A Quadratic Lower Bound for Three-Query Linear Locally Decodable Codes over Any Field

David P. Woodruff

IBM Almaden
dpwoodru@us.ibm.com

Abstract. A linear $(q, \delta, \epsilon, m(n))$-locally decodable code (LDC) $C : \mathbb{F}^n \to \mathbb{F}^{m(n)}$ is a linear transformation from the vector space $\mathbb{F}^n$ to the space $\mathbb{F}^{m(n)}$ for which each message symbol x_i can be recovered with probability at least $\frac{1}{|\mathbb{F}|} + \epsilon$ from $C(x)$ by a randomized algorithm that queries only q positions of $C(x)$, even if up to $\delta m(n)$ positions of $C(x)$ are corrupted. In a recent work of Dvir, the author shows that lower bounds for linear LDCs can imply lower bounds for arithmetic circuits. He suggests that proving lower bounds for LDCs over the complex or real field is a good starting point for approaching one of his conjectures. Our main result is an $m(n) = \Omega(n^2)$ lower bound for linear 3-query LDCs over any, possibly infinite, field. The constant in the $\Omega(\cdot)$ depends only on ε and δ. This is the first lower bound better than the trivial $m(n) = \Omega(n)$ for arbitrary fields and more than two queries.

Keywords: Error-Correcting Codes, Complexity Theory

1 Introduction

Classical error-correcting codes allow one to encode an n-bit message x into a codeword $C(x)$ such that even if a constant fraction of the bits in $C(x)$ are corrupted, x can still be recovered. It is known how to construct such codes of length $O(n)$ that can tolerate a constant fraction of errors, even in such a way that allows decoding in linear time [1]. However, if one is only interested in recovering a few bits of the message, then these codes have the disadvantage that they require reading most of the codeword.

A locally decodable code (LDC) $C : \mathbb{F}^n \to \mathbb{F}^{m(n)}$ is an encoding from the vector space $\mathbb{F}^n$ to the space $\mathbb{F}^{m(n)}$ such that each message symbol x_i can be recovered with probability at least $\frac{1}{|\mathbb{F}|} + \epsilon$ from $C(x)$ by a randomized algorithm that reads only q positions of $C(x)$, even if up to $\delta m(n)$ positions in $C(x)$ are corrupted (here $\frac{1}{|\mathbb{F}|}$ is zero if $\mathbb{F}$ is infinite). If C is a linear transformation, then the LDC is said to be linear. LDCs in their full generality were formally defined by Katz and Trevisan [2]. Linear LDCs were first considered in work by Goldreich et al [3]. There is a vast body of work on LDCs; we refer the reader to Trevisan's survey [4] or to Yekhanin's thesis [5].

While in general an LDC need not be linear, there is good motivation for studying this case. On the practical front, it is easy to encode a message and update a codeword given the generator matrix for a linear code. In applications of error-correcting codes to compressed sensing [6–8], the encoding is defined to be linear because of the physics of an optical lens. In large data streams, sketches are linear because they can be updated efficiently. On the theoretical front, lower bounds for linear 2-query LDCs are useful for polynomial identity testing [9]. These applications consider fields $\mathbb{F}$ of large or infinite size, e.g., in compressed sensing and streaming one has $\mathbb{F} = \mathbb{R}$.

In a surprising recent development, Dvir [10] shows that lower bounds for linear locally self-correctable codes and linear locally decodable codes imply lower bounds on the rigidity of a matrix, which in turn imply size/depth tradeoffs for arithmetic circuits [11]. In Section 5.1 of [10], the author suggests that proving lower bounds on linear locally correctable or linear locally decodable codes over the complex or real field is a good starting point for approaching one of his conjectures.

1.1 Results

Our main result is that for any (possibly infinite) field $\mathbb{F}$, any 3-query linear LDC requires $m(n) = \Omega(n^2)$, where the constant in the $\Omega(\cdot)$ notation depends only on ε and δ.

The first reason previous work does not give a non-trivial lower bound over arbitrary fields is that it uses a generic reduction from an adaptive decoder to a non-adaptive decoder, which effectively reduces ε to $\varepsilon/|\mathbb{F}|^{q-1}$. For constant q, if $\mathbb{F}$ is of polynomial size, one cannot beat the trivial $m(n) = \Omega(n)$ bound this way. We give a better reduction to a non-adaptive decoder.

Given our reduction, it then seems possible to obtain a field-independent $\Omega(n^{3/2})$ bound by turning the birthday paradox argument of Katz and Trevisan [2] into a rank argument. This is still weaker than our bound by a factor of $\sqrt{n}$.

Also, by using a technique of Kerenidis and de Wolf [12], it seems possible to obtain a bound of $\Omega(n^2/(|\mathbb{F}|^2 \log^2 n))$. This bound becomes trivial when $\mathbb{F} = \mathbb{R}$ or $\mathbb{F} = \mathbb{C}$, or even $|\mathbb{F}| = \mathrm{poly}(n)$. Note that if taking $|\mathbb{F}| = \mathrm{poly}(n)$ were to imply 3-query linear LDCs of linear size, then the encoding would need only a linear number of machine words. Our result rules out this possibility.

While the parameters of the LDCs considered by Dvir [10] over $\mathbb{R}$ or $\mathbb{C}$ are in a different regime than those considered here, e.g., he needs a bound for $q = \log^{2+\Omega(1)}(n)$ queries, our result provides the first progress on this problem for LDCs for more than two queries. We note that our results are not possible for non-linear codes, as one can encode n real numbers into a single real number.

An earlier technical report [13] by the author contains some of the ideas used here. That version of this paper has a weaker $m(n) = \Omega(n^2/\log\log n)$ bound for 3-query linear LDCs over any field. It also shows an $\Omega(n^2/\log n)$ bound for non-linear 3-query LDCs over $\mathbb{F}_2$ using a similar argument to that given here in Section 3.1. It contains polylogarithmic improvements over [12] for any odd

$q \geq 3$ number of queries. We do not know if for constant-sized fields, an $\Omega(n^2)$ bound holds for non-linear codes.

1.2 Techniques

In this section we give an overview of the techniques we use for our lower bound.

Let $C : \mathbb{F}^n \to \mathbb{F}^m$ be a linear 3-query LDC. Then each of its output coordinates $C_i(x)$ equals $\langle v_i, x \rangle$, for a vector $v_i \in \mathbb{F}^n$. As observed by Katz and Trevisan [2] for finite fields, since C can tolerate a large fraction of errors and is locally decodable, for each $i \in [n] \overset{\text{def}}{=} \{1, 2, \ldots, n\}$, there is a large matching (i.e., collection of disjoint sets) M_i of triples $\{v_a, v_b, v_c\}$ for which u_i, the i-th standard unit vector, is in $\text{span}\{v_a, v_b, v_c\}$. We generalize this to infinite fields, which requires some care since the matching sizes of Katz and Trevisan (and subsequent work of [3] and [12]) degrade with the field size for general adaptive decoders. For constant ε and δ (the setting we consider here), we show that for any field, $|M_i| = \Omega(m)$.

