Probabilistic analysis of linear programming relaxations

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

Based on joint work with:

Costis Daskalakis, Alex Dimakis, Richard Karp (UC Berkeley)

Introduction

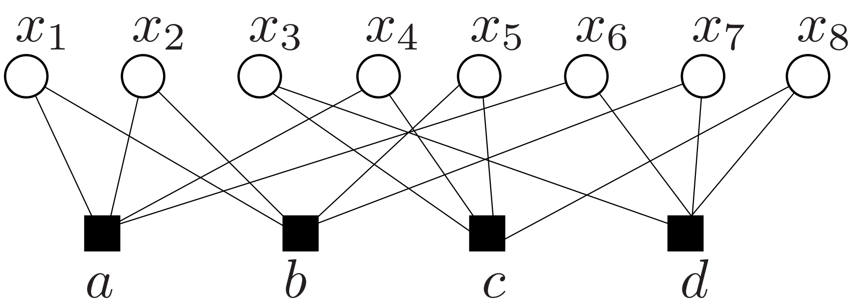
- message-passing: now standard method in various domains (coding, physics, computer vision, computational biology....)
- linear programming (LP) relaxation: standard method in computer science, operations research etc.
- turn out to be numerous connections between these two classes of methods
- some useful features of LP relaxation:
  - certificates of correctness
  - hierarchies of relaxations (guaranteed improvement; increased cost)
  - distinct conceptual perspective on message-passing
  - alternative avenue to finite-length results

Outline

1. Background
   - Motivation
   - First-order (tree-based) relaxation for combinatorial optimization
   - Connections to physics and message-passing
2. LP relaxation for LDPC decoding
   - past and on-going work
   - constant fraction in adversarial setting
   - notion of dual witness
3. Probabilistic analysis of LP decoding
   - combinatorial characterization via hypergraph flow
   - improved dual witness: generalized (p, q) matchings
   - "almost-always" expansion

Combinatorial optimization on factor graphs

- consider a combinatorial optimization problem with objective defined by factor graph $G = (V, F)$



$V \equiv$ variable nodes

$F \equiv$ factor nodes

$E \equiv$ variable–factor edges

- variable $x_i \in \{0, 1, \ldots, m-1\}$ associated with node $i \in V$

- local cost $\psi_a(x_{V(a)})$ at factor a over variable neighbors $V(a)$

- goal: maximize cost formed by product of factors

$$\arg\max_x \; G(x) \quad := \quad \arg \max_{x \in \{0,1,\ldots,m-1\}^n} \left\{ \prod_{i \in V} \psi_i(x_i) \; \prod_{a \in F} \psi_a(x_{V(a)}) \right\}.$$

From integer program to equivalent moment problem

1. Cost function is additive over graph structure:

$$F^* = \max_{x \in \mathcal{X}^n} F(x) \quad = \quad \max_{x} \left\{ \sum_{i \in V} \log \psi_i(x_i) + \sum_{a \in F} \log \psi_a(x_{N(a)}) \right\}.$$

2. Reformulate as equivalent optimization over probability distributions μ with support over $x \in \mathcal{X}^n$

$$F^* \quad = \quad \max_{\mu \in \mathcal{Q}} \sum_{x \in \mathcal{X}^n} \mu(x) \left[\sum_{i \in V} \log \psi_i(x_i) + \sum_{a \in F} \log \psi_a(x_{N(a)}) \right].$$

3. Reformulate again as equivalent optimization over globally consistent marginal distributions $\{\mu_i, i \in V\} \cup \{\mu_a,\ a \in F\}$:

$$F^* \;=\; \max_{\mu_i, \mu_a \in \mathcal{M}} \left[\sum_{i \in V} \sum_{x_i} \mu_i(x_i) \log \psi_i(x_i) + \sum_{a \in F} \sum_{x_a} \mu_a(x_a) \log \psi_a(x_{N(a)}) \right].$$

Marginal polytope for graphical model

- How hard is to an integer program (IP) on the graph G?
- Equivalent question: how hard is to characterize the marginal polytope?

Marginal polytope for factor graph $G = (V, F)$:

$$\begin{aligned}
\mu_i(\cdot) &= \text{local marginal over } x_i, \quad i \in V \\
\mu_a(\cdot) &= \text{local marginal over } x_{N(a)} \text{ at factor } a, \quad a \in F \\
\text{MARG}(G) &= \{\mu_i, i \in V, \text{ and } \mu_a, a \in F \mid (\mu_i, \mu_a) \quad \text{consistent with global } q(\cdot)\}.
\end{aligned}$$

- $\text{MARG}(G)$ has $\mathcal{O}(n)$ facets for trees
- $\mathcal{O}(m^t\, n)$ facets for graphs of treewidth t
- super-exponential # facets for general graphs

(DezLau97, WaiJor03)

Tree-based (1^{st}-order) LP relaxation

- impose *local normalization constraints* on each pseudo-marginal μ_i

$$\sum_{x_i} \mu_i(x_i) \quad = 1.$$

- impose *local marginalization constraints* on each factor pseudomarginal μ_a:

$$\sum_{x_i, i \in N(a) \backslash j} \mu_a(x_{N(a)}) \quad = \mu_j(x_j).$$

- combined with non-negativity constraints, call resulting polytope $\text{LOCAL}_1(G)$

