Three Query Locally Decodable Codes with Higher Correctness Require Exponential Length

Anna Gál* **and Andrew Mills**

Department of Computer Science
University of Texas at Austin
panni@cs.utexas.edu
amills@cs.utexas.edu

Abstract

Locally decodable codes are error correcting codes with the extra property that, in order to retrieve the correct value of just one position of the input with high probability, it is sufficient to read a small number of positions of the corresponding, possibly corrupted codeword. A breakthrough result by Yekhanin showed that 3-query linear locally decodable codes may have subexponential length.

The construction of Yekhanin, and the three query constructions that followed, achieve correctness only up to a certain limit which is $1 - 3\delta$ for nonbinary codes, where an adversary is allowed to corrupt up to δ fraction of the codeword. The largest correctness for a subexponential length 3-query binary code is achieved in a construction by Woodruff, and it is below $1 - 3\delta$.

We show that achieving slightly larger correctness (as a function of δ) requires exponential codeword length for 3-query codes. Previously, there were no larger than quadratic lower bounds known for locally decodable codes with more than 2 queries, even in the case of 3-query linear codes. Our results hold for linear codes over arbitrary finite fields and for binary nonlinear codes.

Considering larger number of queries, we obtain lower bounds for q-query codes for $q > 3$, under certain assumptions on the decoding algorithm that have been commonly used in previous constructions. We also prove bounds on the largest correctness achievable by these decoding algorithms, regardless of the length of the code. Our results explain the limitations on correctness in previous constructions using such decoding algorithms. In addition, our results imply tradeoffs on the parameters of error correcting data structures.

Digital Object Identifier 10.4230/LIPIcs.STACS.2011.673

1 Introduction

Locally decodable codes are error correcting codes with the extra property that, in order to retrieve the correct value of just one position of the input with high probability, it is sufficient to read a sublinear or even just a constant number of positions of the corresponding, possibly corrupted, codeword. The formal definition was given by Katz and Trevisan [9] in 2000.

▶ **Definition 1.1.** (Katz and Trevisan [9]) For reals δ and ϵ, and a natural number q, we say that $\mathbf{C}\colon \Sigma^n \to \Gamma^m$ is a (q, δ, ϵ)-*Locally Decodable Code (LDC)* if there exists a probabilistic algorithm A such that: in every invocation, A reads at most q positions of y; and for every $x \in \Sigma^n$ and $y \in \Gamma^m$ with $d(y, \mathbf{C}(x)) \leq \delta m$, and for every $i \in [n]$, we have $\Pr[A^y(i) = x_i] \geq \frac{1}{|\Sigma|} + \epsilon$, where the probability is taken over the internal coin tosses of A.

We will refer to the value $\frac{1}{|\Sigma|} + \epsilon$ in Definition 1.1 as the *correctness* of the given decoding algorithm A, while ϵ can be thought of as the *advantage* over random guessing.

[*] Supported in part by NSF Grants CCF-0830756 and CCF-1018060

28th Symposium on Theoretical Aspects of Computer Science (STACS'11).
Editors: Thomas Schwentick, Christoph Dürr; pp. 673–684
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany
SYMPOSIUM ON THEORETICAL ASPECTS OF COMPUTER SCIENCE

Locally decodable codes have interesting applications, both in complexity theory and in practical areas. Locally decodable codes are especially useful in situations where we want to encode large amounts of data to protect against errors, but need to be able to access individual units; for example, individual patient records of a large hospital. Encoding each unit separately would give less protection against errors, and encoding the whole data set with a traditional error correcting code would require reading the whole encoded database just to access small parts of it. Locally decodable codes are closely related to private information retrieval: constructions of good locally decodable codes yield efficient protocols for private information retrieval. Private information retrieval schemes allow users to retrieve information from databases without revealing information about which data items the user is retrieving. Other applications and related structures include self correcting computations, random self-reducibility, probabilistically checkable proofs. See [15] for a survey. More recently, [6] related LDCs to polynomial identity testing for arithmetic circuits, and [4] to matrix rigidity and circuit lower bounds.

It is quite remarkable that such codes exist at all for constant number of queries. A simple example is the Hadamard code, which has the property that any input bit can be recovered with probability at least $1 - 2\delta$ from codewords possibly corrupted in up to δm positions, by a randomized algorithm that in every invocation reads no more than 2 bits of the code. However, the code is very large: the length of the codewords is 2^n for encoding n bit inputs.

Of course it would be desirable to have much more efficient, in particular polynomial length codes, but this seems to be currently out of reach for constant number of queries. Efficient constructions are known for large number of queries. See [15] for a survey.

It is known that for large enough n, 1-query locally decodable codes (that read at most one bit) cannot do better than random guessing [9].

For 2-query linear codes essentially tight bounds are known: Goldreich, Karloff, Schulman and Trevisan [8] proved exponential lower bounds for 2-query linear codes over finite fields up to a certain field size. This was later extended by Dvir and Shpilka [6] to give exponential lower bounds for 2-query linear codes over arbitrary fields. Further improvements for the 2-query linear case were given by [12, 14]. Shiowattana and Lokam [14] prove a lower bound of $\Omega(2^{4\delta n/(1-2\epsilon)})$, which is tight within a constant factor, for 2-query binary linear locally decodable codes.

