Tutorial: Locally decodable codes

Anna Gál

UT Austin
Locally decodable codes

Error correcting codes with extra property:
Recover (any) one message bit, by reading only a small number of codeword bits.

Motivation: large data sets – sublinear decoding time

Applications: Group Testing ? :-)  
Private Information Retrieval  
Probabilistically Checkable Proofs  
Data Structures  
Polynomial Identity Testing  
Matrix Rigidity and circuit complexity
Traditional error correcting codes:

\[ C : \{0, 1\}^k \rightarrow \{0, 1\}^n \]

\( x \in \{0, 1\}^k \) can be uniquely reconstructed from any \( y \in \{0, 1\}^n \) within Hamming distance \( \delta n \) from \( C(x) \).

Decoding involves reading all of \( y \).

What if we only need to recover a small part of \( x \)?
Locally decodable codes: no need to read all encoded data to recover individual parts

Examples:
Phone book only want one phone number
Hospital records only want record of one patient

Don’t want to read encoding of the whole phonebook or all records of the entire hospital
First attempt Encode each phone number (each patient record) separately.

This would work if the errors follow a nice pattern: no more than $\delta$ fraction of every record

BUT:
$\delta$ fraction of all of the encoded data can be corrupted

Some records can get completely destroyed this way: $\delta = 1/10$, 10000 patients, 1000 can be lost.
Examples:

Phone book only want one phone number
Hospital records only want record of one patient

Don’t want to read encoding of the whole phonebook or all records of the entire hospital

But want to be able to get record of ANY one of the patients (ANY phone number)
Ideally: amount of encoded data to read
“proportional” to how many records I want to get.

Or, at least:
to get ONE record, read only a sublinear part

BUT:
δ fraction of all of the encoded data can be corrupted

Can we do this?
Decoder must be probabilistic.

Sublinear part of the data - much less than $\delta n$ bits.

If the decoder is deterministic, possibly all the bits it reads are corrupted.
Examples of some trivial probabilistic decoders.

We try to recover the bit $x_1$.

1. Flip a coin, return the result (don’t read anything)
Correct with probability $1/2$

2. Suppose the code has many copies of $x_1$ in the codeword (repetition code for bit $x_1$) Randomly pick a position.
Correct with probability $1 - \delta$. 
$(q, \delta, \epsilon)$-LDC

decoder reads at most $q$ bits
tolerates up to $\delta$ fraction of errors in codeword
correct with probability at least $1/2 + \epsilon$

Better than random guessing if $\epsilon > 0$
Locally decodable codes [Katz, Trevisan, 2000]

$C : \{0, 1\}^k \rightarrow \{0, 1\}^n$ is a $(q, \delta, \epsilon)$-LDC if there exists a probabilistic $A$ such that:

1. $\forall i \in [k], x \in \{0, 1\}^k, y \in \{0, 1\}^n$ s.t. $d(C(x), y) \leq \delta n$

   $Pr[A(y, i) = x_i] \geq \frac{1}{2} + \epsilon$

2. $A$ reads $\leq q$ bits of $y$

Advantage over random guessing $\epsilon$

Correctness $\frac{1}{2} + \epsilon$

For codes over $\Sigma$, $\frac{1}{|\Sigma|} + \epsilon$. 
Efficient constructions:

Linear $n = O(k)$ length, sublinear $q = k^{\beta}$ queries (based on ideas from [Beaver, Feigenbaum 1990]).

[Babai, Fortnow, Levin, Szegedy 1991]:

nearly linear length $n = k^{1+\beta}$, $q = \text{polylog}(k)$

No polynomial length codes known for constant number of queries.
1-query Locally Decodable Code not possible

[Katz, Trevisan 2000]

For any $\delta, \epsilon > 0$ there exist $k_{\delta,\epsilon}$ s.t. for $k > k_{\delta,\epsilon}$

$(1, \delta, \epsilon)$-LDC does not exist.
Hadamard code

Based on ideas in [Blum, Luby, Rubinfeld 1990]:

**Hadamard code** is $(2, \delta, 1/2 - 2\delta)$ - LDC for any $\delta$

**Correctness** $1 - 2\delta$, nontrivial when $\delta < 1/4$

$n = 2^k$

Columns of generator matrix:

all possible vectors of length $k$. 
Hadamard code

\[ x_1, \ldots, x_k \rightarrow C(x)_1, \ldots, C(x)_n \]

Each codeword bit \( C(x)_j \) is some linear combination:
\[ C(x)_j = a_{1,j}x_1 + a_{2,j}x_2 + \ldots + a_{k,j}x_k \]

2-query decoder to find \( x_i \):
Pick random \( j \in [n] \), read \( y_j \) and \( y_j' \)
s.t. \( C(x)_{j'} = C(x)_j - x_i \)

Correct with prob. at least \( 1 - 2\delta \)
Lower bounds for 2-query codes

Goldreich, Karloff, Schulman, Trevisan ’02
For 2-query linear LDCs
\[ n \geq 2^{\Omega(k)} \] holds for any field [Dvir, Shpilka]

Kerenidis, de Wolf ’03
For 2-query binary LDCs
\[ n \geq 2^{\Omega(k)} \]

Still open: nontrivial lower bounds for 2-query nonlinear codes over alphabets of size \( \Omega(\sqrt{k}) \).
General lower bounds

Katz, Trevisan ’00
\[ n \geq \Omega(k^{\frac{q}{q-1}}) \]

Kerenidis, de Wolf ’03
\[ n \geq \Omega(\left(\frac{k}{\log k}\right)^{\frac{q+1}{q-1}}) \]

Woodruff ’07
\[ n \geq \Omega(k^{\frac{q+1}{q-1}}) / \log k \]

All below quadratic for \( q \geq 3 \).
3-query linear codes

Woodruff 2010

\[ n \geq \Omega(k^2) \] for 3-query linear codes over any field.

