$ (d) $\rho(T^3) = \frac{1}{3}$

In this example: $\rho_b = 1$; $\rho_e = \frac{2}{3}$; $\rho_f = \frac{1}{3}$.

The vector $\boldsymbol{\rho_e} = \{ \rho_e \mid e \in E \}$ must belong to the *spanning tree polytope*, denoted $\mathbb{T}(G)$.

Properties of tree-reweighted max-product (TRMP)

- TRMP updates can be understood as a iterative method for solving the LP dual
- any TRMP message fixed point specifies a collection of pseudo-max-marginals ν_s^* for each node $s \in V$ and ν_a^* for each check $a \in C$.

Tree agreement: Vector $\mathbf{x}^* \in \{0,1\}^n$ satisfiies tree agreement if:

(a) for each node s, the bit x_s^* is optimal for ν_s^* (i.e., $\nu_(^* x_s^*) = \max_{u \in \{0,1\}} \nu_s^*(u)$)

(b) for each check a, the subvector $x_{V(a)}^*$ is optimal for ν_a^*.

Theorem: Any vector $\mathbf{x}^*$ that satifies tree agreement with respect to ν^* is an ML optimal codeword.

§3. Properties of LP decoding

A desirable feature of LP decoding is its amenability to analysis:

A. behavior completely determined by *set of pseudocodewords*

B. *stopping set characterization* for binary erasure channel (BEC)

C. guarantees for the BSC based on the *fractional distance*

D. stronger guarantees for codes based on *expander graphs*

A. Pseudocodewords

- other researchers have identified "pseudocodewords" for different channels and codes:

  (a) deviation sets for LDPCs (e.g.,Wiberg, 1996; Horn, 1999)

  (b) pseudocodewords for tail-biting trellises (Forney et al., 2001)

  (c) stopping sets for the BEC (e.g., Luby et al., 1999)

  (d) signal space characterization of decoding (Frey et al., 2001)

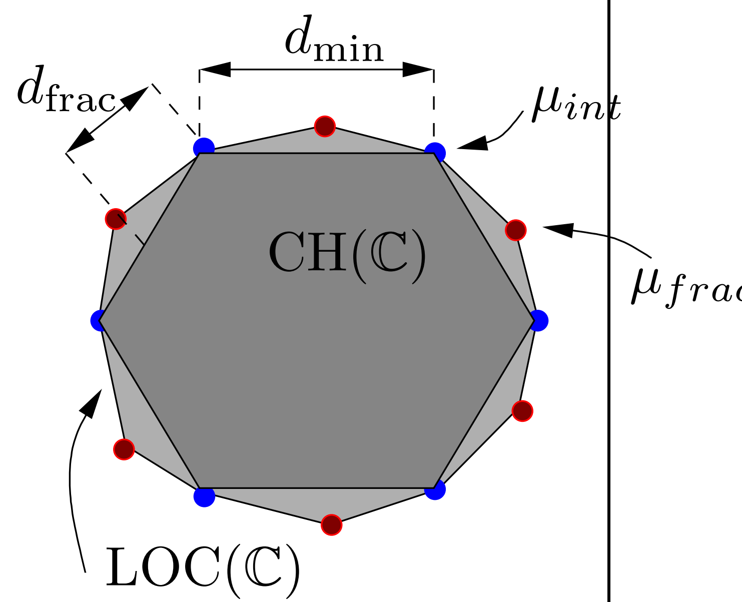
  (e) near codewords (McKay et al., 2002)

- the polytope view (i.e., fractional versus integral vertices) unifies these various notions

- pseudocodewords provide a geometrically intuitive distinction between success and failure for LP decoding

LP decoding finds optimum pseudocodeword

Two vertex types in relaxed polytope:



integral: codewords

(e.g., $\begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}$)

fractional: *pseudo*codewords

(e.g., $\begin{bmatrix} 1 & \frac{1}{2} & 1 & \frac{1}{2} & 1 & \frac{1}{2} \end{bmatrix}$)

Proposition: Given the channel cost vector θ, the LP decoder finds the pseudocodeword with minimum weight $\sum_s \theta_s \mu_s$. Therefore, there are two possible outcomes:

(a) if it finds a codeword, it must be ML optimal.

(b) otherwise it finds a pseudocodeword (acknowledged failure).

Construction of a pseudocodeword

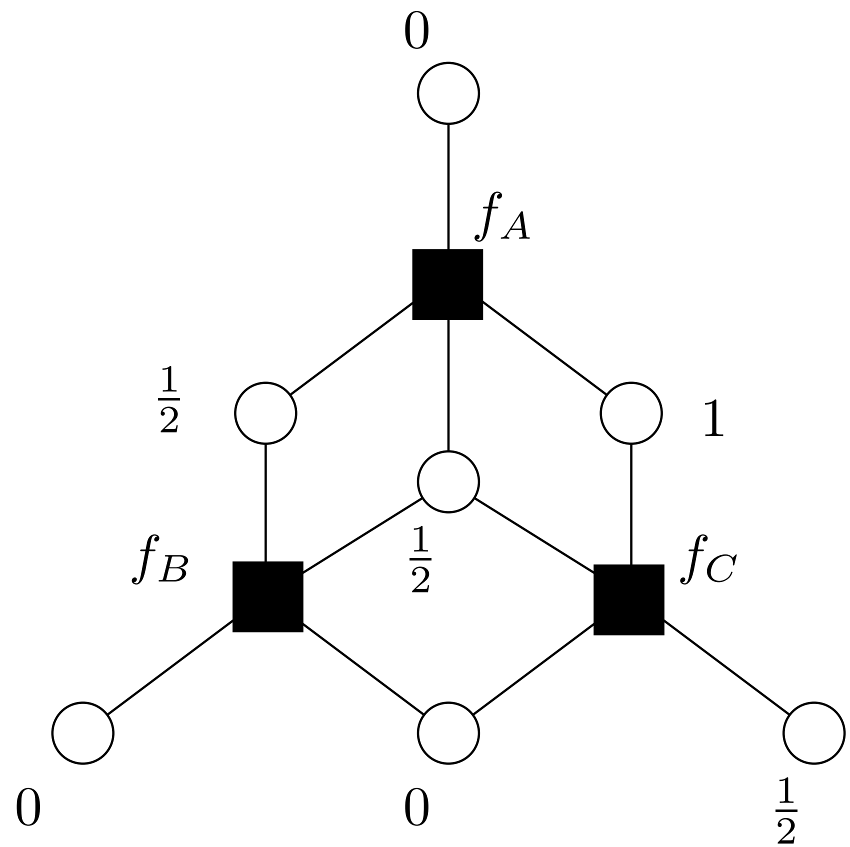
Refer to a fractional vertex of the relaxed codeword polytope LOC($\mathbb{C}$) as a *pseudocodeword*.

