Note Title 4/9/2008

598 Lecture 17

Today: Graph-Based Codes

- Distance + Rate
- Decoding

Review:

Part I : Existential & Universal bounds on codes (Random, Packing, Orthogonality ...)

Part II : Algebraic Codes : Meet many limitation bounds for some settings of parameters

Part III : Algebraic Decoding : Large (Optimal?) fraction of errors, worst-case.

Why study yet more codes?

- Can ask for faster decoding
  - linear time?
  - Sublinear time?
  - concrete parameters ... not "pure" asymptotics?

Today's Motivation: Linear-time algorithms.

Approach: use codes based on graphs.

History:

1. [Gallager '63]: Introduced codes based on graphs as a means of getting Good binary codes, with Efficient Decoding.

Constructions: (a) Random procedure to construct codes

(b) Decoding Algorithm (seemingly efficient)

Theorem: W.h.p. code meets GV bound.

2. [Tanner '84]: Richer class of graph-theoretic codes.

3. [Sipser Spielman '94]: Provably efficient Algorithms! (Worst-case errors)

Motivation:

- Codewords of linear code recognized by parity check matrix H

$$\begin{bmatrix} x_1 & \cdots & x_n \end{bmatrix} \begin{bmatrix} & & \\ & h_{ij} & \\ & & \end{bmatrix} \stackrel{?}{=} \begin{bmatrix} 0 \end{bmatrix}$$

$$\underbrace{h_1 \cdots h_m}_{\uparrow} \qquad\qquad m = n-k$$

- Can it also help recover from error?

Say $(x \cdot H)_j \neq 0$. i.e., $\langle x, h_j \rangle \neq 0$

$\Rightarrow \exists$ error in one of the coordinates where h_j is non-zero.

- Is this (algorithmically) useful?

- For random matrix H, this is virtually useless. Half the coordinates are "flagged" by h_j. Correlation with error $\to 0$.

- But if H is sparse then
  $\Gamma(j) = \{i \mid h_{ij} \neq 0\}$ is small & saying one of those bits is in error could be quite useful.

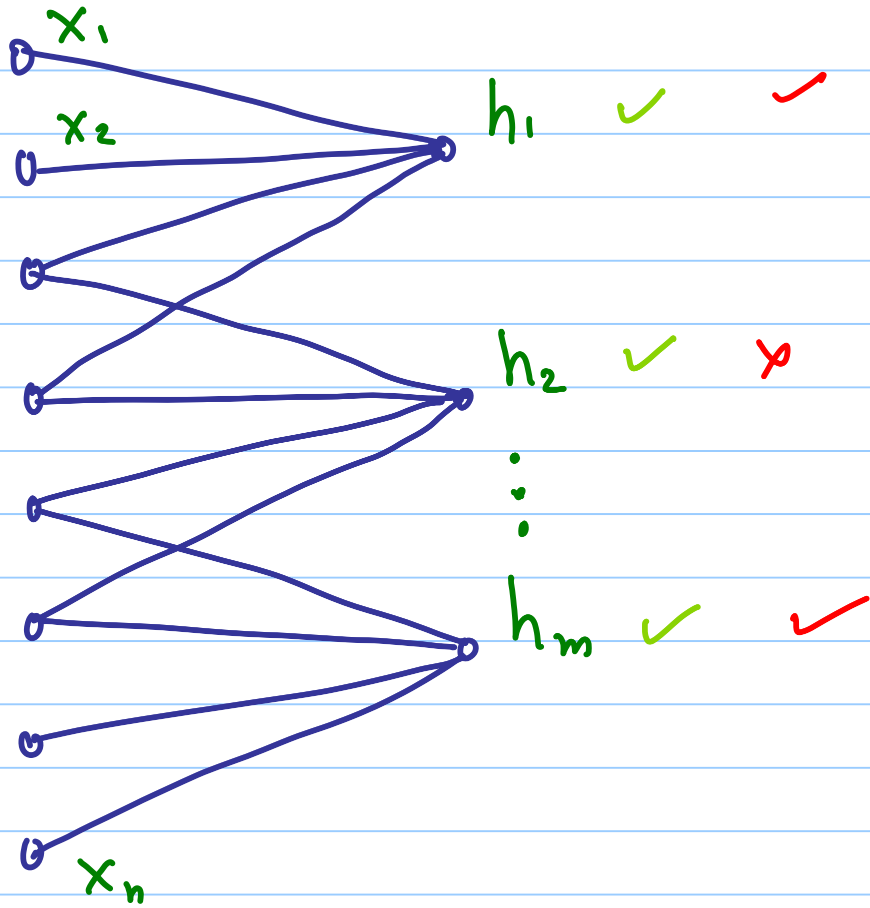
- Motivates LDPC = Low Density Parity Check Codes. ($\equiv H$ is sparse)

