Local Codes for Insertions and Deletions

Elena Grigorescu

Based on joint work with:

Alex Block, Jeremiah Blocki, Kuan Cheng, Shubhang Kulkarni, Xin Li, Yu Zheng, Minshen Zhu.

Insertion-Deletion (Insdel) Codes

For $x \in \Sigma^s, y \in \Sigma^t$

$ED(x, y)$ =minimum number of insertions and deletions in x to obtain y.

Example: $x = 0101;\ y = 1111$ then $ED(x, y) = 4$

$Ham(x, y) = |\#i, s.t. x_i \neq y_i|$

Observation: $ED(x, y) \leq 2\ Ham(x, y)$

Message $x \in \Sigma^n$, Code $C: \Sigma^n \to \Sigma^m$, Decoder $D\colon \Sigma^m \to \Sigma^n, D(C(x) \circ error) = x$.

Relative min distance of code: $\delta_{Ham}(C) = \frac{\min_{\{x_1,x_2\}} Ham\left(C(x_1), C(x_2)\right)}{m}$; $\delta_{ED}(C) = \frac{\min_{\{x_1,x_2\}} ED\left(C(x_1), C(x_2)\right)}{2m}$

Rate: $r(C) = \frac{n}{m}$

Goals of constructions: $r = \Theta(1), \delta = \Theta(1)$. (``good'' codes)

Algorithmic goals: efficient encoder/decoder.

Brief History of Insdel Codes

[Levenstein 66] introduced insdel/edit codes

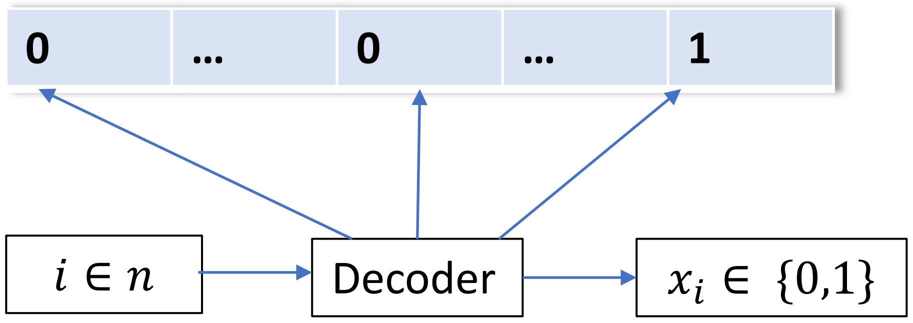
[Schulman, Zuckerman 99] first constructions of good codes with efficient encoding/decoding schemes

Lots of recent follow-up works [MatousekKiwiLoebl05, GuruswamiWang17, HaeuplerShahrasbi17, GuruswamiLi18, BrakensiekGuruswamiZbarsky18, ChengJinLiWu18, HaeuplerShahrasbiSudan18, HaeuplerRubinsteinShahrasbi19, LiuTjuawinataXing19, ChengLi21, ChenZhang22].

Excellent surveys

[Sloane02, MercierBhargavaTarokh10, Mitzenmacher08, HaeuplerShahrasbi21]

Local Codes



A code $C: \{0,1\}^n \to \{0,1\}^m$ is a Locally Decodable Code (LDC) if there exists a decoder Dec such that

- Dec makes at most q queries (``locality'' of code)
- $\forall x$ and y with $\mathrm{dist}(C(x), y) \leq \delta$ and for every message bit i

$$\Pr[Dec(y,i) = x_i] \geq ½ + \epsilon$$

Example of Hamming LDC: Hadamard code $\boldsymbol{Had: F_2^n \to F^{2^n}, Had(x) = (\langle\, a, x\, \rangle)_{a\in F_2^n}}$

$\boldsymbol{x \cdot (0..0)}$	$\boldsymbol{x \cdot (0..01)}$		...		$\boldsymbol{x \cdot (0..01) + 1}$		$\boldsymbol{x \cdot (11..1)}$

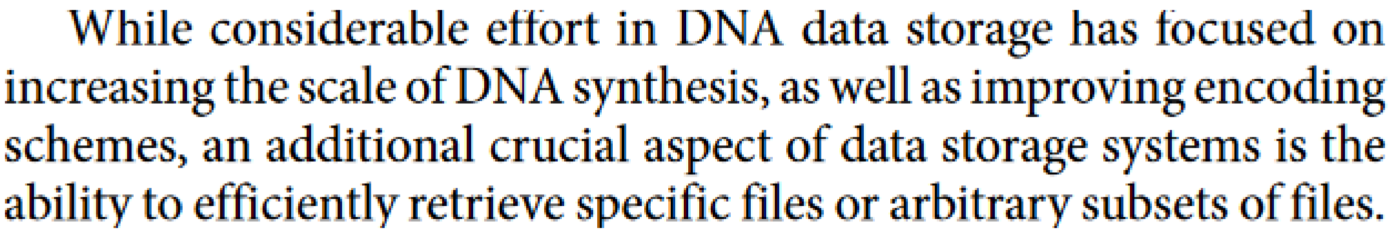
$\boldsymbol{y = Had(x) + err}$

Dec(i): pick $\boldsymbol{a} \in \{\mathbf{0,1}\}^{\boldsymbol{n}}$ u.a.r and query $\boldsymbol{a}$ and $\boldsymbol{a} + \boldsymbol{e_i}$. Output $\boldsymbol{y}(\boldsymbol{a}) + \boldsymbol{y}(\boldsymbol{a} + \boldsymbol{e_i})$.