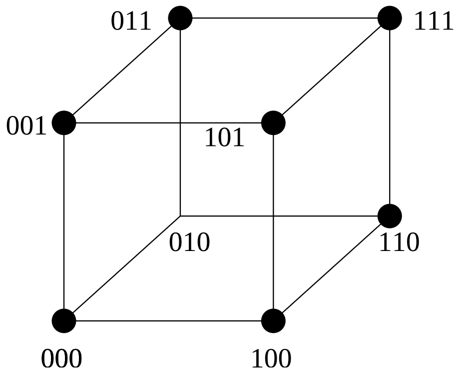
Some observations:

1. For any tree, $\text{LOCAL}_1(T) = \text{MARG}(T)$.
2. For general graphs, $\text{MARG}(G) \subsetneq \text{LOCAL}_1(G)$

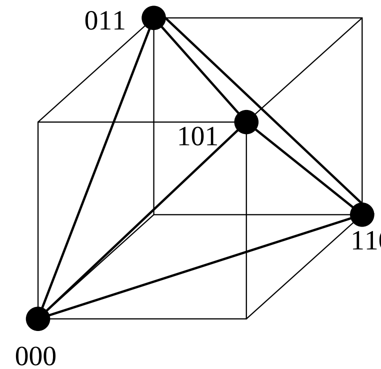
Codeword polytope

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

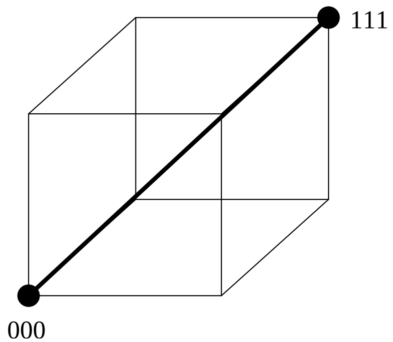
$$\mathrm{CH}(\mathbb{C}) = \Big\{\mu \in [0,1]^n \mid \mu_s = \sum_{\mathbf{x}\in\mathbb{C}} p(\mathbf{x})\, x_s\Big\}$$



(a) Uncoded



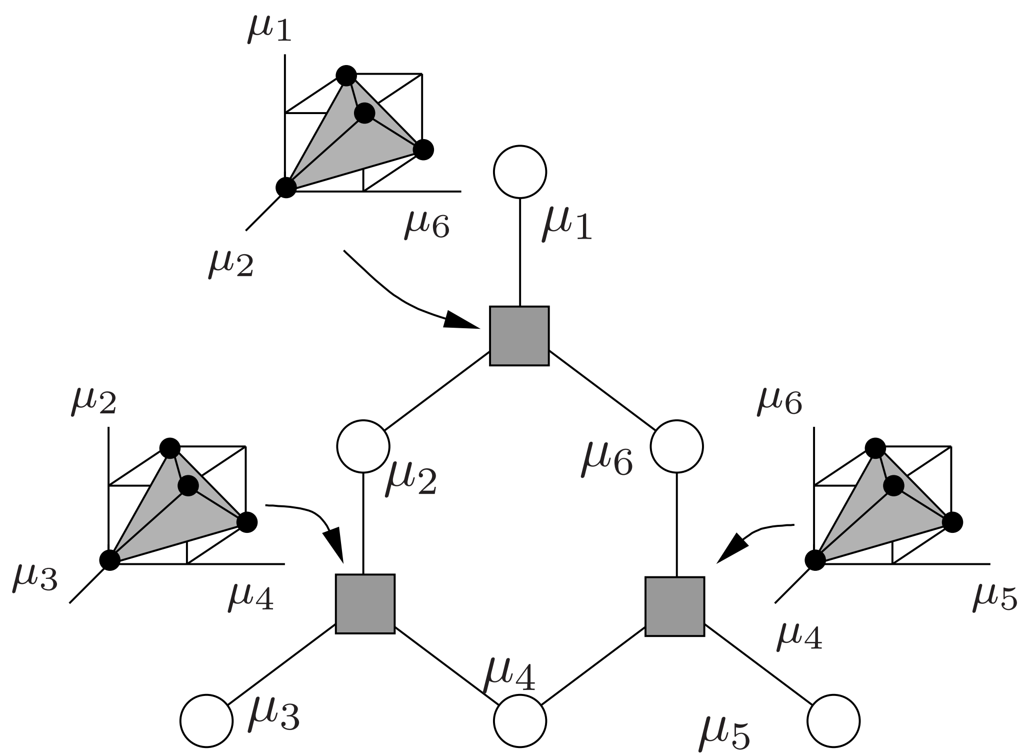
(b) One check



(c) Two checks

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

First-order relaxation for decoding



- each parity check $a \in C$ defines a *local codeword polytope* $\mathrm{LOCAL}_1(a)$
- first-order relaxation obtained by imposing all local constraints:

$$\mathrm{LOCAL}_1(\mathbb{C}) \;:=\; \cap_{a \in C} \, \mathrm{LOCAL}_1(a).$$

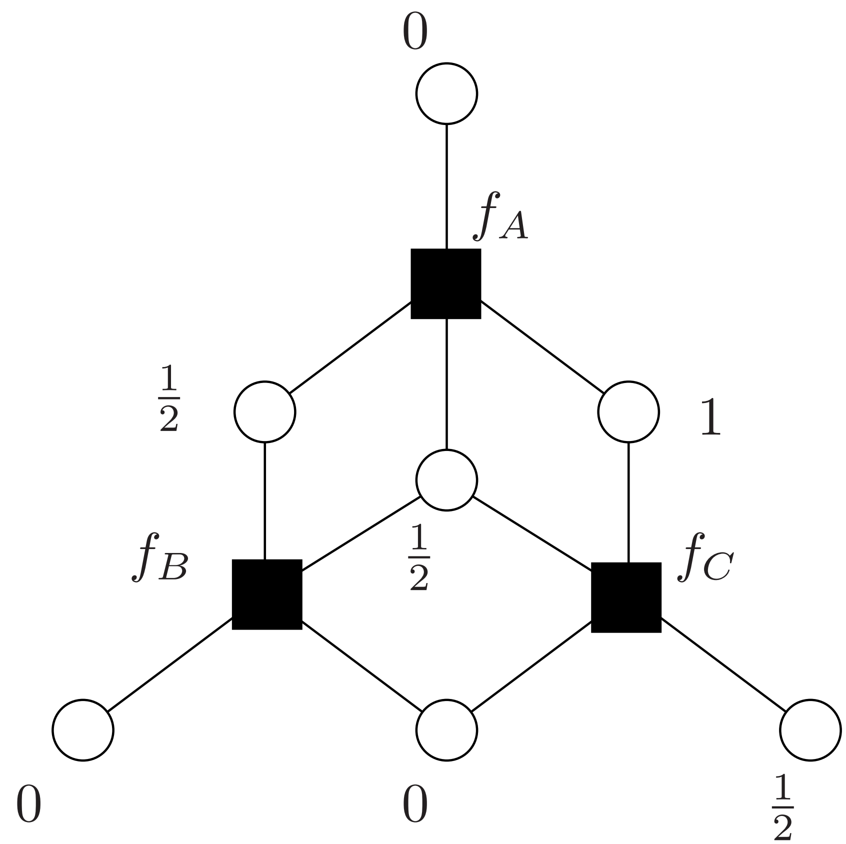
Illustration of fractional vertex

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 belongs to the first-order relaxed polytope $\mathrm{LOCAL}_1(\mathbb{C})$.

Some connections to physics and message-passing

- relaxed polytope $\text{LOCAL}_1(G)$ is constraint set in the Bethe variational principle (YedFreWei02)

- Kikuchi and cluster variational principles: exploit higher-order relaxations $\text{LOCAL}_k(G)$ in a hypertree sequence

- for any tree T, max-product (Viterbi) is a dual algorithm for solving linear program over $\text{LOCAL}_1(T)$

- general connection between ordinary max-product and relaxed LP? not valid in general (WaiJaaWil05)

- zero-temperature limits of sum-product $\longrightarrow$ LP solutions? not in general, but valid for "convexified" entropy approximations

Tree-reweighted max-product algorithm

Modified message update from node t to node s: (WaiJaaWil02)

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

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.

Reweighted max-product and linear programming

Theorem: For "suitable choice" of edge weights $\boldsymbol{\rho_e}$, reweighted max-product has the properties:

(a) Any fixed point M^* for which the pseudo-max-marginals $\tau^*_s(x_s) \propto \psi_s(x_s) \prod_{t \in N(s)} [M_{ts}(x_s)]^{\rho_{st}}$ have unique optimum specifies an integral optimum LP solution. (WaiJaaWil05)

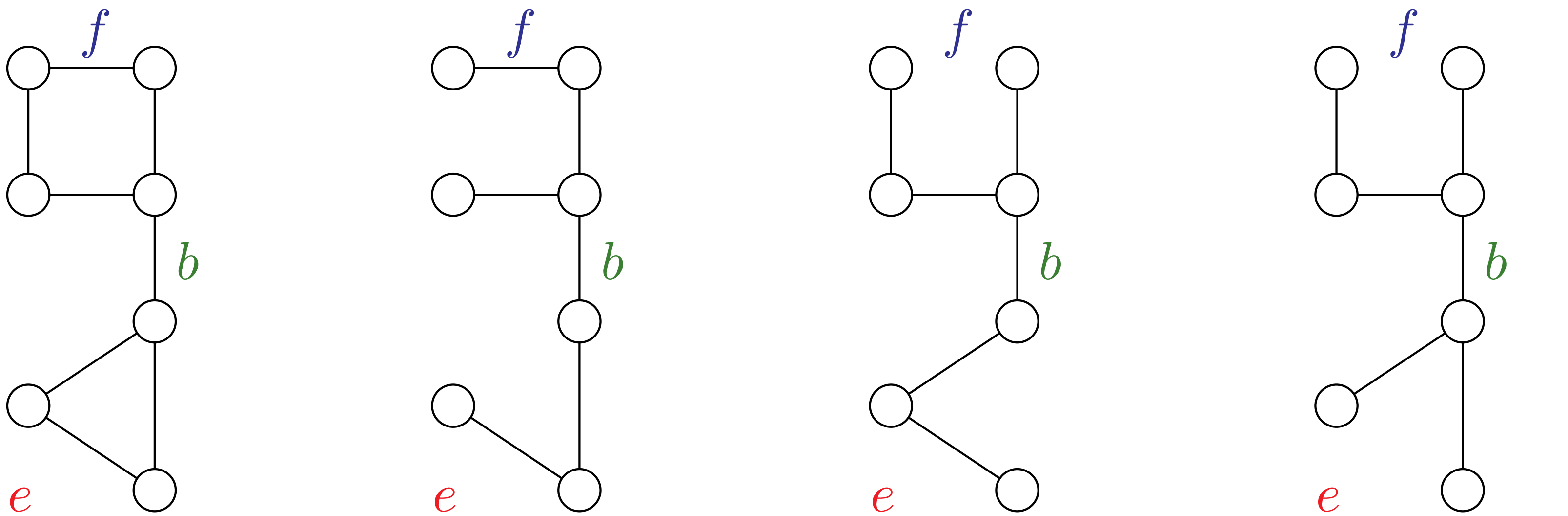
(b) For binary problems (with pairwise interactions), any fixed point M^* is an optimal solution to the dual LP. (KolWai05).

