A Crash Course on Coding Theory

Madhu Sudan
MIT

Topic: Linear time decoding

Algebraic codes give neat decoding algorithms, decoding lots of errors, in polynomial time. But suppose we want much faster algorithms?

Say linear time? Answers:

1. Yes, with a smaller fraction of errors.

2. Yes, provided errors are not adversarial.

Codes and decoding based on graph theoretic principles.

LDPC Codes

Defn: LDPC codes are codes with Low Density Parity Check matrices.

History

- Introduced: [Gallager'63]. Showed existence of codes with efficient decodability when error is prob.

- Rejuvenated: [Tanner'84]. Explicit constructions and a graph-theoretic study.

- Rediscovered: [Sipser+Spielman'95] Linear time decodability with adversarial error. (Also renamed Expander codes.)

Our presentation follows [SS'95].

Basic LDPC codes

Binary codes based on bipartite graphs: $G = (L, R, E)$.

- L = variable nodes. $|L| = n$.
  L assoc. with coordinates of codewords.
- R = constraint nodes. $|R| = m$.
  Each vertex of R imposes a linear constraint on its neighbors.
- Codeword $\mathcal{C}_G$= Boolean assignments to L such that for every vertex in R the parity of its neighbors is 0.

Prop: Code above is a linear code with information length $k \geq n - m$.

Note: G sparse $\equiv$ PC matrix is of low-density.

Distance

Depends on properties of G.

Prop 1: G is a random graph, then code is a random linear code.

Prop 2: Also holds for random sparse graphs.

Defn: G is (c, d)-bounded if every left vertex has degree $\leq c$ and every right vertex has degree $\leq d$. ((c, d)-regular if degrees equal.)

Defn: G is an (α, δ)-expander if for every set $S \subseteq L$, s.t. $|S| \leq \delta n$, the neighborhood of S, denoted $\Gamma(S)$, has cardinality $\geq \alpha |S|$.

Theorem: If G is (c, d)-bounded and an (α, δ)-expander, then $\mathcal{C}_G$ has distance rate at least $\frac{2\alpha\delta}{c}$, provided $2\alpha > c$.

Proof

Consider x with $\mathrm{wt}(x) < \frac{2\alpha\delta}{c}) \cdot n$.
Will show $x \notin \mathcal{C}_G$.

- Let $S = \{i | x_i = 1\}$.
- Let $\Gamma(S) = A \cup B$, where
  $A = \{j \in R | j$ has one neighbor in $S\}$.
  $B = \{j \in R | j$ has ≥ 2 neighbors in $S\}$.

$|A| > 0 \Rightarrow$ some constraint not satisfied.

Case: $\delta n \leq |S| < \frac{2\alpha\delta}{c} n$.)

From boundedness on S-side, we get:
(1) $|A| + 2|B| \leq c|S| < 2\alpha\delta n$.

From expansion, we get:
(2) $|A| + |B| \geq \alpha\delta n$.

Putting above together get $|A| > 0$.

Proof (contd).

Case: $|S| < \delta n$.

From boundedness on S-side, get:
(1) $|A| + 2|B| \leq c|S|$.

From expansion, get:
(2) $|A| + |B| \geq \alpha |S|$.

Putting above together, get
$|A| \geq (2\alpha - c)|S| > 0$.

Decoding

Given: Assignment $\vec{a}$ to variables.
Task: Find nearby codeword sat. all constraints.

The algorithm:

- While $\exists$ variable i with more satisfied ngbrs than unsat. ones, flip a_i.
- If none exists, output $\vec{a}$.

Prop: Algorithm can be implemented in linear time, provided $c, d = O(1)$. (Always reduces # unsat. constraints!)

Thm: Corrects up to $((\frac{2\alpha - c}{c})\delta)$-fraction errors, provided $\alpha / c > 3/4$.

(If $\alpha = (1 - \beta)c$, then distance $= (2 - 2\beta)\delta$ and fraction of errors $= (1 - 2\beta)\delta$.)

Proof Steps

Let $\vec{a}$ have ϵn ones for $\epsilon < \frac{2\alpha - c}{c}\delta$. Will show alg. terminates with all zero vector. At any stage of algorithm:

- Let $S \subseteq L$ be vars. set to 1, $s = |S|$.
- Let $U \subseteq R$ be unsat. constraints, $u = |U|$.

Key Lemma: $0 < s \leq \delta n \;\; \Rightarrow \;\; u > (2\alpha - c)s$.

Corollary 1: $s < \delta n$
Proof: Initially, $u \leq c\epsilon n$. Further algorithm always reduces u. So $s \leq \frac{c}{2\alpha - c}\epsilon n < \delta n$.

Corollary 2: $s > 0$ implies $\exists j \in S$ with more than $c/2$ neighbors in U.
Proof: Averaging + $\alpha > 3c/2$.

Together yield the theorem.