Claim: Hadamard is a 2-query LDC for $\delta = ¼ - \epsilon/2$ Hamming errors

(Open) Question: Do $O(1)$-query **insdel** LDCs exist?

Motivation: DNA Storage


While considerable effort in DNA data storage has focused on increasing the scale of DNA synthesis, as well as improving encoding schemes, an additional crucial aspect of data storage systems is the ability to efficiently retrieve specific files or arbitrary subsets of files.

[BSBHRABB] (Nature materials, 2021)

Construction of Insdel LDCs from Hamming LDCs

Theorem [Ostrovky, Paskin-Cherniavsky 2015]:

q-query Hamming LDC $C: \Sigma^n \to \Sigma^m$ correcting δ-fraction of Hamming errors

⇓

$q\ poly \log(m)$-query Insdel LDC $C: \{0,1\}^n \to \{0,1\}^{m'}$ with $m' = \Theta(m)$ that can correct from $\Theta(\delta)$-fraction of Insdel Errors

[Block, Blocki, G., Kulkarni, Zhu 2020] reprove this result using different techniques.

Other works on insdel LDCs: [Cheng, Li, Zheng 20, Block Blocki 21]

Corollaries to OPC: Hamming vs Insdel LDCs

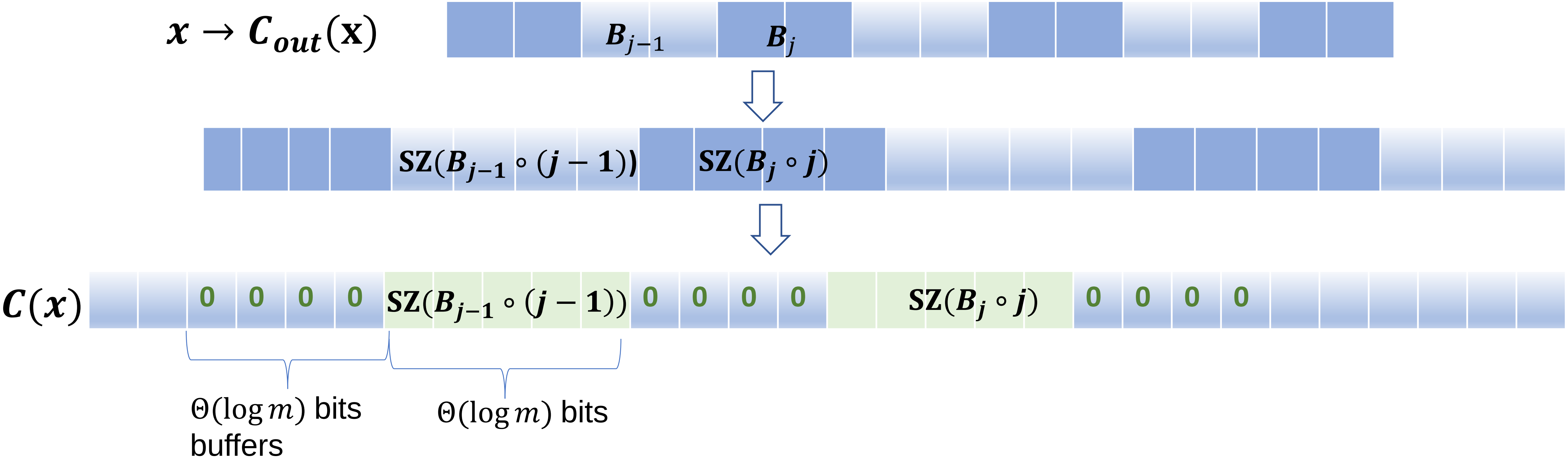
Hamming LDCs		**Insdel LDCs**
• $q = 2;\ \ m = 2^n$	$\Longrightarrow$	• $q = poly\, n;\ m = O(2^n)$
• $q \geq\ 3$ (constant); $\ m = \exp\left(n^{o(1)}\right)$ [Yek08, DKY11, Efr12]	$\Longrightarrow$	• $q = n^{o(1)}\ ;\ m = \exp(n^{o(1)})$
• $q = poly \log n;\ \ m = n^{1+\frac{1}{c}}$	$\Longrightarrow$	• $q =\ poly \log n;\ m = o(n^2)$
• $q = n^{\epsilon};\ n^{o(1)};\ m = O(n)$; [KMRS17]	$\Longrightarrow$	• $q =\ (\log n)^{\log\log n};\ m = O(n)$