Given the matchings, we work in the 3-uniform multi-hypergraph G on vertex set $\{v_1, \ldots, v_m\}$ whose 3-edgeset is $\cup_{j=1}^n M_j$. The average degree of a vertex in G is $\Omega(n)$, and by standard arguments (iteratively remove the minimum degree vertex in the hypergraph and stop once the minimum degree is larger than the original average degree), we can find an induced sub-multi-hypergraph G' with minimum degree βn for a constant $\beta > 0$. In particular, it is easy to show that we can find a set T of αn linearly independent vertices of G' collectively incident to $\Omega(n^2)$ distinct 3-edges, where α is a constant satisfying $0 < \alpha < \beta$.

We now provide a new way to project 3-query LDCs down to 2-query LDCs. Suppose we extend T to a basis $T \cup U$ of $\mathbb{F}^n$ by greedily adding a set U of standard unit vectors. Consider the linear projection P for which T is in the kernel, but P restricted to U is the identity map. Suppose we apply P to every vertex in G'. Let $N(T)$ denote the set of vertices incident to T via a 3-edge $\{a, b, c\}$ in G', i.e., the neighborhood of T. Suppose $\{a, b, c\} \in M_i$. The key point is that after application of P, either the projection of a, b, or c is equal to 0, since one of these vertices is in the kernel of P. But if $u_i \in U$, then $P(u_i) = u_i$. Hence, either $u_i \in \text{span}(P(a), P(b))$, $u_i \in \text{span}(P(a), P(c))$, or $u_i \in \text{span}(P(b), P(c))$. We can thus obtain large matchings of edges (as opposed to 3-edges), for which a standard unit vector is in the span of the endpoints. Notice that since $|U| \geq n - \alpha n$, whereas the minimum degree of each vertex in T is $\beta n > \alpha n$, each vertex is still incident to at least $(\beta - \alpha)n$ edges for different $i \in U$, which is already enough to prove an $\Omega(n^2 / \log n)$ lower bound by now resorting to known techniques for lower bounding 2-query LDCs [9].

The next and harder part is improving the bound to a clean $\Omega(n^2)$. Our lower bound comes from bounding the cardinality of the neighborhood $N(T)$ of T. Suppose this cardinality really were $\Theta(n^2 / \log n)$. Then there are $\Omega(n^2)$ hyperedges from T to its neighborhood. This means that the average degree of a vertex in $N(T)$ using the edges from T to $N(T)$ is $\Omega(\log n)$. By standard arguments we can find a set A of $\alpha' n$ vertices in $N(T)$ incident to a set B of

$\Omega(n \log n)$ vertices in $N(T)$ via the edges from T to $N(T)$. Now if we augment the kernel of our projection to additionally include the vertices in A, as well as more standard unit vectors, we can put most of B into the kernel of our projection. We could not do this a priori, since putting a set B of more than n vertices in the kernel of a projection could make the projection equal to zero. Here, though, it is important that a constant fraction of standard unit vectors are preserved under projection.

We assumed that $N(T) = \Theta(n^2/\log n)$, when it could have been anywhere from $\omega(n^2/\log n)$ to $o(n^2)$. However, we can iteratively apply the above procedure, gradually enlarging the kernel while preserving a large number of standard unit vectors under projection. After $O(\log \log n)$ iterations, we show that the neighborhood of our resulting kernel has size $\Omega(n^2 \log n)$. We can then use lower bound techniques developed in the 2-query setting to deduce that $m = \Omega(n^2)$.

1.3 Related Work

Katz and Trevisan [2] show that 1-query LDCs do not exist.

For linear 2-query LDCs, Dvir and Shpilka [9] show that $m(n) \geq \exp(n)$ for[1] any field $\mathbb{F}$, and the Hadamard code shows this is optimal (see also [3], [14], [15]). We note that for non-linear 2-query LDCs, if the field $\mathbb{F}$ has constant size, then $m(n) \geq \exp(n)$ is also known to hold [12].

For more than 2 queries, there is a large gap between upper and lower bounds. This may, in part, be explained by the recent connections of Dvir [10]. The upper bounds for q-query LDCs are linear and have the form $m(n) = \exp(\exp(\log^{c/\log q} n \log^{1-c/\log q} \log n))$ for an absolute constant $c > 0$ ([16], [17], [18]). While the initial constructions were over finite fields, recently it was shown that similar upper bounds hold also over the real or complex numbers ([19], [20]).

The lower bounds are the aforementioned bounds of Katz and Trevisan [2] and of Kerenidis and de Wolf [12].

2 Preliminaries

Definition 2.1. *([2]) Let $\delta, \epsilon \in (0,1)$, q an integer, and $\mathbb{F}$ a field. A linear transformation $C : \mathbb{F}^n \to \mathbb{F}^m$ is a linear (q, δ, ϵ)-*locally decodable code *(LDC for short) if there is a probabilistic oracle machine A such that:*

- *For every $x \in \mathbb{F}^n$, for every $y \in \mathbb{F}^m$ with $\Delta(y, C(x)) \leq \delta m$, and for every $i \in [n]$, $\Pr[A^y(i) = x_i] \geq \frac{1}{|\mathbb{F}|} + \epsilon$, where the probability is taken over the internal coin tosses of A. Here $\Delta(C(x), y)$ refers to the number of positions in $C(x)$ and y that differ.*
- *In every invocation, A makes at most q queries (possibly adaptively).*

In Section 4, we prove the following.

[1] Here $\exp(n)$ denotes $2^{\Theta(n)}$.

Theorem 2.1. *Let $C : \mathbb{F}^n \to \mathbb{F}^m$ be a linear $(3, \delta, \epsilon)$-LDC. Then C is also a linear $(3, \delta/9, 2/3 - 1/|\mathbb{F}|)$-LDC with a non-adaptive decoder.*

This improves known reductions to non-adaptive codes since it holds for any $\mathbb{F}$. Thus, we may assume that we have a non-adaptive decoder by changing δ and ϵ by constant factors.

By known results described in Appendix A, for every $i \in [n]$ there is a matching M_i of $\{v_1, \ldots, v_m\}$ of size $\Omega(m)$ (where the constant depends on ε, δ, and q) such that, if $e \in M_i$, then $u_i \in \text{span}(v \mid v \in e)$, where u_i denotes the unit vector in direction i. Consider the multi-hypergraph G with vertex set $\{v_1, \ldots, v_m\}$ and hyperedge set $\uplus_{i=1}^{n} M_i$, that is, a hyperedge e occurs in G once for each M_i that it occurs in. For readability, we use the term hypergraph to refer to a multi-hypergraph, that is, a hypergraph which may have repeated hyperedges (which we sometimes just refer to as edges).