Proof of Key Lemma

- Let $\Gamma(S) = A \cup B$, where
  $A = \{j \in R | j$ has one neighbor in $S\}$.
  $B = \{j \in R | j$ has ≥ 2 neighbors in $S\}$.

Recall:
(1) $|A| + 2|B| \leq c|S|$.
(2) $|A| + |B| \geq \alpha|S|$.

Together yield:
$|A| \geq (2\alpha - c)|S|$

Lemma follows since $A \subseteq U$.

Expanders

Need: (c, d)-bounded (α, δ)-expander graphs, with $\frac{\alpha}{c} > \frac{1}{2}$ for distance ($\frac{\alpha}{c} > \frac{3}{4}$ for decoding).

Prop: Random graphs satisfy such properties for positive δ.

Unfortunately:

- No explicit constructions known.
- No tests known.

Explicit constructions give:

Thm: For every α, there exists $c, d < \infty$ and $\delta > 0$, s.t. (c, d)-bounded, (α, δ)-expanding graphs can be constructed in polynomial time.

How to use these?

Extended LDPC codes

Reexamine: Key property used in analysis:
Every constraint vertex needs to have
≥ 2 neighbors set to 1 to be satisfied.
Hence, the requirement $2\alpha > c$.

Suppose: Every constraint vertex needs
$\geq \Delta$ neighbours set to 1 to be satisfied.
Requirement weakens to $\Delta\alpha > c$.

How to set up such constraint?
Error-correcting codes!

New interpretation of constraint vertex:
Assignment to neighbors must be from B,
for some $[d, ?, \Delta]$ error-correcting code B.
(Enumerate neighbors in canonical order.)

Extended LDPC codes

Defn: For (c,d)-regular graph G, and $[d,l,\Delta]$ code B, the Extended LDPC code $\mathcal{C}_{G,B}$ has as codewords all assignments to the variable vertices such that the ngbrs of every constraint vertex form codewords of B.

Specializes/Generalizes LDPC.

Prop: Information length of $\mathcal{C}_{G,B}$ is $n-m(d-l)$.

Thm: If G is an (α,δ)-expander, then the code has distance rate at least $\frac{\Delta\alpha}{c}\delta$, provided $\Delta\alpha > c$.

Proof of Distance

As earlier consider x with $\mathrm{wt}(x) < \frac{\Delta\alpha\delta}{c}) \cdot n$. Will show $x \notin \mathcal{C}_G$.

- Let $S = \{i | x_i = 1\}$.
- Let $\Gamma(S) = A \cup B$, where
  $A = \{j \in R | j$ has $< \Delta$ neighbors in $S\}$.
  $B = \{j \in R | j$ has $\geq \Delta$ neighbors in $S\}$.

$|A| > 0 \Rightarrow$ some constraint not satisfied.

Case: $\delta n \leq |S| < \frac{\Delta\alpha\delta}{c} n.$)

From boundedness on S-side, we get:
(1) $|A| + \Delta|B| \leq c|S| < \Delta\alpha\delta n$.

From expansion, we get:
(2) $|A| + |B| \geq \alpha\delta n$.

Putting above together get $|A| > 0$.

Proof (contd).

Case: $|S| < \delta n$.

From boundedness on S-side, get:
(1) $|A| + \Delta|B| \leq c|S|$.

From expansion, get:
(2) $|A| + |B| \geq \alpha|S|$.

Putting above together, get
$|A| \geq \frac{\Delta\alpha - c}{\Delta - 1}|S| > 0$.

Decoding

Not the same algorithm!

Parallel decoding algorithm:

- Parameter ϵ.
- Repeat
  - If check vertex has less than $\epsilon\Delta$ distance from codeword
    Send flip message to $\epsilon\Delta$ ngbrs.
  - Flip all bits that rec'd flip message.
- Until no flip messages sent.

Analysis omitted.

Encoding?

The LDPC codes are extremely fast to decode, but how easy are they to encode?

Definitely, polynomial time encodable.

But not necessarily linear time!

Need new idea.

Spielman codes

[Spielman'95]

Comes in two steps.

Phase I: Error-reducing codes.

Phase II: Linear-time encodable and decodable codes.

Error-reduction codes

Defn: For bipartite graph $G = (L, R, E)$, the Reducer Code, R_G, is defined as follows:

- L = message bits, $|L| = k$.
- R = check bits, $|R| = n - k$.
- Codewords = n-bit assignments to $L \cup R$ s.t. the assignment to every check bit equals the parity of its neighbors.

Prop 1: If G is $(c, ?)$-bounded, then encoding is linear time.

Prop 2: If G is $(c, ?)$-bounded, then distance is at most $c + 1$.

But in fact, if we fix check bits, then get good code on message side! So will hope check bits are mostly right, and hope to fix message bits.

