List-decoding algorithms for lifted codes

Alan Guo * Swastik Kopparty †

March 2, 2016

Abstract

Lifted Reed-Solomon codes are a natural affine-invariant family of error-correcting codes which generalize Reed-Muller codes. They were known to have efficient local-testing and local-decoding algorithms (comparable to the known algorithms for Reed-Muller codes), but with significantly better rate. We give efficient algorithms for list-decoding and local list-decoding of lifted codes. Our algorithms are based on a new technical lemma, which says that codewords of lifted codes are low degree polynomials when viewed as univariate polynomials over a big field (even though they may be very high degree when viewed as multivariate polynomials over a small field).

1 Introduction

By virtue of their many powerful applications in complexity theory, there has been much interest in the study of error-correcting codes which support "local" operations. The operations of interest include local decoding, local testing, local correcting, and local list-decoding. Error correcting codes equipped with such local algorithms have been useful, for example, in proof-checking, private information retrieval, and hardness amplification.

The canonical example of a code which supports all the above local operations is the Reed-Muller code, which is a code based on evaluations of low-degree polynomials. Reed-Muller codes have nontrivial local algorithms across a wide range of parameters. In this paper, we will be interested in the constant rate regime. For a long time, Reed-Muller codes were the only known codes in this regime supporting nontrivial locality. Concretely, for every constant integer m and every constant $R < \frac{1}{m!}$, there are Reed-Muller codes of arbitrarily large length n, rate R, constant relative distance δ, which are locally decodable/testable/correctable from $(\frac{1}{2} - \epsilon) \cdot \delta$ fraction fraction errors using $O(n^{1/m})$ queries. In particular, no nontrivial locality was known for Reed-Muller codes (or any other codes, until recently) with rate $R > 1/2$.

In the last few years, new families of codes were found which had interesting local algorithms in the high rate regime (i.e., with rate R near 1). These codes include multiplicity codes [KSY11, Kop12], lifted codes [GKS13, Guo13], expander codes [HOW13] and tensor codes [Vid10]. Of these, lifted codes are the only ones that are known to be both locally decodable and locally testable. This paper gives new and improved decoding and testing algorithms for lifted codes.

*CSAIL, Massachusetts Institute of Technology, 32 Vassar Street, Cambridge, MA, USA. aguo@mit.edu. Research supported in part by NSF grants CCF-0829672, CCF-1065125, and CCF-6922462, and an NSF Graduate Research Fellowship

†Department of Mathematics & Department of Computer Science, Rutgers University. swastik.kopparty@rutgers.edu. Research supported in part by a Sloan Fellowship and NSF CCF-1253886.

1.1 Lifted Codes and our Main Result

Lifted codes are a natural family of algebraic, affine-invariant codes which generalize Reed-Muller codes. We give a brief introduction to these codes now[1]. Let q be prime power, let $d < q$ and let $m > 1$ be an integer. Define alphabet $\Sigma = \mathbb{F}_q$. We define the lifted code $\mathcal{C} = \mathcal{C}(q, d, m)$ to be a subset of $\Sigma^{\mathbb{F}_q^m}$, the space of functions from $\mathbb{F}_q^m$ to $\Sigma = \mathbb{F}_q$. A function $f : \mathbb{F}_q^m \to \mathbb{F}_q$ is in $\mathcal{C}$ if for every line $L \subseteq \mathbb{F}_q^m$, the restriction of f to L is a univariate polynomial of degree at most d. Note that if f is the evaluation table of an m-variate polynomial of degree $\leq d$, then f is automatically in $\mathcal{C}$. The surprising (and useful) fact is that if d is large and $\mathbb{F}_q$ has small characteristic, then $\mathcal{C}$ has significantly more functions, but has the same distance as the Reed-Muller code. This leads to its improved rate relative to the corresponding Reed-Muller code, which only contains the evaluation tables of low degree polynomials.

Our main result is an algorithm for list-decoding and local list-decoding of lifted codes. We show that lifted codes of distance δ can be efficiently list-decoded and locally list-decoded (in sublinear-time) upto their "Johnson radius" $(1 - \sqrt{1-\delta})$. Combined with the local testability of lifted codes, this also implies that lifted codes can be locally tested in the high-error regime, upto the Johnson radius.

It is well known that Reed-Muller codes can be list decoded and locally list-decoded upto the Johnson radius [PW04, STV99][2] [3]. Our result shows that a lifted code, which is a natural algebraic supercode of Reed-Muller codes, despite having a vastly greater rate than the corresponding Reed-Muller code, loses absolutely nothing in terms of any (local) algorithmic decoding / testing properties.

In the appendix, we also prove two other results as part of the basic toolkit for working with lifted codes.

- Explicit interpolating sets: For a lifted code $\mathcal{C}$, we give a strongly explicit subset S of $\mathbb{F}_q^m$ such that for every $g : S \to \mathbb{F}_q$, there is a unique lifted codeword $f : \mathbb{F}_q^m \to \mathbb{F}_q$ from $\mathcal{C}$ with $f|_S = g$. The main interest in explicit interpolating sets for us is that it allows us to convert the *sublinear-time* local correction algorithm for lifted codes into a *sublinear-time* local decoding algorithm for lifted codes (earlier the known sublinear-time local correction, only implied low-query-complexity local decoding, without any associated sublinear-time local decoding algorithm).

- Simple local decoding upto half the minimum distance: We note that there is a simple algorithm for local decoding of lifted codes upto half the minimum distance. This is a direct translation of the elegant weighted-lines local decoding algorithm for matching-vector codes [BET10] to the Reed-Muller code / lifted codes setting.

1.2 Methods

We first discuss our (global) list-decoding algorithm, which generalizes the list-decoding algorithm for Reed-Muller codes due to Pellikaan-Wu [PW04]. The main technical lemma underlying our algorithm says that codewords of lifted codes are low-degree when viewed as univariate polynomials.

[1] Technically we are talking about lifted Reed-Solomon codes, but for brevity we refer to them as lifted codes.

[2] To locally list-decode all the way upto the Johnson bound, one actually needs a variant of [STV99] given in [BK09].

[3] There is another regime, where q is constant, in which the Reed-Muller codes can be list-decoded beyond the Johnson bound, upto the minimum distance. See [GKZ08, Gop10, BL14]

This generalizes the classical fact due to Kasami-Lin-Peterson [KLP68] underlying the Pellikaan-Wu decoding algorithm: that multivariate polynomials are low-degree when viewed as univariate polynomials ("Reed-Muller codes are subcodes of Reed-Solomon codes").

The codewords of a lifted code are in general very high degree as m-variate polynomials over $\mathbb{F}_q$. There is a description of these codes in terms of spanning monomials [GKS13], but it is not even clear from this description that lifted codes have good distance. The handle that we get on lifted codes arises by considering the big field $\mathbb{F}_{q^m}$, and letting ϕ be an $\mathbb{F}_q$-linear isomorphism between $\mathbb{F}_{q^m}$ and $\mathbb{F}_q^m$. Given a function $f : \mathbb{F}_q^m \to \mathbb{F}_q$, we can consider the composed function $f \circ \phi$, and view it as a function from $\mathbb{F}_{q^m} \to \mathbb{F}_q$. Our technical lemma says that this function $f \circ \phi$ is low-degree as a univariate polynomial over $\mathbb{F}_{q^m}$ (irrespective of the choice of the map ϕ).