In Appendix A, we show there is a non-empty hypergraph $G' \subseteq G$ with minimum degree βn, where β is such that the number of hyperedges in G is at least βmn.

3 Lower Bounds for 3-Queries over Any Field

3.1 The basic projection

Assume we have a linear $(3, \delta, \epsilon)$-LDC $C : \mathbb{F}^n \to \mathbb{F}^m$ for an arbitrary (possibly infinite) field $\mathbb{F}$. Throughout this section we shall use the term edge to denote a 3-edge (i.e., there are 3 endpoints) for ease of notation.

Let G be the hypergraph on vertex set $\{v_1, \ldots, v_m\}$ and G' the non-empty sub-hypergraph of G with minimum degree βn defined in Section 2. Let v be an arbitrary vertex in G', and let $T = \{v\} \cup N(v)$, where $N(v)$ denotes the set of neighbors of v in G' (i.e., the vertices in a 3-edge containing v). Remove vertices from T so that we are left with a set T of exactly αn linearly independent vectors, where $\alpha < \beta$ is a small enough constant specified by the analysis below. This is always possible because $\{v\} \cup N(v)$ spans βn linearly independent vectors.

We may assume, by increasing m by a factor of at most 3, that every edge in M_i has size exactly 3, and moreover, for every such edge $\{v_{j_1}, v_{j_2}, v_{j_3}\} \in M_i$, we have $u_i = \gamma_1 v_{j_1} + \gamma_2 v_{j_2} + \gamma_3 v_{j_3}$, where $\gamma_1, \gamma_2, \gamma_3$ are non-zero elements of $\mathbb{F}$. Indeed, we may append $2m$ constant functions which always output 0 to the end of C. Then, if an edge in M_i either has size less than 3 or has size 3 and has the form $\{v_{j_1}, v_{j_2}, v_{j_3}\}$, but satisfies $u_i = \gamma_1 v_{j_1} + \gamma_2 v_{j_2} + \gamma_3 v_{j_3}$ for some $\gamma_k = 0$, we can replace the γ_k with 1 and replace j_k with an index corresponding to one of the zero functions.

Let $v_1, \ldots, v_T$ denote the vectors in T. Extend $\{v_1, \ldots, v_T\}$ to a basis of $\mathbb{F}^n$ by adding a set U of $n - \alpha n$ standard unit vectors. Define a linear projection L as follows:

$$L(v) = 0 \text{ for all } v \in T \text{ and } L(v) = v \text{ for all } v \in U.$$

Since L is specified on a basis, it is specified on all of $\mathbb{F}^n$.

Let M_i' denote the collection of edges in M_i that are incident to some vertex in T. Let $e = \{v_{j_1}, v_{j_2}, v_{j_3}\}$ be an edge in some M_i'. Then there are non-zero $\gamma_1, \gamma_2, \gamma_3 \in \mathbb{F}$ for which $\gamma_1 v_{j_1} + \gamma_2 v_{j_2} + \gamma_3 v_{j_3} = u_i$. By linearity, $L(u_i) = L(\gamma_1 v_{j_1} + \gamma_2 v_{j_2} + \gamma_3 v_{j_3}) = \gamma_1 L(v_{j_1}) + \gamma_2 L(v_{j_2}) + \gamma_3 L(v_{j_3})$. By definition of M_i', $|\{v_{j_1}, v_{j_2}, v_{j_3}\} \cap T| > 0$, so one of the following must be true: $L(u_i) \in \text{span}(L(v_{j_1}), L(v_{j_2}))$, $L(u_i) \in \text{span}(L(v_{j_1}), L(v_{j_3}))$, or $L(u_i) \in \text{span}(L(v_{j_2}), L(v_{j_3}))$.

Thus, for each such edge $e = \{v_{j_1}, v_{j_2}, v_{j_3}\}$, by removing exactly one vector $v_{j_\ell} \in \{v_{j_1}, v_{j_2}, v_{j_3}\}$ for which $L(v_{j_\ell}) = 0$, we may define matchings W_i of disjoint pairs $\{v_j, v_k\}$ of $\{v_1, \ldots, v_m\}$ such that if $\{v_j, v_k\} \in W_i$, then $L(u_i) \in \text{span}(L(v_j), L(v_k))$. Moreover, $\sum_{i=1}^n |W_i| = \sum_{i=1}^n |M_i'|$.

Say an index $i \in [n]$ *survives* if $L(u_i) = u_i$, and say an edge e *survives* if $e \in M_i'$ for an i that survives. If i survives, then $u_i \in U$, as otherwise we would have $u_i = \sum_{v \in T} \gamma_v v + \sum_{u \in U} \gamma_u u$ for some coefficients $\gamma_v, \gamma_u \in \mathbb{F}$. Applying L to both sides we would obtain $u_i = L(u_i) = \sum_{u \in U} \gamma_u L(u) = \sum_{u \in U} \gamma_u u$, which is impossible unless $u_i \in U$.

Recall that each of the αn vertices v in T has degree at least βn in G'. For any such $v \in T$, there are at least $\beta n - \alpha n$ edges e in the disjoint union of the M_i' for the i the survive. Thus, since each edge that survives can be incident to at most 3 elements of T, and since $\alpha < \beta$,

$$\sum_{i \text{ that survive}} |W_i| \geq \alpha n(\beta - \alpha)n/3 = \Omega(n^2).$$

For i that do not survive, we set $W_i = \emptyset$. We need a theorem due to Dvir and Shpilka [9].

Theorem 3.1. *([9]) Let $\mathbb{F}$ be any field, and let $a_1, \ldots, a_m \in \mathbb{F}^n$. For every $i \in [n]$, let M_i be a set of disjoint pairs $\{a_{j_1}, a_{j_2}\}$ such that $u_i \in span(a_{j_1}, a_{j_2})$. Then, $\sum_{i=1}^n |M_i| \leq m \log m + m$.*

Applying Theorem 3.1 to our setting, we have m vectors $L(v_j) \in \mathbb{F}^n$ and matchings W_i with $\sum_i |W_i| = \Omega(n^2)$. We conclude that,

Theorem 3.2. *For $\delta, \epsilon \in (0,1)$, if $C : \mathbb{F}^n \to \mathbb{F}^m$ is a linear $(3, \delta, \epsilon)$-locally decodable code, then $m = \Omega_{\delta,\epsilon}(n^2/\log n)$, independent of the field $\mathbb{F}$.*

3.2 Recursing to get the $\Omega(n^2)$ bound

We assume that $\beta > 2\alpha$ and w.l.o.g., that $(\beta - 2\alpha)n$ is a power of 2 and αn is an integer. For a set $A \subseteq \mathbb{F}^n$, let $ex(A)$ denote a maximal linearly independent subset of A.