OPEN: 3-query linear locally decodable codes with
\[ n = O(k^2) \].
Subexponential size 3-query LINEAR codes

Yekhanin ’07 assuming infinitely many Mersenne primes

Efremenko ’08 unconditional $n = 2^{2^{O(\sqrt{\log k \log \log k})}}$

$2^{O(\sqrt{\log k \log \log k})}$

$k^{(\log k)^c} \leq 2^{k^\alpha}$

Correctness $1 - 3\delta$ for nonbinary codes.

Woodruff ’08 Correctness $1 - 3\delta - \eta$ for binary codes.
Tradeoffs between length and correctness

Gal, Mills 2011

3-query codes require exponential size for higher correctness

Correctness $1 - 3\delta + 6\delta^2 - 4\delta^3 + \gamma$ requires size

$$n \geq 2^{\Omega(\gamma k)}$$

Holds for: LINEAR codes over any finite field

BINARY NONLINEAR codes
Recovery graphs and matchings

Hadamard code For each $i \in [k]$ the possible pairs we query for $x_i$ form a matching.

Pick random $j \in [n]$, read $y_j$ and $y_{j'} = y_j + x_i$

$C(x)_j + C(x)_{j'} = x_i$

Given $j$, the corresponding $j'$ is unique (for fixed $i$) → we get a “matching”.
Graph $G = (V, E)$:

vertices – codeword positions $\{1, \ldots, n\}$

edges – pairs of positions $\{j_1, j_2\}$ we query with probability $> 0$

Graph $G$ of the Hadamard code:

union of $k$ matchings, each of size $n/2$
Linear codes: vector $a_j$ for each position $C(x)_j$ of the codeword: $C(x)_j = a_j \cdot x$.

$a_j$: $j$-th column of the generator matrix $A$: $C(x) = A \cdot x$.

Binary codes: $a_j \in \{0, 1\}^k$.
$e_i \in \{0, 1\}$ $i$-th unit vector (1 in position $i$, 0 everywhere else.)

$e_i = a_{j_1} + a_{j_2} \rightarrow x_i = C(x)_{j_1} + C(x)_{j_2}$ for every $x$.

edges of the Boolean hypercube

Hadamard code uses such pairs.
Combinatorial Lemma [GKST]

Let $M_i$ be a set of disjoint pairs $(j_1, j_2)$ such that $e_i = a_{j_1} + a_{j_2}$.

Then $n \geq 2^{2\alpha k}$ where $\alpha = \frac{\sum_{i=1}^{k} |M_i|}{kn}$.

[Katz, Trevisan 2000] The graph of every “good” 2-query locally decodable code must contain a large matching for EVERY $i \in [k]$. 
Extend to $q$-query codes

$q$-tuples of positions $\rightarrow$ “hypergraphs”
or “set systems”.

A $q$-tuple is a $q$-edge for $x_i$ if
- queried with prob. $> 0$
- gives nontrivial advantage over random guessing.
Reed Muller codes

$r$ (prime power) alphabet size
$m$ number of variables of the polynomials
$d$ degree $(d < r - 1)$

$k = \binom{m+d}{d}$ message length $= \# \ of \ possible \ monomials$
message $x \in F_r^k$ specifies a polynomial $P_x$

$n = r^m$ codeword length
codeword $C(x) \in F_r^n$ values of $P_x$ evaluated on all possible inputs from $F_r^m$. 
Reed Muller code is \( d + 1 \)-query locally decodable code for any \( \delta \), with correctness \( 1 - (d + 1)\delta \).

**Decoder** For any \( w \in F^m_r \), pick random \( v \in F^m_r \) and consider \( L = \{ w + \lambda v | \lambda \in F_r \} \).

Query \( d + 1 \) codeword positions, corresponding to points of \( L \) for \( \lambda \in S \), where \( S \subseteq F_r \) and \( |S| = d + 1 \).

Recover unique univariate polynomial \( h(\lambda) \) (degree of \( h \) is \( \leq d \))

If no errors, \( h(0) = P_x(w) \)
Locally correctable codes

$C : \{0, 1\}^k \rightarrow \{0, 1\}^n$ is a $(q, \delta, \epsilon)$-LCC if $\exists$ probabilistic $A$ s.t.

1. $\forall i \in [n], x \in \{0, 1\}^k, y \in \{0, 1\}^n$ s.t. $d(C(x), y) \leq \delta n$

$$\Pr[A(y, i) = C(x)_i] \geq \frac{1}{2} + \epsilon$$

2. $A$ reads $\leq q$ bits of $y$

Reed Muller codes are locally correctable