Through this lemma, we reduce the problem of list-decoding lifted codes over the small field $\mathbb{F}_q$ to the problem of list-decoding univariate polynomials (i.e., Reed-Solomon codes) over the large field $\mathbb{F}_{q^m}$. This latter problem can be solved using the Guruswami-Sudan algorithm [GS99].

Our local list-decoding algorithm uses the above list-decoding algorithm. Following [AS03, STV99, BK09], local list-decoding of m-variate Reed-Muller codes over $\mathbb{F}_q$ reduces to (global) list-decoding of t-variate Reed-Muller codes over $\mathbb{F}_q$ (for some $t < m$). For the list-decoding radius to approach the Johnson radius, one needs $t \geq 2$. This is where the above list-decoding algorithm gets used.

Organization of this paper Section 2 introduces notation and preliminary definitions and facts to be used in later proofs. Section 3 proves our main technical result, that lifted RS codes over domain $\mathbb{F}_q^m$ are low degree when viewed as univariate polynomials over $\mathbb{F}_{q^m}$, as well as the consequence for global list decoding. Section 4 presents and analyzes the local list decoding algorithm for lifted RS codes, along with the consequence for local testability. Appendix A describes the explicit interpolating sets for arbitrary lifted affine-invariant codes. Appendix B presents and analyzes the local correction algorithm upto half the minimum distance.

2 Preliminaries

2.1 Notation

For a positive integer n, we use $[n]$ to denote the set $\{1, \ldots, n\}$. For sets A and B, we use $\{A \to B\}$ to denote the set of functions mapping A to B.

For a prime power q, $\mathbb{F}_q$ is the finite field of size q. We think of a code $\mathcal{C} \subseteq \{\mathbb{F}_Q^m \to \mathbb{F}_q\}$ as a family of functions $f : \mathbb{F}_Q^m \to \mathbb{F}_q$, where $\mathbb{F}_Q$ is an extension field of $\mathbb{F}_q$, but each codeword is a vector of evaluations $(f(x))_{x \in \mathbb{F}_Q^m}$ assuming some canonical ordering of elements in $\mathbb{F}_Q^m$; we abuse notation and say $f \in \mathcal{C}$ to mean $(f(x))_{x \in \mathbb{F}_Q^m} \in \mathcal{C}$.

If $f : \mathbb{F}_q^m \to \mathbb{F}_q$ and line ℓ is a line in $\mathbb{F}_q^m$, this formally means ℓ is specified by some $a, b \in \mathbb{F}_q^m$ and the restriction of f to ℓ, denoted by $f|_\ell$, means the function $t \mapsto f(a + bt)$. Similarly, if P is a plane, then it is specified by some $a, b, c \in \mathbb{F}_q^m$ and the restriction of f to P, denoted by $f|_P$, means the function $(t, u) \mapsto f(a + bt + cu)$.

2.2 Interpolating sets and decoding

Definition 2.1 (Interpolating set)**.** A set $S \subseteq \mathbb{F}_Q^m$ is an *interpolating set* for $\mathcal{C}$ if for every $\widehat{f} : S \to \mathbb{F}_q$ there exists a unique $f \in \mathcal{C}$ such that $f|_S = \widehat{f}$.

Note that if S is an interpolating set for $\mathcal{C}$, then $|\mathcal{C}| = q^{|S|}$.

Definition 2.2 (Local decoding)**.** Let Σ be an alphabet and let $\mathcal{C} : \Sigma^k \to \Sigma^n$ be an encoding map. A (ρ, l)-*local decoding algorithm for* $\mathcal{C}$ is a randomized algorithm $D : [k] \to \Sigma$ with oracle access to an input word $r \in \Sigma^n$ and satisfies the following:

1. If there is a message $m \in \Sigma^k$ such that $\delta(\mathcal{C}(m), r) \leq \rho$, then for every input $i \in [k]$, we have $\Pr[D^r(i) = m_i] \geq \frac{2}{3}$.
2. On every input $i \in [k]$, $D^r(i)$ always makes at most l queries to r.

We call ρ the fraction of errors decodable, or the decoding radius, and we call l the query complexity.

Definition 2.3 (Local correction)**.** Let $\mathcal{C} \subseteq \Sigma^n$ be a code. A (ρ, l)-*local correction algorithm for* $\mathcal{C}$ is a randomized algorithm $C : [n] \to \Sigma$ with oracle access to an input word $r \in \Sigma^n$ and satisfies the following:

1. If there is a codeword $c \in \mathcal{C}$ such that $\delta(c, r) \leq \rho$, then for every input $i \in [n]$, we have $\Pr[C^r(i) = c_i] \geq \frac{2}{3}$.
2. On every input $i \in [n]$, $C^r(i)$ always makes at most l queries to r.

As before, ρ is the decoding radius and l is the query complexity.

The definition and construction of interpolating sets is motivated by the fact that if we have an explicit interpolating set for a code $\mathcal{C}$, then we have an explicit systematic encoding for $\mathcal{C}$, which allows us to easily transform a local correction algorithm into a local decoding algorithm.

Definition 2.4 (List decoding)**.** Let $\mathcal{C} \subseteq \Sigma^n$ be a code. A (ρ, L)-*list decoding algorithm for* $\mathcal{C}$ is an algorithm which takes as input a received word $r \in \Sigma^n$ that outputs a list $\mathcal{L} \subseteq \Sigma^n$ of size $|\mathcal{L}| \leq L$ containing all $c \in \mathcal{C}$ such that $\delta(c, r) \leq \rho$. The parameter ρ is the *list-decoding radius* and L is the *list size.*

Definition 2.5 (Local list decoding)**.** Let $\mathcal{C} \subseteq \Sigma^n$ be a code. A (ρ, L, l)-*local list decoding algorithm for* $\mathcal{C}$ is a randomized algorithm A with oracle access to an input word $r \in \Sigma^n$ and outputs a collection of randomized oracles $A_1, \ldots, A_L$ with oracle access to r satisfying the following:

1. With high probability, it holds that for every $c \in \mathcal{C}$ such that $\delta(c, r) \leq \rho$, there exists a $j \in [L]$ such that for every $i \in [n]$, $\Pr[A_j^r(i) = c_i] \geq \frac{2}{3}$.
2. A makes at most l queries to r, and on any input $i \in [n]$ and for every $j \in [L]$, A_j^r makes at most l queries to r.

As before, ρ is the *list decoding radius*, L is the *list size*, and l is the *query complexity.*

2.3 Affine-invariant codes

Definition 2.6 (Affine-invariant code)**.** A code $\mathcal{C} \subseteq \{\mathbb{F}_Q^m \to \mathbb{F}_q\}$ is *affine-invariant* if for every $f \in \mathcal{C}$ and affine permutation $A : \mathbb{F}_Q^m \to \mathbb{F}_Q^m$, the function $x \mapsto f(A(x))$ is in $\mathcal{C}$.

