Codes on Graphs

CS598 SGT, April 7, 2015

Jonathan Ligo

Overview

1) Introduction to codes

2) Bounds on codes

3) Codes from Graphs

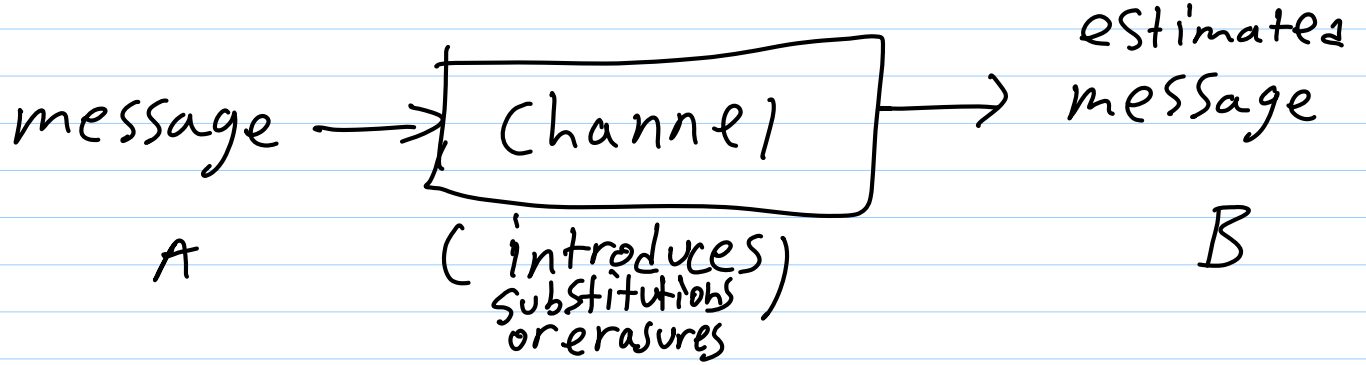
4) Decoding Expander Codes

5) LDPC codes and other cool stuff

Why do we need codes?

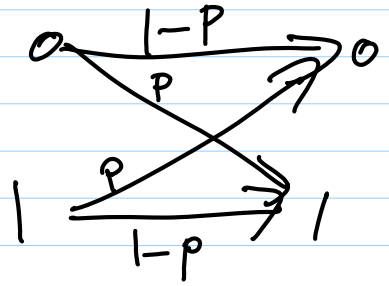
Motivation: Communications

Want to send message from A to B over channel



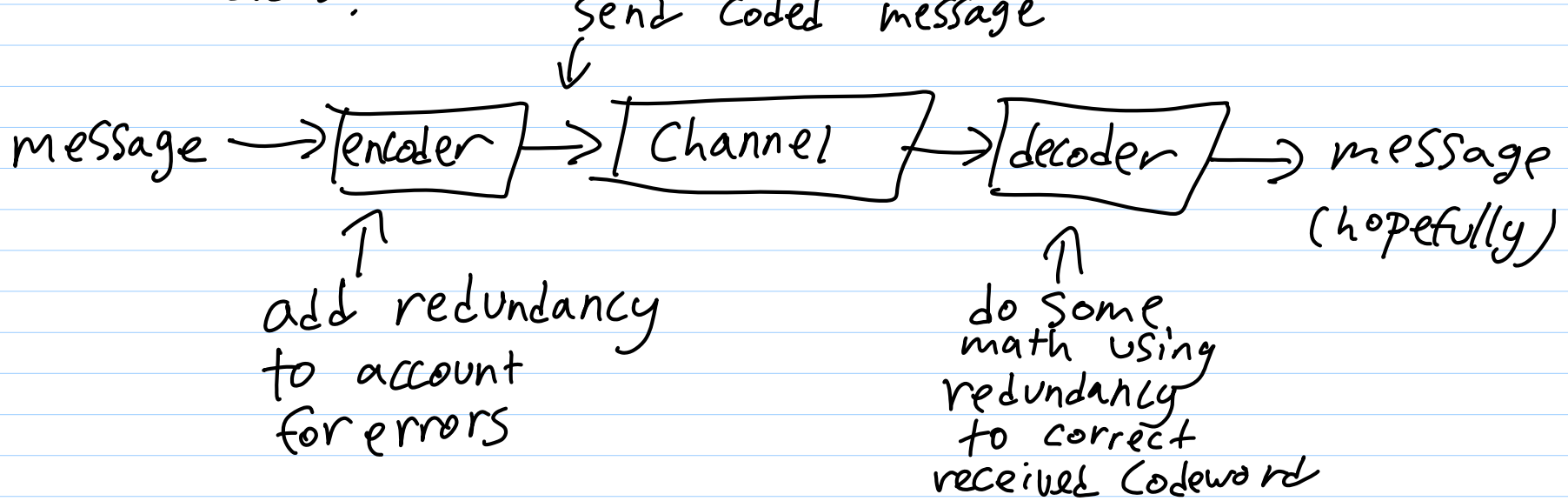
If we just send message over channel, we can't account for errors introduced by channel!

Example of a Channel: (Binary Symmetric Channel)



flip an input bit with Probability

Solution: Codes!



What is a code?

A (linear) code C with parameters $[n, k, d]$ over a finite field $\mathbb{F}$ is a subspace of $\mathbb{F}^n$ of dimension k. Its elements are called codewords.

n = code length

k = # of information symbols

$n-k$ = redundancy

$$\text{Rate} = \frac{k}{n}$$

$$d = \min_{\substack{x,y \in C \\ x \neq y}} d_H(x,y) = \min_{x \neq 0, x \in C} w_H(x)$$

d: distance of the code

$d_H(x,y)$: hamming distance (# of places where x & y differ)

$w_H(x)$: hamming weight (# of non-zeros)

What does d mean?

