Strong Locally Testable Codes with Relaxed Local Decoders

Oded Goldreich       Tom Gur       Ilan Komargodski

Weizmann Institute
An **Error-Correcting Code** is a mapping \( C : \Sigma^k \to \Sigma^n \) such that every two codewords \( C(x), C(y) \) are "far" apart.

In this talk we focus on codes that are:

1. Binary \((\Sigma = \{0, 1\})\);
2. Linear \((C \text{ is a linear function})\);
3. With constant relative distance \( \left( \frac{\text{HD}(x, y)}{n} = \Omega(1) \right) \).
Locally Testable Codes

(Weak) LTCs

<table>
<thead>
<tr>
<th>( w_1 )</th>
<th>( w_2 )</th>
<th>( w_3 )</th>
<th>\cdots</th>
<th>( w_n )</th>
</tr>
</thead>
</table>

Query access

Tester

\( \varepsilon \)

\( T \) makes \( \text{poly}(1/\varepsilon) \) queries.

- If \( w \in C \), \( T^w(\varepsilon) = 1 \).
- If \( w \) is \( \varepsilon \)-far from \( C \),
  \[ \Pr_{T}[T^w(\varepsilon) = 0] \geq 2/3. \]

Strong LTCs

<table>
<thead>
<tr>
<th>( w_1 )</th>
<th>( w_2 )</th>
<th>( w_3 )</th>
<th>\cdots</th>
<th>( w_n )</th>
</tr>
</thead>
</table>

Query access

Tester

\( T \) makes \( O(1) \) queries.

- If \( w \in C \), \( T^w = 1 \).
- If \( w \notin C \),
  \[ \Pr_{T}[T^w = 0] \geq \text{poly}(\delta_C(w)). \]

The relative distance of \( w \) from \( C \).
Locally Testable Codes

Main parameter: **code length** (amount of redundancy).

There are strong-LTCs with pretty good parameters:

### Goldreich and Sudan (2006)

∃ strong-LTC with linear distance and nearly-linear length ($k^{1+\alpha}$ for an arbitrarily small constant $\alpha > 0$).

### Viderman (2013)

∃ strong-LTC with linear distance and quasi-linear length (i.e., length $k \cdot \text{polylog}k$).
**Locally Decodable Codes**

<table>
<thead>
<tr>
<th>LDCs</th>
<th>Decoder</th>
</tr>
</thead>
<tbody>
<tr>
<td>(w_1)</td>
<td>(w_2)</td>
</tr>
</tbody>
</table>

- \(O(1)\) queries
- Index \(i\)
- Decoder
- \(x_i\)

**Despite much attention, the best known LDCs are only of super-polynomial length.**


**Any \(q\)-query LDC must be of length \(\Omega\left(k^{1+\frac{1}{q-1}}\right)\)**

To bypass this barrier, BGHSV introduced relaxed-LDCs, which allow recovery failure on a small fraction of the bits.

"It says they’re worried that these files won’t be readable by future technologies."
Locally Decodable Codes

**LDCs**

<table>
<thead>
<tr>
<th>$w_1$</th>
<th>$w_2$</th>
<th>$w_3$</th>
<th>...</th>
<th>$w_n$</th>
</tr>
</thead>
</table>

Decoder

$O(1)$ queries

$\text{index } i \xrightarrow{} x_i$

- $\forall x \in \{0, 1\}^k \ D^C(x)(i) = x_i$.
- $\forall w \in \{0, 1\}^n \ \delta$-close to $C$,
  $\Pr_D[D^w(i) = x_i] \geq 2/3$.

(Where $\delta$ is the decoding radius.)

**Relaxed LDCs**

<table>
<thead>
<tr>
<th>$w_1$</th>
<th>$w_2$</th>
<th>$w_3$</th>
<th>...</th>
<th>$w_n$</th>
</tr>
</thead>
</table>

Relaxed Decoder

$O(1)$ queries

$\text{index } i \xrightarrow{} \{x_i, \perp\}$

- $\forall x \in \{0, 1\}^k \ D^C(x)(i) = x_i$.
- $\forall w \in \{0, 1\}^n \ \delta$-close to $C$,
  $\Pr_D[D^w(i) \in \{x_i, \perp\}] \geq 2/3$.
- $\forall w$ that is $\delta$-close to $C$ and for most $i \in [k], D^w(i) \neq \perp$. 
There exist relaxed-LDCs with nearly-linear length.

It is still an open problem whether there exist shorter relaxed-LDCs.