Base Case: As before, let G' be the hypergraph on 3-edges with minimum degree βn, and let $T_1 = T$ be the set of αn linearly independent vertices defined in Section 3.1. We extend T_1 to a basis of $\mathbb{F}^n$ by greedily adding a set U of $n - \alpha n$ standard unit vectors to T_1. Set $B_1 = U$. Since each vertex in T_1 has degree at least βn, since $|T_1| = \alpha n$, and since each matching edge can be counted at most

3 times, the set E of 3-edges labeled by a $u \in B_1$ and incident to T_1 has size at least $\alpha n(\beta - \alpha)n/3$.

For each $u \in B_1$, let f_u denote the number of edges in E labeled by u, i.e., in the matching M_u. Order the unit vectors so that $f_{u_1} \geq f_{u_2} \geq \cdots \geq f_{u_{|B_1|}}$, and let $E_1 \subset E$ be the subset of edges incident to T_1 labeled by a unit vector in the set U_1 of the first $\frac{(\beta-2\alpha)n}{2}$ unit vectors. Set $V_1 = T_1$.

Inductive Step: We construct sets T_i, B_i, U_i, E_i, and V_i, $i \geq 2$, as follows. The proof works provided i satisfies $i \leq \min(\lfloor \log_2(\alpha n/2^{i-1}) \rfloor, \log_2(\beta - 2\alpha)n)$, which holds for $i = O(\log n)$. The intuition for the sets is as follows:
- T_i is the set of vertices that are projected to zero by the i-th projection L_i that we construct.
- B_i is a maximal set of standard unit vectors that have not been projected to zero by the projection L_i that we construct.
- U_i is a subset of B_i of the most frequent standard unit vectors, that is, many of the 3-edges incident to a vertex in T_i are labeled by a vector in U_i.
- E_i is a subset of 3-edges incident to T_i that are labeled by a vector in U_i.
- V_i is a small set of vertices that when projected to zero, project T_i to zero.

Let $N(T_{i-1})$ be the neighborhood of vertices of T_{i-1}, that are not themselves in T_{i-1} (so $N(T_{i-1})$ and T_{i-1} are disjoint). We define a multigraph G_{i-1} on vertex set $N(T_{i-1})$ where we connect two vertices by a 2-edge if and only if they are included in a 3-edge in E_{i-1}. Let $r[i-1]$ be the number of connected components of G_{i-1}. Let $C_{i-1,1}, \ldots, C_{i-1,r[i-1]}$ be the connected components of G_{i-1}, where $|C_{i-1,1}| \geq |C_{i-1,2}| \geq \cdots \geq |C_{i-1,r[i-1]}|$. For each connected component $C_{i-1,j}$, arbitrarily choose a vertex $v_{i-1,j} \in C_{i-1,j}$.

Let $T_i = \cup_{j=1}^{\lfloor \alpha n/2^{i-1} \rfloor} C_{i-1,j}$, where $C_{i-1,j} = \emptyset$ if $j > r[i-1]$, and let

$$V_i = V_{i-1} \cup \{v_{i-1,1}, \ldots, v_{i-1,\lfloor \alpha n/2^{i-1} \rfloor}\} \quad \text{(recall that } V_1 = T_1).$$

Extend $ex(V_i \cup (\cup_{j=1}^{i-1} U_j))$ to a basis of $\mathbb{F}^n$ by greedily adding a subset B_i of unit vectors in B_{i-1}. Let E be the set of 3-edges incident to some vertex in T_i, labeled by a $u \in B_i$. We will inductively have that $|U_j| = (\beta - 2\alpha)n/2^j$ for all $j \leq i-1$. Notice that this holds for our above definition of U_1. Notice that

$$\begin{aligned}
|B_i| &\geq n - |V_i| - |\cup_{j=1}^{i-1} U_j| \geq n - \sum_{j=1}^{i} \left\lfloor \frac{\alpha n}{2^{j-1}} \right\rfloor - \sum_{j=1}^{i-1} \frac{(\beta - 2\alpha)n}{2^j} \\
&\geq n - \alpha n - \sum_{j=1}^{i-1} \frac{\alpha n}{2^j} - \sum_{j=1}^{i-1} \frac{(\beta - 2\alpha)n}{2^j} \\
&= n - \alpha n - \sum_{j=1}^{i-1} \frac{\beta n - \alpha n}{2^j} \\
&= n - \alpha n - \beta n + \alpha n + \frac{(\beta - \alpha)n}{2^{i-1}} \\
&= n - \beta n + \frac{(\beta - \alpha)n}{2^{i-1}}
\end{aligned}$$

Each vertex in T_i has degree at least βn, since all vertices in G' have degree at least βn. It follows that each vertex in T_i is incident to at least $\beta n - (n - |B_i|) \geq \frac{(\beta-\alpha)n}{2^{i-1}}$ edges in E, since a vertex cannot be incident to two different edges of the same label. Since an edge can be counted at most 3 times, $|E| \geq |T_i| \cdot \frac{(\beta-\alpha)n}{3 \cdot 2^{i-1}}$. For each $u \in B_i$, let f_u denote the number of edges in E labeled by u, and order the unit vectors so $f_{u_1} \geq \cdots \geq f_{u_{|B_i|}}$. Let $E_i \subset E$ be the subset of edges incident to T_i labeled by a unit vector in the set U_i of the first $\frac{(\beta-2\alpha)n}{2^i}$ unit vectors. Notice that our earlier assumption that $|U_j| = (\beta - 2\alpha)n/2^j$ for all $j \leq i - 1$ holds by this definition of U_i.

Recursive projection: $|T_1| = \alpha n$, and for $i > 1$, $|T_i| = \sum_{j=1}^{\lfloor \alpha n/2^{i-1} \rfloor} |C_{i-1,j}|$. Also, for all $i \geq 1$, $|U_i| = (\beta - 2\alpha)n/2^i$. We turn to bounding $|E_i|$. Since we chose the $(\beta - 2\alpha)n/2^i$ most frequent unit vectors (in terms of the number of their occurrences in E) to include in the set U_i, and since E_i is the set of edges in E labeled by a unit vector in U_i, we have that $|E_i|$ must be at least a $(\beta - 2\alpha)/2^i$ fraction of $|E|$ (there are only n possible unit vectors). That is, we have

$$|E_i| \geq \frac{(\beta - 2\alpha)}{2^i} \cdot |E| \geq \frac{(\beta - 2\alpha)}{2^i} \cdot |T_i| \cdot \frac{(\beta - \alpha)n}{3 \cdot 2^{i-1}} = \left[\frac{2(\beta - 2\alpha)(\beta - \alpha)}{3}\right] \cdot \frac{|T_i|n}{4^i}.$$

We define a sequence of linear projections L_i for $i \geq 1$ as follows. We set $L_i(ex(V_i \cup (\cup_{j=1}^{i-1} U_j))) = 0$, and $L_i(u) = u$ for all $u \in B_i$.