A code of distance d can do Error Correction:

— Correct up to any $\left\lfloor \frac{(d-1)}{2} \right\rfloor$ errors

Proof: Send codeword x, receive y and $d_H(x,y) \leq \frac{d-1}{2}$

Decode y to the nearest codeword $c \in C$.

Then, $d_H(c,y) \leq d_H(x,y) \leq \frac{d-1}{2}$

By triangle ineq, if $c \neq x$,

$$d \leq d_H(c,x) \leq d_H(c,y) + d_H(y,x) \leq d-1$$

Which is a contradiction, so we decode y to x correctly.

Example: Repetition code $([n,1,n])$

Want to send symbol $x \in \mathbb{F}$

Send $(\underbrace{x, \ldots, x}_{n})$

Decoder: Choose most frequent received symbol

If $\leq \frac{n-1}{2}$ errors, majority vote is right!

A code of distance of distance d can do Error detection:

- Can detect any $d-1$ errors

Proof: Since $d_H(x,y) \geq d$ for $x, y \in C$

it takes $\geq d-1$ errors to turn

One codeword into another

So, if received codeword $y \notin C$,

declare error

Example (Single parity Check $[n, n-1, 2]$):

Want to send $(X_1, \ldots, X_{n-1})$

Map to codeword $(X_1, \ldots, X_{n-1}, -\sum_{i=1}^{n-1} X_i)$

(Each codeword sums to 0)

Over $\mathbb{F}_2$: message = 0 1 1

codeword = 0 1 1 0

Receive = 0 1 1 1

↳ $0+1+1+1 = 1 \neq 0$

So detect 1 error

A code of distance d can do Erasure correction:

- An erasure is when a known position of a received codeword is missing

  e.g. 01111 was sent, 01_11 was received but where _ is is known
- Can correct up to $d-1$ erasures

  Proof: If y is received, decode $c \in C$ where c agrees with y on non-erased entries

Example: Single parity check

0_11 received

0011 decoded

So basically, we need:

- 2 distance / error corrected
- 1 distance / error or erasure detected Corrected

Good codes

A sequence of $[n_i, k_i, d_i]$ codes over $\mathbb{F}$ with $n_i \to \infty$ is said to have good distance if $\frac{d_i}{n_i} \to r > 0$

good rate if $\frac{k_i}{n_i} \to \delta > 0$

We will construct codes with good distance & rate (good codes).

Generator and Parity Check Matrices

A **generator matrix** for a $[n,k,d]$ code C is a $k \times n$ matrix whose rows are a basis for C.

Example: Parity Code $[n,n-1,2]$ Repetition Code $[n,1,n]$

$$G = \left[\, I_{n-1 \times n-1} \;\middle|\; \begin{matrix} -1 \\ \vdots \\ -1 \end{matrix} \right] \qquad G = [1 \cdots 1]$$

To encode a message $v = [v_1, v_2, \ldots, v_k]$, we multiply by G:

$$\text{message } v \longrightarrow \text{codeword } vG$$

(typically $O(kn)$ complexity)

A parity check matrix H is a $(n-k) \times n$ matrix such that $c \in C \iff Hc^T = 0$.

Receive $y = c + e \Rightarrow Hy^T = He^T$ (Syndrome)

A generator matrix is systematic if it is of the form $[I \mid A]$. A corresponding systematic parity check matrix is $[-A^T \mid I]$.

If code C has generator matrix G, parity check matrix H, its dual code $C^\perp$ consists of codewords $\perp$ to those of C, and has generator matrix H and parity check matrix G. $C^\perp = [n, n-k, d^\perp]$ and $(C^\perp)^\perp = C$

Example!

The $[n, 1, n]$ repetition code has

$$G_2 = [1 \cdots 1]$$

The $[n, n-1, 2]$ parity code has

$$G_1 = \left[\begin{array}{c|c} I & \begin{matrix} -1 \\ \vdots \\ -1 \end{matrix} \end{array}\right]$$

These form a dual pair.

Note: If H is parity check matrix for $[n, k, d]$ code, any $\leq d-1$ columns are linearly independent.

Bounds on Codes

Singleton bound: $d \leq n-k+1$ (necessary)

Proof: Take a systematic generator matrix $\left[I_{k\times k} \mid A \right]$.

Each row has $\leq n-k+1$ non-zero entries,

So $d \leq d_H(\text{row}, 0) \leq n-k+1$.

If $d = n-k+1$, code is called **maximum distance separable**.

Examples: $\mathbb{F}^n$, $[n, n-1, 2]$ parity, $[n, 1, n]$ repetition code

Reed-Solomon codes (most useful)

Application: RAID, CD's, etc.

Sphere packing bound: Volume in a sphere in $\mathbb{F}_q^n$ of radius $t = V_q(n,t) = \sum_{i=0}^{t} \binom{n}{i} (q-1)^i$

For a $[n,k,d]$ code, $V_q\left(n, \left\lfloor \frac{d-1}{2} \right\rfloor\right) \leq q^{n-k}$ (necessary)

Proof: Spheres of radius $\left\lfloor \frac{d-1}{2} \right\rfloor$ around each codeword are disjoint (else, 2 codewords have dist $< d$).

There are q^k codewords and q^n vectors in $\mathbb{F}_q^n$, so $q^k \cdot V_q\left(n, \left\lfloor \frac{d-1}{2} \right\rfloor\right) \leq q^n$.

Codes with equality are perfect codes.

Gilbert-Varshamov bound: (sufficient)

Let $n, k, d \in \mathbb{N}$ be s.t. $V_q(n-1, d-2) < q^{n-k}$.

Then, there exists a $[n, k, \geq d]$ code over $\mathbb{F}_q$.

Proof: Greedily construct a $(n-k) \times n$ parity check matrix H starting with $I_{n-k \times n-k}$.