Kerenidis and de Wolf [10] proved exponential lower bounds for arbitrary binary (not necessarily linear) 2-query locally decodable codes, based on quantum arguments. They also extended their lower bounds to codes over larger alphabets, but the bound decreases with the alphabet size. The strongest lower bounds so far for nonlinear codes from $\{0,1\}^n$ to $\Sigma^m = (\{0,1\}^\ell)^m$ were proved by Wehner and de Wolf [16], and are of the form $2^{\Omega(\delta\epsilon^2 n/(2^{2\ell}))}$. A proof of the 2-query lower bound for binary codes is given in [3] without using quantum arguments. It is still open to obtain nontrivial lower bounds for 2-query nonlinear codes over alphabets of size $\Omega(\sqrt{n})$.

For larger number of queries, there is still a huge gap between the known upper and lower bounds, even for binary linear codes. For codes over small (constant size) alphabets Katz and Trevisan in [9] gave a general lower bound that holds for any q showing that q-query locally decodable codes must have length $\Omega(n^{q/(q-1)})$. This bound was slightly improved by Kerenidis and de Wolf [10] to $\Omega((n/\log n)^{\frac{q+1}{q-1}})$, and by Woodruff [18] to $\Omega(n^{\frac{q+1}{q-1}})/\log n$. Woodruff [20] proved $\Omega(n^2)$ lower bounds on the length of 3-query linear codes over any field. Prior to our work, no larger than n^2 lower bound was known for locally decodable codes that allow more than 2 queries, even in the case of 3-query linear codes.

A breakthrough result of Yekhanin [21] showed that subexponential length 3-query linear locally decodable codes exist, under assumptions about the existence of infinitely many Mersenne primes. Raghavendra [13] gave some simplifications to Yekhanin's codes. Building on these works, Efremenko [7] gave a construction of subexponential length 3-query linear locally decodable codes without any unproven assumptions. All these constructions have a limit on the correctness achieved by the algorithm as a function of δ where an adversary can corrupt up to δ fraction of the codeword positions. Efremenko's construction [7] gives $1-3\delta$ correctness for a 3-query nonbinary code. For 3-query binary codes, the best dependence between the parameters is achieved in a paper by Woodruff [19], which yields 3-query binary linear locally decodable codes with correctness close to, but still below, $1 - 3\delta$. Note that these results do not provide correctness larger than $1/2$, that is they do not give better correctness than random guessing for binary codes, if the fraction of corrupted positions δ is larger than $1/6$. Recent results of Ben-Aroya, Efremenko, and Ta-Shma [2] give subexponential length locally decodable codes that can do better than random guessing for binary codes for δ fraction of corruption up to $\delta = 1/2-\alpha$ for any $\alpha > 0$, but the number of queries needed gets larger as δ gets closer to $1/2$.

1.1 Three query codes

Our main results show that achieving slightly larger than $1 - 3\delta$ correctness for 3-query locally decodable codes requires exponential length. We prove this for arbitrary (possibly nonlinear) binary codes and for linear codes over arbitrary finite fields. Note that larger, e.g. $1 - 2\delta$ correctness can be achieved even by 2-query linear codes: the Hadamard code is an example. With significantly larger number of queries, the correctness can be much higher as a function of δ (of the form $1 - \delta^{\Omega(q)}$): again the Hadamard code is an example. But this comes at the cost of having large length in the known constructions. Our results show that for 3-query codes, this increase in length cannot be avoided.

Here we give a somewhat simplified statement of the result for binary codes, without specifying the precise constants.

▶ **Theorem 1.2.** *Let* $\mathbf{C}$*:* $\{0,1\}^n \to \{0,1\}^m$ *be an arbitrary (possibly nonlinear) binary* $(3,\delta,\epsilon)$*-LDC with a nonadaptive decoder, and* n *large enough. If* $\frac{1}{2}+\epsilon > 1-3\delta+6\delta^2-4\delta^3+\phi(n)+\mu$*, where* $\phi(n) = O(1/n^{1/9})$*, then* $m \geq 2^{\Omega(\mu n^{1/3})}$*.*

We state the precise values hidden in the notation later in Theorem 3.1. We wanted to start with a more compact statement of our bound, showing that as soon as the correctness achieved by the code is above a certain threshold, the length of the codewords must be exponential. For binary codes this threshold is around $1 - 3\delta + 6\delta^2 - 4\delta^3$, which is just slightly larger than $1 - 3\delta$ for small values of δ. The value $1 - 3\delta$ is interesting, since there are subexponential length constructions of 3-query linear LDCs that achieve correctness $1-3\delta$ [7] and 3-query binary linear LDCs that achieve correctness slightly below $1-3\delta$ [19]. The value $1 - 3\delta + 6\delta^2 - 4\delta^3$ corresponds to the probability that the number of corrupted positions in a given triple is even, where the probability is over the distribution that corrupts each bit independently with probability δ.

For linear codes over arbitrary finite fields, we obtain stronger lower bounds. In our results for nonbinary codes, the value of the threshold is close to the threshold for the binary case, but slightly depends on the field size. We obtain exponential lower bounds for arbitrary finite fields, even if the field size depends on n.

We note that our bound holds for any $\delta \geq 0$ and any $0 < \epsilon \leq 1 - 1/|F|$, where δ and ϵ may be $o(1)$. For the bound to be nontrivial, we need $\delta \leq 1-1/|F|$, because of Observation

2.1. For n to be large enough for our purposes, it is sufficient if $\delta > \Omega(1/n^{1/9})$ and $\epsilon > \Omega(\frac{1}{n})$.