Definition 2.7 (Degree set)**.** For a function $f : \mathbb{F}_Q \to \mathbb{F}_q$, written as $f = \sum_{d=0}^{Q-1} f_d X^d$, its *support* is $\mathrm{supp}(f) := \{d \in \{0, \ldots, Q-1\} \mid f_d \neq 0\}$. If $\mathcal{C} \subseteq \{\mathbb{F}_Q \to \mathbb{F}_q\}$ is an affine-invariant code, then its *degree set* $\mathrm{Deg}(\mathcal{C})$ is

$$\mathrm{Deg}(\mathcal{C}) := \bigcup_{f \in \mathcal{C}} \mathrm{supp}(f).$$

Proposition 2.8 ([BGM+11])**.** *If $\mathcal{C} \subseteq \{\mathbb{F}_{q^m} \to \mathbb{F}_q\}$ is a linear affine-invariant code, then $\dim_{\mathbb{F}_q}(\mathcal{C}) = |\mathrm{Deg}(\mathcal{C})|$.*

In particular, if S is an interpolating set for an affine-invariant code $\mathcal{C} \subseteq \{\mathbb{F}_{q^m} \to \mathbb{F}_q\}$, then $|S| = |\mathrm{Deg}(\mathcal{C})|$. Proposition 2.8 will be used in Appendix A.

2.4 Lifted codes

Definition 2.9 (Lift)**.** Let $\mathcal{C} \subseteq \{\mathbb{F}_q \to \mathbb{F}_q\}$ be an affine-invariant code. For integer $m \geq 2$, the *m-th dimensional lift of $\mathcal{C}$*, $\mathrm{Lift}_m(\mathcal{C})$, is the code

$$\mathrm{Lift}_m(\mathcal{C}) := \{f : \mathbb{F}_q^m \to \mathbb{F}_q \mid f|_\ell \in \mathcal{C} \text{ for every line } \ell \text{ in } \mathbb{F}_q^m\}$$

Let $\mathrm{RS}(q, d)$ be the Reed-Solomon code of degree d over $\mathbb{F}_q$,

$$\mathrm{RS}(q, d) := \{f : \mathbb{F}_q \to \mathbb{F}_q \mid \deg(f) \leq d\}.$$

Definition 2.10 (Lifted Reed-Solomon code)**.** The *m-variate lifted Reed-Solomon code of degree d over $\mathbb{F}_q$* is the code

$$\mathrm{LiftedRS}(q, d, m) := \mathrm{Lift}_m(\mathrm{RS}(q, d)).$$

For positive integers d, e, we say *e is in the p-shadow of d*, or $e \leq_p d$, if d dominates e digit-wise in base p: in other words, if $d = \sum_{i \geq 0} d^{(i)} p^i$ and $e = \sum_{i \geq 0} e^{(i)} p^i$ are the p-ary representations, then $e^{(i)} \leq d^{(i)}$ for all $i \geq 0$. We define the notion of p-shadow for vectors recursively as follows. A vector $(e_1, \ldots, e_m)$ is in the p-shadow of d, denoted by $(e_1, \ldots, e_m) \leq_p d$, if $e_1 \leq_p d$ and $(e_2, \ldots, e_m) \leq_p d - e_1$. It follows easily from the definition that if $(e_1, \ldots, e_m) \leq_p d$, then $\sum_{i=1}^m e_i \leq d$. The following fact motivates these definitions.

Proposition 2.11 (Lucas' theorem)**.** *Let $e_1, \ldots, e_m$ be positive integers and $d = e_1 + \cdots + e_m$ and let p be a prime. The multinomial coefficient $\binom{d}{e_1, \ldots, e_m} = \frac{d!}{e_1! \cdots e_m!}$ is nonzero modulo p if only if $(e_1, \ldots, e_m) \leq_p d$.*

For integers $a \geq 0$ and $Q > 1$, we define the mod-star operator by $a \pmod{^* Q} = 0$ if $a = 0$ and $a \pmod{^* Q} = b \in [Q-1]$ if $a \neq 0$ and $a \equiv b \pmod{Q-1}$. This is motivated by the fact that X^d defines the same function as $X^{d \pmod{^* q}}$ over $\mathbb{F}_q$.

Remark 2.12. For $b \in [Q-1]$, note that $a \pmod{^* Q} \leq b$ if and only if there is some integer $k \geq 0$ such that $a \in [k \cdot (Q-1) + 1, k \cdot (Q-1) + b]$.

Proposition 2.13 ([GKS13])**.** *The lifted Reed-Solomon code $\mathrm{LiftedRS}(q, d, m)$ is spanned by monomials $\prod_{i=1}^m X_i^{d_i}$ such that for every $e_i \leq_p d_i$, $i \in [m]$, we have $\sum_{i=1}^m e_i \pmod{^* q} \leq d$.*

Proposition 2.14 ([GKS13])**.** *The lifted Reed-Solomon code $\mathrm{LiftedRS}(q, d, m)$ has distance*

$$\delta(\mathrm{LiftedRS}(q, d, m)) \geq \delta(\mathrm{RS}(q, d)) - q^{-1}.$$

2.5 Finite field isomorphisms

Let $\mathrm{Tr} : \mathbb{F}_{q^m} \to \mathbb{F}_q$ be the $\mathbb{F}_q$-linear trace map $z \mapsto \sum_{i=0}^{m-1} z^{q^i}$. Let $\alpha_1, \ldots, \alpha_m \in \mathbb{F}_{q^m}$ be linearly independent over $\mathbb{F}_q$ and let $\phi : \mathbb{F}_{q^m} \to \mathbb{F}_q^m$ be the map $z \mapsto (\mathrm{Tr}(\alpha_1 z), \ldots, \mathrm{Tr}(\alpha_m z))$. Since Tr is $\mathbb{F}_q$-linear, ϕ is an $\mathbb{F}_q$-linear map and in fact it is an isomorphism. Observe that ϕ induces a $\mathbb{F}_q$-linear isomorphism $\phi^* : \{\mathbb{F}_q^m \to \mathbb{F}_q\} \to \{\mathbb{F}_{q^m} \to \mathbb{F}_q\}$ defined by $\phi^*(f)(x) = f(\phi(x))$ for all $x \in \mathbb{F}_{q^m}$.

3 Global list decoding

In this section, we present an efficient global list decoding algorithm for $\mathrm{LiftedRS}(q, d, m)$. Define $\alpha_1, \ldots, \alpha_m \in \mathbb{F}_{q^m}$, ϕ, and ϕ^* as in Section 2.5. The key new structural result, Theorem 3.2, states that $\mathrm{LiftedRS}(q, d, m) \subseteq \{\mathbb{F}_q^m \to \mathbb{F}_q\}$ is isomorphic to a subcode of $\mathrm{RS}(q^m, (d+m)q^{m-1}) \subseteq \{\mathbb{F}_{q^m} \to \mathbb{F}_q\}$. In particular, this lets us list decode $\mathrm{LiftedRS}(q, d, m)$ by list decoding $\mathrm{RS}(q^m, (d+m)q^{m-1})$ up to the Johnson radius. We will use this algorithm for $m = 2$ as a subroutine in our local list decoding algorithm in Section 4.