The OPC Reduction: Overview of the Encoder

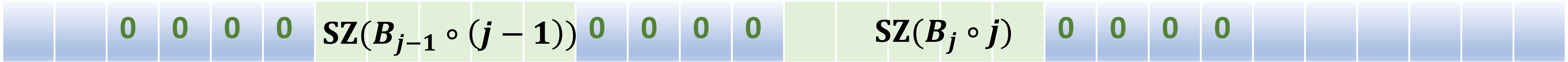
C_{out} is a Hamming LDC: $C_{out}: \Sigma^{n} \to \Sigma^{m}$

C_{in} is the Schulman-Zuckerman code: $SZ: \Sigma^{\log m} \times [m] \to \{0,1\}^{\Theta(\log m \log \Sigma)}$

$C: \{0,1\}^{n \log \Sigma} \to \{0,1\}^{\Theta(m \log \Sigma)}$ is the new Insdel LDC.



The Local Decoder


0 0 0 0
$\boldsymbol{SZ(B_{j-1} \circ (j-1))}$
0 0 0 0
$\boldsymbol{SZ(B_j \circ j)}$
0 0 0 0

To decode x_i:

- Simulate local Hamming outer decoder to find respective block of query q_j
- Decode block using SZ code and output the bits corresponding to query q_j
- Output the output of outer Hamming LDC decoder for the queries $q_{1,}\, q_2, \dots, q_t$

Challenges:

- Searching for an index j in an insdel-corrupted codeword
- Finding the buffers

Helpful Properties of the SZ Code

		0	0	0	0	1	1	1	0	1	0	0	0	0	0	1	0	1	1	0	0	0	0						

1. Constant rate: $SZ\colon \{0,1\}^k \rightarrow \{0,1\}^{\Theta(k)}$
2. Decoder can decode from δ_{in} fraction of insdel errors
3. Encoder and decoder are poly-time algorithms
4. For every x, every interval of length $\Theta(k)$ of $SZ(x)$ has fractional Hamming weight $2/5$.

Implications:

- 1, 2 and 3 implies decoding of queries in (poly log m) time.
- 4 implies can locally distinguish between high-weight and small-weight segments of the code, so can distinguish the buffers from information segments

Question: How to find block j in the codeword corrupted with insdel errors?

Useful idea: Most blocks are in correct order, so in increasing order of indices j

Solution: Perform a ``noisy'' binary search

Noisy Binary Search

Toy problem: Find element j in an array that is δ-sorted, i.e. array that becomes sorted after removing δn elements

Example: $\sigma = (1, 6, 3, 4, 5, 10, 7, 8, 9, 2, 11, 15)$ is 0.25-sorted. Find element 9.

Theorem: Can perform a ``noisy binary search” to locate element j in δ-sorted array of size n in $polylog\ n$ time (caveat: for a constant fraction of j's)

Proof idea: most indices are in order. Pick a random set of indices and perform binary search around the median.

Can find the j’s that are “locally good”, i.e. every interval containing j has small error density.

Summarizing

- Can transform Hamming LDCs into Insdel LDCs with (poly log m) loss in query complexity, $\Theta(1)$ loss in rate, $\Theta(1)$ loss in decoding radius.