Any $d-1$ columns must be linearly independent, so when adding column h_ℓ, it cannot be a linear combination of $d-2$ of the columns $1, \ldots, \ell-1$.

I.e. $h_\ell \neq [h_1 \cdots h_{\ell-1}] X$, $w_H(x) \leq d-2$

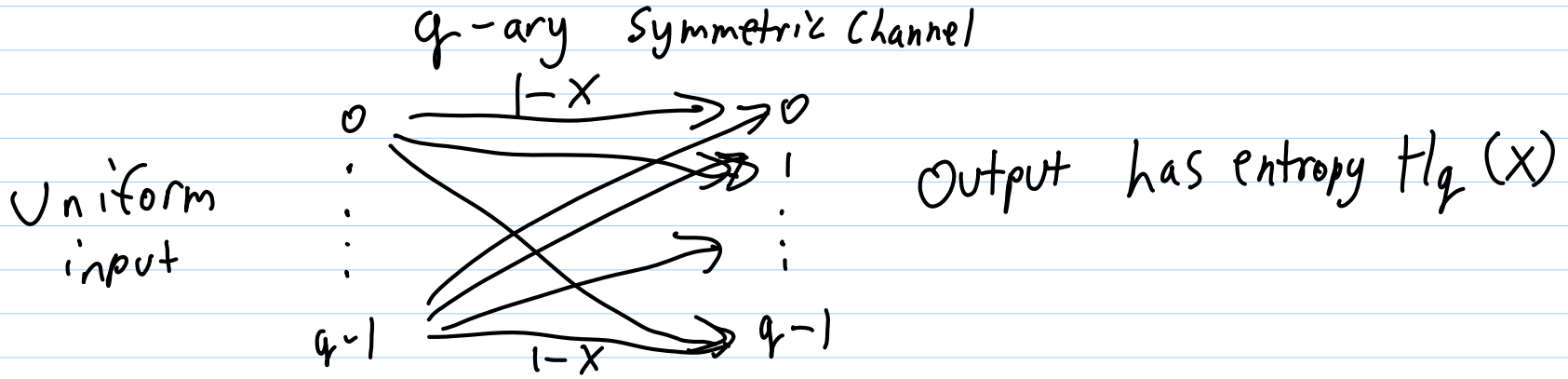
There are $V_q(\ell-1, d-2)$ such X for all $\ell \leq n$, and $V_q(\ell-1, d-2) < q^{n-k}$

So we can construct H.

Gilbert-Varshamov (Asymptotic):

$$H_q(x) \triangleq -x \log_q x - (1-x) \log_q (1-x) + x \log_q (q-1)$$

q-ary entropy function



Output same symbol wp. $1-x$

other symbol uniformly wp. x

$H_q(0) = 0$

$H_q(1) = \log_q(q-1)$

H_q is strictly concave, ≥ 0, maximized at $1 - 1/q$. So, $H_q^{-1} : [0, 1-1/q] \to [0,1]$ exists.

Let $n, nR \in \mathbb{N}$, $\delta \in (0, 1-1/q]$ be s.t.

$$R \leq 1 - H_q(\delta).$$

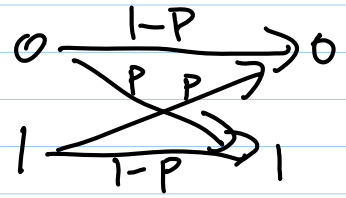
Then, there exists a $[n, nR, \geq \delta n]$ code over $\mathbb{F}_q$.

A little detour into why perfect codes are bad and distance isn't all important

Consider $\mathbb{F} = \{0,1\}$.

The Gilbert-Varshamov conjecture states $d \approx n H_2^{-1}(1-R)$ for the highest distance $[n, nR, d]$ code.

Consider the BSC channel



and bounded distance decoder: Receive y

- Return c if $\exists\ c \in C$ s.t. $d_H(c,y) \leq \left\lfloor \frac{d-1}{2} \right\rfloor$
- Else, error

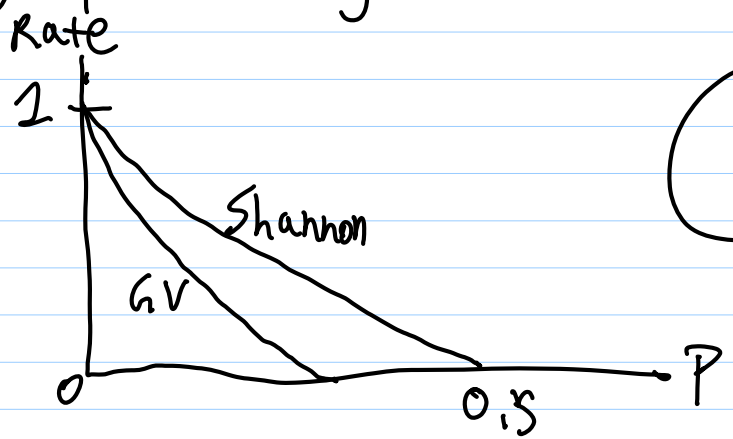
This can compensate for $p \approx \frac{1}{2}\frac{d}{n}$