Claim. For any $i \geq 2$, if $j \leq \lfloor \alpha n/2^{i-1} \rfloor$, then all vertices $b \in C_{i-1,j}$ satisfy $L_i(b) = 0$.

Proof. We prove this by induction on $i \geq 2$. For the base case $i = 2$, consider any vertex b in $C_{1,j}$, and let $v_{1,j} = a_0, a_1, a_2, \ldots, a_k = b$ be a path from $v_{1,j}$ to b in $C_{1,j}$. Since $\{a_0, a_1\}$ is an edge in $C_{1,j}$, we have $a_0, a_1 \in N(T_1)$ and so there is a 3-edge $e = \{w, a_0, a_1\} \in E_1$ with $w \in T_1$ and labeled by a $u_j \in U_1$. But then $L_2(w) = 0$ since $w \in T_1 = V_1$. Moreover, $L_2(u_j) = 0$ since $u_j \in U_1$. But, for non-zero $\gamma_1, \gamma_2, \gamma_3 \in \mathbb{F}$, $\gamma_1 w + \gamma_2 a_0 + \gamma_3 a_1 = u_j$. These conditions imply that $\gamma_2 L_2(a_0) + \gamma_3 L_2(a_1) = 0$. Now, notice that $v_{1,j} \in V_2$ since $j \leq \lfloor \alpha n/2^{i-1} \rfloor$, and so $L_2(v_{1,j}) = L_2(a_0) = 0$. It follows that $L_2(a_1) = 0$. By repeated application on the path from $v_{1,j}$ to $a_k = b$, we get $L_2(b) = 0$.

Inductively, suppose it is true for all values from 2 up to $i-1$. We prove it for i. Consider any vertex b in $C_{i-1,j}$ and let $v_{1,j} = a_0, a_1, \ldots, a_k = b$ be a path from $v_{1,j}$ to b in $C_{i-1,j}$. Since $\{a_0, a_1\}$ is an edge in $C_{i-1,j}$, we have $a_0, a_1 \in N(T_{i-1})$ and so there is a 3-edge $e = \{w, a_0, a_1\} \in E_{i-1}$ with $w \in T_{i-1}$ and labeled by a $u_j \in U_{i-1}$. But then $L_i(w) = 0$ since $w \in T_{i-1}$ and so $w \in C_{i-2,j}$ for some $j \leq \lfloor \alpha n/2^{i-2} \rfloor$, which by the inductive hypothesis means $L_{i-1}(w) = 0$, and the kernel of L_{i-1} is contained in the kernel of L_i. Now also $L_i(u_j) = 0$ since $u_j \in U_{i-1}$. For non-zero $\gamma_1, \gamma_2, \gamma_3 \in \mathbb{F}$, we have $\gamma_1 w + \gamma_2 a_0 + \gamma_3 a_1 = u_j$, and so $\gamma_2 L_i(a_0) + \gamma_3 L_i(a_1) = 0$. Notice that $v_{1,j} \in V_i$ since $j \leq \lfloor \alpha n/2^{i-1} \rfloor$, and so $L_i(v_{1,j}) = L_i(a_0) = 0$. Hence, $L_i(a_1) = 0$, and by repeated application on the path from $v_{1,j}$ to $a_k = b$, we get $L_i(b) = 0$. This completes the induction.

For each component $C_{i-1,j}$ for any i and j, let $c_{i-1,j}$ denote $|C_{i-1,j}|$ for notational convenience.

Lemma 3.1. *For any $i \geq 2$, if $j \leq \lfloor \alpha n/2^{i-1} \rfloor$, then the number of edges in $C_{i-1,j}$ is at most $c_{i-1,j} \log c_{i-1,j} + c_{i-1,j}$.*

Proof. Let $\{a, b\}$ be an edge in $C_{i-1,j}$. Then there is an edge $e = \{a, b, c\} \in E_{i-1}$ with $c \in T_{i-1}$. Then $\gamma_1 a + \gamma_2 b + \gamma_3 c = u_k$ for some $u_k \in U_{i-1}$, for non-zero $\gamma_1, \gamma_2, \gamma_3$ in $\mathbb{F}$. Since $e \in E_{i-1}$, we have $u_k \in U_{i-1} \subseteq B_{i-1}$, and so we have $L_{i-1}(u_k) = u_k$. Now, $c \in T_{i-1}$, and by Claim 3.2, L_{i-1} vanishes on all of T_{i-1} In particular, $L_{i-1}(c) = 0$. By linearity, $\gamma_1 L_{i-1}(a) + \gamma_2 L_{i-1}(b) = u_k$. Moreover, for each $k' \in [n]$, each vertex in $C_{i-1,j}$ can occur in at most one 3-edge labeled by $u_{k'}$ (by definition of the matchings in G'), so we obtain matchings $W_{k'}$, where an edge $\{a, b\}$ in $C_{i-1,j}$ is in $W_{k'}$ iff there is an $e \in E_{i-1}$ labeled by $u_{k'}$. By Theorem 3.1, $\sum_{k'} |W_{k'}| \leq c_{i-1,j} \log c_{i-1,j} + c_{i-1,j}$. But the number of edges in $C_{i-1,j}$ is at most the sum of matching sizes $|W_{k'}|$ for $u_{k'} \in U_{i-1}$.

Define the constant $\gamma = 2(\beta - 2\alpha)(\beta - \alpha)/3$. It follows that for all i, we have the constraints

1. $\frac{\gamma |T_{i-1}| n}{4^{i-1}} \leq |E_{i-1}| \leq \sum_{j=1}^{r[i-1]} (c_{i-1,j} \log c_{i-1,j} + c_{i-1,j})$
2. $|T_i| = \sum_{j=1}^{\lfloor \alpha n/2^{i-1} \rfloor} c_{i-1,j}$

Lemma 3.2. *Suppose for $i = 1, 2, \ldots, \Theta(\log \log n)$, we have $|T_i| > 8|T_{i-1}|$. Then $m = \Omega(n^2)$.*

Proof. By induction, $|T_i| > 8^{i-1}|T_1| = 8^{i-1}\alpha n$ for $i = 1, 2, \ldots, \Theta(\log \log n)$. We thus have,

$$|E_i| \geq \gamma \cdot \frac{|T_i| n}{4^i} \geq \frac{\gamma \alpha}{8} \cdot 2^i n^2.$$

Hence, for $i = \Theta(\log \log n)$, we have $|E_{i-1}| = \Omega(n^2 \log n)$. Using that $\Omega(n^2 \log n) = |E_{i-1}| \leq \sum_{j=1}^{r[i-1]} (c_{i-1,j} \log c_{i-1,j} + c_{i-1,j})$, we have

$$m \geq \sum_{j=1}^{r[i-1]} c_{i-1,j} = \Omega(n^2 \log n / \log n) = \Omega(n^2),$$

where we have used that $c_{i-1,j} \leq n^2$ for all i and j, as otherwise $m \geq c_{i-1,j} = n^2$ for some i and j, and we would already be done. Hence, we can use $\log c_{i-1,j} = O(\log n)$.