3.1 Lifted Reed-Solomon codes are subcodes of Reed-Solomon codes

We begin with a lemma on monomials in lifted Reed-Solomon codes. We postpone the proof of this lemma to Section 3.2.

Lemma 3.1. *Let $h_1, \ldots, h_m$ satisfy $\prod_{i=1}^m X_i^{h_i} \in \mathrm{LiftedRS}(q, d, m)$, where $d < q - m$. Write $\sum_{i=1}^m h_i = a(q-1) + b$, where $0 \le a \le m$ and $0 \le b \le d$. Then $a \le b$.*

We now state and prove our main structural theorem, which shows that codewords of an m-variate lifted Reed-Solomon code over $\mathbb{F}_q$ are low degree when viewed as univariate polynomials over $\mathbb{F}_{q^m}$.

Theorem 3.2. *Let $d < q - m$. If $f \in \mathrm{LiftedRS}(q, d, m)$, then $\deg(\phi^*(f)) \le (d+m)q^{m-1}$.*

Proof. By Proposition 2.13 and linearity, it suffices to prove this for a monomial $f(X_1, \ldots, X_m) = \prod_{i=1}^m X_i^{d_i}$, where $d_1, \ldots, d_m$ have the property that for every $e_1, \ldots, e_m$ with $e_i \le_p d_i$, we have $\sum_{i=1}^m e_i \pmod{^*\, q} \le d$.

For $z \in \mathbb{F}_{q^m}$, by the multinomial theorem we get the following expansion:

$$\begin{aligned}
\phi^*(f)(z) &= f(\phi(z)) \\
&= \prod_{i=1}^{m} (\mathrm{Tr}(\alpha_i z))^{d_i} \\
&= \prod_{i=1}^{m} (\sum_{k=0}^{m-1} (\alpha_i z)^{q^k})^{d_i} \\
&= \prod_{i=1}^{m} \left(\sum_{e_{i,0}, e_{i,1}, \dots, e_{i,m-1} \text{ s.t. } \sum_j e_{i,j} = d_i} \binom{d_i}{e_{i,0}, \dots, e_{i,m-1}} \cdot \prod_{j=1}^{m} (\alpha_i z)^{e_{i,j} q^j} \right) \\
&= \sum_{(e_{i,j})_{1 \le i \le m, 0 \le j \le m-1} \text{ s.t. } \sum_j e_{i,j} = d_i} \left(\prod_i \binom{d_i}{e_{i,0}, \dots, e_{i,m-1}} \prod_j (\alpha_i z)^{e_{i,j} q^j} \right) \\
&= \sum_{(e_{i,j})_{1 \le i \le m, 0 \le j \le m-1} \text{ s.t. } \sum_j e_{i,j} = d_i} \left(\left(\prod_i \binom{d_i}{e_{i,0}, \dots, e_{i,m-1}} \prod_j (\alpha_i)^{e_{i,j} q^j} \right) \cdot z^{\sum_{j=0}^{m-1} (\sum_{i=1}^{m} e_{i,j}) \cdot q^j} \right).
\end{aligned}$$

We now use Lucas' theorem to understand the multinomial coefficients, (in a similar manner to Lemma B.2 and Proposition 2.8 in [GKS13]), and this tells us that many terms in this sum equal 0. So we get that $\phi^*(f)(z)$ is of the form:

$$\phi^*(f)(z) = \sum_{(e_{ij})_{1 \le i \le m, 0 \le j \le m-1} \text{ s.t. } e_{ij} \le_p d_i} (\dots) \cdot z^{\sum_{j=0}^{m-1} (\sum_{i=1}^{m} e_{i,j}) q^j}.$$

To conclude the proof of this theorem, we just need to show that the only monomials z^t that appear in the above expression are all such that $t \pmod{^*} q^m$ is at most $(d+m) \cdot q^{m-1}$. Concretely, we need to show that whenever $(e_{i,j})_{1 \le i \le m, 0 \le j \le m-1}$ satisfy (1) $e_{i,j} \le_p d_i$ for all i, j, and (2) $\sum_{j=0}^{m-1} e_{i,j} = d_i$, then we have the bound

$$E := \sum_{j=0}^{m-1} \left(\sum_{i=1}^{m} e_{i,j} \right) q^j \pmod{^*} q^m \le (d+m) q^{m-1}.$$

Recall that Proposition 2.13 allowed us to assume that $d_1, \dots, d_m$ have the property that for every $e_i \le_p d_i$, $i \in [m]$, we have $\sum_{i=1}^{m} e_i \pmod{^*} q \le d$. Therefore, $\sum_{i=1}^{m} e_{i,m-1} = a(q-1) + b$ for some $0 \le a \le m$ and $0 \le b \le d$.

We now proceed to give upper and lower bounds on E, which will then enable us to show that

$E \pmod{^* q^m} \leq (d+m)q^{m-1}$. We start with the upper bound:

$$\begin{aligned}
E &= q^{m-1}\sum_{i=1}^{m} e_{i,m-1} + \sum_{j=0}^{m-2}\sum_{i=1}^{m} e_{i,j}q^j \\
&\leq q^{m-1}\sum_{i=1}^{m} e_{i,m-1} + q^{m-2}\sum_{j=0}^{m-2}\sum_{i=1}^{m} e_{i,j} \\
&\leq q^{m-1}\cdot(a(q-1)+d) + q^{m-2}\sum_{j=0}^{m-2}\sum_{i=1}^{m} q \\
&= aq^{m-1}(q-1) + (d+m)q^{m-1} \\
&\leq a(q^m-1) + (d+m)q^{m-1}.
\end{aligned}$$

We proceed with the lower bound. If $a = 0$, then $E \geq 0$. Suppose $a \geq 1$. Since $\leq_p$ is transitive, by Proposition 2.13, the monomial $\prod_{i=1}^m X_i^{e_{i,m-1}} \in \mathrm{LiftedRS}(q,d,m)$. Recall that $\sum_{i=1}^m e_{i,m-1} = a(q-1)+b$. Thus by Lemma 3.1, $a \leq b$. Therefore,

$$\begin{aligned}
E &= q^{m-1}\sum_{i=1}^{m} e_{i,m-1} + \sum_{j=0}^{m-2}\sum_{i=1}^{m} e_{i,j}q^j \\
&\geq q^{m-1}\sum_{i=1}^{m} e_{i,m-1} \\
&= q^{m-1}(a(q-1)+b) \\
&= aq^m + (b-a)q^{m-1} \\
&= a(q^m-1) + (b-a)q^{m-1} + a \\
&\geq a(q^m-1) + 1.
\end{aligned}$$

To summarize, if $a = 0$, then $0 \leq E \leq (d+m)\cdot q^{m-1}$, and if $a \geq 1$, then $a(q^m-1)+1 \leq E \leq a(q^m-1)+(d+m)q^{m-1}$. In both cases, we get that $E \pmod{^* q^m} \leq (d+m)q^{m-1}$, as desired. □

Corollary 3.3. *There is a polynomial time global list decoding algorithm for* $\mathrm{LiftedRS}(q,d,m)$ *that decodes up to* $1-\sqrt{\frac{d+m}{q}}$ *fraction errors. In particular, if* $m = O(1)$ *and* $d = (1-\delta)q$*, then* $\delta(\mathrm{LiftedRS}(q,d,m)) = \delta - o(1)$ *and the list decoding algorithm decodes up to* $1-\sqrt{1-\delta}-o(1)$ *fraction errors as* $q \to \infty$*.*