1.2 Arbitrary number of queries

We obtain similar results for arbitrary number of queries, under some assumptions on the decoding algorithm. We note that the types of decoding algorithms we consider have been commonly used in recent constructions. Our results explain the limitations on correctness of these constructions.

Unless otherwise noted, a q-query decoder is allowed to use less than q queries. So the correctness thresholds for requiring exponential length for q-query codes are never going to be smaller than the correctness thresholds for the same class of 3-query codes. In the special cases below, we show that the same thresholds to require exponential length as for 3-query codes also apply for arbitrary number of queries.

It remains open what is the correctness threshold (as a function of δ) to require exponential length for general q-query codes. Note that it will have to be a value larger than the threshold in our 3-query results: a q-query code for $q > 3$ can always do at least as well, as a 3-query code. We will see below, that if we require the query sets to be exactly of size q, then this is not necessarily the case.

1.2.1 Linear Decoders

One of the starting points of our approach was the observation that using larger number of queries does not help to tolerate errors if the decoder returns a fixed linear combination of the positions read. Moreover, the probability of error increases with the number of positions used with nonzero coefficients by a linear decoder. We formalize these ideas in our results about linear decoders.

▶ **Definition 1.3.** Let $\mathbf{C}$: $F^n \to F^m$ be an arbitrary (possibly nonlinear) code. We say that an algorithm A is a *linear decoder* for $\mathbf{C}$ if for any fixing of the outcomes of the coin flips of A, the value it returns is a fixed linear combination of the codeword positions it reads.

We show that linear decoders that use exactly q positions cannot achieve larger correctness than $1 - q\delta + o(\delta) + O(1/n)$, *regardless of the length of the code.* Moreover, we show that the correctness of any linear decoder, for any number of positions used, is at most $1-2\delta+o(\delta)+O(1/n)$. This holds for arbitrary (possibly nonlinear) codes and over any finite field F. This implies that our exponential length lower bounds extend to linear decoders with arbitrary number of queries, with the same correctness threshold as for 3-query codes.

Linear decoders are commonly used in the known constructions of locally decodable codes. In fact it is noted for example in [10, 18] that any (possibly nonlinear) binary (q, δ, ϵ)-LDC has a linear decoder that achieves correctness $1/2 + \epsilon/2^q$.

In the case of linear *smooth codes* (see [9]), requiring the decoders to be linear is inconsequential: for linear codes, if any algorithm gives nontrivial advantage over random guessing when querying a given set of codeword positions Q, then by Lemma 2.2, $e_i \in \mathrm{span}(Q)$ must hold. Thus, there is a fixed linear combination of the positions in Q that gives the correct value of x_i for any input x. Using the same procedure as the original decoder to choose which positions to query and then returning this fixed linear combination (if it exists) cannot violate the smoothness of the code.

However, for locally decodable codes (both linear and nonlinear), requiring to use only linear decoders may significantly reduce the correctness associated with the code. For example, taking majorities, one can obtain correctness of the from $1 - \delta^{\Omega(q)}$. Our results show that in the recent results of [5, 2] obtaining subexponential length constructions with larger than $1 - q\delta$ correctness for larger values of q, the use of nonlinear operations in the decoding algorithm is important.

Our results on linear decoders imply that there is no significantly better general reduction from smooth codes to locally decodable codes than the current bounds giving at most $1 - q\delta$ correctness for q query locally decodable codes. The possibility of better reductions was raised in [9].

1.2.2 Matching sum decoders

Matching sum decoders are a subclass of linear decoders, thus our results on linear decoders immediately apply. However, for matching sum decoders we can prove stronger results. In particular, we can replace the correctness bounds $1 - q\delta + o(\delta) + O(1/n)$, by simply $1 - q\delta + O(1/n)$ for codes with q-query matching sum decoders regardless of the length of the code, and prove exponential lower bounds on the length of LDCs with matching sum decoders using any number of queries that achieve correctness larger than $1 - 3\delta + O(1/n)$.

Matching sum decoders were formally defined by Woodruff [19]. A q-query matching sum decoder picks a set of size q uniformly at random from a collection of sets that form a matching in the complete q-uniform hypergraph, whose vertices correspond to the positions of the codeword. Then, the decoder reads the positions corresponding to the chosen set, and returns the sum of the positions read. Most known constructions of locally decodable codes have such decoders.

Woodruff [19] proved that LDCs with 2-query matching sum decoders must have exponential length. We show that q-query matching sum decoders cannot achieve larger correctness than $1 - q\delta + O(1/n)$, *regardless of the length of the code.* This holds for arbitrary codes and over any field.

Considering matching sum decoders where the query size is not fixed, we show that for any binary code (possibly nonlinear), and for linear codes over arbitrary finite fields, if a matching sum decoder with query sets of size at most q achieves correctness more than $1 - 3\delta + O(1/n^{1/3})$, then the length of the code must be exponential.

1.2.3 Query sets with large rank

For linear codes our proofs also apply to arbitrary number of queries, and possibly nonlinear decoders as long as the vectors corresponding to the positions queried are linearly independent. This is a property that holds in some of the known constructions of linear locally decodable codes. For such query sets, we show that if the correct value of x_i is spanned by q of the linearly independent vectors with nonzero coefficients, then the correctness of the decoder cannot be larger than $1 - q\delta + o(\delta) + O(1/n)$, *regardless of the length of the code.*

This implies, that for linear codes over arbitrary finite fields, if a q-query decoder (with query sets of size at most q) queries only linearly independent positions of the code and achieves correctness more than $1 - 3\delta + o(\delta) + O(1/n^{1/3})$, then the length of the code must be exponential. The exponential length lower bound extends to query sets that are not fully independent, but have large rank, with a correctness threshold that depends on the rank of the query sets. The results described for query sets with large rank are direct consequences of our proofs for linear codes.