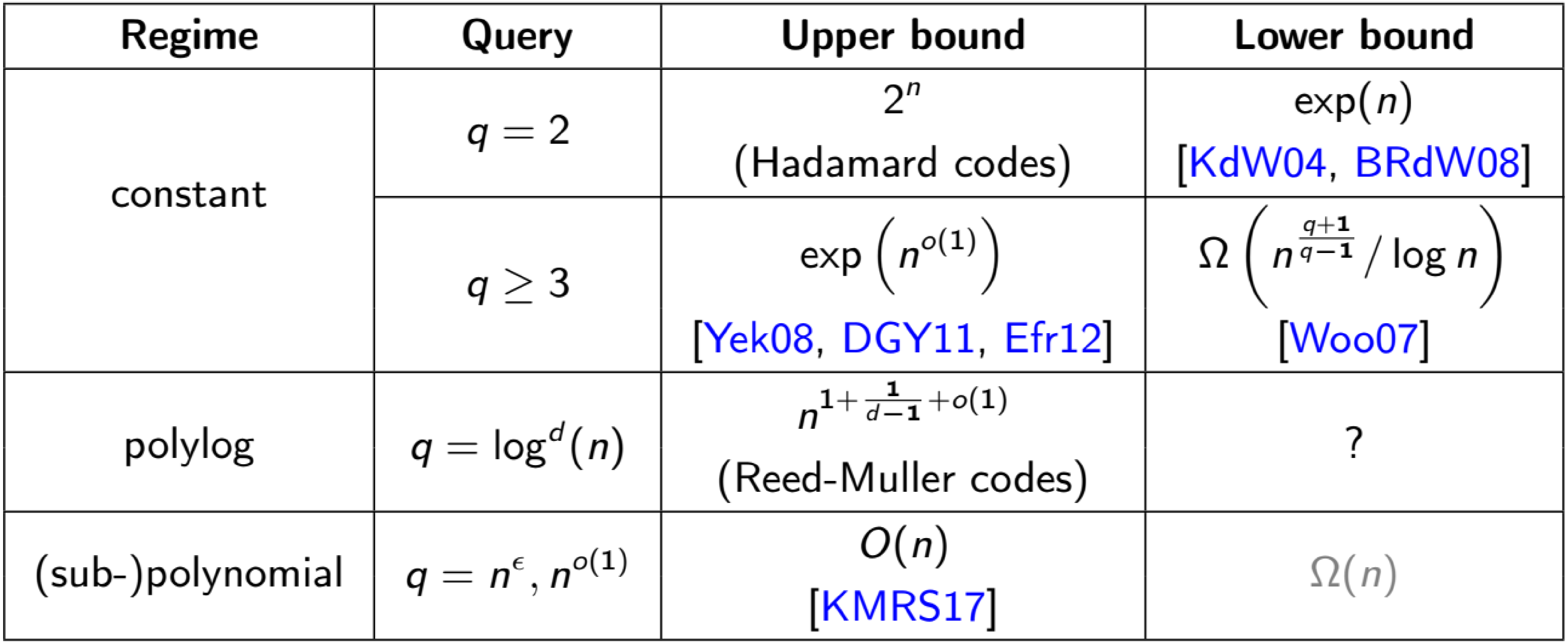
- No implications to the constant-query regime

- **Questions:** Is noisy binary search inherent to insdel LDCs?

  If not, do constant-query insdel LDCs exist?

- Next: strong lower bounds for Insdel LDCs for $q = O(1)$

State-of-the-art for Hamming LDCs



Regime	Query	Upper bound	Lower bound
constant	$q = 2$	2^n (Hadamard codes)	$\exp(n)$ [KdW04, BRdW08]
	$q \geq 3$	$\exp\left(n^{o(1)}\right)$ [Yek08, DGY11, Efr12]	$\Omega\left(n^{\frac{q+1}{q-1}} / \log n\right)$ [Woo07]
polylog	$q = \log^d(n)$	$n^{1+\frac{1}{d-1}+o(1)}$ (Reed-Muller codes)	?
(sub-)polynomial	$q = n^{\epsilon}, n^{o(1)}$	$O(n)$ [KMRS17]	$\Omega(n)$

Our Results for Insdel LDCs [BCGLZZ21]

Regime	Query	Upper bound	Lower bound
constant	$q = 2$	?	$\exp(n)$ ∞ if linear
	$q \geq 3$	?	$\exp\left(n^{\frac{1}{2q-4}}\right)$
polylog	$q \leq \frac{\log n}{2d \log\log n}$	?	$\exp\left(\log^{d-2}(n)\right)$
	$q = \log^{d+5}(n)$	$n^{1+\frac{1}{d-1}+o(1)}$ [OPC15, BBG+20]	?
(sub-)polynomial	$q = n^{\epsilon}, n^{o(1)}$	$O(n)$ [KMRS17] [OPC15, BBG+20]	$\Omega(n)$

Previous Strategies for Hamming Errors for LDCs

LDC implies ``smooth'' decoding: each index $j \in [m]$ is queried w.p. $O(1/m)$

For each $i \in n$, view the q queries as hyperedge in hypergraph with m vertices.
Then ∃ a large matching of hyperedges of size $\Omega(m/q)$.

Remainder of proofs analyze matching via:

- Quantum/information theory [KT00, KdW04, Woo07]
- Matrix hypercontractivity [BRdW08]
- Combinatorial arguments [KT00, BCG20]
- Reduction from q-query to 2-query [Woo07, Woo12]

Insdel LDCs Lower Bounds Ideas

Analyze the sets of Good queries for decoding each index:

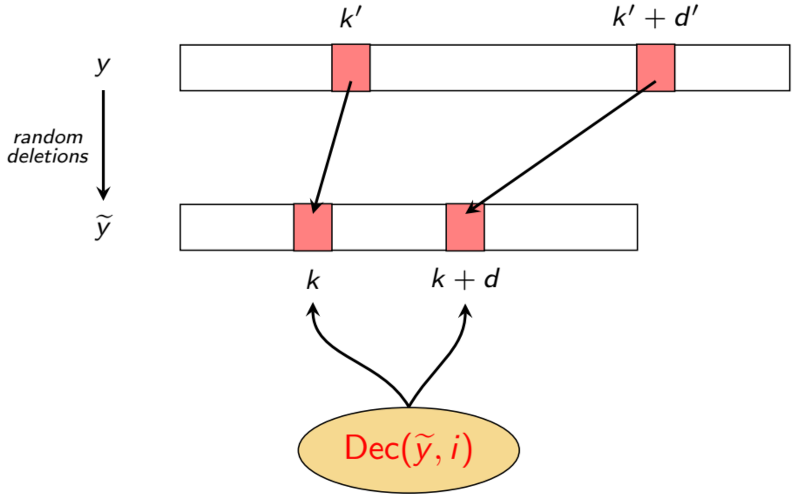
$$\boldsymbol{Good_i = \{Q = (i_1, i_2, \dots, i_q)| \exists\, f: \{0,1\}^q \to \{0,1\}\ s.t.\ Pr_x[f(C(x)_{|Q}) = x_i] \geq \text{½} + \epsilon/4\}}$$

Packing lemma [KatzTrevisan00]: Each $\boldsymbol{Q}$ is good for $\boldsymbol{O(q)}$ many message bits $\boldsymbol{i}$

Hitting Lemma: For every $\boldsymbol{i}$, $\boldsymbol{Dec(i)}$ must hit $\boldsymbol{Good_i}$ with probability $> \boldsymbol{3\,\epsilon/2}$

Next: What deletion patterns make Dec work hard?

Deletion Patterns and Queries (I)



Pattern type 1: Pick random e and delete the first e symbols of codeword.

$$x \rightarrow \quad C(x) = 01100111110001111$$

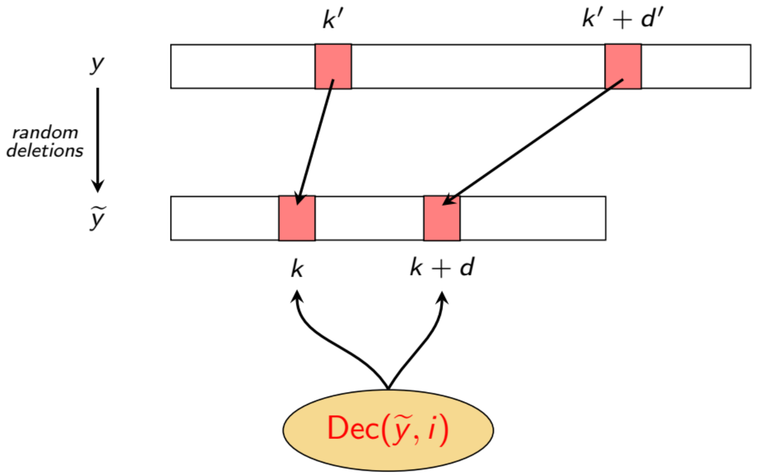
$$\tilde{y} = 11110001111$$

Observation: Query tuple $(\boldsymbol{k}, \boldsymbol{k}+\boldsymbol{d})$ comes from $(\boldsymbol{k}+\boldsymbol{e}, \boldsymbol{k}+\boldsymbol{e}+\boldsymbol{d})$ for some random $\boldsymbol{e}$.

Observation: Distance between queries stays the same, regardless of $\boldsymbol{e}$.

Definition: Distribution of $(\boldsymbol{k}+\boldsymbol{e}, \boldsymbol{k}+\boldsymbol{e}+\boldsymbol{d})$ is the “induced” distribution by query $(\boldsymbol{k}, \boldsymbol{d})$ under error distribution.

LDC for Deletion Pattern 1?



Recall Pattern type 1: Pick random e and delete the first e symbols of codeword.

Important point: Distance between queries stays the same.

Recall Hadamard code: $\boldsymbol{Had}: \boldsymbol{F}_2^n \to \boldsymbol{F}^{2^n}$, $\boldsymbol{Had}(\boldsymbol{x}) = (\langle\, \boldsymbol{a}, \boldsymbol{x}\, \rangle)_{\boldsymbol{a} \in \boldsymbol{F}_2^n}$

$\boldsymbol{x} \cdot (\mathbf{0..0})$	$\boldsymbol{x} \cdot (\mathbf{0..01})$		…				$\boldsymbol{x} \cdot (\mathbf{11..1})$

$\boldsymbol{Dec}(\boldsymbol{i})$: Pick random locations a and $a + e_i$.
$\boldsymbol{Dec}(\boldsymbol{i})$ restated: Pick a random pair at distance 2^{i-1} (assuming lex ordering of indices)

Claim: There is a 2-query LDCs with $m = \exp(n)$, correcting a $\Theta(1)$ fraction of type 1 deletions.

Proof: Hadamard code with some massaging.