Lemma 3.3. *Suppose for a value $i = O(\log \log n)$, $c_{i-1,1} = \Omega(n^2/\log n)$. Then $m = \Omega(n^2)$.*

Proof. Notice that $|T_i| \geq c_{i-1,1} = \Omega(n^2/\log n)$, and also, $|E_i| = \Omega(|T_i| n/4^i) = \Omega(n^3/\text{polylog}(n)) = \Omega(n^2 \log n)$. Using the constraint that $m \geq \sum_{j=1}^{r[i-1]} c_{i-1,j} = \Omega(|E_i|/\log n)$, it follows that $m = \Omega(n^2)$. Here we have again upper bounded $\log c_{i-1,j}$ by $O(\log n)$, justified as in the proof of Lemma 3.2.

Lemma 3.4. *Suppose for a value $i = O(\log \log n)$, $|T_i| \leq 8|T_{i-1}|$. Then $m = \Omega(n^2)$.*

Proof. Let i^* be the smallest integer i for which $|T_i| \leq 8|T_{i-1}|$. It follows that $|T_{i^*-1}| \geq 8^{i^*-2}|T_1| = 8^{i^*-2}\alpha n$. Note that $|E_{i^*-1}| = \Omega(|T_{i^*-1}|n/4^{i^*-1}) = \Omega(n^2 2^{i^*})$. We attempt to maximize the RHS of constraint 1 defined above, namely

$$\sum_{j=1}^{r[i^*-1]} (c_{i^*-1,j} \log c_{i^*-1,j} + c_{i^*-1,j}), \tag{1}$$

subject to a fixed value of $|T_{i^*}|$, where recall $|T_{i^*}| = \sum_{j=1}^{\lfloor \alpha n/2^{i^*-1} \rfloor} c_{i^*-1,j}$. We can assume that

$$c_{i^*-1,1} \geq c_{i^*-1,2} = c_{i^*-1,3} = \cdots = c_{i^*-1,\lfloor \alpha n/2^{i^*-1} \rfloor},$$

as otherwise we could increase $c_{i^*-1,1}$ while replacing the other values with $c_{i^*-1,\lfloor \alpha n/2^{i^*-1} \rfloor}$, which would preserve the value of $|T_{i^*}|$ and only make constraint 1 defined above easier to satisfy (notice that since $|T_{i^*}|$ is fixed, the LHS of constraint 1 remains fixed, as well as both sides of constraint 2). Moreover, constraint 1 is only easier to satisfy if we make

$$c_{i^*-1,\lfloor \alpha n/2^{i^*-1} \rfloor} = c_{i^*-1,\lfloor \alpha n/2^{i^*-1} \rfloor+1} = \cdots = c_{i^*-1,r[i^*-1]}.$$

We can assume that $c_{i^*-1,1} = o(n^2/\log n)$, as otherwise Lemma 3.3 immediately shows that $m = \Omega(n^2)$. In this case, though, $c_{i^*-1,1}$ does not contribute asymptotically to sum (1) since $|E_{i^*-1}| = \Omega(n^2 2^{i^*})$ and so sum 1 must be at least this large. It follows that we can replace constraint 1 with

$$\Omega(|T_{i^*-1}|n/4^{i^*}) \leq rA(\log A + 1), \tag{2}$$

where A is the common value $c_{i^*-1,x}$, where $r = r[i^* - 1]$, and where $x \in \{2, \ldots, r\}$. Using that $i = O(\log \log n)$, so we can ignore the floor operation in constraint 2, constraint 2 becomes $An/2^{i^*} = \Theta(|T_{i^*}|)$, or equivalently, $A = \Theta(|T_{i^*}|2^{i^*}/n)$.

Using that $|T_{i^*}| \leq 8|T_{i^*-1}|$, it follows that $A = O(|T_{i^*-1}|2^{i^*}/n)$. Combining this with our reformulation of constraint 1 in (2), we have

$$r(\log A + 1) = \Omega(n^2/8^{i^*}),$$

or equivalently, $r = \Omega(n^2/(8^{i^*}(\log A + 1)))$. Now,

$$m = \Omega(Ar) = \Omega\left(\frac{n|T_{i^*-1}|}{4^{i^*}(\log(|T_{i^*-1}|2^{i^*}/n) + 1)}\right).$$

This is minimized when $|T_{i^*-1}|$ is as small as possible, but $|T_{i^*-1}| \geq 8^{i^*-2}\alpha n$. Hence, $m = \Omega\left(\frac{n^2 2^{i^*}}{\log 16^{i^*}}\right)$, which is minimized for $i^* = \Theta(1)$, in which case $m = \Omega(n^2)$, as desired.

Combining Lemma 3.2 and Lemma 3.4, we conclude,

Theorem 3.3. *For $\delta, \epsilon \in (0,1)$, if $C : \mathbb{F}^n \to \mathbb{F}^m$ is a linear $(3, \delta, \epsilon)$-locally decodable code, then $m = \Omega_{\delta,\epsilon}(n^2)$, independent of the field $\mathbb{F}$.*

4 From adaptive decoders to non-adaptive decoders

Theorem 4.1. *For given $\delta, \varepsilon \in (0,1)$, if $C : \mathbb{F}^n \to \mathbb{F}^m$ is a linear $(3, \delta, \epsilon)$-LDC, then C is a linear $(3, \delta/9, 2/3 - 1/|\mathbb{F}|)$-LDC with a non-adaptive decoder.*

Proof. Since C is a linear code, each of its coordinates can be identified with a vector $v_j \in \mathbb{F}^n$, with the function for that coordinate computing $\langle v_j, x\rangle$, where the inner product is over $\mathbb{F}$. Define the ordered list of vectors $B = v_1, \ldots, v_m$.

Fix some $i \in [n]$, and let $\mathcal{C}_i$ be the collection of all non-empty sets $S \subseteq \{v_1, \ldots, v_m\}$, with $|S| \leq 3$, for which $u_i \in \text{span}(v_j \mid v_j \in S)$, where u_i denotes the unit vector in direction i. Let $D_i \subseteq \{v_1, \ldots, v_m\}$ be a smallest dominating set of $\mathcal{C}_i$, that is, a set for which for all $S \in \mathcal{C}_i$, $|S \cap D_i| > 0$.

Claim. $|D_i| > \delta m$.

Proof. Suppose not. Consider the following adversarial strategy: given a codeword $C(x)$, replace all coordinates $C(x)_j$ for which $v_j \in D_i$ with 0. Denote the new string $\tilde{C}(x)$. The coordinates of $\tilde{C}(x)$ compute the functions $\langle \tilde{v}_j, x\rangle$, where $\tilde{v}_j = v_j$ if $v_j \notin D_i$, and $\tilde{v}_j = 0$ otherwise. Let $\tilde{B}$ be the ordered list of vectors $\tilde{v}_1, \ldots, \tilde{v}_m$.