Remarks:

1. Some convergence guarantees (but still relatively weak). (Kol06)
2. From case (b): reweighted max-product has same behavior as first-order LP relaxation for various IPs (e.g., Ising ground states; min-cut; matching; vertex cover).

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 $\boldsymbol{\rho}$?



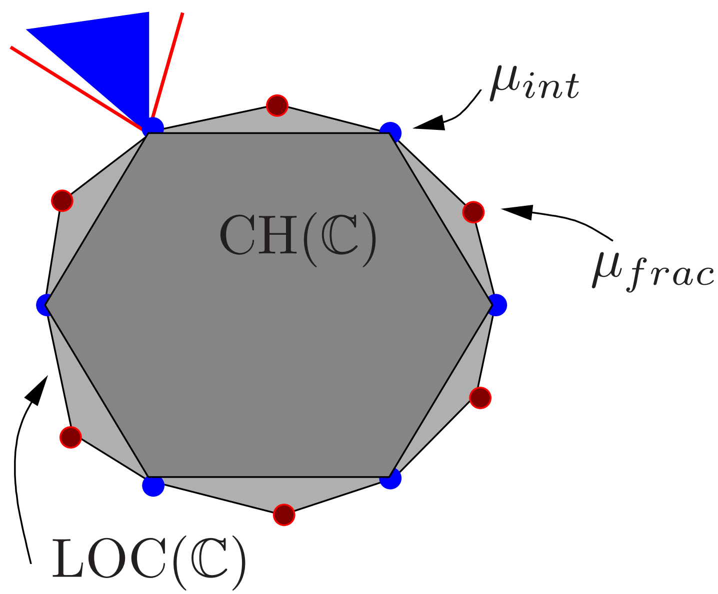
(a) Original (b) $\rho(T^1) = \frac{1}{3}$ (c) $\rho(T^2) = \frac{1}{3}$ (d) $\rho(T^3) = \frac{1}{3}$

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 $\mathbb{T}(G)$.

§2. LP relaxation for decoding

- basic LP decoder: solve first-order LP relaxation (with cost vector defined by channel) (FelWaiKar03)



- two vertex types: integral (codewords) and fractional (pseudocodewords)
- channel-dependent pseudoweight governs performance:

$$\text{BSC pseudoweight} \quad = \quad \min\left\{ k \mid \sum_{i=1}^{k} x_{(i)} \geq \sum_{i=k+1}^{n} x_{(i)} \right\}.$$

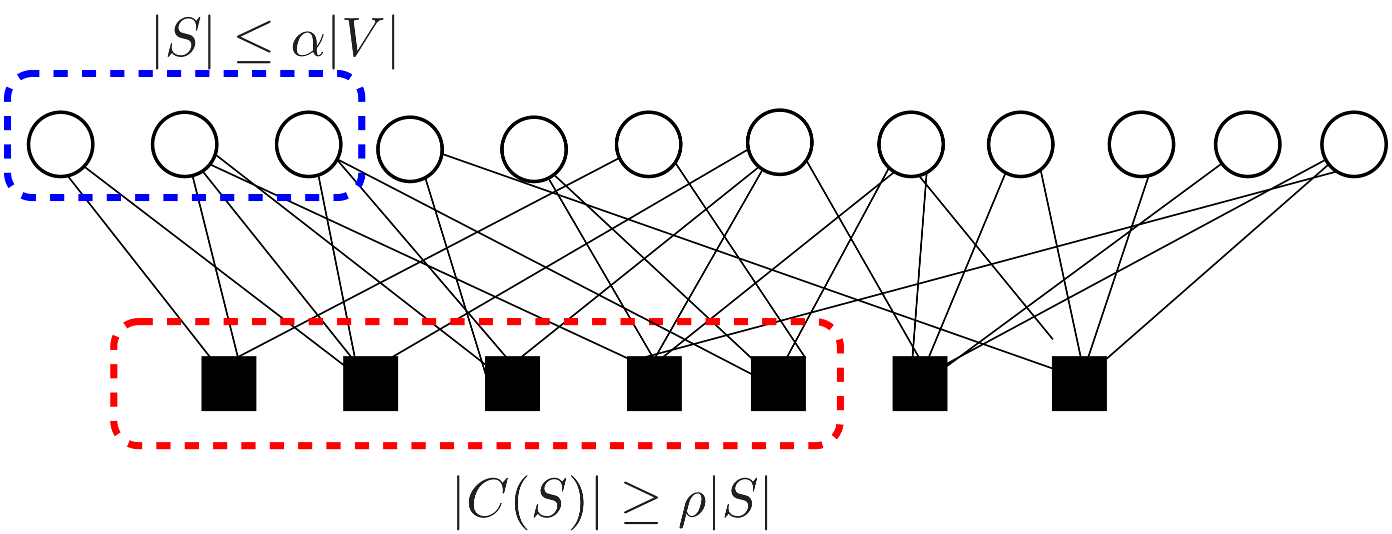
$$\text{AWGN pseudoweight} \quad = \quad \frac{\|x\|_1^2}{\|x\|_2^2}$$