1.3 Error Correcting Data Structures

Error correcting data structures were defined by de Wolf [17]. Such data structures are a variation of the traditional bit-probe model (see e.g. [11]), where the algorithms answering questions about the data are correct with probability at least $1/2 + \epsilon$, as long as at most δ fraction of the database representing the data is corrupted, possibly by adversarial error. It is noted in [17] that error correcting data structures for the membership problem yield locally decodable codes, with the same parameters. [17] showed the existence of error correcting data structures for the membership problem and some of its variants, assuming the existence of locally decodable codes with given parameters. Because of the direct correspondence between the two models, our results rule out the existence of error correcting data structures for membership of subexponential size with larger correctness than our thresholds above, for 3-probe algorithms, as well as for algorithms with arbitrary number of probes, assuming the algorithm only uses linear operations.

1.4 Techniques

We start by noting why some hand waiving arguments and intuition based on smooth codes would fail to explain our most general results. *Smooth codes* were defined by Katz and Trevisan [9], who also gave reductions between smooth codes and locally decodable codes. So up to changes in parameters, smooth codes and locally decodable codes are equivalent. Most of the current lower bounds for locally decodable codes have been proved via proving lower bounds for smooth codes, and the correctness of the known subexponential length constructions of 3-query linear LDCs is analyzed based on their property of having *smooth decoders*, that are correct with large probability if there is no error, and query each position of the code with not too large probability. However, the current techniques to analyze smooth decoders cannot imply larger than $1 - q\delta$ correctness for q-query locally decodable codes. In fact we show that no significantly better general reduction is possible. If we consider larger than $1 - q\delta$ correctness, then the equivalence of smooth codes and locally decodable codes starts to break down.

We elaborate on a few specific points below. One could try to argue that the probability that the decoder does not query any corrupted positions is upper bounded by a function not much larger than $1 - q\delta$, thus the decoder will have to read corrupted positions. However, errors may cancel out, so the fact that some of the positions read by the decoder may contain an error, in itself does not explain our lower bounds.

If the decoder was only working with query sets that form a matching, and the decoder was linear (which is the case in several of the known constructions), then - as we show - $1 - q\delta$ would in fact be a limit on correctness for decoders that query exactly q positions. But these assumptions do not have to hold for every decoding algorithm, and our results cannot be explained by this simplified view.

Our proofs of the 3-query lower bounds are based on a lemma that was central in obtaining the exponential lower bounds for 2-query codes. However, we would like to emphasize that we do not use 2-query lower bounds as a black box. We show that query sets that provide large correctness must contain subsets of size at most 2 that give nontrivial correlation with the input position we try to recover. But this does not imply that the code somehow "reduces" to a 2-query code. Consider the following simple example (many other examples are possible): query 3 positions such that each in itself has large correlation with the position x_i, and take the majority of the answers. Replacing this with reading only a subset of the bits, would preserve the properties of a smooth decoder, but it would reduce the correctness of the decoding algorithm. Thus, the decoder cannot be simply replaced by a 2-query decoder, if we want to preserve the correctness probabilities of the decoder.

Our approach can be summarized as follows: we show that in the case of 3-query codes, if the code is small, we can "force" the decoder to only examine query sets that are vulnerable to error. We achieve this by considering the algorithm's performance over random input x and a specially constructed distribution for the corruption caused by the adversary. We show that over our distribution, the decoder cannot perform much better than a linear decoder.

In all of our results, the probability of error is estimated in terms of the probability - over appropriate random corruption - of the event that the sum of the corruption in the positions of a given query set is nonzero. Intuitively, this probability would indeed give a lower bound on the error if the decoder always returned the sum (or a fixed linear combination) of the positions read, and if this was equal to the correct answer for uncorrupted codewords. For example, this would be the case for linear decoders of a linear code. However, we also consider nonlinear codes, and arbitrary decoders that may involve nonlinear operations. In fact, we do not claim that the probability of having nonzero sum of corruption in the query set is a lower bound on the error in general. Instead, we lower bound the probability of error by a different expression, and show that this expression is lower bounded by the probability mentioned above in the case of random corruption according to our distribution.

A crucial point in our proofs for nonlinear decoders (for both linear and nonlinear codes) is comparing the conditional probabilities of error of the decoder, conditioned on the sum of the values in the corrupted positions. We show that - under appropriate assumptions on the query sets for linear codes - the sum of these conditional probabilities of the decoder being incorrect, is always $|F|-1$. A subtle point of this argument is that the various events we work with are not always independent. Our proof for nonlinear codes is based on a similar property of conditioning on the number of corrupted positions being odd vs. even. However, for nonlinear codes instead of directly considering the conditional probabilities of incorrect decoding, we reduce estimating the probability of error to estimating the probability that the sum of the positions read gives an incorrect answer. This analysis lets us estimate the probability of incorrect decoding even if the decoding algorithm uses nonlinear operations.