Proof. Given $r : \mathbb{F}_q^m \to \mathbb{F}_q$, convert it to $r' = \phi^*(r) \in \mathbb{F}_{q^m}$, and then run the Guruswami-Sudan list decoder for $\mathrm{RS} := \mathrm{RS}(q^m, (d+m)q^{m-1})$ on r' to obtain a list $\mathcal{L}$ with the guarantee that any $f \in \mathrm{RS}$ with $\delta(r', f) \leq 1-\sqrt{\frac{d+m}{q}}$ lies in $\mathcal{L}$. We require that any $f \in \mathrm{LiftedRS}(q,d,m)$ satisfying $\delta(r,f) \leq 1-\sqrt{\frac{d+m}{q}}$ also satisfies $\phi^*(f) \in \mathcal{L}$, and this follows immediately from Theorem 3.2. The fact that $\delta(\mathrm{LiftedRS}(q,d,m)) = \delta - o(1)$ when $m = O(1)$ and $q = (1-\delta)q$ follows immediately from Proposition 2.14. □

3.2 Proof of Lemma 3.1

We begin with three simple claims about the $\leq_p$ relation.

Claim 3.4. *If $e \leq_p h_1 + \cdots + h_m$, then there exist $e_1, \ldots, e_m$ such that $e_i \leq_p h_i$ for each $i \in [m]$ and $e_1 + \cdots + e_m = e$.*

Proof. The coefficient of X^e in $(1+X)^{h_1+\cdots+h_m}$ is $\sum_{e_1+\cdots+e_m=e} \prod_{i=1}^m \binom{h_i}{e_i}$. By Proposition 2.11, the hypothesis implies that this coefficient is nonzero modulo p, hence there is some choice of $e_1 + \cdots + e_m = e$ such that $\prod_{i=1}^m \binom{h_i}{e_i}$ is nonzero modulo p. By Proposition 2.11, $e_i \leq_p h_i$ for each $i \in [m]$. ☐

Claim 3.5. *Let $c \geq 1$ and $k \leq p^c/2$. If $0 \leq x \leq p^c - 2k + 1$, then there exists $0 \leq i \leq k-1$ such that $x + i \leq_p p^c - k$.*

Proof. Let $n := p^c - k$. We have the identity

$$\binom{n+k-1}{x+k-1} = \sum_{i=0}^{k-1} \binom{n}{x+i}\binom{k-1}{i}$$

from the fact that the LHS counts the number of ways of choosing $x+k-1$ elements from $[n+k-1]$, whereas the RHS counts the same thing by picking $x+i$ elements from $[n]$ and picking $(k-1)-i$ elements from $\{n+1, \ldots, n+k-1\}$. The LHS is $\binom{p^c-1}{x+k-1} \not\equiv 0 \pmod{p}$ by Proposition 2.11. Using the identity above, there must be some i such that $\binom{p^c-k}{x+i} = \binom{n}{x+i} \not\equiv 0 \pmod{p}$. Again, by Proposition 2.11, $x + i \leq_p p^c - k$. ☐

Claim 3.6. *If $N = a(q-1) + b$, where $1 \leq b < a \leq m$ and q is a power of prime p, then there exists $e \leq_p N$ such that $b < e \leq m$.*

Proof. Write $q = p^s$ and $a = p^c - r$, where $0 \leq r < p^c$. Then $N = aq - p^c + r + b = (a-1)q + (p-1)\sum_{i=c}^{s-1} p^i + (r+b)$. But $r + b = p^c - (a-b) < p^c$, therefore $r + b \leq_p N$. Therefore, it suffices to find $e \leq_p r + b$ such that $b < e \leq a \leq m$. If $r + b \leq a$, then we can simply take $e := r + b$. Otherwise, if $a < r + b$, then $a - b < p^c/2$, for if not, then $a \geq p^c/2$ and $r + b = p^c - (a-b) \leq p^c/2$ and therefore $r + b \leq a$, a contradiction. By Claim 3.5, there exists $i \in [a-b]$ such that $b + i \leq_p p^c - (a-b) = r + b$. Set $e := b + i$. ☐

We can now complete the proof of Lemma 3.1.

Proof of Lemma 3.1. If $a = 0$, then the result trivially holds. Suppose $a \geq 1$. Then $b \geq 1$. Suppose, for the sake of contradiction, that $a > b$. By Claim 3.6, there exists $e \leq_p h_1 + \cdots + h_m$ such that $b < e \leq m$. By Claim 3.4, there exist $e_1, \ldots, e_m$ such that $e_i \leq_p h_i$ for $i \in [m]$ and $e_1 + \cdots + e_m = e$. For $i \in [m]$, define $b_i := h_i - e_i$. Then $b_i \leq_p h_i$, and so by Proposition 2.13 we have $\sum_{i=1}^m b_i \pmod{^* q} \leq d$. On the other hand, $\sum_{i=1}^m b_i = \sum_{i=1}^m h_i - \sum_{i=1}^m e_i = a(q-1) + b - e$. We can lower bound this by

$$a(q-1) + b - e \geq a(q-1) + b - m \geq (a-1)(q-1) + q - m > (a-1)(q-1) + d$$

and upper bound this by

$$a(q-1) + b - e \leq a(q-1) - 1 < (a-1)(q-1) + (q-1)$$

and so $\sum_{i=1}^m b_i \pmod{^* q} > d$, a contradiction. ☐

4 Local list decoding

In this section, we present a local list decoding algorithm for LiftedRS(q, d, m), where $d = (1-\delta)q$ which decodes up to radius $1 - \sqrt{1-\delta} - \epsilon$ for any constant $\epsilon > 0$, with list size poly$(\frac{1}{\epsilon})$ and query complexity q^3.

Local list decoder: Oracle access to received word $r : \mathbb{F}_q^m \to \mathbb{F}_q$.

1. Pick a random line ℓ in $\mathbb{F}_q^m$.
2. Run Reed-Solomon list decoder (e.g. Guruswami-Sudan) on $r|_\ell$ from $1 - \sqrt{1-\delta} - \frac{\epsilon}{2}$ fraction errors to get list $g_1, \ldots, g_L : \mathbb{F}_q \to \mathbb{F}_q$ of Reed-Solomon codewords.
3. For each $i \in [L]$, output $\mathsf{Correct}(A_{\ell,g_i})$

where $\mathsf{Correct}$ is a local correction algorithm for the lifted codes for 0.1δ fraction errors, and A is an oracle which takes as advice a line and a univariate polynomial and simulates oracle access to a function which is supposed to be $\ll 0.1\delta$ close to a lifted RS codeword.