So, $R \approx 1 - H_2\left(\frac{d}{n}\right)$

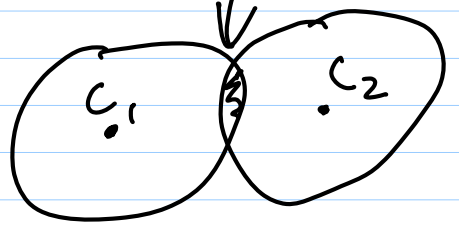
However, Shannon says you can have

$R < 1 - H_2(p)$

with vanishing probability of error



(low probability)
very small in high dimensions,



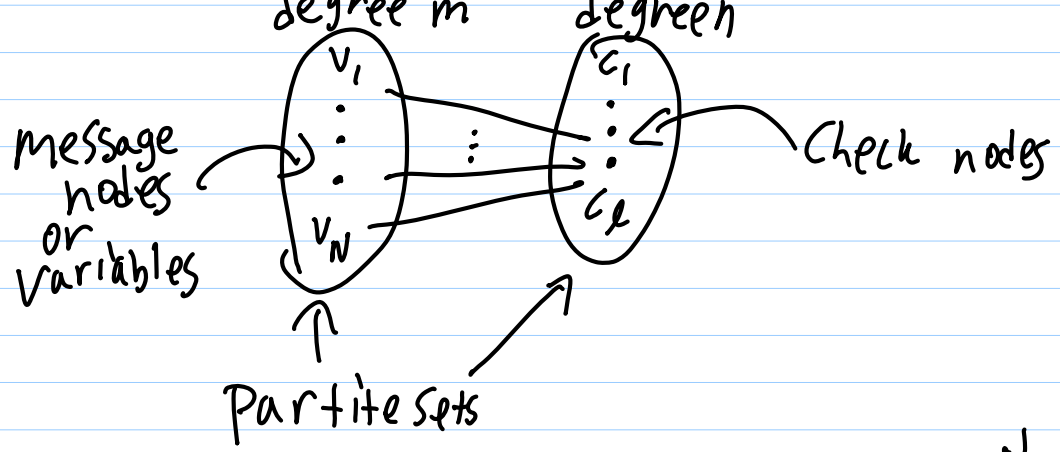
So perfect Codes aren't great

So, in order to achieve near the best possible, we have to decode beyond the worst case guarantee

Codes from Bipartite Graphs

Given $C = [n, k = rn, d = \theta n]$ over $\mathbb{F}$

$G = (m, n)$ – regular bipartite graph

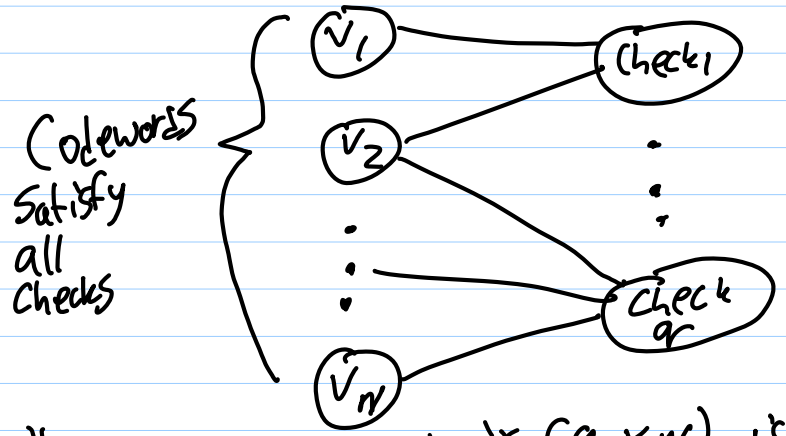


order the vertices adjacent to c_i as $v_{i,1}, \ldots, v_{i,n}$

Define the code $C_G = \{(v_1, \ldots, v_N) \in \mathbb{F}^N : (v_{i,1}, \ldots, v_{i,n}) \in C$ for $i = 1, \ldots, \ell\}$

Example: Low-Density Parity Check Codes (LDPC)

These were the original graph code introduced by Gallager in 1963. Let $\mathbb{F} = \mathbb{F}_2$. Gallager used random graphs here, but Sipser & Spielman replaced them with expanders to get some expander codes.



Check i is satisfied if $\sum$ adjacent vertices $= 0$

(C = parity code)

Parity check matrix ($q \times n$) is the bipartite adjacency matrix:

$$H_{ij} = \begin{cases} 1 & \text{if } v_j \text{ participates in check } i \\ 0 & \text{o.w.} \end{cases}$$

If H is very sparse, the code is called LDPC. If H has a fixed number of 1's in each row and in each column, the code is regular LDPC.

- Decodable by message passing / belief propagation

  Example (bit-flipping):

  Repeat

  1) Send message bits to adjacent check nodes

  2) Check nodes tell adjacent message nodes if parity check is satisfied or not

  3) message node flips its bit if majority of check nodes adjacent to it have parity /

  Until message bits are a valid codeword (all checks = 0)
  or
  max # of iterations (declare failure)

- Easily extended to different #'s of messages participating in each check or different #'s of checks for each message (Irregular LDPC)

Codes from Graphs

Let $G = (V, E)$ be a n-regular graph

$E(u)$ = ordered list of edges incident to $u \in V$

$|E| = N$

$|V| = \frac{2N}{n}$

$\gamma_G = \frac{\text{2nd largest adjacency matrix e.v.}}{n}$

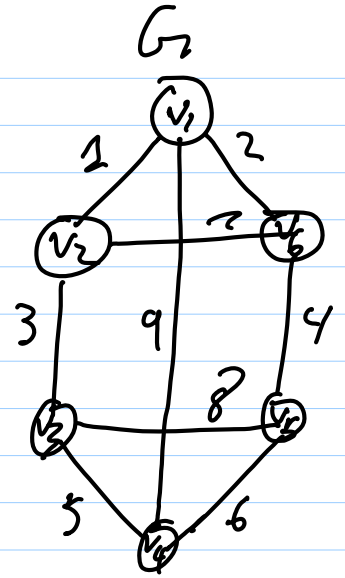
Let C be a $[n, k = rn, d = \theta n]$ code over $\mathbb{F}_q$.

(θ: minimum relative distance)

We form the graph code C_G by putting codewords on the edges of G, and edges adjacent to each vertex form a codeword.