BGHSV (2002)

Relaxed LDCs

\( n = k^{1+\alpha} \)

- \( \forall x \in \{0, 1\}^k \ D^C(x)(i) = x_i. \)
- \( \forall w \in \{0, 1\}^n \ \delta\text{-close to } C, \Pr_D[D^w(i) \in \{x_i, \bot\}] \geq 2/3. \)
- \( \forall w \) exists large \( I_w \) on which \( D \) must not output \( \bot \).
Our Goal: Constructing short codes that are both strong-LTCs and relaxed-LDCs.

Motivation: The promise required for successfully decoding can be verified by the testing procedure.

Theorem 1
There exists a binary linear code that is a relaxed-LDC and a strong-LTC with constant relative distance and nearly-linear length \((n = k^{1+\alpha})\).
**Starting point: The relaxed LDC of BGHSV**

**Key idea:** Each codeword consists of 3 equal-length parts with the following properties.

Let $C$ be a good linear code. Each codeword $C'(x)$ of the relaxed-LDC is given by:

<table>
<thead>
<tr>
<th>$C'(x)$</th>
<th>$x$</th>
<th>PCPs of proximity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Large Distance</td>
<td>Easy to Decode</td>
<td>Consistency Mechanism</td>
</tr>
</tbody>
</table>

**Large Distance**

**Easy to Decode**

**Consistency Mechanism**
PCPs of Proximity

PCPPs are PCPs with query access to both proof & input.

PCPP verifier $V$ for $S$ satisfies:
- If $x \in S$, $\exists \pi$ s.t. $V^{x,\pi} = 1$.
- If $x$ is $\varepsilon$-far from $S$, $\forall \pi$  
  $\Pr_{V}[V^{x,\pi} = 0] > 2/3$.

BGHSV (2002)

There exist PCPPs for NP with nearly-linear length and constant query complexity.
Starting point: The relaxed LDC of BGHSV

Let \( C : \{0, 1\}^k \rightarrow \{0, 1\}^n \) be a good linear code. Each codeword \( C'(x) \) of the relaxed-LDC is given by:

<table>
<thead>
<tr>
<th>( C(x) )</th>
<th>( x )</th>
<th>( \pi_1, \pi_2, \ldots, \pi_k )</th>
</tr>
</thead>
<tbody>
<tr>
<td>High Distance</td>
<td>Easy to Decode</td>
<td>Consistency Mechanism</td>
</tr>
</tbody>
</table>

\( \pi_i \) is a PCPP that asserts that the first part is a codeword of \( C \) that is consistent with the \( i \)'th bit of the second part.

The length of relaxed-LDC above is \textit{quadratic}.

Using further ideas, BGHSV obtain \textit{nearly-linear} length.

- Repetition is used to balance the lengths of the three parts.
Obtaining Weak Testability

Since there exist PCPPs with nearly-linear length for NP, any linear code can be transformed to an LTC, at the cost of a nearly-linear blow-up in the length.

The drawback of this approach is that it results in LTCs that are inherently weak.
Instead, we consider a stronger notion of PCPPs, called **Strong Canonical PCPPs** (scPCPP), which satisfy two additional requirements:

- **Canonicity**: \( \forall x \in S \) there exists a unique \( \pi(x) \) (the canonical proof) that the verifier is **required** and **allowed** to accept.

- **Strong Soundness**: The verifier is **proximity oblivious** and must reject any \( (x', \pi') \) w.p. that is polynomial in its distance from \( \{(x, \pi(x))\}_{x \in S} \).

\[
\begin{align*}
\Pr[V \wedge \pi \wedge x = 1] &= 1. \\
\Pr[V(x', \pi') = 0] &> \frac{2}{3}. \\
\Pr[V \wedge x' = 0] &\geq \varepsilon^c
\end{align*}
\]
Strong Canonical PCPPs

Given adequate scPCPPs, appending codewords with an efficient scPCPP allows to transform codes to strong-LTCs.

Unfortunately, there are no known general-purpose scPCPPs (let alone such with nearly-linear length).

We construct scPCPPs for the specific statements that we need, albeit with polynomial length.

Theorem 2

Let $C$ be a good linear code. Then, there exists a scPCPP with polynomial proof length for membership in $C$. 
Obtaining Strong Testability

(i.e., replacing the PCPPs in the construction of BGHSV with our scPCPPs)

Applying our scPCPPs in a *naive* way yields *polynomial* length codes, whereas we aim for *nearly-linear* length.

Instead, we use an alternative approach that relies on *tensor codes*.

Tensor codes allow us to use scPCPPs only for short statements, such that even with the polynomial blow-up, the length of each proof would still be sub-linear.

Instead of providing PCPPs that assert the consistency of each $x_i$ with the entire codeword — we provide PCPPs that assert the consistency of each $x_i$ with all lines in the tensor that cross it.

We prove that for tensor codes, these local consistency statements propagate to global consistency statements.
Questions?