2 Preliminaries

The definition of locally decodable codes allows the decoding algorithm to be adaptive. Lower bounds for nonadaptive decoders can be translated to lower bounds for arbitrary decoders with the same number of queries but larger correctness: it is noted in the paper by Katz and Trevisan [9] that any adaptive (q, δ, ϵ) decoding algorithm for a code $\mathbf{C}\colon \Sigma^n \to \Gamma^m$, can be transformed to a nonadaptive $(q, \delta, \epsilon/|\Gamma|^{q-1})$ decoding algorithm for the same code.

We only consider nonadaptive decoding algorithms in the rest of the paper. We will refer to the (at most q) positions the algorithm chooses to read in a given invocation as a *query set*. In a nonadaptive algorithm, the choice of the query set only depends on the coin flips of the algorithm.

The following simple observation means that for proving lower bounds we may assume that $\delta < 1 - \frac{1}{|\Sigma|}$, since otherwise, no algorithm can do better than random guessing for *any* of the input positions.

▶ **Observation 2.1.** *Let A be a decoding algorithm for any code $\mathbf{C}\colon \Sigma^n \to \Sigma^m$. If $\delta \geq 1-\frac{1}{|\Sigma|}$, then for any $i \in [n]$, $\min_{x\in\Sigma^n} \left(\min_{y\in\Gamma^m \,:\, d(y,\mathbf{C}(x))\leq\delta m} \Pr\left[A^y(i) = x_i\right]\right) \leq \frac{1}{|\Sigma|}$.*

For a linear code $\mathbf{C}\colon F^n \to F^m$, it is convenient to represent the function that determines a given codeword position by a vector: for $j \in [m]$, define $a_j \in F^n$ as the vector satisfying

$\forall x \in F^n$, $\mathbf{C}_j(x) = a_j \cdot x$. For vectors $a, x \in F^n$, we use $a \cdot x$ to denote their inner product over F. (We omit F from the notation.)

For a query set $Q = \{j_1, \ldots, j_q\} \subset [m]$, we use the notation $\mathrm{span}(Q)$ to represent the linear span of the vectors $a_{j_1}, \ldots, a_{j_q}$ corresponding to the positions in Q. We denote the i'th unit vector with length n by e_i. e_i has 1 in its i-th coordinate and 0 everywhere else.

The following lemma was stated in [8] for two query binary linear codes. Its extension to arbitrary fields and any number of queries is straightforward, but important for our arguments.

▶ **Lemma 2.2.** *(implicit in [8]) Let $\mathbf{C}$: $F^n \to F^m$ be a linear code. Let $i \in [n]$ and let $Q = \{j_1, j_2, ... j_q\} \subset [m]$ be a query set that the algorithm A queries with nonzero probability when trying to recover the value of input position i. Suppose $\Pr_{x \epsilon_U F^n} \left[A^{\mathbf{C}(x)}(i) = x_i \mid A \text{ queries } Q\right] > \frac{1}{|F|}$ where the probability is taken over letting x be uniformly random from F^n and over the internal coin tosses of A. Then $e_i \in \mathrm{span}(Q)$ must hold.*

We will use the following simple fact as well as Lemma 2.2 throughout our proof for linear codes.

▶ **Fact 2.3.** *(implicit in [1]) Let $a_1, \ldots, a_t$ be vectors from F^n. For x uniformly random from F^n, the corresponding random values $a_1 \cdot x, \ldots, a_t \cdot x$ are t independent uniformly distributed values from F, if and only if the vectors $a_1, \ldots, a_t$ are linearly independent over F.*

The following theorem of Goldreich, Karloff, Schulman and Trevisan [8] is a crucial ingredient of our proofs.

▶ **Theorem 2.4.** *[8] Let a_1, ... a_m be a sequence of (not necessarily distinct) elements of $\{0,1\}^n$ such that for every $i \in [n]$ there is a set M_i of disjoint pairs of indices $\{j_1, j_2\}$ such that $e_i = a_{j_1} \oplus a_{j_2}$. Then $m \geq 2^{2\alpha n}$, where $\alpha \triangleq \frac{\sum_{i=1}^{n} |M_i|}{nm}$.*

This theorem was extended to arbitrary finite fields in [8]. The dependence on the field size in the bound was removed by Dvir and Shpilka in [6]. We will use the following version (see Corollary 2.9 in [6]).

▶ **Theorem 2.5.** *[6] Let F be a field. Let a_1, ... a_m be a sequence of (not necessarily distinct) elements of F^n such that for every $i \in [n]$ there is a set M_i of disjoint pairs of indices $\{j_1, j_2\}$ such that $e_i \in \mathrm{span}(a_{j_1}, a_{j_2})$. Then $m \geq 2^{\alpha n - 1}$, where $\alpha \triangleq \frac{\sum_{i=1}^{n} |M_i|}{nm}$.*

A version of the theorem applicable to binary nonlinear codes is given in the "non-quantum" proof of the exponential lower bounds for 2-query binary nonlinear codes by Ben-Aroya, Regev and de Wolf [3].

▶ **Theorem 2.6.** *(implicit in Theorem 11 of [3]) Let $0 < \epsilon, \alpha < 1/2$. Let a_1, ... a_m be a sequence of (not necessarily distinct) functions from $\{0,1\}^n$ to $\{0,1\}$ such that for at least τn indices $i \in [n]$ there is a set M_i of disjoint pairs of indices $\{j_1, j_2\}$ such that $|M_i| \geq \alpha m$ and*

$$|\Pr_x[x_i = a_{j_1}(x) \oplus a_{j_2}(x)] - \Pr_x[x_i \neq a_{j_1}(x) \oplus a_{j_2}(x)]| \geq \epsilon$$

where the probability is over uniform $x \in \{0,1\}^n$. Then $m \geq 2^{\tau \alpha^2 \epsilon^2 n}$.