Graph Theoretic View

- H: Adjacency matrix of bipartite graph
- Rows of H = Variables - left vertices
- Columns of H = Constraints - right vertices
- Edges $i \leftrightarrow j \iff h_{ij} = 1$.
- Boolean assignment to variables forms codeword iff all constraints satisfied
- Constraint j satisfied if assignment to neighbors has parity 0.

EXAMPLE

0	0
1	1
0	1
1	0
1	0
1	1
0	1
0	0



codeword

non-codeword

[GALLAGER]

Construction: - Given: n, k, + integer parameter c

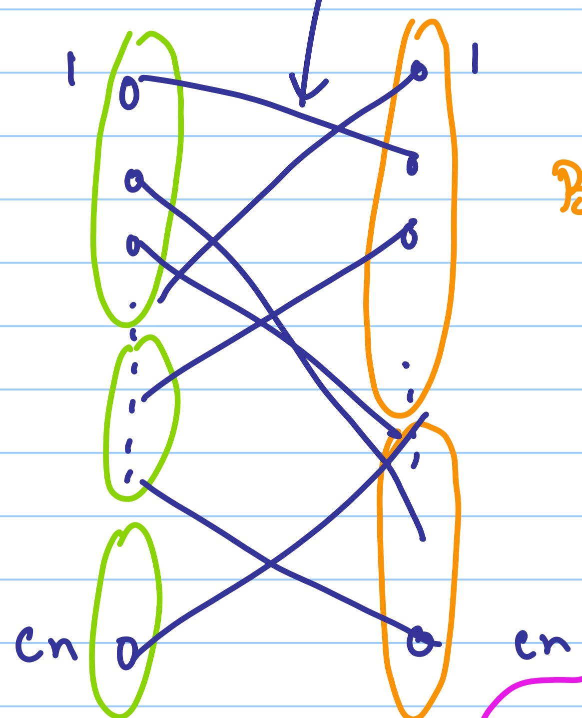
- Let $m = n-k$ & $d = \frac{cn}{m}$

  (assume d = integer)

- Pick random permutation

  $$\sigma : [cn] \longrightarrow [cn]$$

Blocks of c



Blocks of d

$h_{ij} = 1$

iff

$\sigma(c(i-1)+\ell)$

$\in \{d(j-1)+1 \dots dj\}$

for some $\ell \in [c]$

Theorem: Code reaches GV bound as $c \to \infty$.

Algorithm (Vaguely): Iterate in rounds ....

ℓ^{th} Round: • Start with prob. estimates p_i $\triangleq$ prob. i^{th} variable is 1.

• Constraints compute prob. they are satisfied.