Deletion Patterns and Queries (II)

Pattern type 2: fix $p < \delta$ and delete each bit with probability p.

$$x \rightarrow \ C(x) = 01100111110001111 \text{ , } p = \frac{1}{3}$$

$$\tilde{y} = \ 010011101111$$

Observation: Distribution induced by query $(k, k + d)$ corresponds to $(k', k' + d')$ where d' is concentrated around $d/(1-p)$; probability of any d' around mean

$\sim 1/\sqrt{d}$. (negative binomial distribution)

Pattern type 3: Pick p uniformly from $[\delta/8, \delta/4]$ and delete each bit of the codeword with probability p independently.

Our deletion distribution: type 1, then type 3.

Outcome: Flatter and better distribution

Properties of the Hard Error Distribution
(Towards a Lower Bound for 2-Query Insdel LDCs)

Lemma: For every (k, d), under our error distribution:

- $d' \in [d, 20d]$ w.p. $1 - \epsilon$ (concentration)
- For any $j \in [d, 20d]$, $\Pr[d' = j] = O(1/d)$ (anticoncentration/smoothness)

Recall Hitting Lemma: For every i, $Dec(i)$ must hit $Good_i$ with probability $> 3\,\epsilon/2$

Hitting lemma implies: ∃ “good query” (k, d) such that

$$\boldsymbol{Pr[(k', d') \in Good_i] > 3\,\epsilon/2}$$

Next: What does the induced distribution (k', d') look like?

Towards a Lower Bound for 2-Query Insdel LDCs

$$Good_i = \{(k', k' + d')|\exists\, f: \{0,1\} \to \{0,1\}^2 \ s.t.\, Pr_x\left[f\left(C(x)_{|Q}\right)\right] \geq \tfrac{1}{2} + \epsilon/4\}$$

For bit $i \in [n]$, let (k, d) be good for i.

What does the (support of) induced distribution (k', d') look like?

Split m into $t = \log_{20} m$ intervals $[20^{j-1}, 20^{j})$.

Let $P_j = [m] \times [20^{j-1}, 20^{j})$

Let $r = r(i)$ be such that $20^{r-1} \leq d < 20^{r}$.

Claim: $|Good_i \cap P_r| = \Omega(md)$

Towards a Lower Bound for 2-Query Insdel LDCs

$Good_i = \{(k', k' + d')| \exists\, f: \{0,1\}^n \to \{0,1\}\ s.t. Pr_x[f(C(x)_{|Q})] \geq ½ + \epsilon/4\}$

For bit $i \in [n]$, let (k, d) be good for i.

What does the (support of) induced distribution (k', d') look like?

Split m into $t = \log_{20} m$ intervals $[20^{j-1}, 20^j)$.