We also use the following theorem of Katz and Trevisan [9].

▶ **Theorem 2.7.** *(Theorem 2 in [9]) Let $\mathbf{C} : \{0,1\}^n \to R$ be a function. Assume there is an algorithm A such that for every $i \in [n]$, we have $\Pr_x[A(\mathbf{C}(x), i) = x_i] \geq \frac{1}{2} + \epsilon$, where the probability is taken over the internal coin tosses of A and uniform $x \in \{0,1\}^n$. Then $\log |R| \geq (1 - H(1/2 + \epsilon))n$.*

2.1 Notation

Let F be an arbitrary finite field. We denote by F^* the set of nonzero elements of F. Arithmetic operations involving field elements are over F. This should be clear from the context, and will be omitted from the notation.

For a code $\mathbf{C}\colon F^n \to F^m$, we can represent any vector $y \in F^m$ with $d(y, \mathbf{C}(x)) \le \delta m$ as a sum of the form $y = \mathbf{C}(x) + B$, where $B \in F^m$, such that the number of nonzero entries in B is at most δm.

We will use the notation $\Pr_{x,B,A}$ to indicate probabilities over uniformly random input x from F^n, B chosen at random from a given distribution for corruption, and the random coin tosses of the given algorithm A.

Note that while in general the corruption may be produced by an arbitrary adversary, we will only consider distributions for B that do not depend on the input x or on the distribution for the coin tosses of the algorithm. This is sufficient for our purposes, since we are proving lower bounds on the length of the code.

3 Lower Bounds for Three Query Codes

3.1 Lower Bounds for Three Query Binary Codes

We state the precise version of our lower bound for arbitrary (possibly nonlinear) binary codes.

▶ **Theorem 3.1.** *Let $\mathbf{C}\colon \{0,1\}^n \to \{0,1\}^m$ be a $(3,\delta,\epsilon)$-LDC with a nonadaptive decoder, and n large enough. Let $\alpha \triangleq \delta - (\frac{1}{2} - (\frac{\epsilon}{4})^{1/3}) - (3/n^{1/3} + \frac{36}{n})^{1/3} - \nu$, and $\nu \triangleq \frac{10}{n(1-H(1/2+1/n^{1/3}))} = O(1/n^{1/3})$. If $\alpha > 0$, then $m \ge 2^{0.225\alpha^2 n^{1/3}}$.*

▶ Remark. We will show that $\alpha > 0$ when $\frac{1}{2} + \epsilon > 1 - 3\delta + 6\delta^2 - 4\delta^3 + \phi(n)$, where $\phi(n) = 4((3/n^{1/3} + \frac{36}{n})^{1/3} + \nu)$. Moreover, $\alpha > \frac{\mu}{4}$ when $\frac{1}{2} + \epsilon > \mu + 1 - 3\delta + 6\delta^2 - 4\delta^3 + \phi(n)$ for some $\mu \ge 0$. This implies the version of the bound stated in Theorem 1.2 for binary codes. Note that we could also obtain a lower bound of the form $2^{\Omega(n)}$ by setting ϵ_2 in the proof to a constant, but then the correctness required for the bound would be larger, roughly by $4(\epsilon_2)^{1/3}$.

Sketch of proof.

For the case of binary (possibly nonlinear) codes, using the Fourier representation of Boolean functions, and properties of correlation, we show that for any decoding algorithm, and for any query set Q, the advantage of the algorithm over random guessing when reading the values of the query set Q is at most the sum of the advantages obtained by all possible fixed linear functions over the given query set. This observation has been implicitly used also in the arguments of [10] and [20] showing the existence of linear decoders with correctness $1/2 + \epsilon/2^q$ for any binary (q,δ,ϵ)-LDC.

In all our proofs, we use a distribution for the adversary that corrupts each codeword position in a particular set S independently, and chooses the corruption over the remaining set of positions so that the total fraction of corrupted positions is still below δ. We construct the set S so that for query sets that do not intersect the set S, the contribution of sums over subsets of size at most 2 towards the advantage over random guessing is small. However, the size of S has to be small to keep the total fraction of corrupted positions below δ.

To achieve this, we first argue that in any LDC, the number of codeword positions that have large correlation with a given input bit x_i over random input x must be small for most input positions $i \in [n]$. This is straightforward for linear codes. Note however that for nonlinear codes, it is possible that a given codeword position has significant correlation with more than one input bit. We show the desired statement using Theorem 2.7.

Next we consider pairs of codeword positions, such that the sum of their values gives large correlation with a given input bit x_i over random input x. Using Theorem 2.6, we show that if the length of the code is small, then for most $i \in [n]$, all such pairs of positions can be covered by a small number of codeword positions.

This allows us to conclude that *if the length of the code is small*, then for at least one index $i \in [n]$, there exist a set S of small size, such that for query sets that do not intersect the set S, the contribution of sums over subsets of size at most 2 towards the advantage over random guessing the bit x_i is small.

We show that for any LDC with correctness $1/2 + \epsilon$, there is a decoding algorithm that never reads any of the positions in S, but is correct with probability at least $1/2 + \epsilon$ on average over random input x, and the random corruption of the above distribution. Note that the algorithm may not achieve the required correctness on every input and for every string within distance δm of $\mathbf{C}(x)$. We only claim a bound on its probability of being correct over uniformly random x and over random corruption according to our distribution.