Check A:

$$\begin{bmatrix} 0 \\ \frac{1}{2} \\ \frac{1}{2} \\ 1 \end{bmatrix} = \frac{1}{2}\begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \end{bmatrix} + \frac{1}{2}\begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}$$

Check B:

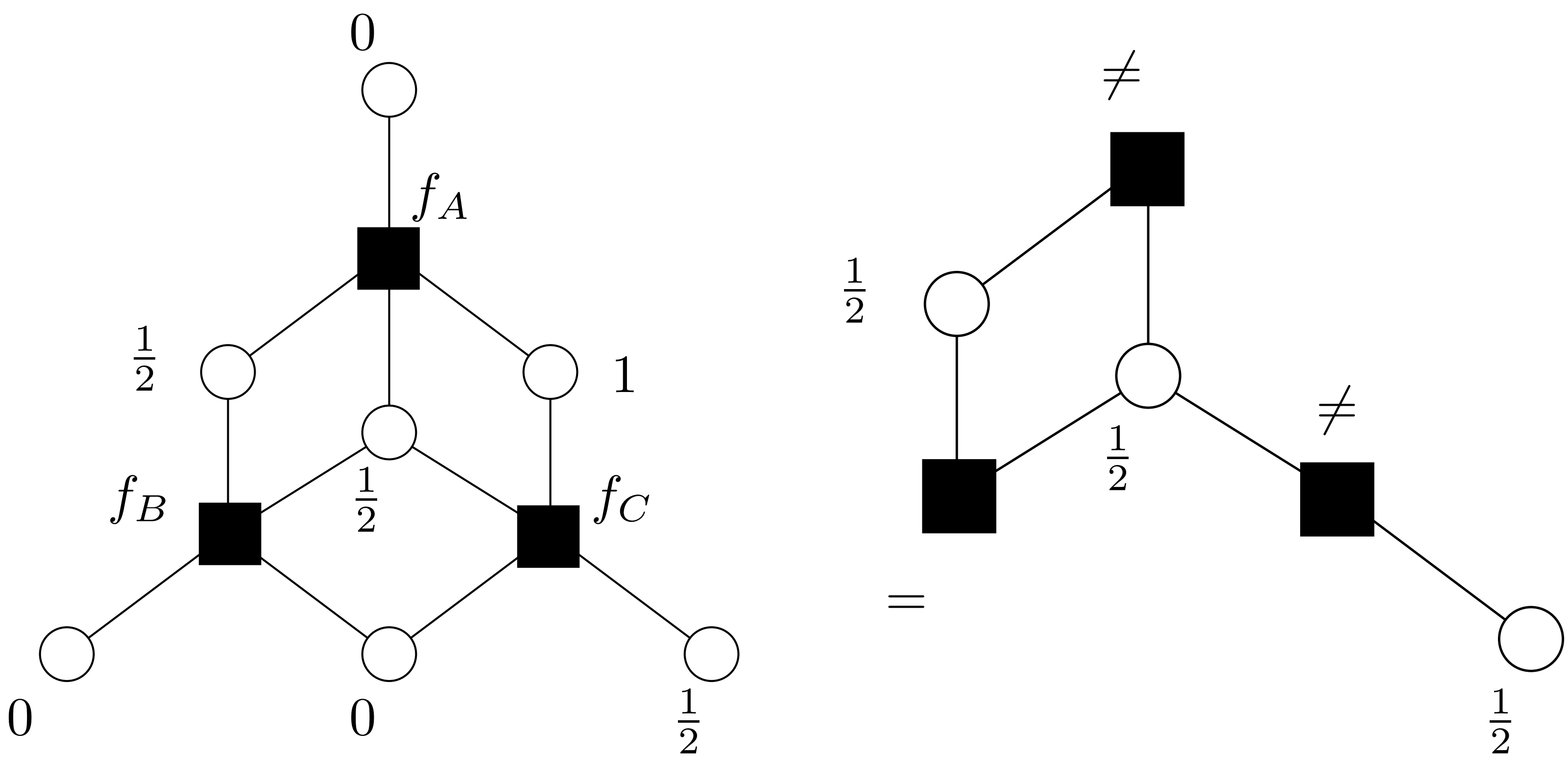
$$\begin{bmatrix} \frac{1}{2} \\ \frac{1}{2} \\ 0 \\ 0 \end{bmatrix} = \frac{1}{2}\begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} + \frac{1}{2}\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$



The pseudocodeword is locally-consistent for each check $\Longrightarrow$ it *does belong* to the first-order relaxed polytope LOC($\mathbb{C}$).