Define 3-span$(\tilde{B})$ to be the (possibly infinite) list of all vectors in the span of each subset of $\tilde{B}$ of size at most 3. We claim that $u_i \notin$ 3-span$(\tilde{B})$. Indeed, if not, then let $S \subseteq \{\tilde{v}_1, \ldots, \tilde{v}_m\}$ be a smallest set for which $u_i \in \text{span}(S)$. Then $|S| \leq 3$. This is not possible if $|S| = 0$. It follows that $S \cap D_i \neq \emptyset$. This implies that 0 is a non-trivial linear combination of vectors in S. Indeed, there is an ℓ for which $\tilde{v}_\ell \in S$ and $v_\ell \in D_i$, implying $\tilde{v}_\ell = 0$. Hence, $u_i \in \text{span}(S \setminus \tilde{v}_\ell)$. But $|S \setminus \{\tilde{v}_\ell\}| < |S|$, which contradicts that S was smallest.

Let A be the decoder of C, where A computes $A^y(i, r)$ on input index $i \in [n]$ and random string r. Here, for any $x \in \mathbb{F}^n$, we let the string $y = y(x)$ be defined by the adversarial strategy given above. For any $x \in \mathbb{F}^n$, $A^y(i, r)$ first probes coordinate j_1 of y, learning the value $\langle \tilde{v}_{j_1}, x\rangle$. Next, depending on the answer it receives, it probes coordinate j_2, learning the value $\langle \tilde{v}_{j_2} x\rangle$. Finally, depending on the answer it receives, it probes coordinate j_3, learning the value $\langle \tilde{v}_{j_3} x\rangle$. Consider the affine subspace V of dimension $d \geq n - 2$ of all $x \in \mathbb{F}^n$ which cause $A^y(i, r)$ to read positions j_1, j_2, and j_3. Let V_0 be the affine subspace of V of all x for which $A^y(i, r)$ outputs x_i. Since the output of $A^y(i, r)$ is fixed given that it reads positions j_1, j_2, and j_3, and since $u_i \notin \text{span}(\tilde{v}_{j_1}, \tilde{v}_{j_2}, \tilde{v}_{j_3})$, it follows that the dimension of V_0 is at most $d - 1$.

Suppose first that $\mathbb{F}$ is a finite field. Then for any fixed r, the above implies $A^y(i, r)$ is correct on at most a $\frac{1}{|\mathbb{F}|}$ fraction of $x \in \mathbb{F}^n$ since $\frac{|V_0|}{|V|} \leq \frac{1}{|\mathbb{F}|}$ for any set of three indices j_1, j_2, and j_3 that A can read. Thus, by averaging, there exists an $x \in \mathbb{F}^n$ for which $\Pr[A^y(i) = x_i] \leq \frac{1}{|\mathbb{F}|}$, where the probability is over the random coins r of A. This contradicts the correctness of A.

Now suppose that $\mathbb{F}$ is an infinite field. We will show that there exists an $x \in \mathbb{F}^n$ for which $\Pr[A^y(i) = x_i] = 0$, contradicting the correctness of the decoder.

For each random string r, there is a finite non-empty set G_r of linear constraints over $\mathbb{F}$ that any $x \in \mathbb{F}^n$ must satisfy in order for $A^y(i, r) = x_i$. Consider the union $\cup_r G_r$ of all such linear constraints. Since the number of different r is finite, this union contains a finite number of linear constraints.

Since $\mathbb{F}$ is infinite, we claim that we can find an $x \in \mathbb{F}^n$ which violates all constraints in $\cup_r G_r$. We prove this by induction on n. If $n = 1$, then the constraints have the form $x_1 = c_1, x_1 = c_2, \ldots, x_1 = c_s$ for some finite s. Thus, by choosing $x_1 \notin \{c_1, c_2, \ldots, c_s\}$, we are done. Suppose, inductively, that our claim is true for $n-1$. Now consider $\mathbb{F}^n$. Consider all constraints in $\cup_r G_r$ that have the form $x_1 = c$ for some $c \in \mathbb{F}$. There are a finite number of such constraints, and we can just choose x_1 not to equal any of these values c, since $\mathbb{F}$ is infinite. Now, substituting this value of x_1 into the remaining constraints, we obtain constraints (each depending on at least one variable) on $n-1$ variables $x_2, \ldots, x_n$. By induction, we can choose the values to these $n-1$ variables so that all constraints are violated. Since we haven't changed x_1, the constraints of the form $x_1 = c$ are still violated. This completes the proof.

It follows that since $|D_i| > \delta m$ and D_i is a smallest dominating set of $\mathcal{C}_i$, we can greedily construct a matching M_i of $\delta m/3$ disjoint triples $\{v_{j_1}, v_{j_2}, v_{j_3}\}$ of $\{v_1, \ldots, v_m\}$ for which $u_i \in \text{span}(v_{j_1}, v_{j_2}, v_{j_3})$.

Consider the new behavior of the decoder: on input $i \in [n]$, choose a random triple $\{v_{j_1}, v_{j_2}, v_{j_3}\} \in M_i$, and compute x_i as $\gamma_1\langle v_{j_1}, x\rangle + \gamma_2\langle v_{j_2}, x\rangle + \gamma_3\langle v_{j_3}, x\rangle$, where $u_i = \gamma_1 v_{j_1} + \gamma_2 v_{j_2} + \gamma_3 v_{j_3}$. Since the adversary can now corrupt at most $\delta m/9$ positions, it follows that with probability at least $2/3$, the positions queried by the decoder are not corrupt and it outputs x_i. Note that the new decoder also makes at most 3 queries.

This can be extended straightforwardly to any constant $q > 3$ number of queries:

Theorem 4.2. *For given $\delta, \varepsilon \in (0, 1)$, if $C : \mathbb{F}^n \to \mathbb{F}^m$ is a linear (q, δ, ϵ)-LDC, then C is a linear $(q, \delta/(3q), 2/3 - 1/|\mathbb{F}|)$-LDC with a non-adaptive decoder.*

Acknowledgment: The author thanks Johnny Chen, Anna Gal, Piotr Indyk, Swastik Kopparty, Alex Samorodnitsky, C. Seshadhri, Ronald de Wolf, Sergey Yekhanin, and the anonymous referees for many helpful comments.

