An Exponential Lower Bound for Linear 3-Query Locally Correctable Codes

Peter Manohar
Carnegie Mellon University

Based on joint work with:
Pravesh K. Kothari (IAS, Princeton, CMU)
Locally Correctable Codes

LCC:

$\begin{array}{cccccc}
0 & 1 & 0 & 1 & 1 & 1 \\
\end{array}$

$b \in \{0,1\}^k$

$(q, \delta, \varepsilon)$-LCC

Input: $y \in \{0,1\}^n$, $\Delta(x,y) \leq \delta n$

$u \in [n]$

Read $\leq q$ bits of $y$

Output $x_u$ w.p. $\geq 1 - \varepsilon$

$x \in \{0,1\}^n$

$\delta n$ errors

$\begin{array}{cccccc}
0 & 1 & 0 & 1 & 1 & 1 \\
\end{array}$

$y \in \{0,1\}^n$

$\hat{x}_u \in \{0,1\}$

$q$

Used in: program checking, PCPs, PIR, avg-case to worst-case, explicit rigid matrices, additive combinatorics, block designs
Locally Decodable Codes

LDC:

\[ b \in \{0,1\}^k \]

\[ x \in \{0,1\}^n \]

\[ \Delta(x, y) \leq \delta n \]

\( (q, \delta, \varepsilon) \)-LDC

Input: \( y \in \{0,1\}^n \), \( \Delta(x, y) \leq \delta n \), \( i \in [k] \)

Read \( \leq q \) bits of \( y \)

Output \( b_i \) w.p. \( \geq 1 - \varepsilon \)

Errors: \( \delta n \)

Decoder

Used in: program checking, PCPs, PIR, avg-case to worst-case, explicit rigid matrices, additive combinatorics, block designs
Locally Correctable Codes

LCC:

\[
\begin{array}{cccccc}
0 & 1 & 0 & 1 & 1 & 1 \\
\end{array}
\]

\[b \in \{0,1\}^k\]

\[(q, \delta, \epsilon)\text{-LCC}\]

Input: \(y \in \{0,1\}^n\), \(\Delta(x, y) \leq \delta n\)

\[u \in [n]\]

Read \(\leq q\) bits of \(y\)

Output \(x_u\) w.p. \(\geq 1 - \epsilon\)

\[x \in \{0,1\}^n\]

\[\delta n\text{ errors}\]

\[
\begin{array}{ccccccccc}
0 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\
\end{array}
\]

\[y \in \{0,1\}^n\]

\[u \in [n]\]

\[\hat{x}_u \in \{0,1\}\]

Decoder

Used in: program checking, PCPs, PIR, avg-case to worst-case, explicit rigid matrices, additive combinatorics, block designs
Do LCCs/LDCs exist?

Purely combinatorial question
No natural random model for probabilistic method
Reed-Muller Codes

Simple $q$-LCC: Reed-Muller codes, $n = \exp(k^{1/(q-1)})$, and thus $q$-LDC

For $3$-LCC: take deg 2 polys $f(z_1, \ldots, z_t)$ on $\mathbb{F}_4$

To decode $f(z)$, pick $z' \in \mathbb{F}_4^t$, look at $L_{z, z'} = \{z + \lambda z' : \lambda \in \mathbb{F}_4\}$

Binary code via $\text{Tr}: \mathbb{F}_4 \to \mathbb{F}_2$

Parameters: $k \approx t^2/2$, $n = 4^t = 2^{O(\sqrt{k})} = 2^{\sqrt{2k}}$

Can we do better?
The Story in 2022

Simple $q$-LCC: Reed-Muller codes, $n = \exp(k^{1/(q-1)})$, and thus $q$-LDC

Can we do better?

$q = 2$ : achieves $n \leq 2^k$, linear 2-LCC
  tight $n \geq 2^{\Omega(k)}$ 2-LDC lower bound  
  
  $[\text{Katz Trevisan 00}]
  \ [\text{Goldreich, Karloff, Schulman, Trevisan 02}]
  \ [\text{Kerenidis, de Wolf 04}]

$q = 3$ : achieves $n \leq 2^{2\sqrt{2k}}$, linear 3-LCC
Matching vector code $n \leq 2^{\Omega(\sqrt{\log k \log \log k})}$, linear 3-LDC of subexp length!
Is it a 3-LCC? Open Q  
  $[\text{Yekhanin 12}]
  \ [\text{Yekhanin 07}]
  \ [\text{Efremenko 08}]

Weak $n \geq k^2$ 3-LDC lower bound  
  $[\text{Kerenidis, de Wolf 04}]

“Techniques cannot beat $k^2$ ”  
  $[\text{Dvir, Gopi, Gu, Wigderson 19}]
  \ [\text{Alrabiah Guruswami 21}]

Can we do better?
The Story in 2022

Simple $q$-LCC: Reed-Muller codes, $n = \exp(k^{1/(q-1)})$, and thus $q$-LDC

Can we do better?