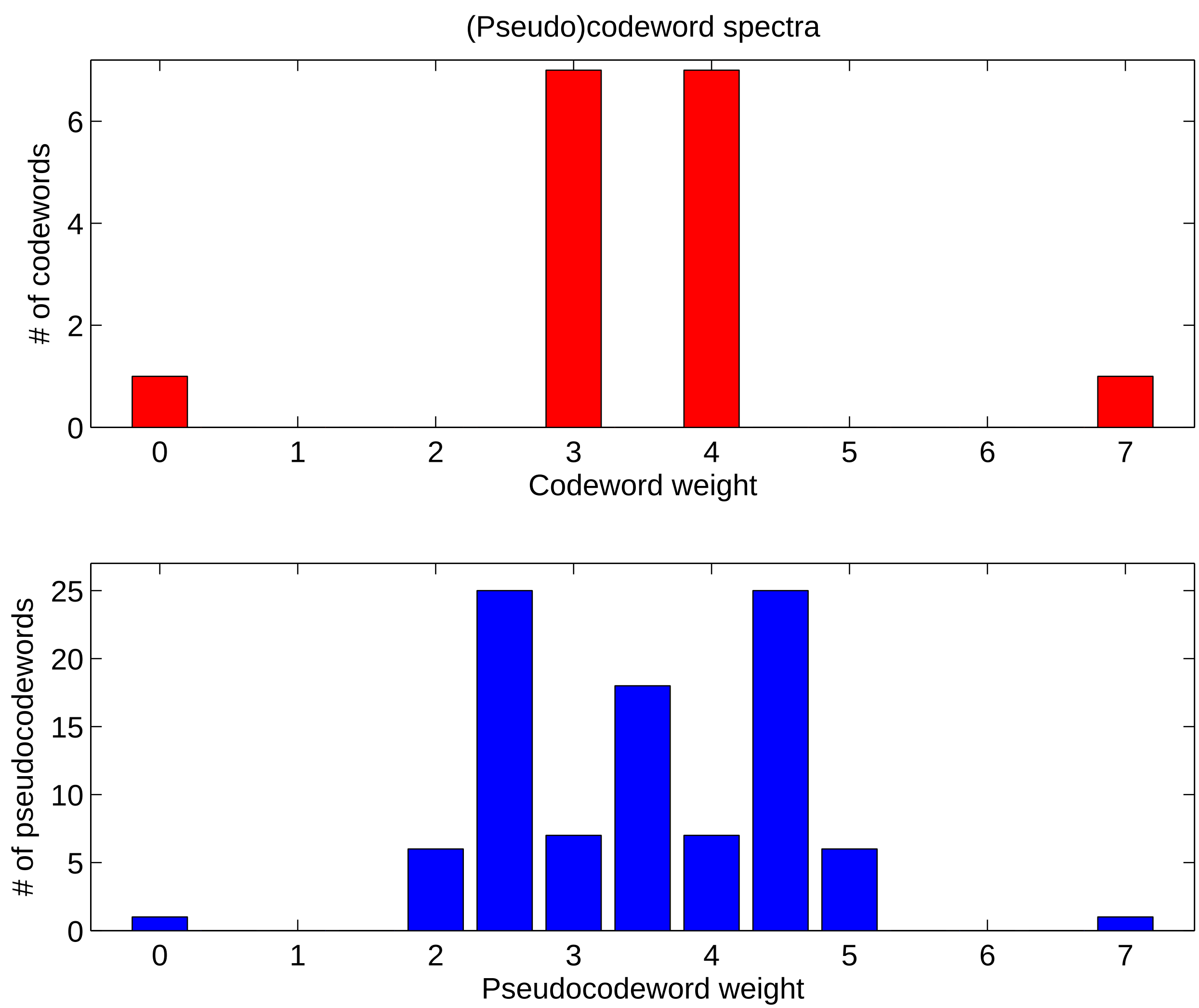
Verifying global inconsistency

- first set all non-fractional bits to their preferred values



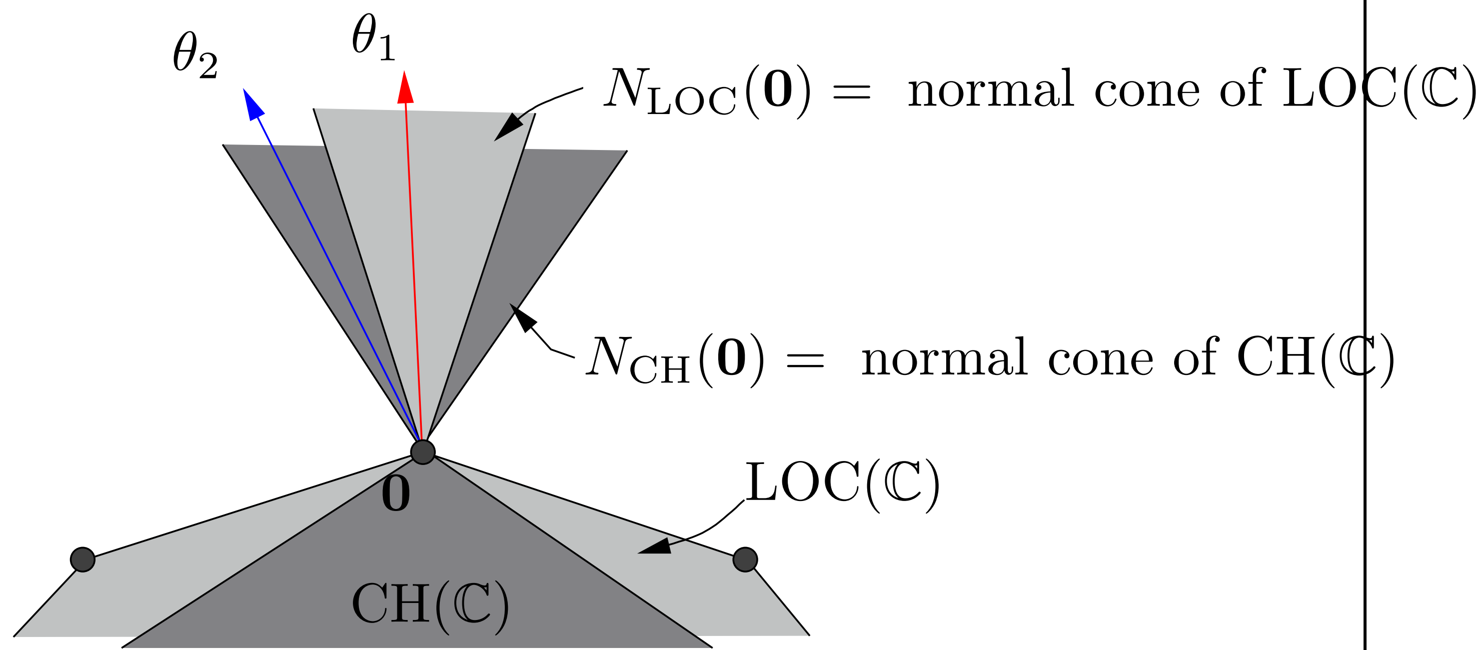
- this generates an inconsistent set of requirements for the remaining bits $\Longrightarrow$ vector does *not* belong to exact codeword polytope $\mathrm{CH}(\mathbb{C})$