Error-reduction

Defn: A is an (ϵ, γ)-error-reduction alg. if

$$\forall \quad s, t, \ (m, c) \in R_G$$
$$(x, y) \in \{0, 1\}^n$$
$$\text{s.t. } \Delta(m, x) = s \leq \gamma n$$
$$\text{and } \Delta(c, y) = t \leq \gamma n,$$
$$x' = A(x, y) \text{ satisfies } \Delta(m, x') \leq \epsilon t.$$

If $t = 0$, then must correct all errors!

Error-reduction (contd.)

Algorithm

- Set $x' = x$.
- While $\exists$ message vertex i with more satisfied ngbrs than unsat. ones, flip x'_i.
- If none exists, output x'.

Prop: Algorithm can be implemented in linear time, provided $c, d = O(1)$.

Thm: If G is a $(c, 2c)$-regular and a $(\frac{7}{8}c, \delta)$-expander for some $\delta > 0$, then alg. above is an (ϵ, γ)-error-reduction alg. for $\epsilon = \frac{4}{c}$ and $\gamma = \frac{c\delta}{2(c+2)}$.

Analysis

Fix x', y, m, c and let:

- $S' = \{i | x'_i \neq m_i\}$ and $s' = |S'|$
- $T = \{j | y_j \neq c_j\}$ and $t = |T|$
- $U = \{j | j\text{th chkbit unsat.}\}$ and $u = |U|$.
- $A = \{j \in \Gamma(S) | j \text{ has one ngbr in } S\}$.
- $B = \{j \in \Gamma(S) | j \text{ has } \geq 2 \text{ ngbr in } S\}$.

Prop: $A - T \subseteq U \subseteq A \cup T$.

Key Lemma:

$$0 < s' \leq \delta k \Rightarrow u > (2\alpha - c)s' - t.$$

(Proved as in earlier cases.)

Analysis (contd.)

Corollary 1: $s' \leq \delta k$.

Proof:

- Initially, $u \leq cs + t \leq \frac{2}{3}(c+1)\gamma k$.
- Algorithm always reduces u.
- So $s' \leq \frac{cs+2t}{2\alpha - c} \leq \delta k$.

Corollary 2: $s > \frac{4t}{c}$ implies $\exists j \in S$ with more than $c/2$ neighbors in U.
Proof: Averaging + $\alpha = 7c/8$.

Together yield the theorem.

Phase II

Given: Sequence of error-reduction codes
$R_2, R_4, R_8, \ldots, R_{k=2^i}, \ldots$.
R_k has k message bits + $k/2$ checkbits.

Will construct: Seq. of Error-Correcting codes:
$C_2, C_4, C_8, \ldots, C_{k=2^i}, \ldots$.
C_k has k message bits + $3k$ checkbits.

Given: k-bit message m,
Checkbits of $C_k = c_1 \circ c_2 \circ c_3$, where
c_1 = checkbits of $R_k(m)$.
c_2 = checkbits of $C_{k/2}(c_1)$.
c_3 = checkbits of $R_{2k}(c_1 \circ c_2)$.

Verify: c_1 has $k/2$-bits, $c_1 \circ c_2$ has $2k$-bits.
$c_1 \circ c_2 \circ c_3$ has $3k$ bits.

Encoding & Decoding

Prop: If $R_2, R_4, \ldots$ is linear time encodable, then so is $C_2, C_4, \ldots$

Decoding Algorithm

Given: $x \circ y_1 \circ y_2 \circ y_3$

Step 1: Error-reduce R_{2k} on $y_1 \circ y_2, y_3$ and get $y_1' \circ y_2'$.

Step 2: Error-correct $C_{k/2}$ on $y_1' \circ y_2'$ and get y_1''.

Step 3: Error-reduce R_k on x, y_1'' and get x'.

Step 4: Return x'.

Decoding (Analysis)

Prop: If the error-reduction algorithm for $R_2, R_4, \ldots$ runs in linear time, then the error-correction alg. also runs in linear time.

Theorem: If, for $\gamma > 0$, the codes $R_2, R_4, \ldots$ have an $(\frac{1}{2}, \gamma)$-error-reduction algorithm, Then the decoding algorithm above corrects $\gamma/4$-fraction errors.

Proof

Proof: Suppose $\Delta(x \circ y_1 \circ y_2 \circ y_3, m \circ c_1 \circ c_2 \circ c_3) \leq \gamma/4n = \gamma k$.

- Then the following hold:
  $\Delta(x, m) \leq \gamma k$
  $\Delta(y_1 \circ y_2, c_1 \circ c_2) \leq \gamma k$
  $\Delta(y_3, c_3) \leq \gamma k$
- Can decode $R_{k/2}$. Yields $\Delta(y_1' \circ y_2', c_1 \circ c_2) \leq (\gamma/2)k$.
- Error in $C_{k/2}$ small. Can correct it. Thus $y_1'' = c_1$.
- All checkbits of R_k correct! Thus $x' = m$!

Summarizing

Theorem: There exists a family of linear-time encodable and decodable error-correcting codes.

Theorem: Such a family can be constructed in poly time.