Yes for LDCs. What about LCCs? **We don’t know!**

 Conj: RM codes are optimal LCCs

True for $q = 2$. Hard to prove $q \geq 3$  

[Hamada 74] Conj: RM codes are optimal design LCCs

Barrier: **all lower bounds apply to Locally Decodable Codes**

Recall: Matching Vector codes, 3-LDC, $n = \exp(k^{o(1)})$  

[Yekhanin 07]  
[Efremenko 08]
The Story in 2024

(1) [Alrabiah, Guruswami, Kothari, M 23]: 3-LDC has $n \geq \tilde{\Omega}(k^3)$
(2) [Kothari, M 23]: linear 3-LCC has $n \geq 2^{\Omega(k^{1/8})} \rightarrow 2^{\Omega(k^{1/4})} \rightarrow 2^{\tilde{\Omega}(\sqrt{k})}$

Techniques do not work for LDCs!
MV codes are not LCCs

Via connection to “rainbow cycles”
[Hsieh Kothari Mohanty Munhá Sudakov 24]
“Rainbow cycle bound”
[Alon Bucić Sauermann Zakharov Zamir 23]
The Story in 2024

(1) [Alrabiah, Guruswami, Kothari, M 23]: 3-LDC has \( n \geq \tilde{\Omega}(k^3) \)

(2) [Kothari, M 23]: linear 3-LCC has \( n \geq 2^{\Omega(k^{1/8})} \) → \( 2^{\Omega(k^{1/4})} \) → \( 2^{\tilde{\Omega}(\sqrt{k})} \)

Via connection to “rainbow cycles”

MV codes are not LCCs

(3) [Kothari, M 24]: design 3-LCC has \( n \geq 2^{(1-o(1))\sqrt{k}} \)

Proves Hamada’s conj for 4-designs up to \( 2\sqrt{2} \)-factor

“Rainbow cycle bound”

(4) [Kothari, M 24]: nonlin \((3,\delta, \varepsilon)\)-LCC has \( n \geq \tilde{\Omega}(k^{1/2\varepsilon}) \) → \( k^{\Omega(\log k)} \)

Techniques do not work for LDCs!

\[
\begin{align*}
k^2 & \text{ barrier} \\
& \begin{array}{c}
[DGGW 19] [AG 21] \\
[AGKM 23] [KM 24]
\end{array}
\end{align*}
\]

“LDC barrier” \( \exp(k^{o(1)}) \)

\[
\begin{align*}
k^2 & \quad k^3 & \quad k^{1/2\varepsilon} & \quad k^{\Omega(\log k)} & \quad 2^{\Omega(k^{1/8})}
\end{align*}
\]

3-LCCs

Today

\[
\begin{align*}
[\text{Today}] & \quad 2\Omega(k^{1/4}) & \quad 2\Omega(\sqrt{k/\log k}) & \quad 2^{1-o(1))\sqrt{k}} & \quad 2^{2\sqrt{2k}}
\end{align*}
\]

3-LDCs
Proof Strategy

Theorem [KM 23]:
Let $C$ be a linear $(3, \delta, \epsilon)$-locally correctable code. Then, $n \geq \exp((\delta^2 k)^{1/8})$

Approach of [AGKM 23]:
1. Reduce to proving unsatisfiability of XOR formulas with minimal randomness
2. Use algorithm for CSPs: [Abascal, Guruswami, Kothari 21], [Guruswami, Kothari, M 22]

Semirandom CSP refutation
(CSPs with minimal randomness)

“Theory analogue” of using SAT solvers to prove theorems

Today: present ideas as reduction to 2-LDC
Linear 3-LCCs in “Combinatorial Form”

To decode $x_u$, the decoder (1) queries some entries $C \subseteq [n]$ of $y$
(2) computes a “predicate” on $y|_C$

For linear codes…

(1) “Predicate” is linear: output $\sum_{v \in C} y_v$

(2) Query sets $H_u$ is $\Omega(n)$-sized 3-unif matching ($C$’s are disjoint, $|C| = 3$)

```
C_1, C_2, C_3, C_4 \in H_u
```

```
0 1 0 1 1 1 0 0 0 1 1 0 0 0 0 1
```

“4-Sparse Parity check matrix”:

for each $u \in [n], C = \{v_1, v_2, v_3\} \in H_u$

$$x_{v_1} + x_{v_2} + x_{v_3} = x_u$$
for all codewords $x$
From 3-LCCs to 2-LDC

Given: “parity check matrix” \( H_1, \ldots, H_n \), 3-unif matchings of size \( \Omega(n) \)

Goal: construct 2-LDC of length \( n^{O(\ell)} \)

Then by 2-LDC lower bound, \( k \leq O(\ell \log n) \)

2-LDC encoding (high level):

1. map \( x \in \{0,1\}^n \) to \( y \in \{0,1\}^N, N = \binom{n}{\ell} \)
2. set \( y_S = \sum_{v \in S} x_v \)

2-LDC if there exist matchings \( M_1, \ldots, M_n \) on \( N \) of size \( \Omega(N) \) s.t.