Codeword and pseudocodeword spectra



Geometry of LP decoding

Proposition: The LP relaxation is code-symmetric. Therefore, for the purposes of analysis, can assume that codeword $\mathbf{0}$ was sent.



$$
\begin{aligned}
\text{Prob. of successful ML decoding} \quad &= \quad \Pr\left[\theta \in N_{\mathrm{CH}}(\mathbf{0})\right] \\
\text{Prob. of successful LP decoding} \quad &= \quad \Pr\left[\theta \in N_{\mathrm{LOC}}(\mathbf{0})\right]
\end{aligned}
$$

B. Performance for the BEC

- standard iterative decoding (sum-product; belief propagation) takes a very simple form in the BEC: (e.g., Luby et al., 2001)

```
While there exists at least one erased (*) bit:
1. Find check node with exactly one erased bit nbr.
2. Set erased bit neighbor to the XOR of other bit neighbors.
3. Repeat.
```

- success/failure is determined by presence/absence of stopping sets in the erased bits (Di et al., 2002)

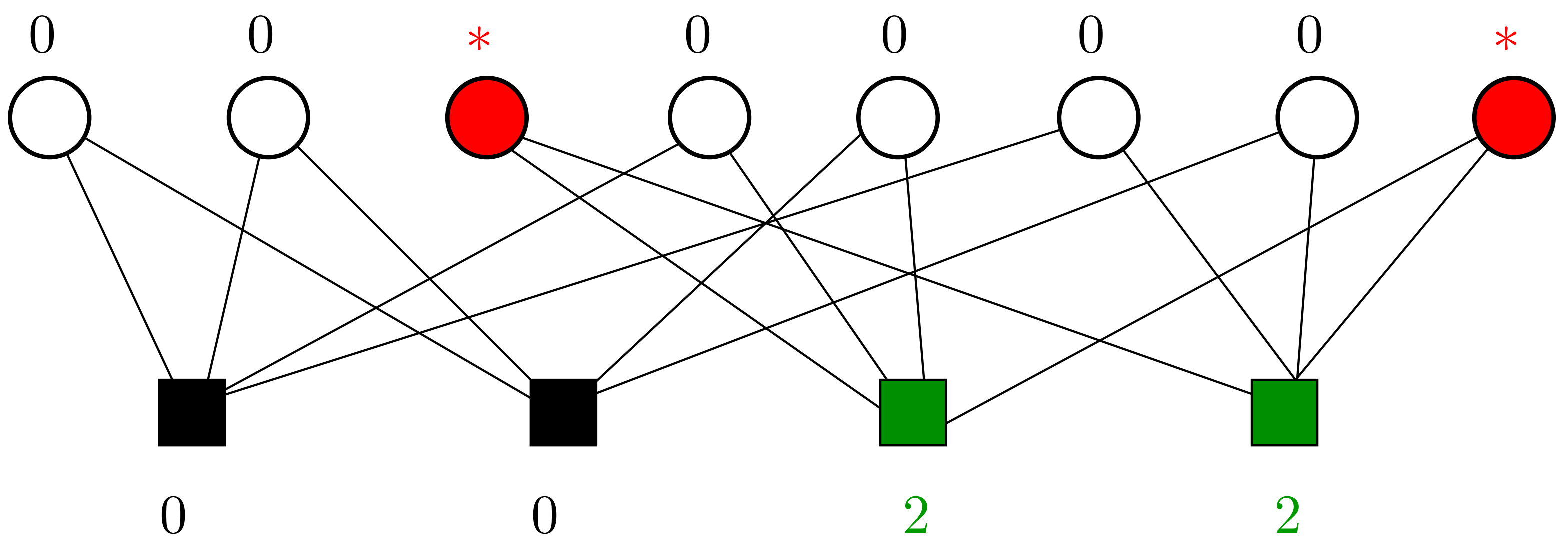
- for LP decoding, cost vector takes form $\theta_s = \begin{cases} -1 & \text{if } y_s = 1 \\ 1 & \text{if } y_s = 0 \\ 0 & \text{if } y_s \text{ erased} \end{cases}$.