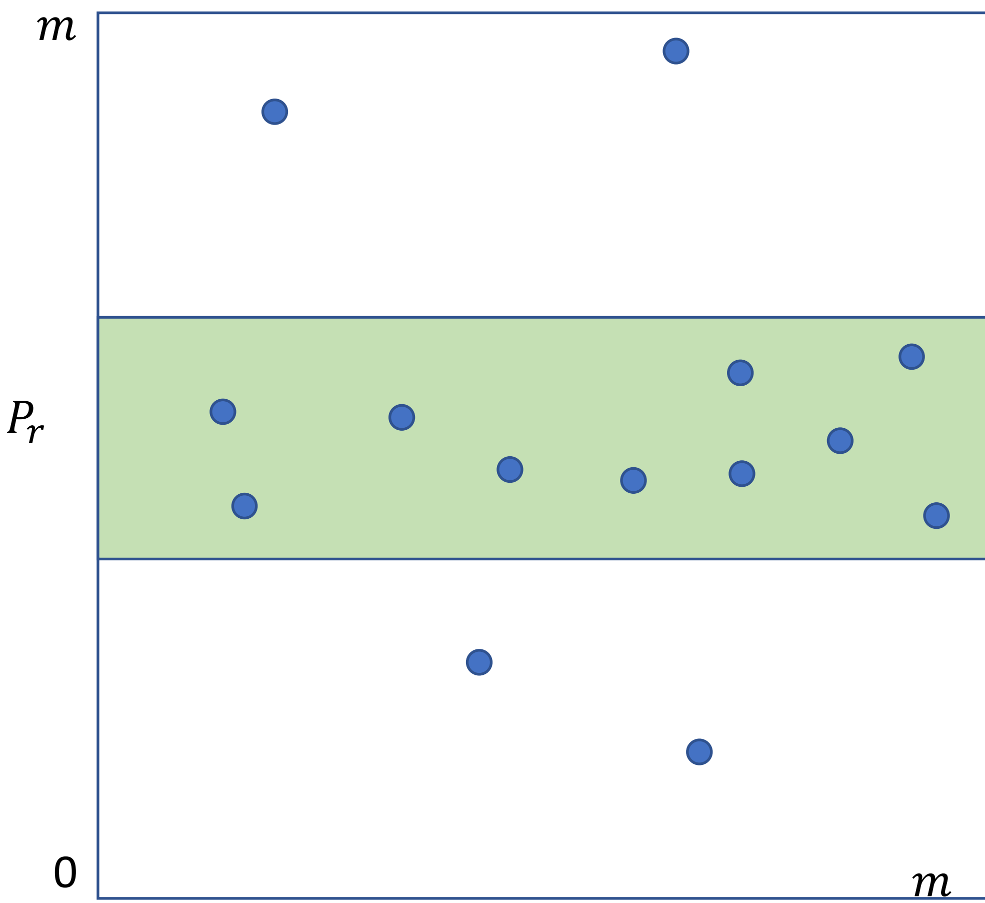
Let $P_j = [m] \times [20^{j-1}, 20^j)$

Let $r = r(i)$ be such that $20^{r-1} \leq d < 20^r$.

Claim: $|Good_i \cap\ P_r| = \Omega(md)$

Proof: Consequence of concentration, hitting lemma and smoothness.

Corollary: $\exists\, r$ s.t. $\frac{|Good_i \cap P_r|}{|P_r|} = \Omega(1)$



Wrapping up the Lower Bound

Theorem: Any 2-query Insdel LDC has $m = \exp(n)$

Bounding $\sum_{j=1}^{t} \sum_{i=1}^{n} \frac{|Good_i \cap P_j|}{|P_j|}$ from above and below we get:

$$\sum_{j=1}^{t} \sum_{i=1}^{n} \frac{|Good_i \cap P_j|}{|P_j|} \geq \sum_{i=1}^{n} \frac{|Good_i \cap P_r|}{|P_r|} \geq \Omega(n)$$

Since each (k, d) is good for $O(1)$ many i's then for every j: $\sum_{i=1}^{n} \frac{|Good_i \cap P_j|}{|P_j|} \leq O(1)$. So:

$$\sum_{j=1}^{t} \sum_{i=1}^{n} \frac{\left|Good_i \cap P_j\right|}{\left|P_j\right|} \leq O(t) = O(\log m)$$

Hence $m = \Omega(\exp(n))$. QED.

Generalizations

- Observation: The $m = \Omega(\exp(n))$ lower bound [Kerenidis-deWolf] for Hamming 2-query LDC does hold for insdel LDCs. Uses quantum arguments.

  Our proof is via classical arguments.

- Proof generalizes to non-existence of affine/linear 2-query insdel LDCs

- Proof generalizes to $q \geq 3$ queries

- Strategy: Need generalized smoothness properties of the deletion distributions

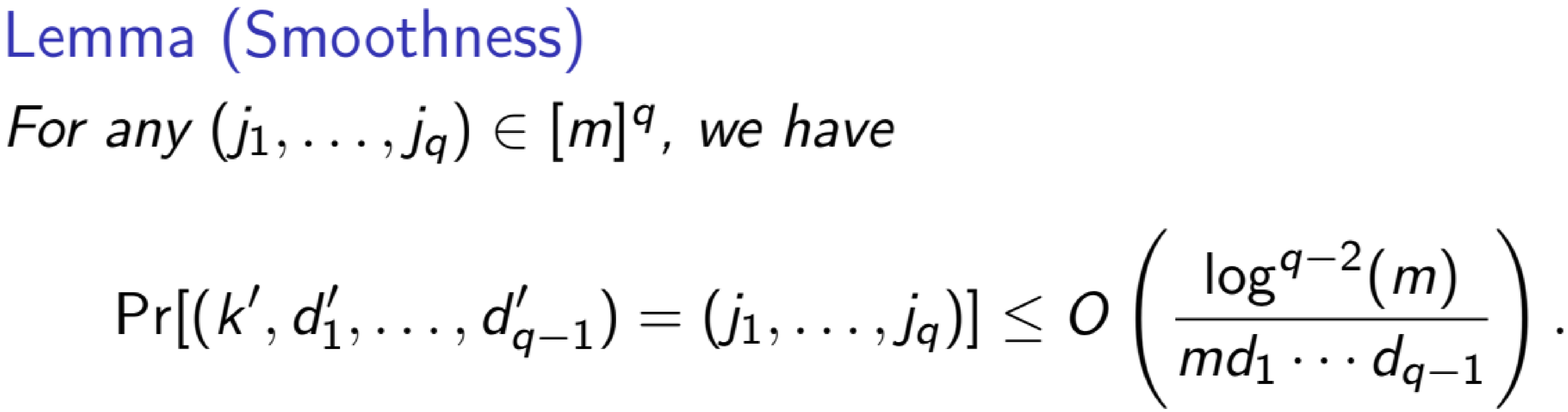
Nice Properties of the Deletion Process

View deletion pattern as set $\boldsymbol{S} \subseteq [\boldsymbol{m}]$.

