Local Testability and Decodability of Sparse Linear Codes

Madhu Sudan
MIT

Joint work with Tali Kaufman (IAS & MIT).
Local (Sublinear-time) Algorithmics

- Data getting ever-larger
  - Need algorithms that can infer “global” properties from “local” observations ...

- Led to
  - Property testing, Sublinear-time algorithms

- Common themes:
  - Oracle-access to input, implicit output.
  - Answers of the form: “input close to having property”
Error-Correcting Codes

- **Code:** \( C \subseteq \{0, 1\}^n \) image of \( E : \{0, 1\}^k \rightarrow \{0, 1\}^n \)

- **Distance ...**
  - ... between sequences: \( \delta(x, y) = \Pr_i [x_i \neq y_i] \)
  - ... of code: \( \delta(C) = \min_{x \neq y \in C} \{ \delta(x, y) \} \)

- **Algorithmic Problems:**
  - **Encode:** Compute \( E \)
  - **Detect Errors:** Given \( r \in \{0, 1\}^n \), is \( r \in C \)?
    - Or \( \exists x \in C \) s.t. \( \delta(r, x) \leq \epsilon \)?
  - **Decode:** Given \( r \in \{0, 1\}^n \) s.t. \( \exists x \in C \)
    - with \( \delta(r, x) \leq \epsilon \), compute \( x \).
Local Algorithmics in Coding

- **Encoding**: Can not be performed “locally”
  - Single bit change in input should alter constant fraction of output!

- Testing, Decoding, Error-correcting ... can be performed locally. Furthermore
  - They are very natural problems.
  - Have many applications in theory (PCP, PIR, Hardness amplification).
  - Lots of interesting effects are achievable.
Local Algorithmic Problems

- **Common framework:** Fixed code $C \in \{0, 1\}^n$; Oracle access to $r \in \{0, 1\}^n$; Only $k$ queries allowed.

- **Local Testing:** accept if $r \in C$
  
  reject (with $\Omega(1)$ prob.) if $\delta(r, C) \geq \epsilon$.

- **Local Self-Correction:**
  
  Promise: $\exists c \in C$ s.t. $\delta(c, r) \leq \epsilon$.
  
  Given $i \in [n]$, compute $c_i$

- **Local Decoding:**
  
  Setup: Fix $E : \{0, 1\}^k \rightarrow \{0, 1\}^n$ s.t. $C = \text{Image}(E)$.

  Promise: $\exists m$ s.t. $\delta(E(m), r) \leq \epsilon$.

  Given $i \in [k]$, compute $m_i$
Example: Hadamard Codes

- **Encoding:** Given \( m \in \{0, 1\}^{\log n} \), and \( x \in \{0, 1\}^{\log n} \)
  \[
  E(m)_x = \sum_{i=1}^{\log n} m_i x_i \pmod{2}
  \]

- **Test:** Accept iff \( r_x + r_y = r_{x+y} \)

- **Correction:** Given \( x \in \{0, 1\}^{\log n} \), pick \( y \in \{0, 1\}^{\log n} \)
  uniformly and output \( r_{x+y} - r_y \)

- **Decoding:**
  1. 1st bit of message is \( e\text{1st coordinate of its encoding.} \)
Brief History

- **Local Decoding/Self-Correcting:**
  - [Beaver-Feigenbaum], [Lipton], [Blum-Luby-Rubinfeld] – instances of Local Decodability.
  - ...

- **Locally Testable Codes:**
  - [Arora], [Rubinfeld-Sudan], [Spielman], [Goldreich-Sudan] – definitions.
  - ...

Constructions of Locally X-able Codes

- Basic codes: Algebraic in nature.
  - Analysis:
    - Decoding: typically simple, uses algebra.
    - Testing: more complex.

- Better codes: Careful compositions of basic codes.
  - Exception: [Meir ‘08] – not algebraic.

- Questions:
  - Do we need all this algebra/careful constructions?
  - Can we derive local algorithms from “classical” parameters?
  - Can randomly chosen codes have local algorithms?
Our Results

- Theorem (Informal): Every “sparse”, “linear” code of “large distance” is locally testable, correctible.

- Linear? $C$ linear if $x, y \in C \Rightarrow x + y \in C$
- Sparse? $C$ is $t$-sparse if $|C| \leq n^t$
- Large Distance? $C$ has $\gamma$-large-distance if $\delta(C) \geq \frac{1}{2} - n^{-\gamma}$

Theorem 1: $\forall \gamma > 0$, $t < \infty$, $\exists k < \infty$ such that if $C$ is $t$-sparse, linear and has $\gamma$-large-distance then $C$ is $k$-locally testable.
Our Results (contd.)