Oracle $A_{\ell,g}(x)$:

1. If ℓ contains x, i.e. $\ell = \{a + bt \mid t \in \mathbb{F}_q\}$ for some $a, b \in \mathbb{F}_q^m$ and $x = a + bt$, then output $g(t)$.
2. Otherwise, let P be the plane containing ℓ and x, parametrized by $\{a+bt+(x-a)u \mid t, u \in \mathbb{F}_q\}$.
   (a) Use the global list decoder for bivariate lifted RS code given above to list decode $r|_P$ from $1 - \sqrt{1-\delta} - \frac{\epsilon}{2}$ fraction errors and obtain a list $\mathcal{L}$.
   (b) If there exists a unique $h(t, u) \in \mathcal{L}$ such that $h|_\ell = g$, output $h(0, 1)$, otherwise fail.

Analysis: To show that this works, we just have to show that, with high probability over the choice of ℓ, for every lifted RS codeword f such that $\delta(r, f) \leq 1 - \sqrt{1-\delta} - \epsilon$, there is $i \in [L]$ such that $\mathsf{Correct}(A_{\ell,g_i}) = f$, i.e. $\delta(A_{\ell,g_i}, f) \leq 0.1\delta$.

We will proceed in two steps:

1. First, we show that with high probability over ℓ, there is some $i \in [L]$ such that $f|_\ell = g_i$.
2. Next, we show that $\delta(A_{\ell,f|_\ell}, f) \leq 0.1\delta$.

For the first step, note that $f|_\ell \in \{g_1, \ldots, g_L\}$ if $\delta(f|_\ell, r|_\ell) \leq 1 - \sqrt{1-\delta} - \frac{\epsilon}{2}$. Note that $\delta(f|_\ell, r|_\ell)$ has mean $1 - \sqrt{1-\delta} - \epsilon$ with variance less than $\frac{1}{q}$ (by pairwise independence of points on a line), so by Chebyshev's inequality the probability that $\delta(f|_\ell, r|_\ell) \leq 1 - \sqrt{1-\delta} - \frac{\epsilon}{2}$ is $1 - o(1)$.

For the second step, we want to show that $\Pr_{x \in \mathbb{F}_q^m}[A_{\ell,f|_\ell}(x) \neq f(x)] \leq 0.1\delta$. First consider the probability when we randomize ℓ as well. We get $A_{\ell,f|_\ell}(x) = f(x)$ as long as $f|_P \in \mathcal{L}$ and no element $h \in \mathcal{L}$ has $h|_\ell = f|_\ell$. With probability $1 - o(1)$, ℓ does no contain x, and conditioned on this, P is a uniformly random plane. It samples the space $\mathbb{F}_q^m$ well, so with probability $1 - o(1)$ we have $\delta(f|_P, r|_P) \leq 1 - \sqrt{1-\delta} - \frac{\epsilon}{2}$ and hence $f|_P \in \mathcal{L}$. For the probability that no two codewords in $\mathcal{L}$ agree on ℓ, view this as first choosing P, then choosing ℓ within P. The list size $|\mathcal{L}|$ is a constant, polynomial in $1/\epsilon$. So we just need to bound the probability that two bivariate lifted RS codewords agree on a uniformly random line. The key observation is that every line of agreement must divide the difference of the two bivariate polynomials, which has degree $2q$. Thus there are at most $2q$ such lines, and so the probability that a uniformly random line is one of these lines is at most $2/q$. Thus, with probability $1 - o(1)$, $f|_P$ is the unique codeword in $\mathcal{L}$ which is consistent with $f|_\ell$ on ℓ. Therefore,

$$\begin{aligned}\Pr_{\ell}\left[\delta(A_{\ell,f|_\ell}, f|_\ell) > 0.1\delta\right] &= \Pr_{\ell}\left[\Pr_{x}[A_{\ell,f|_\ell}(x) \neq f(x)] > 0.1\delta\right] \\ &\leq \frac{\Pr_{\ell,x}[A_{\ell,f|_\ell}(x) \neq f(x)]}{0.1\delta} \\ &= o(1).\end{aligned}$$

As a corollary, we get the following testing algorithm.

Theorem 4.1. *For any $\alpha < \beta < 1 - \sqrt{1-\delta}$, there is an $O(q^4)$-query algorithm which, given oracle access to a function $r : \mathbb{F}_q^m \to \mathbb{F}_q$, distinguishes between the cases where r is α-close to* $\mathrm{LiftedRS}(q, d, m)$ *and where r is β-far.*

Proof. Let $\rho = (\alpha + \beta)/2$ and let $\epsilon = (\beta - \alpha)/8$, so that $\alpha = \rho - 4\epsilon$ and $\beta = \rho + 4\epsilon$. Let T be a local testing algorithm for $\mathrm{LiftedRS}(q, d, m)$ with query complexity q, which distinguishes between codewords and words that are ϵ-far from the code. The algorithm is to run the local list decoding algorithm on r with error radius ρ such that $\alpha < \rho < \beta$, to obtain a list of oracles $M_1, \ldots, M_L$. For each M_i, we use random sampling to estimate the distance between r and the function computed by M_i to within ϵ additive error, and keep only the ones with estimated distance less than $\rho + \epsilon$. Then, for each remaining M_i, we run T on M_i. We accept if T accepts some M_i, otherwise we reject.

If r is α-close to $\mathrm{LiftedRS}(q, d, m)$, then it is α-close to some codeword f, and by the guarantee of the local list decoding algorithm there is some $j \in [l]$ such that M_j computes f. Moreover, this M_j will not be pruned by our distance estimation. Since f is a codeword, this M_j will pass the testing algorithm and so our algorithm will accept.

Now suppose r is β-far from $\mathrm{LiftedRS}(q, d, m)$, and consider any oracle M_i output by the local list decoding algorithm and pruned by our distance estimation. The estimated distance between r and the function computed by M_i is at most $\rho + \epsilon$, so the true distance is at most $\rho + 2\epsilon$. Since r is β-far from any codeword, that means the function computed by M_i is $(\beta - (\rho + 2\epsilon)) > \epsilon$-far from any codeword, and hence T will reject M_i.

All of the statements made above were deterministic, but the testing algorithm T and distance estimation are randomized procedures. However, at a price of constant blowup in query complexity, we can make their failure probabilities arbitrarily small constants, so that by a union bound the distance estimations and tests run by T simultaneously succeed with large constant probability. $\square$

Acknowledgements

We thank the anonymous reviewers for their helpful and insightful comments.

References

[AS03] Sanjeev Arora and Madhu Sudan. Improved low-degree testing and its applications. *Combinatorica*, 23:365–426, 2003.

[BET10] Avraham Ben-Aroya, Klim Efremenko, and Amnon Ta-Shma. Local list decoding with a constant number of queries. In *51th Annual IEEE Symposium on Foundations of Computer Science, FOCS*, pages 715–722, 2010.

[BGM+11] Eli Ben-Sasson, Elena Grigorescu, Ghid Maatouk, Amir Shpilka, and Madhu Sudan. On sums of locally testable affine invariant properties. In *APPROX-RANDOM*, pages 400–411, 2011.

[BK09] K. Brander and S. Kopparty. List-decoding Reed-Muller over large fields upto the Johnson radius. *Manuscript*, 2009.

[BL14] Abhishek Bhowmick and Shachar Lovett. List decoding Reed-Muller codes over small fields. *CoRR*, abs/1407.3433, 2014.

[GKS13] A. Guo, S. Kopparty, and M. Sudan. New affine-invariant codes from lifting. In *ITCS*, pages 529–540, 2013.