- stopping sets correspond to cost vectors that lie outside the relaxed normal cone $N_{\mathrm{LOC}}(\mathbf{0})$

Stopping sets for the BEC

Definition: A *stopping set* S is a set of bits such that:

- every bit in S is erased
- every check that is adjacent to S has degree at least two (with respect to S)



LP decoding in the BEC

The performance of the LP decoder in the BEC is completely characterized by stopping sets:

Theorem: (Feldman et al., 2003)

(a) LP decoding succeeds in the BEC if and only the set of erasures does *not* contain a stopping set.

(b) Therefore, the performance of (first-order) LP decoding is equivalent to sum-product/belief propagation decoding in the BEC.

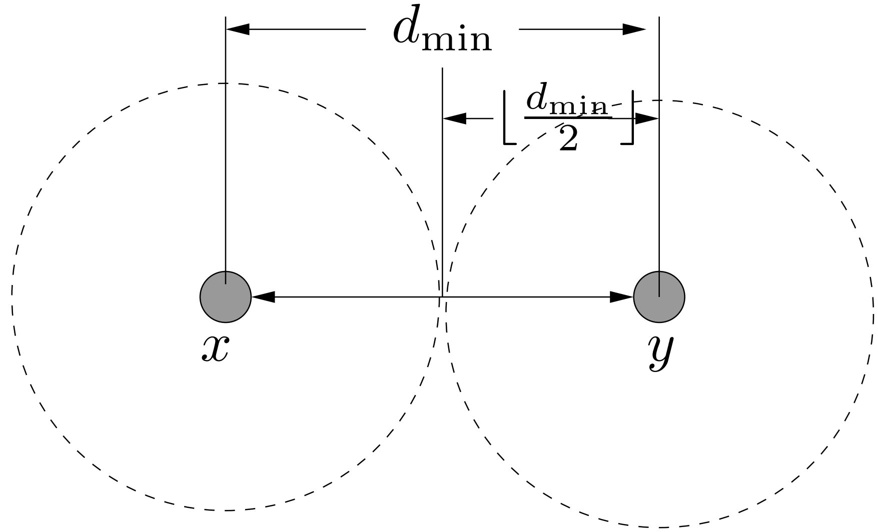
Corollary: With appropriate choices of low-density parity check codes, LP decoding can achieve capacity in the BEC.

C. Guarantees based on fractional distance

- the *minimum distance* of a code is given by

$$d_{\min} = \min_{x,y \in \mathbb{C},\ x \neq y} \|x - y\|_1$$

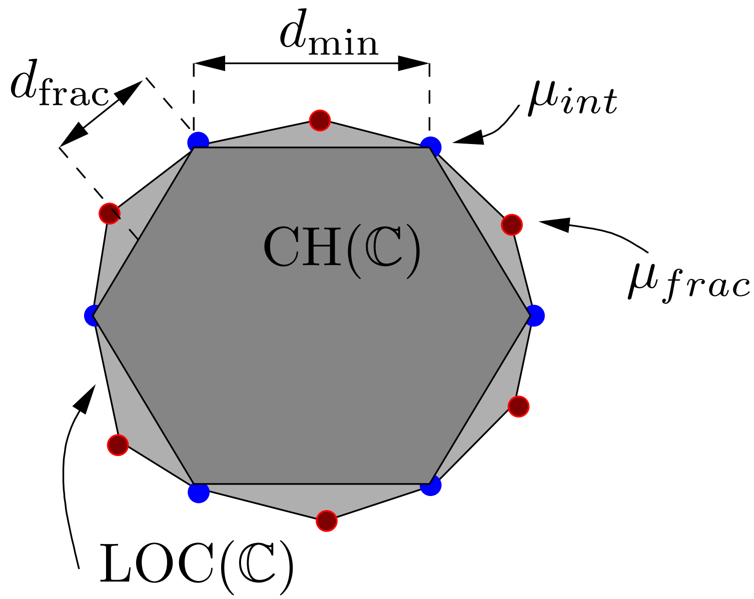
- for a linear code, this reduces to $d_{\min} = \min_{x \neq 0} \|x\|_1$.



Classical result: optimal maximum-likelihood decoding (ML) can correct up to $\lfloor \frac{d_{\min}}{2} \rfloor$ bit flips (in the BSC).

Polytope-based view of minimum distance

- classical minimum distance is smallest ℓ_1 norm between vertices of the codeword polytope $\mathrm{CH}(\mathbb{C})$
- natural to define an analogue for the *relaxed* polytope $\mathrm{LOC}(\mathbb{C})$



- **Definition:** Define the *fractional distance* d_{frac} to be the minimum ℓ_1-distance between any pair of vertices of $\mathrm{LOC}(\mathbb{C})$.
- for a code-symmetric polytope and linear code, the fractional distance is the ℓ_1 distance from $\mathbf{0}^n$ and the *nearest pseudocodeword*

Error-correction in terms of frac. distance

Theorem:

(a) In the binary symmetric channel, the LP decoder will succeed as long as no more than $\lfloor \frac{d_{\text{frac}}}{2} \rfloor$ bits are flipped.