- **Linear?** $C$ linear if $x, y \in C \Rightarrow x + y \in C$
- **Sparse?** $C$ is $t$-sparse if $|C| \leq n^t$
- **Large Distance?**
  $C$ has $\gamma$-large-distance if $\delta(C) \geq \frac{1}{2} - n^{-\gamma}$
- **Balanced?**
  $C$ is $\gamma$-balanced if $\forall x \neq y \in C,$
  \[ \frac{1}{2} - n^{-\gamma} \leq \delta(x, y) \leq \frac{1}{2} + n^{-\gamma}. \]

**Theorem 2:** $\forall \gamma > 0, t < \infty, \exists k < \infty$ such that if $C$ is $t$-sparse, linear and is $\gamma$-balanced then $C$ is $k$-locally correctible.
Corollaries

- Reproduce old results: Hadamard, dual-BCH
- New codes:
  - Random sparse linear codes (decodable under any linear encoding).
  - dual-BCH variants
    \[
    \left\{ \text{Trace}(c_1 x^{i_1} + \cdots + c_t x^{i_t}) | c_1, \ldots, c_t \in \mathbb{F}_{2^{\log n}} \right\},
    \]
    \[i_1, \ldots, i_t < \sqrt{n}\]
- Nice closure properties: (Subcodes, Addition of new coordinates, removal of few coordinates)
Previously ...

- [Kaufman-Litsyn] Similar result + techniques. Main differences:
  - Required $\gamma \geq \frac{1}{2}$. So $\delta(C) \geq \frac{1}{2} - \frac{1}{\sqrt{n}}$
  - Worked only for balanced codes.
  - Only proved local testability ... no correctibility
Proof Techniques

- Modifying (simplifying? extending?) the proofs of [Kaufman Litsyn ’05] (some ideas go back to [Kiwi 95]).

- Buzzwords: Duality, MacWilliams Identities, Krawtchouk Polynomials, Johnson bounds.
Linearity, Duality, & Testing

- **Dual of a Code:**
  \[ C^\perp = \{ y \in \{0, 1\}^n | \langle x, y \rangle \triangleq \bigoplus_{i=1}^n x_i y_i = 0, \forall x \in C \} \]

- **Canonical (only) test for membership in C:**
  Pick low-weight \( y \in C^\perp \)
  Test \( \langle r, y \rangle = \bigoplus_{i \in 1_y} r_i = 0 \)
  \( \text{wt}(y) = k \Rightarrow \text{Test is } k\text{-local} \)

- **Canonical self-corrector:**
  To compute \( c_i \), pick low-weight \( y \) s.t. \( y_i = 1 \)
  output \( \bigoplus_{j \in 1_y - \{i\}} r_j \)
Questions:

- Does $C^\perp$ even have any low-weight codewords?

- Is the distribution of non-zero coords. of low-weight $y$ s.t. $y_i = 1$ roughly uniform?

- How to even analyze the test?
Path to answers

- Need “weight distribution” of some codes:
  Weight distribution: $C_0, \ldots, C_n$, where
  \[ C_i = \# \text{ codewords in } C \text{ of weight } i. \]

- Testing + Correcting: Weight distribution of $C^\perp$
  Specifically $C_k^\perp$

- Testing: [Kiwi, KL]
  Also need weight distribution of $(C \cup (C + r))^\perp$.
  Specifically, $(C \cup (C + r))_{k}^\perp$

- Correcting: [New]
  Wt. distribution of $C^{-i}, C^{-\{i,j\}}$
  $(C^{-i} : C \text{ with } i\text{th coordinate deleted.})$
Dual Weight Distribution?

- MacWilliams Identities: Can compute weight distribution of dual from weight distribution of primal ... exactly!

- Don’t have primal distribution exactly ... Can coarse information suffice?
  - [Kiwi] - Manages to compute primal info. exactly.
  - [Kaufman-Litsyn] – Find out a lot about primal distribution.
MacWilliams Identities: Precise Form

- Krawtchouk Polynomials

\[ P_k(i) = \sum_{j=0}^{k} (-1)^j \binom{i}{j} \binom{n-i}{k-j} \]

- Dual Weight Distribution

\[ C_{k}^\perp = \frac{1}{|C|} \cdot \sum_{i=0}^{n} P_k(i) C_i \]

- Double summation! Many negative terms. Cancellations?
Primal Weight Distribution (Balanced)

\[ C_i \leq n \pm n^{1-\gamma} \]

\[ \sum_i C_i \leq n^t \]
Krawtchouk Polynomial (k odd)

\[ i \in \frac{n}{2} \pm n^{1-\gamma} \]

\[ C_i \]

Zeros \[ \frac{n}{2} \pm \sqrt{kn} \]
Krawtchouk Polynomial (k odd)

\[ i \in \frac{n}{2} \pm n^{1-\gamma} \]

\[ C_i \]
Low-weight codewords in dual

- Can conclude: constant weight codewords exist.
  \[ C_k^\perp \approx \frac{1}{|C|} \cdot \binom{n}{k} \cdot (1 \pm n^{t-k}) \]

- Very tight bound (If \( k \gg t/\gamma \))

- Leads to self-corrector
Analysis of self-corrector

- Need to understand
  
  \[ C_{k,i}^\perp = |\{y \in C^\perp | \text{wt}(y) = k \text{ and } y_i = 1\}| \]

- New Code: \( C^{-i} = C \) with \( i \)th coordinate deleted.
  
  \[ = \{\pi(y) | y \in C\}. \]

- Claim: \( (C^{-i})^\perp = \{\pi(y) | y \in C^\perp \text{ s.t. } y_i = 0\} \)
  
  and so \( C_{k,i}^\perp = C_k^\perp - (C^{-i})_{k}^\perp \)

- But \( C^{-i} \) is sparse and balanced
  
  and so can determine \( (C^{-i})_{k}^\perp \)
Analysis of self-corrector (contd.)