\((S, T) \in E(M_u)\) implies \( y_S + y_T = \sum_{v \in S} x_v + \sum_{v \in T} x_v = x_u \) for all codewords \( x \)
From 3-LCCs to 2-LDC

Given: “parity check matrix” $H_1, \ldots, H_n$, 3-unif matchings of size $\Omega(n)$
Goal: construct 2-LDC of length $n^{O(\ell)}$
Then by 2-LDC lower bound, $k \leq O(\ell \log n)$

2-LDC encoding (high level): $y_S = \sum_{v \in S} x_v$, $|S| = \ell$, $N = \binom{n}{\ell}$

2-LDC if there exist *matchings* $M_1, \ldots, M_n$ on $N$ of size $\Omega(N)$ s.t.
\[(S, T) \in E(M_u) \text{ implies } y_S + y_T = \sum_{v \in S} x_v + \sum_{v \in T} x_v = x_u \text{ for all codewords } x\]

Define $G_u$: edge $(S, T)$ if $y_S + y_T = x_u$ for all codewords $x$

1) Clearly need $G_u$ to have avg deg $d_u \gg 1$
   If $G_u$ has max deg $O(d_u)$, then “greedy” matching $\geq d_u N / O(d_u) = \Omega(N)$

This is **false**!

2) “Row pruning”: find dense $G'_u$ with max deg $O(d_u)$
“Degree Heuristic”

Given: “parity check matrix” $H_1, \ldots, H_n$, 3-unif matchings of size $\Omega(n)$
Goal: construct 2-LDC of length $n^{O(\ell)}$
Then by 2-LDC lower bound, $k \leq O(\ell \log n)$

(1) Clearly need $G_u$ to have $\text{avg} \ \deg d_u \gg 1$

\[ S \xrightarrow{T} \sum_{v \in S \oplus T} x_v = \sum_{v \in S} x_v + \sum_{v \in T} x_v = x_u \]

“Labeled by $R$“: $S \xrightarrow{T} \sum_{v \in R} x_v = x_u$, $S \oplus T = R$

Each $R$ contributes $\approx N \cdot (\ell/n)^{|R|/2}$ edges, or $(\ell/n)^{|R|/2}$ to the density

Recall: start with $H_u$ s.t. $\sum_{v \in C} x_v = x_u$ for all $C \in H_u$, $|C| = 3$, $|H_u| \geq \Omega(n)$

\[ |E(G_u)|/N \approx (\ell/n)^{3/2} \cdot \Omega(n) \quad \rightarrow \quad \text{Need } d_u \approx (\ell/n)^{3/2} n \gg 1 \quad \rightarrow \quad \ell = n^{1/3} \]

\[ \rightarrow \quad \text{Get } k \leq O(\ell \log n) = \tilde{O}(n^{1/3}) \quad \text{[AGKM 23]} \]
Boosting Density with 2-Chains

Given: “parity check matrix” $H_1, …, H_n$, 3-unif matchings of size $\Omega(n)$
Goal: construct 2-LDC of length $n^{O(\ell)}$
Then by 2-LDC lower bound, $k \leq O(\ell \log n)$

(1) Clearly need $G_u$ to have avg deg $d_u \gg 1$

Suppose we form a “chain”:

$$x_{v_1} + x_{v_2} + x_w = x_u$$
$$x_{v_3} + x_{v_4} + x_{w'} = x_w$$
$$C = \{v_1, v_2, w\} \in H_u$$
$$C \in H_w$$

$H_u^{(2)}$ produces $N \cdot (\ell/n)^{2.5} \Omega(n^2)$ edges in $G_u$

So, $\ell = n^{1/5} \implies k^5 \leq n$ Small win! Beat $k^3$!
Boosting Density with Long Chains

Given: “parity check matrix” $H_1, \ldots, H_n$, 3-unif matchings of size $\Omega(n)$

Goal: construct 2-LDC of length $n^{O(\ell)}$

Then by 2-LDC lower bound, $k \leq O(\ell \log n)$

(1) Clearly need $G_u$ to have $\text{avg} \ \text{deg} \ d_u \gg 1$

Suppose we form a “long chain” $r$ steps:

\[
|H_u^{(r)}| = \Omega(n)^r, \text{ arity } 2r + 1 \quad \text{always odd!}
\]