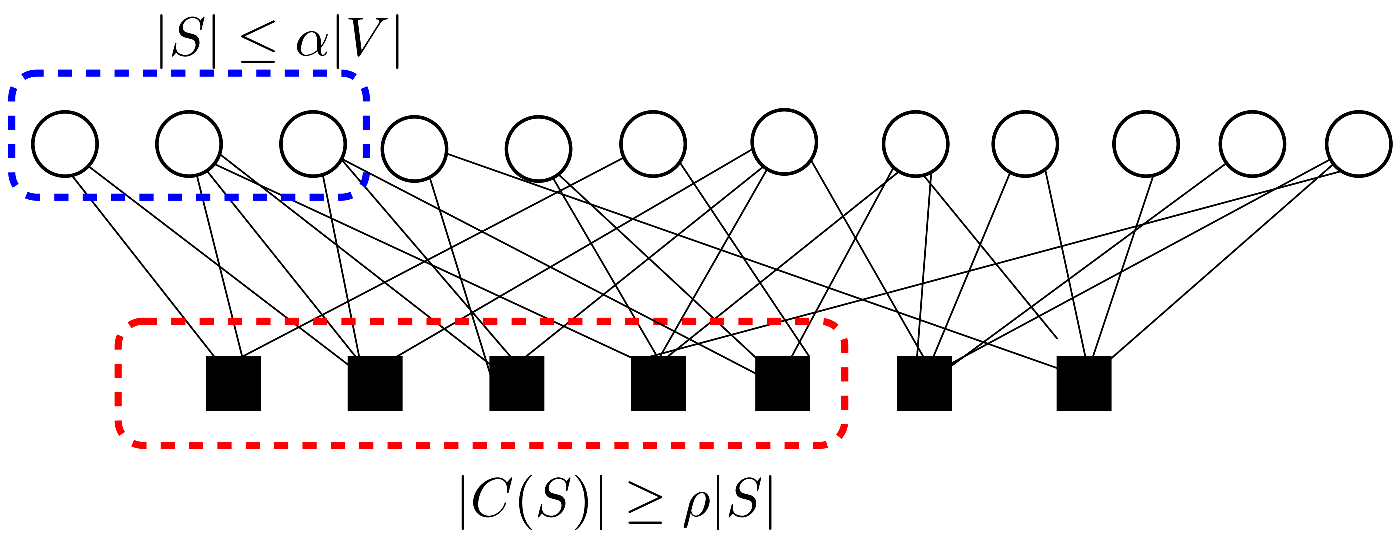
(b) For any factor graph with variable degree $\Delta_v \geq 3$, check degree $\Delta_c \geq 2$ and girth g, the fractional distance satisfies

$$d_{\text{frac}} \quad \geq \quad \frac{2}{\Delta_c}(\Delta_v - 1)^{\lceil \frac{g}{4} - 1 \rceil}.$$

(a), (b), Feldman, Karger & Wainwright, IEEE Trans. Info Theory (to appear)

D. Guarantees for expander graph codes

- exploit graph expansion properties to obtain stronger results beyond girth
- previous work on expander codes (Spielman et al., 1995; Burshtein & Miller, 2002; Barg & Zemor, 2002)



- **Definition:** Let $\alpha \in (0, 1)$. A factor graph $G = (V, C, E)$ is a (α, ρ)-*expander* if for all subsets $S \subset V$ with $|S| \leq \alpha|V|$, at least $\rho|S|$ check nodes are incident to S

LP decoding corrects a constant fraction of errors

- let $\mathbb{C}$ be an LDPC described by a factor graph G with regular variable (bit) degree Δ_v.

Theorem: Suppose that G is an $(\alpha, \delta\Delta_v)$-expander, where $\delta > 2/3 + 1/(3\Delta_v)$ and $\delta\Delta_v$ is an integer.
Then the LP decoder can correct at least $\frac{3\delta-2}{2\delta-1}(\alpha n - 1)$ bit flips in the binary symmetric channel. (Feldman et al., ISIT 2004)

- idea of proof:
  - given a code-symmetric polytope, can assume that $\mathbf{0}$ was sent.
  - decoder works if and only if primal LP optimum $p^* = 0$.
  - dual certificate of optimality: use expansion to construct a dual-optimal solution with cost $q^* = 0$
- "dual certificate" proof technique is more generally applicable (e.g., capacity-achieving expander codes: Feldman & Stein, SODA 2005)

Dual certificate proof technique

Primal decoding LP:

$$\text{min.} \quad \sum_i \theta_i \mu_i \quad \text{s.t.} \begin{cases} w_{a,J} \geq \; 0 \\ \sum\limits_{J \in \mathbb{C}(a)} w_{a,J} = \; 1 \\ \sum\limits_{J \in \mathbb{C}(a), J_v = 1} w_{a,J} \quad = \; \mu_v \end{cases}$$

Dual LP:

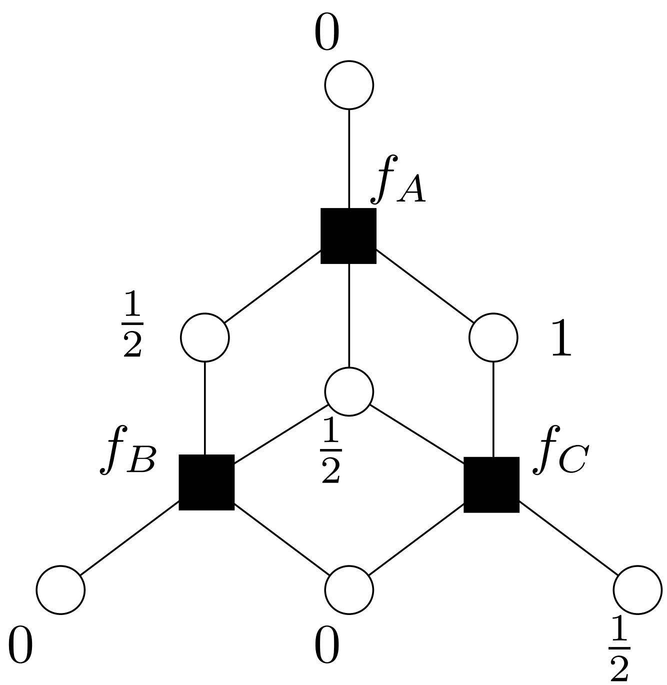
$$\text{max.} \quad \sum_a v_a \quad \text{s.t.} \begin{cases} v_a \forall a \in C, \quad \tau_{ia} \forall (i,a) \in E \quad \text{free} \\ \sum_{i \in S} \tau_{ia} \geq v_a \text{ for all} \qquad a \in C, J \in \mathbb{C}(a), \; S \in C(a) \\ \sum_{a \in N(i)} \tau_{ia} \leq \theta_i \qquad \text{for all } i \in V \end{cases}$$