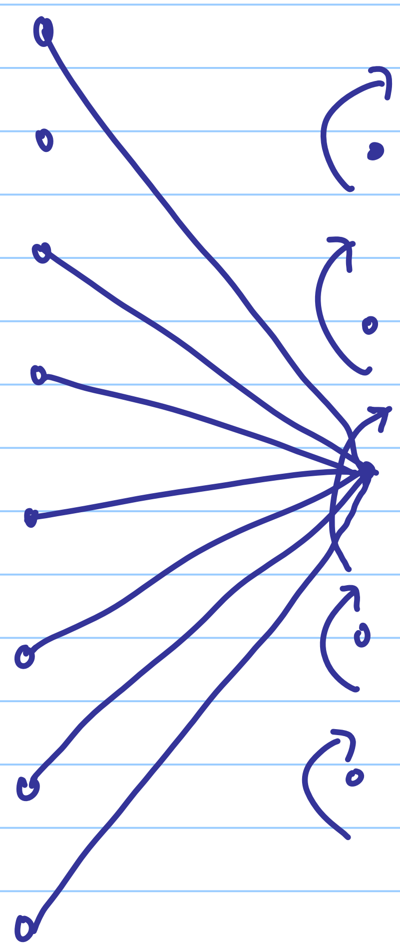
• Variables update their prob. p_i to improve prob. constraints are satisfied.

Rough assertion: Algorithm converges with random errors.

Precise theorems?

[Tanner] : Use more general constraints !

e.g.



in this order variables should be from [7,4,3] Hamming Code.

More general ? Not really, since this is still an LDPC Code ....

But gives better idea how to construct such codes ....

[Tanner]: Distance lower bounded by "girth" of underlying graph. Leads to concrete results.

("girth" = length of shortest cycle in graph)

[Sipser-Spielman]:
- Distance lower bounded by "expansion" of underlying graph.
- Expansion can also be used to <u>prove</u> performance under worst case errors.

Basic Graph Theoretic Definitions

- Degree $(u) \triangleq$ # vertices adjacent to u
- (c,d) - **bounded** graph : left degrees $\leq c$

  Right degrees $\leq d$.
- Neighborhood $\Gamma(S) = \{ v \mid \underset{u \leftrightarrow v}{\exists u \in S} \}$
- (γ, δ) - **expander** : $G = (L, R, E)$

  $|L| = n$, $|R| = m$

  $\forall S \subseteq L$ $\quad |S| \leq \delta n$

  $\Rightarrow |\Gamma(S)| \geq \gamma \cdot |S|$

Expansion ?

- One measure of connectivity
- Good expanders $\equiv$ large γ, δ

  $\Rightarrow$ well-connected
- How large can γ, δ be ?
  - Cleary $\delta < 1$.
  - Also $\delta < \frac{1}{\gamma} \cdot \frac{m}{n}$

More critical parameter : γ

- $\gamma > 0$ non-trivial to get (with $m < n$)
- $\gamma \leq c$ even single vertex does not expand by more.
- random graph [Gallager] gets

  $\gamma \to c$; $\delta = \Omega(\frac{1}{c} \cdot \frac{m}{n})$.
- Explicitly .... getting there ... Assume "YES WE CAN"

Distance via Expansion

Theorem [Sipser-Spielman]: If bip. graph $G = (L, R, E)$ is (c, d) - bounded
(γ, δ) - expander
& $\gamma > \frac{c}{2}$ then C_G, code associated with G has distance $\geq \delta$.

Key Concept: Unique Neighborhood $\Gamma^+(S)$

for $S \subseteq L$, $\Gamma^+(S) = \{ v \in R \mid \exists !\, u \in S \text{ s.t. } u \leftrightarrow v \}$

G is a $(\tilde{\gamma}, \delta)$ unique-expander if

$\forall S \subseteq L$, $|S| \leq \delta n \Rightarrow |\Gamma^+(S)| \geq \tilde{\gamma} \cdot |S|$.