$H_u^{(r)}$ produces $N \cdot (\ell/n)^{r+0.5} \Omega(n)^r$ edges in $G_u$

So, $\ell = n^{1/2r} \implies k \leq n^{1/2r}$

avg deg is $(\ell/n)^{r+0.5} \cdot n^r \sim \ell^r n^{0.5}$

Take $r = O(\log n) \implies k \leq \log^8 n$

Big win: $n \geq \exp(k^{1/8})$!
Recall: 3-LCC to 2-LDC Reduction

Given: “parity check matrix” $H_1, \ldots, H_n$, 3-unif matchings of size $\Omega(n)$

Goal: construct 2-LDC of length $n^{O(\ell)}$

Then by 2-LDC lower bound, $k \leq O(\ell \log n)$

2-LDC encoding (high level): $y_S = \sum_{v \in S} x_v$, $|S| = \ell$, $N = \binom{n}{\ell}$

2-LDC if there exist matchings $G_1, \ldots, G_n$ on $N$ of size $\Omega(N)$ s.t.

$(S, T) \in E(G_u)$ implies $\sum_{v \in S} x_v + \sum_{v \in T} x_v = x_u$ for all codewords $x$

Form chains!

Main technical part!

(1) Clearly need $G_u$ to have $\text{avg deg } d_u \gg 1$

If $G_u$ has $\text{max deg } O(d_u)$, then “greedy” matching $\geq d_u N/O(d_u) = \Omega(N)$

This is false!

(2) “Row pruning”: find dense $G'_u$ with $\text{max deg } O(d_u)$
“Row Pruning”

Given: “parity check matrix” $H_1, \ldots, H_n$, 3-unif matchings of size $\Omega(n)$
Goal: construct 2-LDC of length $n^{O(\ell)}$

(2) “Row pruning”: find dense $G'_u$ with max deg $O(d_u)$

[KM 23]: random $S$ (vertex) has degree $O(d_u)$ w.h.p.
Proof uses Kim-Vu style conc inequality, “high moments”, needs $\ell \geq \log^5 n$
+ $\log^2 n$ loss from “heavy pairs” = $\log^8 n$

[Yan 24] (rephrased): second moment $\mathbb{E}_S[\deg(S)^2]$ suffices!
Requires $\ell \geq \log^2 n + \log n$ loss from “heavy pairs” = $\log^4 n$

[KM 24]: if $H$ is a block design, can take $\ell = \log n$
Necessarily requires some changes to $G_u$ in [KM 23, Yan 24]
3-LCC to 2-LDC Reduction

Given: “parity check matrix” $H_1, \ldots, H_n$, 3-unif matchings of size $\Omega(n)$
Goal: construct 2-LDC of length $n^{O(\ell)}$
Then by 2-LDC lower bound, $k \leq O(\ell \log n)$

2-LDC encoding (high level): $y_S = \sum_{v \in S} x_v$, $|S| = \ell$, $N = \binom{n}{\ell}$

2-LDC if there exist matchings $G_1, \ldots, G_n$ on $N$ of size $\Omega(N)$ s.t.
$(S, T) \in E(G_u)$ implies $\sum_{v \in S} x_v + \sum_{v \in T} x_v = x_u$ for all codewords $x$

Form chains!

Main technical part!

(1) Clearly need $G_u$ to have $\text{avg deg } d_u \gg 1$
If $G_u$ has $\text{max deg } O(d_u)$, then “greedy” matching $\geq d_u N/O(d_u) = \Omega(N)$

This is false!

(2) “Row pruning”: find dense $G_u'$ with $\text{max deg } O(d_u)$
Open Problems

What happens for larger $q$?

Degree heuristic: $\ell = n^{1-2/(q-1)} \ll n^{1-2/q}$

“Plug in $q - 1$ in LDC lower bound”

Non-linear codes?

Form “adaptive chains”, loses success prob: $1 - \varepsilon$ to $1 - r\varepsilon$

Cannot use “standard reductions” from [Katz Trevisan 00]

Need full power of spectral refutation

Conj: RM codes are optimal linear 3-LCCs

[KM 24]: design case up to $2\sqrt{2}$

[AG 24]: almost $2^{\Omega(\sqrt{k}/\log k)}$ vs $2^{O(\sqrt{k})}$

(1) Exp lower bounds for non-linear 3-LCCs?
(2) Beat $k^3$ for (linear) 3-LDCs or 4-LCCs?

Thanks! (arXiv 2311.00558 and 2404.06513)