§4. Beyond the first-order relaxation: Hierarchies of LP decoders

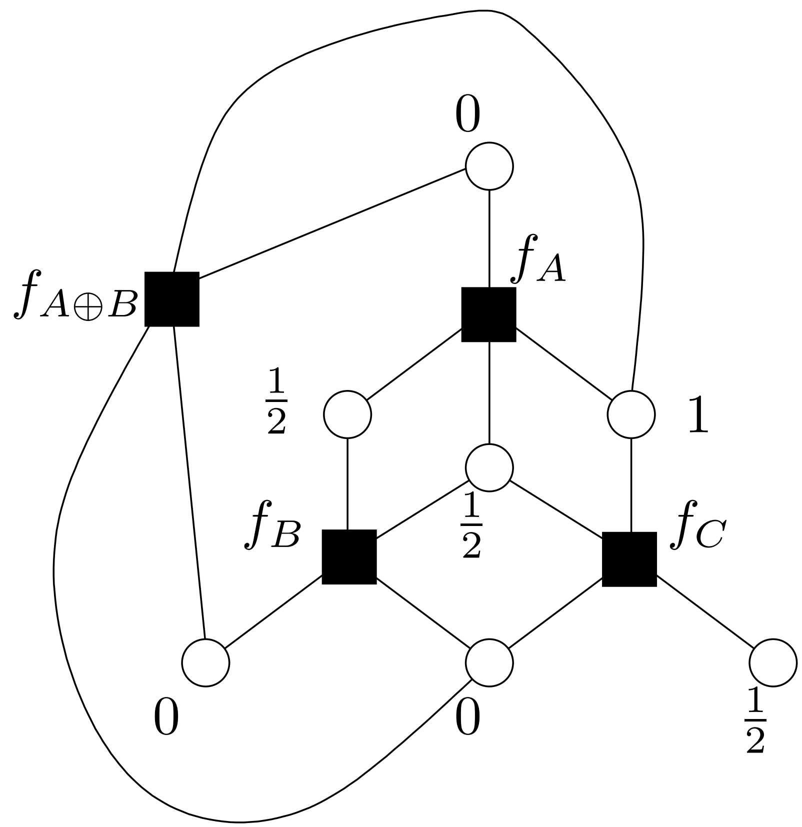
Intuition: pseudocodewords can be "pruned" by adding constraints.

- several natural ways to generate constraints:
  1. generating additional checks: redundant for the code, but tighten the LP relaxation
  2. other "lift-and-project" methods (e.g., Lovasz & Schrijver, 1990)
- similar in spirit to generalized belief propagation procedures (Yedidia et al., 2002)
- desirable property: decoding performance is guaranteed to improve (or at least not degrade) for any channel

Illustration: Hamming code



(a) First-order relaxation



(a) Higher-order relaxation

Key: Adding the additional check $f_{A\oplus B}$ removes a subset of pseudocodewords from the first-order relaxation.

A conjecture

Canonical full relaxation: add a local codeword polytope for every possible check (i.e., one for each dual codeword).

Illustration (Hamming code):

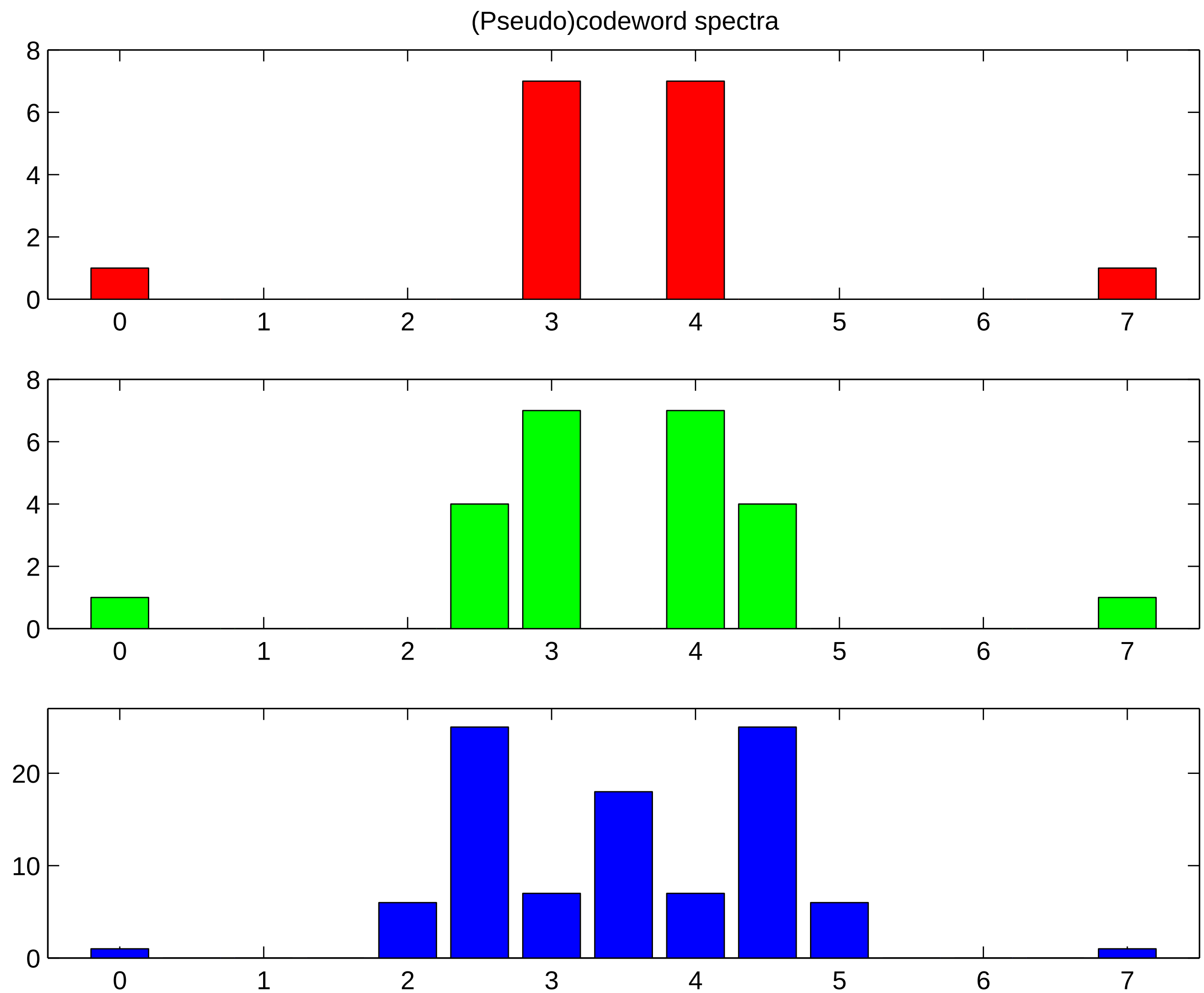
$$H_1 = \begin{bmatrix} A: & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ B: & 0 & 1 & 1 & 0 & 1 & 1 & 0 \\ C: & 0 & 0 & 1 & 1 & 0 & 1 & 1 \end{bmatrix} \qquad H_2 = \begin{bmatrix} A \oplus B: & 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ B \oplus C: & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ A \oplus C: & 1 & 1 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