Example:

C is a code with $n=3$



If a vector c in $\mathbb{F}^6$ has

$E(v_1)=(1,2,9)$

$E(v_2)=(1,3,7)$

$E(v_3)=(3,5,8)$

$E(v_4)=(5,6,9)$

$E(v_5)=(6,8,4)$

$E(v_6)=(4,7,2)$

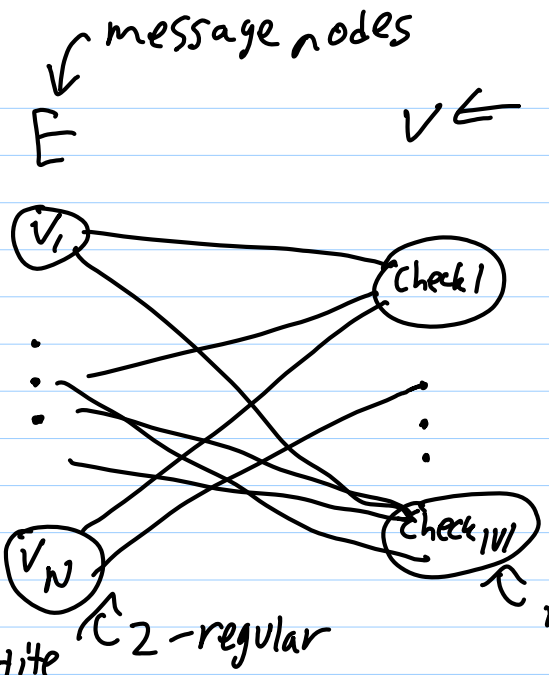
and $[c]_{E(v_i)} \in C$ $i=1,\dots,6$

then, $c \in C_G$.

Let $E(u)$ be an ordered list of the edges incident to u.

Then, $C_G = \{ c \in \mathbb{F}^N : [c]_{E(u)} \in C \ \forall \ u \in V \}$

As a bipartite graph:



Check i is satisfied if the appropriately ordered adjacent vertices are in C. Codewords satisfy all checks.

So, C_G is the bipartite graph code corresponding to the (2,n)-regular edge-vertex incidence graph:

$$(E : V, \{(e, v) : v \in V \text{ is endpoint of } e \in E\}).$$

We will use C_G to denote these types of codes from now on.

Properties of C_G:

C_G is a linear code with rate $R \geq 2r-1$:

Proof: C has a $(1-r)n \times n$ parity check matrix.

So, C_G can be written as $|V|(1-r)n$ linear constraints (since there are $|V|$ check nodes).

$$C_G = \{ c \in \mathbb{F}^n : \underbrace{H [c]_{E(v)}^T = 0}_{(1-r)n \text{ linear constraints}} \quad \forall v \in V \}$$

So C_G is linear $[N, k]$ and

$$N - K \leq |V| n (1-r)$$

$$\Downarrow$$

$$R = \frac{K}{N} \geq 2r-1$$

We will need the following fact:

If G is a n-regular ξ-expander and $S = \sigma|V|$,

$$\frac{2|E_{S,S}|}{|S|} \leq (\xi\sigma + 1 - \xi)n$$

Proof: $\sum_{u \in S} \deg_G(u) = n|S| = 2|E_{S,S}| + |\partial S|$

$$|\partial S| \geq \xi n |S| \left(1 - \frac{|S|}{|V|}\right)$$

Lecture 3, Defn of ξ-expander $\xi \in [0, 1-\gamma_G]$

$$n|S| + \xi n |S| \left(\frac{|S|}{|V|} - 1\right) \geq 2|E_{S,S}|$$

$$n(1 + \xi\sigma - \xi) \geq \frac{2|E_{S,S}|}{|S|}$$

(holds for $\xi \in [0, 1-\gamma_G]$)

If G is a ξ-expander, $C_G = [N, K = RN, D = \delta N]$

where $\delta \geq \dfrac{\theta(\theta + \xi - 1)}{\xi}$ Remember, $C = [n, rn, d = \theta n]$

Proof: Let $\xi > 1 - \theta$ and $\tilde{c} \in C_G$, $\tilde{c} \neq 0$.

$Y = \{ e \in E : \tilde{c}_e \neq 0 \}$

Let $S = \{ v \in V :$ An edge in Y has v as an endpoint $\}$

and look at the subgraph of G, (S, Y).

$$\frac{2 W_H(\tilde{c})}{|S|} = \frac{2|Y|}{|S|} \geq d$$ (since $\deg_{(S,Y)} \geq d$ because C has distance d)

Also, $$\frac{2|\psi|}{|S|} \leq \frac{2|E_{S,S}|}{|S|} \leq (q\sigma + 1 - q)n$$

So $$\sigma \geq \frac{\theta + q - 1}{q}$$

Plugging this into

$$w_H(C) \geq \frac{d|S|}{2} = \frac{d}{2}\sigma|U| \geq \frac{\theta n}{2}\,\frac{\theta + q - 1}{q}\,\frac{2N}{n}$$

and letting $w_H(C) = \delta N$

gives the result.

This also implies $$\delta \geq \frac{\theta(\theta - \gamma_G)}{1 - \gamma_G}$$

Good rate and distance!

Comparison: If C attains the Gilbert-Varshamov bound and G is a n-regular Ramanujan expander ($\gamma_G \leq \frac{2\sqrt{n-1}}{n}$)

$$R \geq 2r-1 \geq 1-2H_q(\theta)$$

$$\delta \geq \theta^2 - o(1)$$

So $$R \geq 1 - 2H_q(\sqrt{\delta}) + o(1)$$

If $q=2$, $= 0$ for $\delta \approx 0.012$

Well under what we could get by just using Gilbert-Varshamov!

However, these codes can be decoded in linear time when G is bipartite to correct $\approx \frac{1}{4}\delta N$ errors.