1. **Boundedness:** $|S| \leq \delta m$ with probability $1 - o(1)$.
2. **Concentration:** For any interval $I \subseteq [m]$, $|S \cap I| \leq 0.9|I|$ with probability $\geq 1 - \epsilon$.
3. **Anti-concentration:** For any disjoint intervals $I_1, I_2, \ldots, I_k \subseteq [m]$, and any $0 \leq i_1 \leq |I_1|, \ldots, 0 \leq i_k \leq |I_k|$,

$$\Pr\left[|S \cap I_1| = i_1, \ldots, |S \cap I_k| = i_k\right] \leq \frac{B_{m,k}}{|I_1| \cdot |I_2| \cdots |I_k|}.$$

Generalizing, our Distribution Satisfies



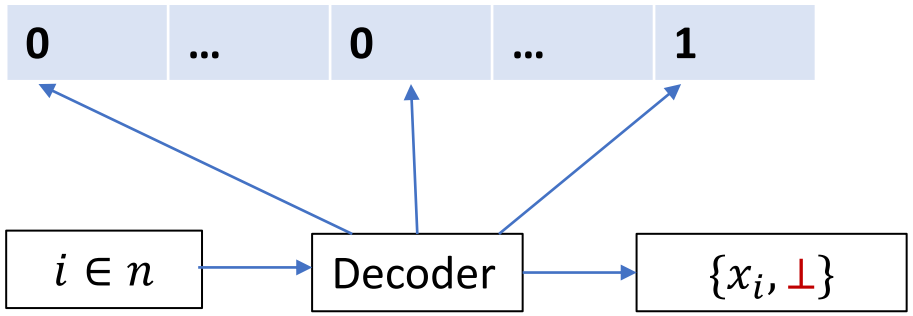
Lemma (Smoothness)

For any $(j_1, \ldots, j_q) \in [m]^q$, *we have*

$$\Pr[(k', d'_1, \ldots, d'_{q-1}) = (j_1, \ldots, j_q)] \leq O\left(\frac{\log^{q-2}(m)}{m d_1 \cdots d_{q-1}}\right).$$

Open question: what is the best upper bound possible?

Variant: Relaxed LDC



A code $C: \{0,1\}^n \rightarrow \{0,1\}^m$ is a Relaxed Locally Decodable Code (RLDC) if there exists a q-query decoder Dec such that
(1) $\forall x, i \;\; \Pr[Dec(C(x), m', i) = x_i] = 1$ (perfect completeness)

(2) $\forall x$ and y with $\text{dist}(C(x), y) \leq \delta m$ and for every i
$$\Pr[Dec(y, m', i) = \{x_i, \perp\}] \geq ½ + \epsilon$$

(3) $\forall x$ and y with $\text{dist}(C(x), y) \leq \delta m$ and $\forall i \in [n], \exists I \subseteq [n]$ with $|I| = \Omega(n)$ such that x_i is output with probability $½ + \epsilon$

Theorem [BenSassonGoldreichHarshaSudanVadhan06, AsadiShinkar21]
$\exists\, q = O(1)$ s.t. there exists q-query Hamming RLDC with $m = n^{1+\gamma}$.

Theorem [BGHSV06]
Condition (1)+(2) (i.e. ``weak'' RLDC) ⇒ Condition (3) (i.e.``strong'' RLDC) for $q = O(1)$ query, and $O(1)$ rate code

Theorem [GurLachish21, Dall'AgnonGurLachish21] Matching lower bounds.

Question [GurLachish21]: For $q = 2$ are there short (Hamming) RLDCs?

Our New Results [BBCGLZZ22]

Theorem: 2-query Hamming RLDC must have $m = \exp(n)$.

Corollary: Phase transition for some $q = O(1)$: m drops from Super-poly(n) at some q, to $poly(n)$ at $q + 1$.

Question: Where?

Question: Constructions/lower bounds for insdel RLDCs?

Other New Results

- ∃ weak $O(1)$-query insdel RLDC with $m = n^{1+\epsilon}$

  Hence, binary search is not inherent to weak insdel RLDCs.

- Strong insdel $O(1)$-query RLDC must have $m = \Omega(\exp(n))$

- Hence weak insdel RLDCs are not equivalent to strong RLDCs (unlike weak and strong Hamming RLDCs)

Final Open Problems about RLDCs and LDCs

- Exact phase-transition thresholds for Hamming/weak insdel RLDC.
- Lower bounds for Hamming 2-query RLDCs without perfect completeness?
- Is noisy binary search inherent to constructions of O(1)-queries insdel LDCs/strong RLDCs?
- Do O(1)-query insdel LDCs exist?
- Constructions of $o(\log^2 n)$-query Insdel LDCs?
- Applications of insdel local codes: analogies to PIR, PCPs, self-correction, fault-tolerant circuits, data structures, quantum computing?

Thanks for your attention!