[GKZ08] Parikshit Gopalan, Adam R. Klivans, and David Zuckerman. List-decoding Reed-Muller codes over small fields. In *Proceedings of the 40th Annual ACM Symposium on Theory of Computing, STOC*, pages 265–274, 2008.

[Gop10] Parikshit Gopalan. A Fourier-Analytic approach to Reed-Muller decoding. In *51th Annual IEEE Symposium on Foundations of Computer Science, FOCS*, pages 685–694, 2010.

[GS99] Venkatesan Guruswami and Madhu Sudan. Improved decoding of Reed-Solomon and algebraic-geometric codes. *IEEE Transactions on Information Theory*, 45:1757–1767, 1999.

[Guo13] A. Guo. High rate locally correctable codes via lifting. *Electronic Colloquium on Computational Complexity (ECCC)*, 20:53, 2013.

[HOW13] B. Hemenway, R. Ostrovsky, and M. Wootters. Local correctability of expander codes. In *ICALP (1)*, pages 540–551, 2013.

[KLP68] Tadao Kasami, Shu Lin, and W. Wesley Peterson. Polynomial codes. *IEEE Transactions on Information Theory*, 14(6):807–814, 1968.

[Kop12] S. Kopparty. List-decoding multiplicity codes. In *Electronic Colloquium on Computational Complexity (ECCC)*, TR12-044, 2012.

[KSY11] S. Kopparty, S. Saraf, and S. Yekhanin. High-rate codes with sublinear-time decoding. In *STOC*, pages 167–176, 2011.

[PW04] R. Pellikaan and X. Wu. List decoding of q-ary Reed-Muller codes. *IEEE Transactions on Information Theory*, 50(4):679–682, 2004.

[STV99] Madhu Sudan, Luca Trevisan, and Salil Vadhan. Pseudorandom generators without the XOR lemma. In *39th ACM Symposium on Theory of Computing (STOC)*, pages 537–546, 1999.

[Vid10] Michael Viderman. A note on high-rate locally testable codes with sublinear query complexity. *Electronic Colloquium on Computational Complexity (ECCC)*, 17:171, 2010.

A Interpolating set for affine-invariant codes

In this section, we present, for any affine-invariant code $\mathcal{C} \subseteq \{\mathbb{F}_q^m \to \mathbb{F}_q\}$, an explicit interpolating set $S_{\mathcal{C}} \subseteq \mathbb{F}_q^m$, i.e. for any $\widehat{f} : S_{\mathcal{C}} \to \mathbb{F}_q$ there exists a unique $f \in \mathcal{C}$ such that $f|_{S_{\mathcal{C}}} = \widehat{f}$.

Define $\alpha_1, \ldots, \alpha_m \in \mathbb{F}_{q^m}$, ϕ, and ϕ^* as in Section 2.5. It is straightforward to verify that if $\mathcal{C} \subseteq \{\mathbb{F}_q^m \to \mathbb{F}_q\}$ and $S \subseteq \mathbb{F}_{q^m}$ is an interpolating set for $\phi^*(\mathcal{C})$, then $\phi(S)$ is an interpolating set for $\mathcal{C}$.

Theorem A.1. *Let $\mathcal{C} \subseteq \{\mathbb{F}_q^m \to \mathbb{F}_q\}$ be a nontrivial affine-invariant code with $\dim_{\mathbb{F}_q}(\mathcal{C}) = D$. Let $\omega \in \mathbb{F}_{q^m}$ be a generator, i.e. ω has order $q^m - 1$. Let $S = \{\omega, \omega^2, \ldots, \omega^D\} \subseteq \mathbb{F}_{q^m}$. Then $\phi(S) \subseteq \mathbb{F}_q^m$ is an interpolating set for $\mathcal{C}$.*

Proof. The map ϕ induces a map $\phi^* : \{\mathbb{F}_q^m \to \mathbb{F}_q\} \to \{\mathbb{F}_{q^m} \to \mathbb{F}_q\}$ defined by $\phi^*(f) = f \circ \phi$. It suffices to show that S is an interpolating set for $\mathcal{C}' \triangleq \phi^*(\mathcal{C})$. Observe that $\mathcal{C}'$ is affine-invariant over $\mathbb{F}_{q^m}$, and let $\mathrm{Deg}(\mathcal{C}') = \{i \mid \exists f \in \mathcal{C} \;\; i \in \mathrm{supp}(f)\}$. By Proposition 2.8, $\dim_{\mathbb{F}_q}(\mathcal{C}') = |\mathrm{Deg}(\mathcal{C}')|$, so suppose $\mathrm{Deg}(\mathcal{C}') = \{i_1, \ldots, i_D\}$. Every $g \in \mathcal{C}'$ is of the form $g(z) = \sum_{j=1}^{D} a_j z^{i_j}$, where $a_j \in \mathbb{F}_{q^m}$. By linearity, it suffices to show that if $g \in \mathcal{C}'$ is nonzero, then $g(z) \neq 0$ for some $z \in S$. We have

$$\begin{bmatrix} \omega^{i_1} & \omega^{i_2} & \cdots & \omega^{i_D} \\ \omega^{2i_1} & \omega^{2i_2} & \cdots & \omega^{2i_D} \\ \vdots & \vdots & \ddots & \vdots \\ \omega^{Di_1} & \omega^{Di_2} & \cdots & \omega^{Di_D} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_D \end{bmatrix} = \begin{bmatrix} g(\omega) \\ g(\omega^2) \\ \vdots \\ g(\omega^D) \end{bmatrix}$$

and the leftmost matrix is invertible since it's a generalized Vandermonde matrix. Therefore, if $g \neq 0$, then the right-hand side, which is simply the vector of evaluations of g on S, is nonzero. ☐

B Local unique decoding upto half minimum distance

Theorem B.1. *Let $\mathcal{C} \subseteq \{\mathbb{F}_q \to \mathbb{F}_q\}$ be an affine-invariant code of distance δ. For every positive integer $m \geq 2$ and for every $\epsilon > 0$, there exists a local correction algorithm for $\mathrm{Lift}_m(\mathcal{C})$ with query complexity $O(q/\epsilon^2)$ that corrects up to $\left(\frac{1}{2} - \epsilon\right)\delta - \frac{1}{q}$ fraction errors.*

Proof. Let $\mathrm{Corr}_{\mathcal{C}}$ be a correction algorithm for $\mathcal{C}$, so that for every $f : \mathbb{F}_q \to \mathbb{F}_q$ that is $\delta/2$-close to some $g \in \mathcal{C}$, $\mathrm{Corr}_{\mathcal{C}}(f) = g$. The following algorithm is a local correction algorithm that achieves the desired parameters.

Local correction algorithm: Oracle access to received word $r : \mathbb{F}_q^m \to \mathbb{F}_q$.

On input $x \in \mathbb{F}_q^m$:

1. Let $c = \lceil \frac{4 \ln 6}{\epsilon^2} \rceil$ and pick $a_1, \ldots, a_c \in \mathbb{F}_q^m$ independently and uniformly at random.
2. For each $i \in [c]$:
   (a) Set $r_i(t) := r(x + a_i t)$.