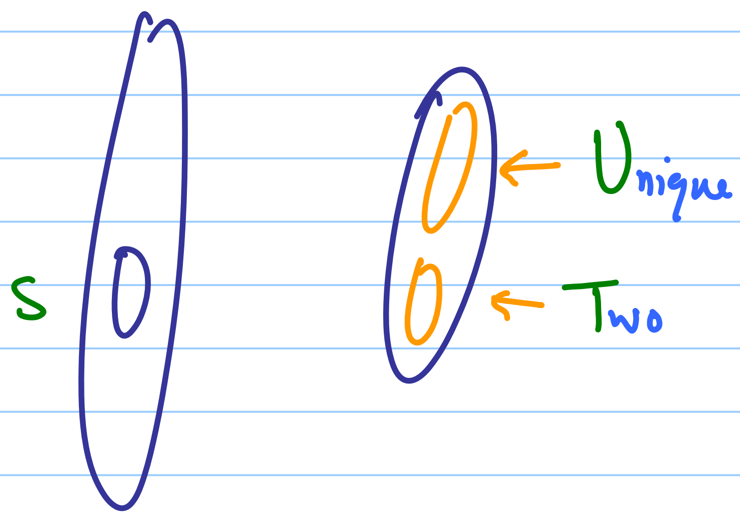
Key Lemma: G is (c,d)-bounded

(γ,δ)-expander

$\Rightarrow$ G is $(2\gamma - c, \delta)$-unique expander.

(useful only if $\gamma > \frac{c}{2}$)

Proof: Fix $S \subseteq L$



Lets count edges F incident to S

$$|F| \leq c \cdot |S|$$

$$|F| \geq |U| + 2 \cdot |T|$$

$$\Rightarrow \quad |U| + 2\cdot|T| \leq c \cdot |S| \qquad — (1)$$

But using expansion, we also have

$$|U| + |T| \geq \gamma \cdot |S| \quad - \quad ②$$

$② \times 2 - ①$

$$|U| \geq (2\gamma - c)|S| \quad \text{as desired} \quad \boxtimes$$

Proof of Distance:

- Let $x_1 \ldots x_n$ be vector/assignment of $\underline{wt < \delta n}$
- $S \triangleq \{ i \mid x_i = 1 \}$; $|S| < \delta n$
- $|\Gamma^+(S)| > (2\gamma - c) \cdot |S| \geq 1$
- $v \in \Gamma^+(S) \Rightarrow v^{th}$ constraint is not satisfied.

$\Rightarrow x_1 \ldots x_n$ <u>not a codeword</u> $\Rightarrow$ codeword has wt. $> \delta n + 1$ $\boxtimes$

Utility:

- At the time (1994) best expanders achieved $\gamma \to \infty$ with $\delta > 0$ as $c \to \infty$.
- No explicit constructions achieving $\gamma > \frac{c}{2}$.
- Two fixes:
  - Roll clock forward to 2002 ....
    [Capalbo, Reingold, Vadhan, Wigderson] construct nice expanders.
  - Use [Tanner]'s generalization.

A Convenient (potentially incorrect) Abstraction of Expander Technology in the 90s.

$\forall \gamma \quad \exists c$ s.t. $\forall d \quad \exists \delta > 0$ s.t. $\forall$ suff. large n

can construct (c,d)- bounded

(γ, δ)- expander on n left vertices.

Theorem: [Tanner + SS]: Combining G that is (c,d)- bounded with (γ, δ)- expander with a $[d, \ell, \Delta]$- code for constraints gives code of distance δ if $\gamma > \frac{c}{\Delta}$

Proof: Similar to before. Fix S, $|S| \leq \delta n$.

Let U_Δ = neighbors with fewer than Δ neighbors in S

T_Δ = rest.

Have $|U_\Delta| + |T_\Delta| \geq \gamma \cdot |S|$

$$|U_\Delta| + \Delta \cdot |T_\Delta| \leq c \cdot |S|$$

$$\Rightarrow |U_\Delta| \geq \frac{1}{\Delta - 1}(\Delta\gamma - c)|S|$$

$\ldots$ $\boxtimes$

Using Theorem + "Abstraction"

- Pick γ as you like (hmm...?)
- Let C be as given by abstraction.
- Pick $\Delta > \frac{C}{\gamma}$
- Pick (d, ℓ) so that $[d, \ell, \Delta]$ code exists & $(d-\ell)m < n$

$$\Leftrightarrow (d-\ell)\, c < d$$

$d - \ell \geq \Delta \log d \Rightarrow$ code exists

$$c\, \Delta \log d < d$$

$$\Rightarrow \frac{d}{\log d} > \Delta C$$

So $\ell = \Delta c \log \Delta c$; $d = \ell + 2\Delta \log \ell$ works.