Some known results

- empirical results on LP decoding: slightly better than max-product, slightly worse than sum-product
- LP decoding equivalent to message-passing for binary erasure channel (stopping sets $\Longleftrightarrow$ pseudocodewords)
- positive result: LP pseudoweight grows linearly for expander codes and the binary symmetric channel (Fel+04)
- negative result: sublinear LP pseudoweight for Gaussian channel (KoeVon03, VonKoe05)
- various extensions to basic LP algorithm
  - adaptive LP decoding (TagSie06)
  - stopping set redundancy for BEC (SchVar06)
  - facet guessing (DimWai06)
  - loop corrections for LP decoding (CheChe06)

Codes based on expander graphs

- previous work on expander codes (e.g., SipSpi02; BurMil02; BarZem02)
- graph expansion: yields stronger results beyond girth-based analysis



- **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

Worst-case constant fraction for expanders

Theorem: Let $\mathbb{C}$ be an LDPC described by a factor graph G with regular variable (bit) degree d_v. Suppose that G is an $(\alpha, \delta d_v)$-expander, where $\delta > 2/3 + 1/(3d_v)$ and δd_v is an integer.

Then the LP decoder can correct any pattern of $\frac{3\delta-2}{2\delta-1}(\alpha n)$ bit flips.

(FelMalSerSteWai, ISIT-04)

Comments:

- key technical device: notice of dual witness for LP success
  - LP succeeds when 0^n sent $\iff$ primal optimum $p^* = 0$
  - suffices to construct dual optimal solution with $q^* = 0$
- caveat: constant fraction very low (e.g., $c = 0.00017$ for $R = 0.5$)
- potential gaps in the analysis
  - analysis adversarial in nature
  - dual witness relatively weak

Proof technique: Construction of dual witness

Primal LP: Vars. $\{\mu_i,\ i \in V\}$, $\{\mu_{a,J},\ a \in F, J \subseteq N(a), \quad |J| \text{ even}\}$

$$\text{min.} \quad \sum_{i \in V} \theta_i \mu_i \quad \text{s.t.} \begin{cases} \mu_{a,J} \geq 0 \\ \sum_{J \in \mathbb{C}(a)} \mu_{a,J} = 1 \\ \sum_{J \in \mathbb{C}(a), J_v = 1} \mu_{a,J} = \mu_v \end{cases}$$

Dual LP: Vars. $\{v_a,\ a \in F\}$ $\{\tau_{ia},\ (i,a) \in E\}$ unconstrained

$$\text{max.} \quad \sum_{a \in F} v_a \quad \text{s.t.} \begin{cases} \sum_{i \in S} \tau_{ia} \geq v_a \text{ for all} \quad a \in C, J \subseteq C(a), |J| \text{ even} \\ \sum_{a \in N(i)} \tau_{ia} \leq \theta_i \qquad \text{for all } i \in V \end{cases}$$

Dual witness to zero-valued primal solution

- assume WLOG that 0^n is sent: suffices to construct a dual solution with value $q^* = 0$
- dual LP simplifies substantially as follows:

Dual feasibility: Find real numbers $\{\tau_{ia},\ (i,a) \in E\}$ such that

$$\begin{aligned} \tau_{ia} + \tau_{ja} &\geq 0 && \forall\, a \in C, \text{ and } i, j \in N(a) \\ \sum_{a \in N(i)} \tau_{ia} &< \theta_i && \text{for all } i \in V \end{aligned}$$

- random weights $\theta_i \in \mathbb{R}$ defined by channel; e.g., for binary symmetric channel

$$\theta_i = \begin{cases} 1 & \text{with prob. } 1-p \\ -1 & \text{with prob. } p \end{cases}$$

§3. Probabilistic analysis of LP decoding over BSC

Consider an ensemble of LDPC codes with rate R, regular vertex degree d_v, and blocklength n. Suppose that the code is a $(\nu, \left(\frac{p}{d_v}\right) d_v)$ expander.

Theorem: For each (R, d_v, n), we specify fractions $\alpha > 0$ and error exponents $c > 0$ such that the LP decoder succeeds with probability $1 - \exp(-cn)$ over the space of bit flips $\leq \lfloor \alpha n \rfloor$. (DasDimKarWai07)

Remarks:

- the correctable fraction α is always larger than the worst case guarantee $\frac{3\frac{p}{d_v}-2}{2\frac{p}{d_v}-1}\nu$.

- concrete example: rate $R = 0.5$, degree $d_v = 8$ and $p = 6$ yields a correctable fraction $\alpha = 0.002$.

Hyperflow-based dual witness

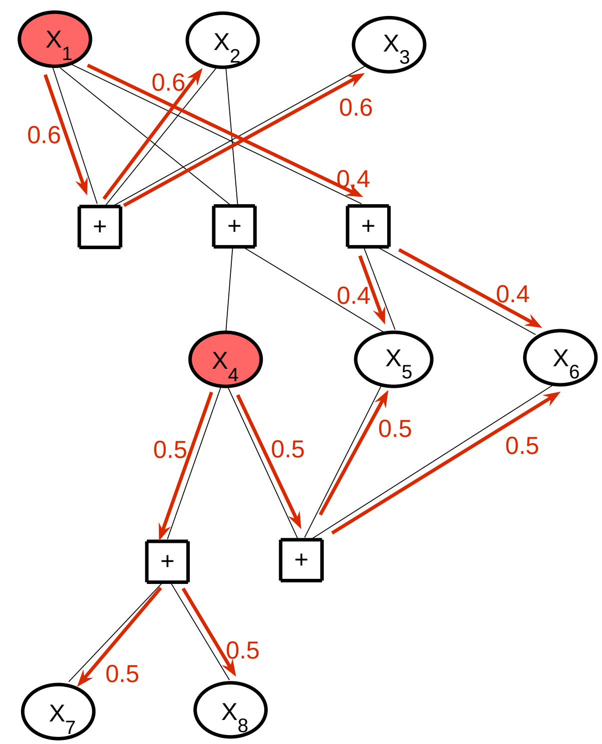
(DasDimKarWai07)