(b) Compute $s_i := \mathrm{Corr}_{\mathcal{C}}(r_i)$ and $\delta_i := \delta(r_i, s_i)$.

(c) Assign the value $s_i(0)$ a weight $W_i := \max\left(1 - \frac{\delta_i}{\delta/2}, 0\right)$.

3. Set $W := \sum_{i=1}^{c} W_i$. For every $\alpha \in \mathbb{F}_q$, let $w(\alpha) := \frac{1}{W} \sum_{i:s_i(0)=\alpha} W_i$. If there is an $\alpha \in \mathbb{F}_q$ with $w(\alpha) > \frac{1}{2}$, output α, otherwise fail.

Analysis: Fix a received word $r : \mathbb{F}_q^m \to \mathbb{F}_q$ that is $(\tau - \frac{1}{q})$-close from a codeword $c \in \mathrm{Lift}_m(\mathcal{C})$, where $\tau = \left(\frac{1}{2} - \epsilon\right)\delta$. The query complexity follows from the fact that the algorithm queries $O(1/\epsilon^2)$ lines, each consisting of q points. Fix an input $x \in \mathbb{F}_q^m$. We wish to show that, with probability at least 2/3, the algorithm outputs $c(x)$, i.e. $w(c(x)) > \frac{1}{2}$.

Consider all lines ℓ passing through x. For each such line ℓ, define the following:

$$\begin{aligned}
\tau_\ell &:= \delta(r|_\ell, c|_\ell) \\
s_\ell &:= \mathrm{Corr}_{\mathcal{C}}(r|_\ell) \\
\delta_\ell &:= \delta(r|_\ell, s_\ell) \\
W_\ell &:= \max\left(1 - \frac{\delta_\ell}{\delta/2}, 0\right) \\
X_\ell &= \begin{cases} W_\ell & s_\ell = c|_\ell \\ 0 & s_\ell \neq c|_\ell. \end{cases}
\end{aligned}$$

Let $p := \Pr_\ell[s_\ell = c|_\ell]$. Note that if $s_\ell = c|_\ell$, then $\delta_\ell = \tau_\ell$, otherwise $\delta_\ell \geq \delta - \tau_\ell$. Hence, if $s_\ell = c|_\ell$, then $W_\ell \geq 1 - \frac{\tau_\ell}{\delta/2}$, otherwise $W_\ell \leq \frac{\tau_\ell}{\delta/2} - 1$.

Define

$$\begin{aligned}
\tau_{\mathrm{good}} &= \mathbb{E}[\tau_\ell \mid s_\ell = c|_\ell] \\
\tau_{\mathrm{bad}} &:= \mathbb{E}[\tau_\ell \mid s_\ell \neq c|_\ell] \\
W_{\mathrm{good}} &:= \mathbb{E}[W_\ell \mid s_\ell = c|_\ell] \geq 1 - \frac{\tau_{\mathrm{good}}}{\delta/2} \\
W_{\mathrm{bad}} &:= \mathbb{E}[W_\ell \mid s_\ell \neq c|_\ell] \leq \frac{\tau_{\mathrm{bad}}}{\delta/2} - 1.
\end{aligned}$$

Observe that

$$\begin{aligned}
\mathbb{E}[\tau_\ell] &\leq \frac{1 + (\tau - \frac{1}{q})(q-1)}{q} \leq \tau \\
\mathbb{E}[X_\ell] &= p \cdot W_{\mathrm{good}} \\
\mathbb{E}[W_\ell] &= p \cdot W_{\mathrm{good}} + (1-p) \cdot W_{\mathrm{bad}}.
\end{aligned}$$

We claim that

$$p \cdot W_{\mathrm{good}} \geq (1-p) \cdot W_{\mathrm{bad}} + 2\epsilon. \tag{1}$$

To see this, we start from

$$\left(\frac{1}{2} - \epsilon\right)\delta = \tau \geq \mathbb{E}[\tau_\ell] = p \cdot \tau_{\mathrm{good}} + (1-p) \cdot \tau_{\mathrm{bad}}.$$

Dividing by $\delta/2$ yields

$$1 - 2\epsilon \geq p \cdot \frac{\tau_{\text{good}}}{\delta/2} + (1-p) \cdot \frac{\tau_{\text{bad}}}{\delta/2}.$$

Re-writing $1 - 2\epsilon$ on the left-hand side as $p + (1-p) - 2\epsilon$ and re-arranging, we get

$$p \cdot \left(1 - \frac{\tau_{\text{good}}}{\delta/2}\right) \geq (1-p) \cdot \left(\frac{\tau_{\text{bad}}}{\delta/2} - 1\right) + 2\epsilon.$$

The left-hand side is bounded from above by $p \cdot W_{\text{good}}$ while the right-hand side is bounded from below by $(1-p) \cdot W_{\text{bad}} + 2\epsilon$, hence (1) follows.

For each $i \in [c]$, let ℓ_i be the line $\{x + a_i t \mid t \in \mathbb{F}_q\}$. Note that the X_ℓ are defined such that line i contributes weight $\frac{X_{\ell_i}}{W}$ to $w(c(x))$, so it suffices to show that, with probability at least $2/3$,

$$\frac{\sum_{i=1}^{c} X_{\ell_i}}{\sum_{i=1}^{c} W_{\ell_i}} > \frac{1}{2}.$$

Each $X_\ell, W_\ell \in [0,1]$, so by Hoeffding's inequality,

$$\begin{aligned}
\Pr\left[\left|\frac{1}{c}\sum_{i=1}^{c} X_{\ell_i} - \mathbb{E}[X_\ell]\right| > \epsilon/2\right] &\leq \exp(-\epsilon^2 c/4) \leq 1/6 \\
\Pr\left[\left|\frac{1}{c}\sum_{i=1}^{c} W_{\ell_i} - \mathbb{E}[W_\ell]\right| > \epsilon/2\right] &\leq \exp(-\epsilon^2 c/4) \leq 1/6.
\end{aligned}$$

Therefore, by a union bound, with probability at least $2/3$ we have, after applying (1),

$$\begin{aligned}
\frac{\sum_{i=1}^{c} X_i}{\sum_{i=1}^{c} W_i} &\geq \frac{\mathbb{E}[X_\ell] - \epsilon/2}{\mathbb{E}[W_\ell] + \epsilon/2} \\
&= \frac{p \cdot W_{\text{good}} - \epsilon/2}{p \cdot W_{\text{good}} + (1-p) \cdot W_{\text{bad}} + \epsilon/2} \\
&\geq \frac{(1-p) \cdot W_{\text{bad}} + 3\epsilon/2}{2(1-p) \cdot W_{\text{bad}} + 5\epsilon/2} \\
&> \frac{1}{2}
\end{aligned}$$

where the second to last inequality follows from (1) and the fact that if $a < b$ and $x \leq y$, then $\frac{x+a}{x+b} \leq \frac{y+a}{y+b}$ (here $a = -\epsilon/2$, $b = (1-p) \cdot W_{\text{bad}} + \epsilon/2$, $x = (1-p) \cdot W_{\text{bad}} + 2\epsilon$, and $y = p \cdot W_{\text{good}}$). $\square$