References

1. Sipser, M., Spielman, D.A.: Expander codes. IEEE Trans. Inform. Theory, 42:1710-1722 (1996)
2. Katz, J., Trevisan, L.: On the efficiency of local decoding procedures for error-correcting codes. In: STOC. (2000)

3. Goldreich, O., Karloff, H.J., Schulman, L.J., Trevisan, L.: Lower bounds for linear locally decodable codes and private information retrieval. In: CCC. (2002)
4. Trevisan, L.: Some applications of coding theory in computational complexity. Quaderni di Matematica 13:347-424 (2004)
5. Yekhanin, S.: Locally Decodable Codes and Private Information Retrieval Schemes. PhD thesis, MIT (2007)
6. Candès, E.J., Romberg, J.K., Tao, T.: Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. IEEE Transactions on Information Theory **52** (2006) 489–509
7. Donoho, D.L.: Compressed sensing. IEEE Transactions on Information Theory **52** (2006) 1289–1306
8. Duarte, M., Davenport, M., Takhar, D., Laska, J., Sun, T., Kelly, K., Baraniuk, R.: Single-pixel imaging via compressing sensing. IEEE Signal Processing Magazine (2008)
9. Dvir, Z., Shpilka, A.: Locally decodable codes with 2 queries and polynomial identity testing for depth 3 circuits. In: Symposium on the Theory of Computing (STOC). (2005)
10. Dvir, Z.: On matrix rigidity and locally self-correctable codes. In: IEEE Conference on Computational Complexity (CCC). (2010)
11. Valiant, L.G.: Graph-theoretic arguments in low-level complexity. In: MFCS. (1977) 162–176
12. Kerenidis, I., de Wolf, R.: Exponential lower bound for 2-query locally decodable codes. In: STOC. (2003)
13. Woodruff, D.P.: New lower bounds for general locally decodable codes. Electronic Colloquium on Computational Complexity (ECCC) **14** (2007)
14. Obata, K.: Optimal lower bounds for 2-query locally decodable linear codes. In: APPROX-RANDOM, 2483: 39-50. (2002)
15. Woodruff, D.P.: Corruption and recovery-efficient locally decodable codes. In: APPROX-RANDOM. (2008) 584–595
16. Efremenko, K.: 3-query locally decodable codes of subexponential length. In: STOC. (2009)
17. Itoh, T., Suzuki, Y.: New constructions for query-efficient locally decodable codes of subexponential length. Manuscript (2009)
18. Yekhanin, S.: Towards 3-query locally decodable codes of subexponential length. J. ACM **55** (2008)
19. Dvir, Z., Gopalan, P., Yekhanin, S.: Matching vector codes. Electronic Colloquium on Computational Complexity (ECCC) (2010)
20. Gopalan, P.: A note on Efremenko's locally decodable codes. Electronic Colloquium on Computational Complexity (ECCC) (2009)
21. Diestel, R.: Graph theory. Springer-Verlag Graduate Texts in Mathematics (2005)

A Basic Reductions

Intuitively, a local-decoding algorithm A cannot query any particular location of the (corrupted) codeword too often, as otherwise an adversary could ruin the success probability of A by corrupting only a few positions. This motivates the definition of a *smooth code*.

Definition A.1. *([2]) For fixed c, ϵ, and integer q, a linear transformation $C : \mathbb{F}^n \to \mathbb{F}^m$ is a linear (q, c, ϵ)-*smooth code *if there exists a probabilistic oracle machine A such that for every $x \in \mathbb{F}^n$,*

- *For every $i \in [n]$ and $j \in [m]$, $\Pr[A^{C(x)}(i)$ reads index $j] \le \frac{c}{m}$.*
- *For every $i \in [n]$, $\Pr[A^{C(x)}(i) = x_i] \ \ge \ \frac{1}{|\mathbb{F}|} + \epsilon$.*
- *In every invocation A makes at most q queries.*

The probabilities are taken over the coin tosses of A. An algorithm A satisfying the above is called a (q, c, ϵ)-smooth decoding algorithm for C (a decoder for short).

Unlike a local-decoding algorithm, a smooth decoding algorithm is required to work only when given access to a valid codeword, rather than a possibly corrupt one. The following reduction from LDCs to smooth codes was observed by Katz and Trevisan.

Theorem A.1. *([2]) Let $C : \mathbb{F}^n \to \mathbb{F}^m$ be a linear (q, δ, ϵ)-LDC that makes non-adaptive queries. Then C is also a linear $(q, q/\delta, \epsilon)$-smooth code.*

We use a graph-theoretic interpretation of smooth codes given in [3] and [2]. Let $C : \mathbb{F}^n \to \mathbb{F}^m$ be a linear (q, c, ϵ)-smooth code, and let algorithm A be a (q, c, ϵ)-smooth decoding algorithm for C. Since C is linear, each of the m positions of C computes $\langle v_i, x \rangle$ for a vector $v_i \in \mathbb{F}^n$. We say that a given invocation of A *reads* a set $e \subseteq \{v_1, \ldots, v_m\}$ if the set of inner prodcuts that A reads in that invocation equals $\{\langle v_i, x \rangle \mid v_i \in e\}$. Since A is restricted to read at most q entries, $|e| \le q$.

We say that e is *good* for i if $\Pr[A^{C(x)}(i) = x_i \mid A \text{ reads } e] \ \ge \ \frac{1}{|\mathbb{F}|} + \frac{\epsilon}{2}$, where the probability is over the internal coin tosses of A. It follows that if e is good for i, then the i-th standard unit vector u_i is in the span of the $|e|$ vectors. Indeed, otherwise, one can find two different inputs x which agree on the inner products that are read but differ in coordinate i.

Definition A.2. *([2]) Fixing a smooth code $C : \mathbb{F}^n \to \mathbb{F}^m$ and a q-query recovery algorithm A, the recovery hypergraphs for $i \in [n]$, denoted G_i, consist of the vertex set $\{v_1, \ldots, v_m\}$ and the hyperedge set $C_i = \{e \subseteq \{v_1, \ldots, v_m\} \mid u_i \in span(e)\}$.*

Lemma A.1. *([2]) Let C be a (q, c, ϵ)-smooth code that is good on average, and let $\{G_i\}_{i=1}^n$ be the set of recovery hypergraphs. Then, for every i, the hypergraph $G_i = (\{v_1, \ldots, v_m\}, C_i)$ has a matching M_i of sets of size q with $|M_i| \ge \frac{\epsilon m}{cq}$.*

Consider the multi-hypergraph G with vertex set $\{v_1, \ldots, v_m\}$ and hyperedge set $\uplus_{i=1}^n M_i$, that is, a hyperedge occurs in G once for each M_i that it occurs in. For readability, we use the term hypergraph to refer to a multi-hypergraph, that is, a hypergraph which may have repeated hyperedges (which we sometimes just refer to as edges). We claim that we can find a non-empty induced sub-hypergraph G' of G with minimum degree βn for a constant $\beta > 0$. The proof is a straightforward generalization of Proposition 1.2.2 in [21] to hypergraphs. For a proof, see Lemma 27 in Appendix 6 of [13] (omitted here due to space constraints).