A *hyperflow* is a collection of weights $\{\tau_{ia}, (i,a) \in E\}$ such that:
(a) for each check $a \in F$, exists some $\gamma_a \geq 0$ and privileged neighbor $i^* \in N(a)$ such that

$$\tau_{ia} = \begin{cases} -\gamma_a & \text{for } i = i^* \\ +\gamma_a & \text{for } i \neq i^*. \end{cases}.$$

(b) $\sum_{a \in N(i)} \tau_{ia} < \theta_i$ for all $i \in V$.

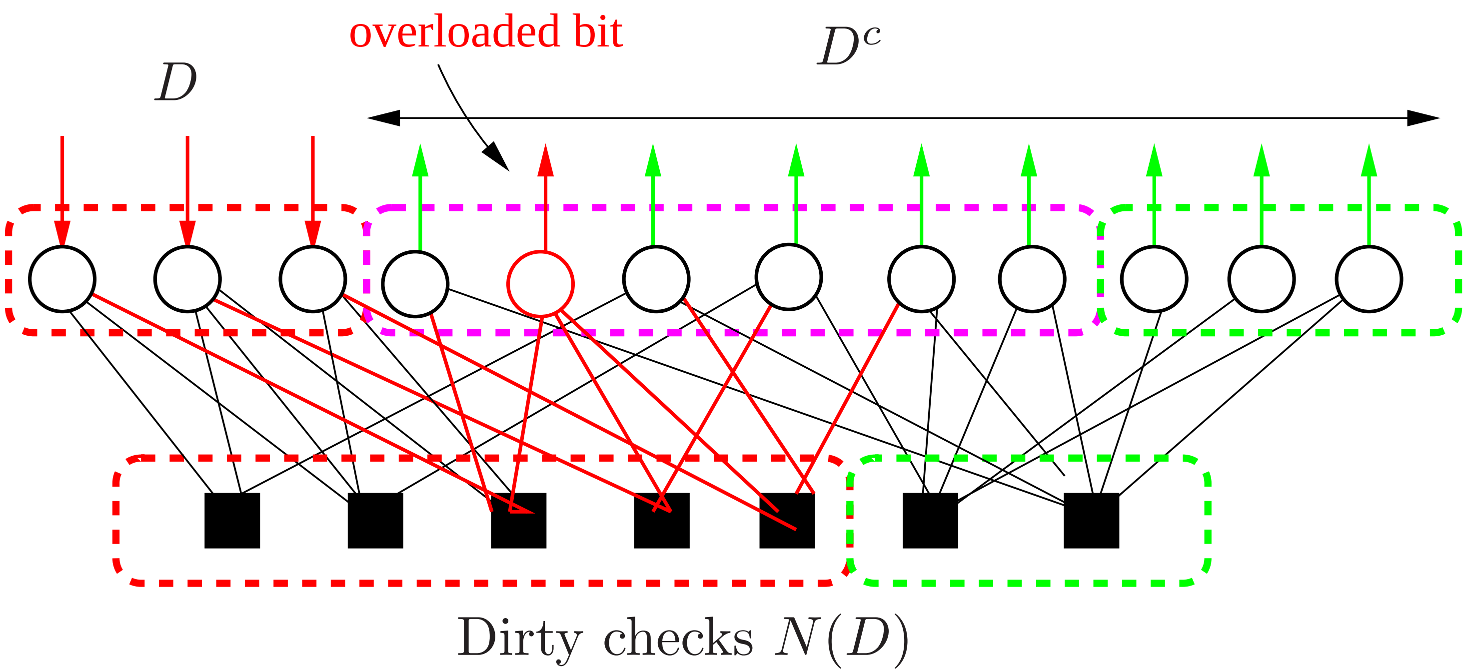
Proposition: A hyperflow exists $\Longleftrightarrow$ $\exists$ a dual feasible point with zero value.



Hyperflow (epidemic) interpretation:

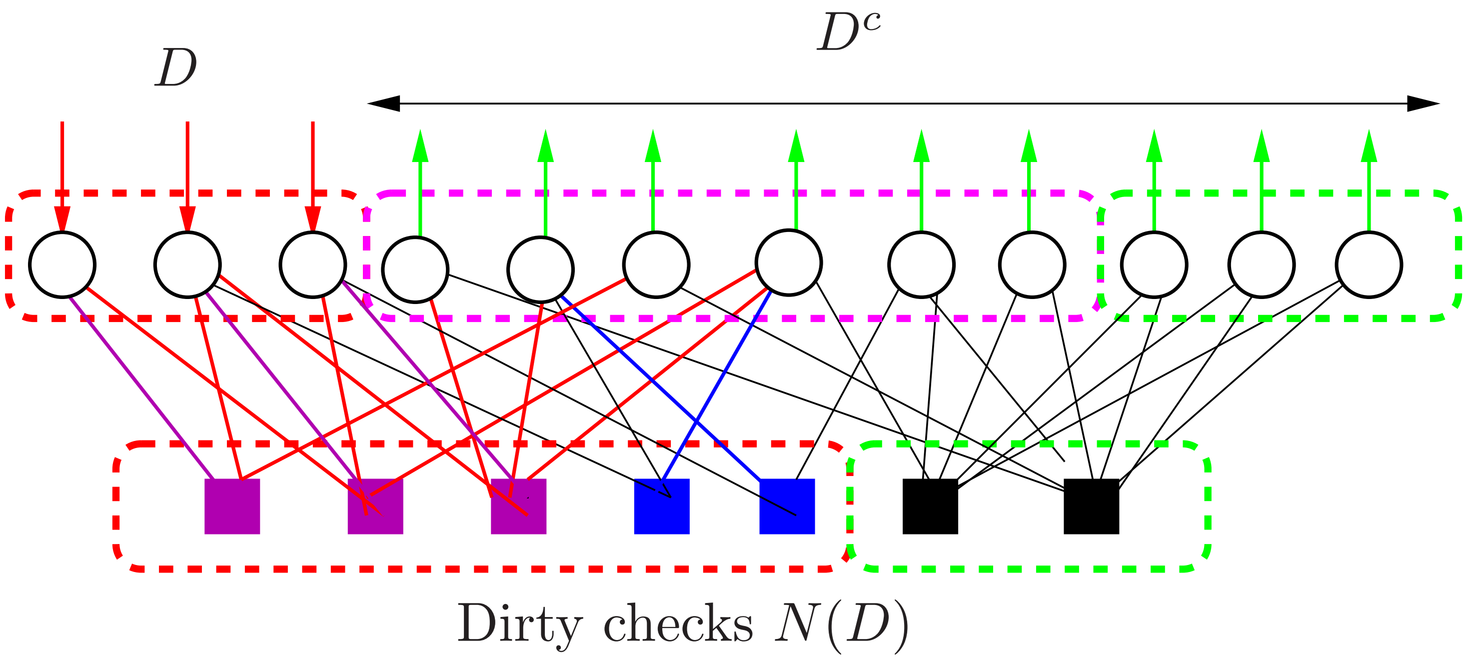
- each flipped bit adds 1 unit of "poison"; each clean bit absorbs at most 1 unit
- each infected check relays poison to all of its neighbors

Naive routing of poison may fail



- need to route 1 unit of poison away from each flipped bit
- each unflipped bit can neutralize at most one unit
- naive routing of poison can lead to overload

Routing poison via generalized matching



Definition: A (p, q)-matching is defined by the conditions:

(i) every flipped bit $i \in D$ is matched with p distinct checks.

(ii) every unflipped bit $j \in D^c$ matched with $\max\{Z_j - (d_v - q),\ 0\}$ checks from $N(D)$, where $Z_j = |N(j) \cap N(D)|$.

Generalized matching implies hyperflow

Lemma: Any (p, q) matching with $2p + q > 2d_v$ can be used to construct a valid hyperflow.

Proof:

- construct hyperflow with each flipped bit routing $\gamma \geq 0$ units to each of p checks
- each flipped bit can receive at most $(d_v - p)\gamma$ units from other dirty checks (to which it is not matched)
- hence we require that $-p\gamma + (d_v - p)\gamma < -1$, or $\gamma > 1/(2p - d_v)$
- each unflipped bit receives at most $(d_v - q)\gamma$ units so that we need $\gamma < 1/(d_v - q)$

High-level overview of key steps

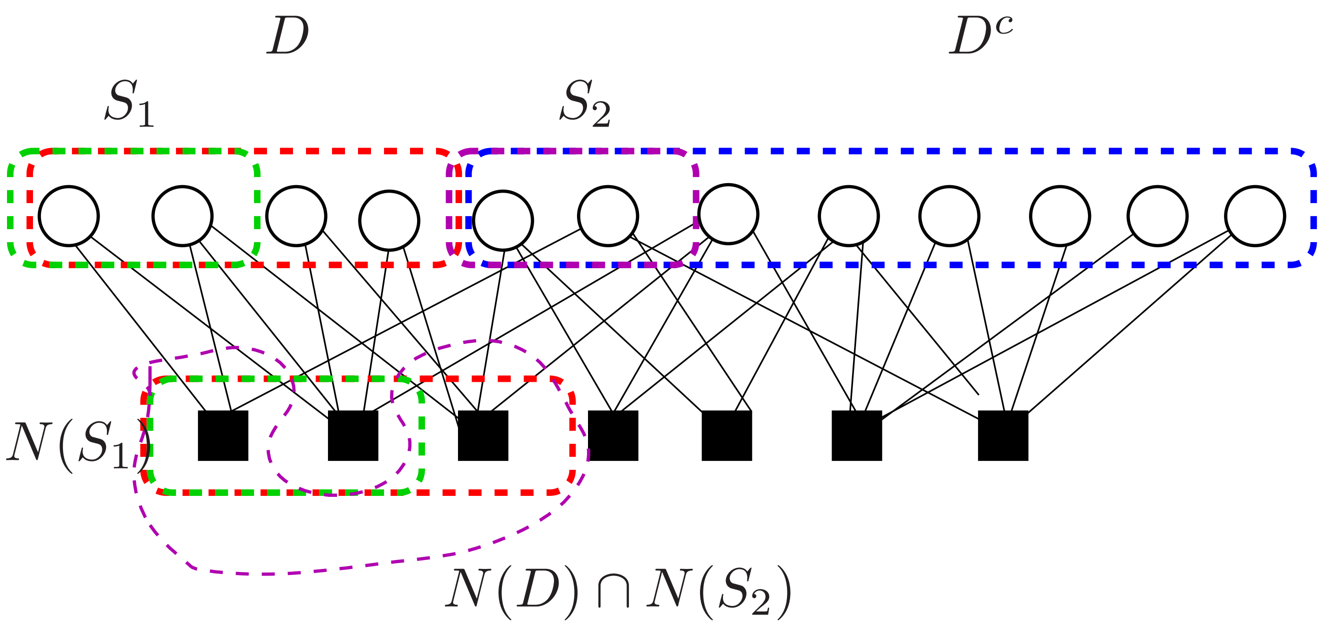
1. Randomly constructed LDPC is "almost-always" expander with high probability (w.h.p.)
    - weaker notion than classical expansion: holds for larger sizes
    - proof: union bounds plus martingale concentration