Decoding graph codes

Assume G is a n-regular bipartite graph with partite sets V', V'' and edgeset E

Let D be a decoder for C which can recover $\lfloor \frac{d-1}{2} \rfloor$ errors (in constant time)

A C_G decoder ($y \in \mathbb{F}^N$ is received vector):

$z \leftarrow y$

For $i = 1, \ldots, r$:

$U = \begin{cases} V', & i = \text{odd} \\ V'', & i = \text{even} \end{cases}$

For $v \in U$:

$[z]_{E(v)} \leftarrow D([z]_{E(v)})$

Return z if $z \in C_G$ else return Failure

We show this algorithm can correct up to $\frac{n\theta}{2}\sigma$ errors where $\sigma < \frac{\frac{\theta}{2} - \gamma_G}{1-\gamma_G}$ for appropriately chosen r

Proof:
Recall the expander mixing lemma (lecture 10):

If $S \subset V$, $T \subset V$, $|S| = \sigma|V|$, $|T| = \tau|V|$ and graph is n-regular

$$\left| |E_{S,T}| - n\sigma\tau|V| \right| \leq \gamma_G n|V| \sqrt{\sigma(1-\sigma)\tau(1-\tau)}$$

In this case, if $S \subseteq V'$, $T \subseteq V''$, note $V = V' \cup V''$ and $|V'| = |V''| = \frac{|V|}{2}$

so, if $S \subseteq V'$, $T \subset V''$, $|S| = \sigma|V|$, $|T| = \tau|V|$, $\sigma + \tau > 0$,

$$|E_{S,T}| \le \gamma_G n|V| \sqrt{\sigma(1-\sigma)\tau(1-\tau)} + n\sigma\tau|V|$$

$$= \sqrt{\sigma\tau}\left(\gamma_G \sqrt{(1-\sigma)(1-\tau)} + \sqrt{\sigma\tau}\right) n|V|$$

$\sqrt{\sigma\tau} + \sqrt{(1-\sigma)(1-\tau)} \le 1$ for $\sigma, \tau \in [0,1]$

$$\le \sqrt{\sigma\tau}\left(\gamma_G\left(1 - \sqrt{\sigma\tau}\right) + \sqrt{\sigma\tau}\right) n|V|$$

$$= \left((1-\gamma_G)\sigma\tau + \gamma_G\sqrt{\sigma\tau}\right) n|V|$$

If we re-define $|S| = \sigma|V'| = \frac{\sigma}{2}|V|$, $|T| = \tau|V''| = \frac{\tau}{2}|V|$,
we can loosen this to

$$|E_{S,T}| \le \left((1-\gamma_G)\sigma\tau + \gamma_G\sqrt{\sigma\tau}\right) n|V'| \quad (*)$$

Lemma: Suppose there exist $S \subset V'$, $T \subset V''$, $|S| = \sigma|V'|$, $|T| = \tau|V''|$ such that $v \in T \Rightarrow |N(v) \cap S| \geq \frac{\theta n}{2}$ for some $\theta > 0$.

Then, $\sqrt{\frac{\sigma}{\tau}} \geq \frac{\theta/2 - (1-\gamma_G)\sigma}{\gamma_G}$

Proof: $|E_{S,T}| = \sum_{v \in T} |N(v) \cap S| \geq \frac{\theta n}{2} |T| = \frac{\theta n}{2} \tau |V'|$

Combining with (*) and dividing through by $\gamma_G \tau n |V'|$ gives the result.

We now show the decoder works by showing the # of coordinates in U violating C's constraints decays exponentially quick in iteration #.

Main Result for decoding

Recall $U_i = \begin{cases} V', & i = \text{odd} \\ V'', & i = \text{even} \end{cases}$ $\quad C$ = transmitted codeword

$Y_i = \{ e \in E : Z_{i,e} \neq C_e \}$ $\quad i = 1, \ldots, r$ (coordinates which are wrong at end of iteration i)

$S_i = \{ v \in U_i : |E(v) \cap Y_i| > 0 \}$ (Vertices which have wrong coordinate incident to them at end of iteration i)

Let $\sigma_i = \frac{|S_i|}{|V'|}$, $\quad \theta > 2\gamma_G > 0$,

$$\beta = \frac{\theta/2 - \gamma_G}{1 - \gamma_G} \qquad \sigma \in (0, \beta)$$

and $d(y, c) \leq \frac{N\theta}{2}\sigma$. Then,

$$\boxed{|S_i| = \sigma_i |V'| \leq \left(\left(1 - \frac{\sigma}{\beta}\right)\left(\frac{\theta}{2\gamma_G}\right)^{i-1} + \frac{\sigma}{\beta} \right) \sigma |V|}$$

Proof: $Y_0 = \{e \in E : y_e \neq c_e\}$

D makes errors if it tries to correct $\geq \frac{d}{2}$ errors

So $v \in S_i \Rightarrow |E(v) \cap Y_{i-1}| \geq \frac{d}{2}$ $\quad i = 1, \ldots, V$ (**)

So for $i = 1$,

$$d(y, c) = |Y_0| \geq \sum_{v \in S_1} |E(v) \cap Y_0| \geq \frac{d}{2} |S_1|$$

So, $\sigma_1 |V'| = |S_1| \leq \frac{2}{d} d(y, c) \leq \frac{2}{\Theta n} \frac{N\theta}{2} \sigma = \sigma |V'|$

So, claim holds for $i = 1$.