This way we can argue that *if the length of the code is small* then there is a decoding algorithm that only uses query sets that either provide only small advantage over random guessing, or they involve 3 codeword positions, such that the sum of the 3 positions gives the correct value of the input bit x_i with large probability over random input and the random corruption according to our distribution.

On the other hand, for this decoding algorithm we can lower bound the probability of error by the probability that the sum of a given triple of codeword positions gives an incorrect value over random input and the random corruption according to our distribution.

Please see the full version of the paper for a detailed proof of the theorem and for the formal description of the distribution for the random corruption.

3.2 Lower Bounds for Three Query Linear Codes over Arbitrary Finite Fields

For linear codes over arbitrary finite fields, we obtain stronger lower bounds than our bounds for nonlinear codes.

Let F be an arbitrary finite field. We denote by F^* the set of nonzero elements of F. It is convenient to state the threshold on correctness in our bounds in terms of the probability of the event that a fixed linear combination of a given triple of coordinates of an appropriate random corruption equals to 0. More precisely, let $Q \subseteq [m]$ with $|Q| = q$ be an arbitrary fixed subset of the coordinates. Let $c_j \in F^*$, for $j \in Q$ and let $\delta \leq 1 - 1/|F|$. For the distributions we work with, the values of c_j will not make a difference, as long as they are all nonzero. Let $P(\delta, q, F) \triangleq \Pr_B \left[(\sum_{j \in Q} c_j B_j = 0)\right]$, where the probability is over $B \in F^m$ randomly chosen according to a distribution that first chooses to corrupt each coordinate in $[m]$ independently with probability δ, and then uniformly and independently assigns a value from F^* to each chosen coordinate of B. The remaining coordinates of B are set to 0. Note that an adversary using this distribution would possibly corrupt more than δm positions with nonzero probability, so this is not the distribution we use in our proofs. But it is convenient to use the probability $P(\delta, q, F)$ in the statement of our bounds, since $P(\delta, q, F)$ only depends on δ, q and $|F|$, it does not depend on m. The theorem also holds using the distribution that chooses δm positions uniformly (instead of independently corrupting the positions). While it is well known that these probabilities are not too far from each other in the two distributions, for our purposes we need more precise estimates.

We start with a simplified statement of the result without specifying the precise constants.

▶ **Theorem 3.2.** *Let* $\mathbf{C}$*:* $F^n \to F^m$ *be a linear* $(3, \delta, \epsilon)$*-LDC with a nonadaptive decoder, and* n *large enough. If* $\frac{1}{|F|} + \epsilon > P(\delta, 3, F) + \phi(n) + \mu$*, where* $\phi(n) = O(1/n^{1/3})$*, then* $m \geq 2^{\Omega(\mu n)}$*.*

We state the precise values hidden in the notation in Theorem 3.3.

For binary linear codes we present a slightly stronger bound in the next section.

▶ **Theorem 3.3.** *Let* $\mathbf{C}$*:* $F^n \to F^m$ *be a linear* $(q = 3, \delta, \epsilon)$*-LDC with a nonadaptive decoder,* $\delta \leq 1 - \frac{1}{|F|}$*, and* n *large enough. Then,* $m \geq 2^{.45\alpha n - 1}$ *where* $\alpha \triangleq \delta - (1 - \frac{1}{|F|} - \epsilon^{1/3}(1 - \frac{1}{|F|})^{2/3}) - (\frac{108}{n})^{1/3} - \frac{10}{n}$*.*

▶ Remark. We will show that $\alpha > 0$ when $\frac{1}{|F|} + \epsilon > 1 - 3\delta(1-\delta)^2 - (1 - \frac{1}{|F|-1})3\delta^2(1-\delta) - (1 - \frac{1}{|F|-1} + \frac{1}{(|F|-1)^2})\delta^3 + \phi(n)$, where $\phi(n) = 4((108/n)^{1/3} + 10/n)$. Moreover, $\alpha > \frac{\mu}{4}$ when $\frac{1}{|F|} + \epsilon > \mu + 1 - 3\delta(1-\delta)^2 - (1 - \frac{1}{|F|-1})3\delta^2(1-\delta) - (1 - \frac{1}{|F|-1} + \frac{1}{(|F|-1)^2})\delta^3 + \phi(n)$ for some $\mu \geq 0$. This implies the version of the bound stated in Theorem 3.2.

Sketch of proof.

Similarly to the proof for binary codes, we construct the set S of positions that we corrupt independently, so that for query sets that do not intersect the set S, the contribution of sums (or linear combinations) over subsets of size at most 2 towards the advantage over random guessing is small. Linear codes have the very strong property that linear combinations of codeword positions are either exactly equal to a given input bit, or give no advantage over random guessing towards recovering the given input bit (see Lemma 2.2). Thus, in the case of linear codes, we can construct the distribution of the adversary so that the decoding algorithm is left with query sets of size 3, such that no subsets of size at most 2 can give any advantage over random guessing. All other query sets that the algorithm can read will give no advantage over random guessing for a given input bit. Thus we don't need the part of the argument using Fourier representation of Boolean functions used in the binary proof to reduce estimating the probability of error to estimating the error over query sets of a special form. However, we still need to deal with the fact that the decoders can use nonlinear operations. We achieve this by considering the conditional probabilities of error of the decoder, conditioned on the sum (more precisely a fixed linear combination) of the values in the corrupted positions. In addition, we show that in the query sets we are left with the positions must correspond to linearly independent vectors in the generator matrix of the code. Based on this, we show that the probability of error (on average using our distribution) is lower bounded by $|F| - 1$ times the minimum over $k \in F$ of the probability that a fixed linear combination (with nonzero coefficients) of the corruption in the positions of the query set equals k.