- Plugging in bounds:
  \[ \Pr_{y \in C_k^\perp} [y_i = 1] \approx k/n (1 \pm n^{-c}) \]

- Similar calculations with \( C^{-i,j} \) yield:
  Events \( y_i = 1 \) and \( y_j = 1 \) roughly independent if \( y \leftarrow C_k^\perp \).

- Conclude: Self-corrector computes \( C_i \) correctly w.p. \( \geq 1 - O(\epsilon \cdot t/\gamma) \) from \( \epsilon \)-corrupted received word.
Analysis of Tester (balanced case)

- Need to analyze $\text{span}(C, r)_{k}^\perp$
  where $\text{span}(C, r) = C \cup (C + r)$

- Specifically, want: $\text{Pr}_{y \in C_{k}^\perp} [y \notin \text{span}(C, r)_{k}^\perp] = \Omega(\epsilon)$.
  $\iff \text{span}(C, r)_{k}^\perp \leq (1 - \Omega(\epsilon)) \cdot C_{k}^\perp$

- Easy fact (from MacWilliams Identities)
  $\text{span}(C, r)_{k}^\perp = \frac{1}{2} \cdot C_{k}^\perp + \frac{1}{2} \cdot \frac{1}{|C|} \cdot \sum_{i=0}^{n} P_{k}(i) \cdot (C + r)_{i}$

- Suffices to analyze second term. But what does the weight distribution of $C + r$ look like? and how does $P_{k}(\cdot)$ interact with this?
Weight Distribution of $C+r$ (vs. $C$)

$i \in \frac{n}{2} \pm n^{1-\gamma}$
Weight Distribution of $C+r$ (vs. $C$)

\[ C_0 = 1 \]

\[ C_i \]

\[ i \in \frac{n}{2} \pm n^{1-\gamma} \]

\[ i \in \frac{n}{2} \pm (\epsilon n + n^{1-\gamma}) \]

[Equation]

\[ \sum_i C_i \leq n^t \]
Inner Product with Krawtchouk’s

\[ C_0 = 1 \]

\[ C_i \]

\[ i \in \frac{n}{2} \pm n^{1-\gamma} \]

\[ i \in \frac{n}{2} \pm (\epsilon n + n^{1-\gamma}) \]

\[ \sum_i C_i \leq n^t \]

Don’t make a difference

Non-positive
Inner Product with Krawtchouk’s

\[ C_i = 1 \]

Helps! But by how much?

Hurt! By how much?
More Bounds

- Some weak Krawtchouk bounds:
  1. $P_k(\epsilon n) \leq (1 - \epsilon)P_k(0)$ (the “helpful” part)
  2. $P_k(i) \leq (n - 2i)^k/k!$ (For $i$ in our range. Useful to limit the “hurt”)

- Bound 2. not sufficient to bound the “hurt” ... but can combine with “Johnson bound”

- Johnson Bound:
  Code of relative distance $1/2 - \tau$ can not have too many codewords in ball of radius $1/2 - \sqrt{\tau}$
Putting all the bounds together

- Can conclude:

\[
\frac{1}{|\mathcal{C}|} \cdot \sum_{i=0}^{n} P_k(i)(C + r)_i \leq (1 - \Omega(\epsilon)) \cdot C_k^\perp
\]

- Implies test rejects $\epsilon$-corrupted codeword with probability $\Omega(\epsilon)$. 
Unbalanced codes?

- Many things break down ...

- E.g., If $\overline{1} \in C$ then $C_{k}^{\perp} = 0$ for odd $k$.

- Our approach:
  - **Step 1**: Codes of max. wt. $\leq 5/8n$ (weakly balanced).
  - **Step 2**: Reduce general case to weakly balanced case.
Weakly balanced codes

- Can now prove $C_{k}^\perp > 0$ for odd $k$.

- But can’t get a precise bound on $C_{k}^\perp$.

- Instead, we bound $C_{k}^\perp - (\text{span}(C, r))^\perp_k$ directly;
  - Show that contribution of any word to both terms is roughly the same (Uses some properties of $P_{k-1}(\cdot)$.)
  - Show that contribution of the coset leader drops by $\Omega(\epsilon)$-factor.
Reducing general codes to w.b. codes

- Write $C = \tilde{C} + \text{span}(x, y, z)$ where $\tilde{C}$ is weakly-balanced.

- Test if $\exists u \in \text{span}(x, y, z)$ such that $r + u \in \tilde{C}$.

- Yields tester for all binary, linear, sparse, high-distance codes.
Conclusions/Questions

- Simpler proof for random codes? (Some work by Shachar Lovett, Or Meir)

- Self-correct imbalanced codes?

- Are random sparse codes locally list-decodable?

- Is this just a logarithmic saving in locality?

- Are there other ways to pick broad classes of testable codes (at “random”)?