$i > 1$: Look at the graph $(V_{i-1} : U_1, E)$

$S = S_{i-1}$ $\quad$ $T = S_i$ $\quad$ By defn of S_i and (**)

$$u \in S_i \Rightarrow |N(u) \cap S_{i-1}| \geq \frac{d}{2} \geq \frac{\theta n}{2}$$

So by our lemma,

$$\sqrt{\frac{\sigma_{i-1}}{\sigma_i}} \geq \underbrace{\frac{\theta}{2\gamma_a} - \frac{1-\gamma_a}{\gamma_a}\sigma_{i-1}}_{\geq 1 \text{ for } i=2 \text{ since } \sigma_1 \leq \sigma \leq \rho}$$

So by induction, $\sigma_i \leq \sigma$ for all i

So

$$\sqrt{\frac{\sigma_{i-1}}{\sigma_i}} \geq \frac{\theta}{2\gamma_a} - \frac{1-\gamma_a}{\gamma_a}\sqrt{\sigma_{i-1}\sigma}$$

$$\Downarrow$$

$$\frac{1}{\sqrt{\sigma_i}} \geq \frac{\theta}{2\gamma_a\sqrt{\sigma_{i-1}}} - \frac{(1-\gamma_a)\sqrt{\sigma}}{\gamma_a}$$

Setting equality overestimates σ_i:

$$\frac{1}{\sqrt{\sigma_i}} = \left(\frac{\theta}{2\gamma_G}\right)\frac{1}{\sqrt{\sigma_{i-1}}} - \frac{(1-\gamma_G)\sqrt{\sigma}}{\gamma_G}$$

which is a linear recurrence in $\frac{1}{\sqrt{\sigma_i}}$

Solving this gives

$$\frac{1}{\sqrt{\sigma_i}} = \left((1-\frac{\sigma}{\rho})\cdot\left(\frac{\theta}{2\gamma_G}\right)^{i-1} + \frac{\sigma}{\rho}\right)\frac{1}{\sqrt{\sigma}}$$

which gives the result.

So choosing $v = O(\log|V'|)$ gives a $O(|V'|\log|V'|)$ decoder for C_G. Doing careful re-analysis will show $O(|V'|)$.

- Sipser & Spielman showed if you allow for $\frac{1}{12}$ th of the errors, a similar decoder works when G isn't bipartite
- This analysis is due to Zémor

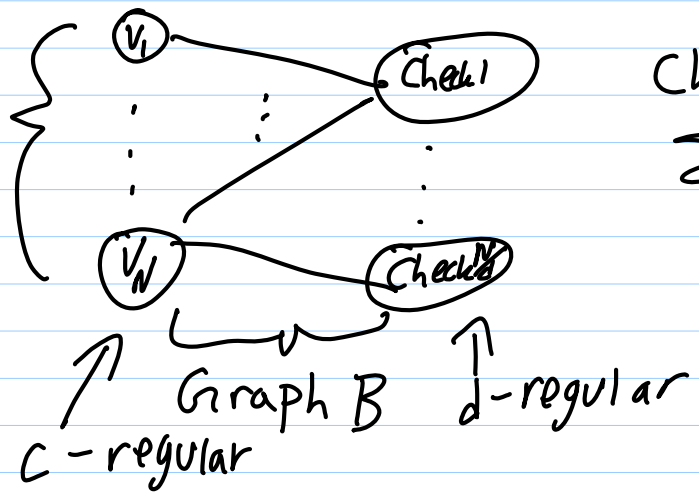
LDPC codes and other cool stuff

Topics:
- Bit-flipping decoder for expander codes
- Repeat-Accumulate codes
- Performance of LDPC codes

Bit-flipping decoder

Consider the LDPC code over $\mathbb{F}_2$

Codewords satisfy all checks



Check i is satisfied if $\sum$ adjacent variables $= 0$

B is a $(C, d, \varepsilon, \delta)$-expander if

1) B is bipartite with a c-regular partite set V' and d-regular partite set V''

2) If $S \subseteq V'$, $|S| \leq \varepsilon |V'|$

$$|\{y : \exists x \in S \text{ s.t. } (x,y) \in E\}| > \varepsilon |S|$$

3) Same as (2) except replace V' with V'' and ε with δ

It turns out the following algorithm:

Repeat

If a variable is in more unsatisfied checks than satisfied checks, flip it

Until ↗ is false for all variables

Will correct all sets of αN errors in the code has the necessary cond. all sets of αN variables have at least

$$\alpha N \left(1 + \frac{2\frac{c-1}{2d}}{3 + \frac{c-1}{2d}} \right) \text{ neighbors,}$$

The proof basically amounts to counting the # of satisfied and unsatisfied checks adjacent to a bit to be flipped.

Also, if B is a $(c, d, \alpha, 3c/4)$ expander, the algorithm can correct upto $\alpha N/2$ errors. However, a non-random construction of such expanders is not known.

Repeat - Accumulate Codes

In general, for a LDPC code, H is sparse.
The generator is not, though, so still need $O(kn) \approx O(n^3)$ operations to encode ($u \to uG$).

Repeat-Accumulate (RA) codes have linear time encoding.

Say you want to send $u = (u_1, \ldots, u_k)$.

1) Repeat u r-times.

$$\underbrace{u_1 \ldots u_1}_{r} \; \underbrace{u_2 \ldots u_2}_{r} \; \ldots \; \underbrace{u_k \ldots u_k}_{r} \qquad (*)$$

2) Choose your favorite (fixed) permutation of length rk and apply it to $(*)$

$t_1 \; t_2 \; \ldots \; t_N \qquad N = rk$

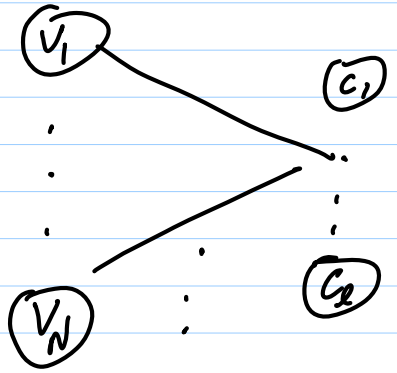
3) Transmit $v_1 \cdots v_N$ where

$$v_1 = t_1$$
$$v_2 = v_1 + t_2 \mod 2$$
$$v_3 = v_2 + t_3 \mod 2$$
$$\vdots$$
$$v_n = v_{n-1} + t_n \mod 2$$
$$\vdots$$
$$v_N = v_{N-1} + t_N \mod 2$$

Decode via message passing.