See the full version for a detailed proof of the Theorem.

3.3 Lower Bounds for Three Query Binary Linear Codes

For binary linear codes, we obtain a slightly stronger bound than what follows from the lower bound for linear codes over arbitrary finite fields.

▶ **Theorem 3.4.** *Let* $\mathbf{C}$*:* $\{0,1\}^n \to \{0,1\}^m$ *be a linear* $(3, \delta, \epsilon)$*-LDC with a nonadaptive decoder, and* n *large enough. Then,* $m \geq 2^{1.8\alpha n}$ *where* $\alpha \triangleq \delta - (\frac{1}{2} - (\frac{\epsilon}{4})^{1/3}) - (\frac{36}{n})^{1/3} - \frac{10}{n}$*.*

▶ Remark. We will show that $\alpha > 0$ when $\frac{1}{2} + \epsilon > 1 - 3\delta + 6\delta^2 - 4\delta^3 + \phi(n)$, where $\phi(n) = 4((36/n)^{1/3} + 10/n)$. Moreover, $\alpha > \frac{\mu}{4}$ when $\frac{1}{2} + \epsilon > \mu + 1 - 3\delta + 6\delta^2 - 4\delta^3 + \phi(n)$ for some $\mu \geq 0$. This implies the version of the bound stated in Theorem 3.2 for binary codes.

The proof is almost identical to the proof in the previous section for arbitrary finite fields. The improvement comes from using Theorem 2.4, and because in the case of binary linear codes we can use a node cover of size $|M_1|$ instead of $2|M_1|$ when defining the distribution.

Acknowledgements We thank David Woodruff for helpful conversations and for pointing us to [3], and the anonymous referees for helpful comments.

References

1 N. Alon, L. Babai and A. Itai: A fast and simple randomized parallel algorithm for the maximal independent set problem. *Journal of Algorithms*, Vol. 7, pp. 567 - 583, 1986.

2 A. Ben-Aroya, K. Efremenko, A. Ta-Shma: Local list-decoding with a constant number of queries. ECCC Report No. 47, 2010.

3 A. Ben-Aroya, O. Regev and R. de Wolf: A hypercontractive inequality for matrix valued functions with applications to quantum computing and LDCs. In *Proceedings of FOCS 2008*, pp. 477 - 486.

4 Z. Dvir: On matrix rigidity and locally self-correctable codes. ECCC Technical Report No. 134, 2009.

5 Z. Dvir, P. Gopalan, S. Yekhanin: Matching Vector Codes. In *Proceedings of FOCS 2010*, pp. 705 - 714.

6 Z. Dvir and A. Shpilka: Locally decodable codes with 2 queries and polynomial identity testing for depth 3 circuits. In *Proceedings of STOC 2005*, pp. 592 - 601.

7 K. Efremenko: 3-query locally decodable codes of subexponential length. In *Proceedings of STOC 2009*, pp. 39-44.

8 O. Goldreich, H. Karloff, L. Schulman and L. Trevisan: Lower bounds for linear locally decodable codes and private information retrieval. *Comput. Complex.*, 15(3):263–296, 2006.

9 J. Katz and L. Trevisan: On the efficiency of local decoding procedures for error-correcting codes. In *Proceedings of STOC 2000*, pp. 80-86.

10 I. Kerenidis and R. de Wolf: Exponential lower bound for 2-query locally decodable codes via a quantum argument. In *Proceedings of STOC 2003*, pp. 106-115.

11 P. Bro Miltersen: Cell probe complexity - a survey. In *Advances in Data Structures Workshop, 1999.*

12 K. Obata: Optimal lower bounds for 2-query locally decodable linear codes. In *Proceedings of RANDOM 2002* pp. 39–50.

13 P. Raghavendra: A note on Yekhanin's locally decodable codes. *ECCC TR07-016*, 2007.

14 D. Shiowattana and S. V. Lokam: An optimal lower bound for 2-query locally decodable linear codes. *Inf. Process. Lett.*, 97(6):244–250, 2006.

15 L. Trevisan: Some applications of coding theory in computational complexity. *ECCC TR04-043*, 2004.

16 S. Wehner and R. de Wolf: Improved lower bounds for locally decodable codes and private information retrieval. In *Proceedings of ICALP 2005* Vol. 3580 of LNCS, pp. 1424 - 1436.

17 R. de Wolf: Error-correcting data structures In *Proceedings of STACS 2009* pp. 313 - 324.

18 D. Woodruff: Some new lower bounds for general locally decodable codes. *ECCC TR07-006*, 2006.

19 D. Woodruff: Corruption and recovery-efficient locally decodable codes. In *Proceedings of RANDOM 2008*, pp. 584–595.

20 D. Woodruff: A Quadratic Lower Bound for Three-Query Linear Locally Decodable Codes over Any Field. In *Proceedings of RANDOM 2010.*

21 S. Yekhanin: Towards 3-query locally decodable codes of subexponential length. In *Proceedings of STOC 2007*, pp. 266–274.