2. Prove that an "almost-always" expander will have a generalized matching w.h.p.
    - requires concentration statements
    - generalized Hall's theorem

3. Generalized matching guarantees existence of hyperflow.

4. Valid hyperflow is a dual witness for LP decoding succcess.

Generalized matching and Hall's theorem



- by generalized Hall's theorem, (p,q)-matching fails to exist if only if there exist subsets $S_1 \subseteq D$ and $S_2 \subseteq D^c$ that *contract*:

$$\underbrace{|N(S_1) \cup [N(S_2) \cap N(D)]|}_{\text{available matches}} \quad \leq \quad \underbrace{p|S_1| + \sum_{j \in S_2} \max\{0, q - (d_v - Z_j)\}}_{\text{total \# requests}}.$$

Analysis over a simpler random ensemble

- analysis in standard ensemble: complicated due to coupling between $N(D)$ and number of requests from D^c

- consider simplified (but equivalent) ensemble:
  - each node in D^c chooses $Z_j \sim \text{Bin}(d_v, \frac{|N(D)|}{m})$
  - chooses a subset from $N(D)$ of size Z_j

- LP error prob. (over random subset D) bounded by probability of existing contractive subsets $S_1 \subseteq D$ and $S_2 \subseteq D^c$:

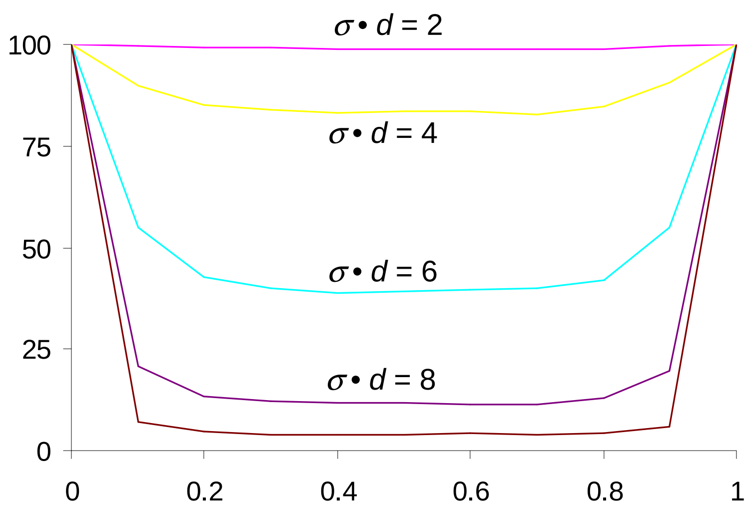
$$\mathbb{P}\Big[\exists \quad S_1 \subseteq D, \quad S_2 \subseteq D^c \;\mid\; |N(S_1) \cup [N(S_2) \cap N(D)]| \leq p|S_1| + \sum_{j \in S_2} R_j\Big]$$

- argument establishes existence of "almost-always expanders" (with parameters much larger than worst-case sense)

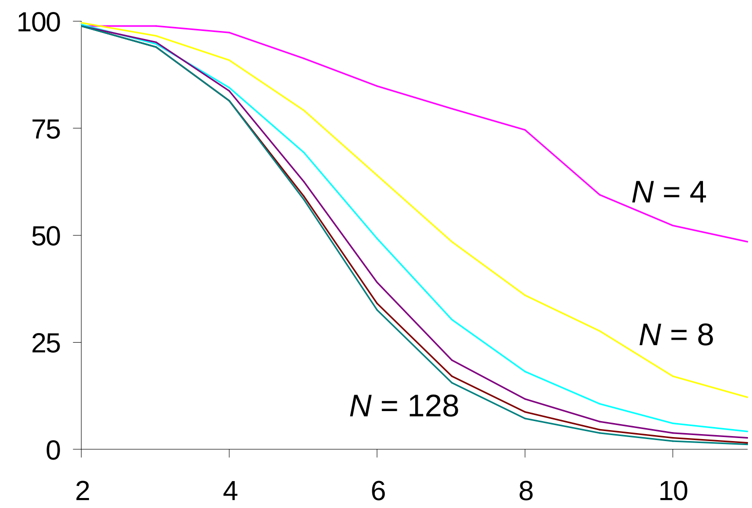
Summary

- linear programming relaxations for optimization in graphical models
  - various connections to message-passing
  - alternative route for non-asymptotic results
- probabilistic analysis of LP decoding for BSC
  - hyperflow characterization of dual LP
  - yields improved error-correction guarantees
  - exploits “almost-always” expander (other applications?)
- various open directions:
  - average-case analysis for other problems, ensembles?
  - polytope structure for survey-propagation and SAT?
  - guarantees on approximation hierarchies?

LP relaxation for "near-sub-modular" problems



(a) Increased frustration



(b) Increased coupling