Performance of LDPC codes

Irregular LDPC codes:



Check i is satisfied if $\sum_k$ adj. $v_k = 0$

The partite sets are not regular.

Intuition: If a message node has high degree, it quickly gets the right value.

⇓

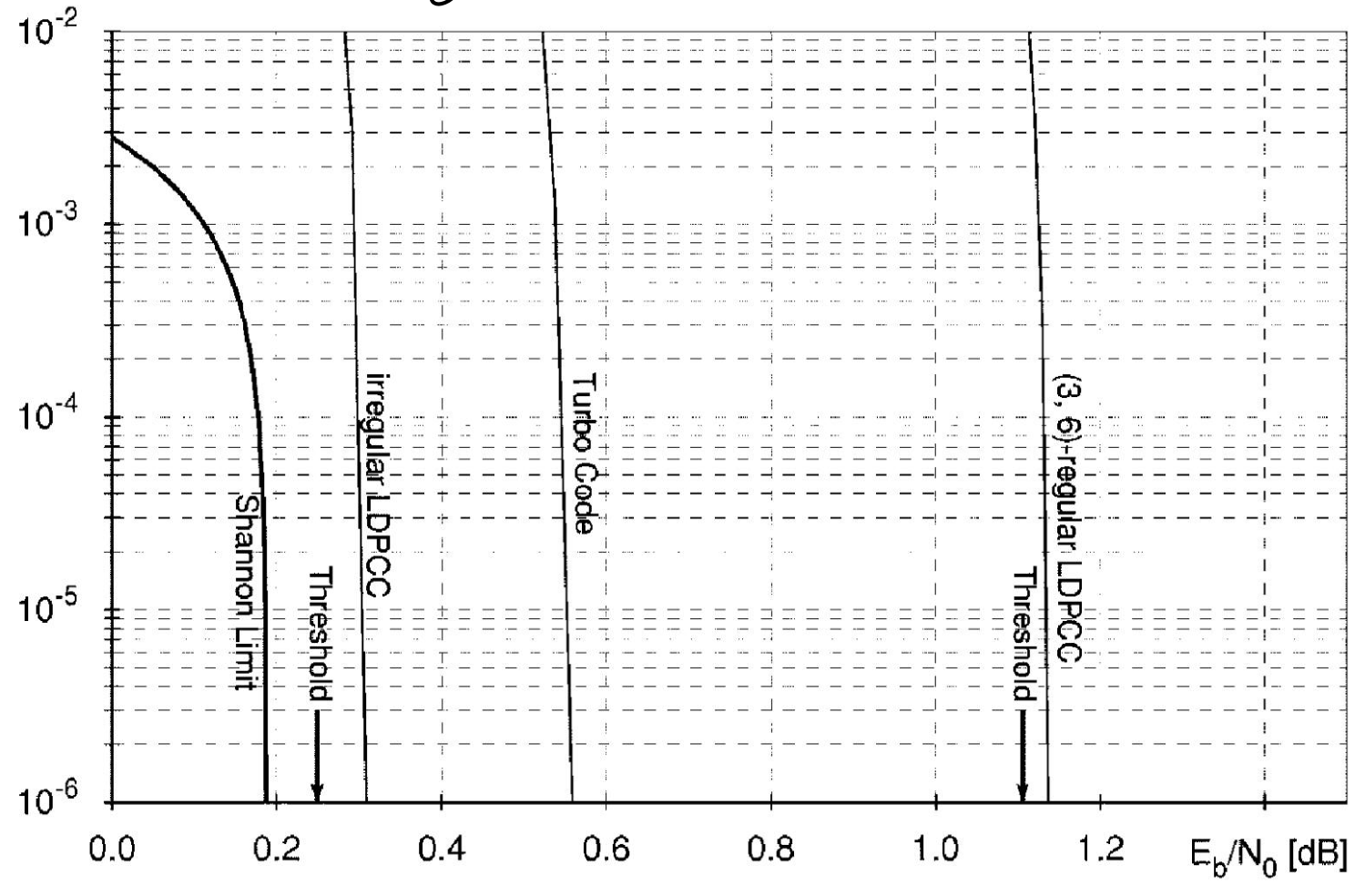
Good information to check nodes

⇓

Good information to correct lower degree nodes

These come very close to capacity!

Rate = 1/2, $N = 10^6$
BPSK + AWGN



From:
Design of Capacity-Approaching Low-Density Parity-Check Codes by Richardson et.al.
IEEE trans. IT, Vol 47, No. 2, Feb. 2001 pp. 619-637

References

Books

1) R.M. Roth, Introduction to Coding theory, CUP 2006

   Covers everything except sec. 5, LDPC

2) D.J.C. Mackay, Information Theory, Inference and Learning Algorithms, CUP 2003

   Covers LDPC, sec 5

3) Cover & Thomas, Elements of Information Theory, 2e (capacity)

Review articles

1) A. Shokrollahi, LDPC Codes: An Introduction

2) S. Johnson, Introducing Low-Density Parity Check Codes

Papers

1) M. Sipser, D. Spielman, Expander Codes, IEEE Trans. IT, Vol 42, No 6, Nov. 1996 pp. 1710 - 1722

   Sections 3, 4

2) G. Zemor, On Expander codes. IEEE Trans. IT, Vol 47, No. 2, Feb 2001 pp. 835-837

The use of $G=(V':V'', E)$ is due to this in Sec. 4.

3) M. Luby et. al. Analysis of Low Density Codes and improved designs Using irregular graphs, STOC 98, pp. 249-258

Intuition for irregular LDPC