$$H_3 = \begin{bmatrix} A \oplus B \oplus C: & 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}$$

Add a local codeword polytope constraint for each such check.

Conjecture: This relaxation provides an exact description of the codeword polytope.

Higher-order pseudocodeword spectra



Counterexample: Dual of (7,4,3) Hamming code

- consider the dual $\mathbb{C}^{\perp}$ of the $(7,4,3)$-Hamming code:

$$\begin{bmatrix}0 & 0 & 0 & 0 & 0 & 0 & 0\end{bmatrix} \quad \begin{bmatrix}1 & 1 & 1 & 1 & 0 & 0 & 0\end{bmatrix}$$

$$\begin{bmatrix}0 & 1 & 1 & 0 & 1 & 1 & 0\end{bmatrix} \quad \begin{bmatrix}0 & 0 & 1 & 1 & 0 & 1 & 1\end{bmatrix}$$

$$\begin{bmatrix}1 & 0 & 0 & 1 & 1 & 1 & 0\end{bmatrix} \quad \begin{bmatrix}0 & 1 & 0 & 1 & 1 & 0 & 1\end{bmatrix}$$

$$\begin{bmatrix}1 & 1 & 0 & 0 & 0 & 1 & 1\end{bmatrix} \quad \begin{bmatrix}1 & 0 & 1 & 0 & 1 & 0 & 1\end{bmatrix}$$

- can show that the point $\mu^* = (\frac{2}{3}, \ldots, \frac{2}{3})$ satisfies all constraints in the canonical full relaxation

- moreover, there holds

$$(-\mathbf{1}^T)\mu^* \;=\; -\frac{14}{3} \quad < \quad -4 \;=\; \min_{\mathbf{x} \in \mathbb{C}^{\perp}} (-\mathbf{1})^T \mathbf{x}$$

so that μ^* is a vertex (i.e, a pseudocodeword)

Sum-of-circuits property

- for a subclass of binary linear codes, the full metric relaxation is *exact*

- based on matroids with the "sum-of-circuits" property (Seymour, 1981)

- the subclass is characterized by forbidding three particular subcodes obtained via sequence of
  (a) code puncturing
  (b) code shortening

- includes as special cases:
  (a) all tree and trellis codes
  (b) all cycle codes
  (c) all cutset codes on planar graphs

Polynomial-time algorithms

- full relaxation involves imposing an exponential number of constraints (a local polytope for each dual codeword)
- naively might expect that resulting LP not polynomial-time solvable
- for sum-of-circuits codes, there exists a separation oracle for canonical full relaxation $\Longrightarrow$ ellipsoid algorithm is applicable (Groetschel et al., 1987)
- hence, binary linear codes satisfying sum-of-circuits are ML decodable in polynomial time

Various open questions

- provides a considerably larger class of ML-decodable codes
  (a) are any such codes useful (in isolation)?
  (b) which are useful in a concatenated or turbo setting?

- multi-stage adaptive decoding methods
  - solve first-order relaxation
  - stop if ML correct; else refine set of constraints and re-solve

- other techniques for forming hierarchies: complexity versus decoding performance

Summary

- LP relaxations for error-correcting decoding
- amenable to analysis in finite-length setting
- provides some insight into standard iterative methods

Open directions:

1. beyond worst-case: average-case performance analysis
2. extremely fast methods for solving LP relaxations? (e.g., flow-based formulations; combinatorial algorithms)
3. stronger relaxations (e.g., semidefinite) and performance guarantees
4. study of trade-off complexity of LP decoder and error probability

Contact information

Martin Wainwright

`wainwrig@{eecs,stat}.berkeley.edu`

Papers at: http://www.eecs.berkeley.edu/~wainwrig