Electronic Colloquium on Computational Complexity, Report No. 148 (2012)

A new family of locally correctable codes based on degree-lifted algebraic geometry codes

Eli Ben-Sasson [*] Ariel Gabizon [†] Yohay Kaplan [‡] Swastik Kopparty [§]
Shubhangi Saraf [¶]

November 7, 2012

Abstract

We describe new constructions of error correcting codes, obtained by "degree-lifting" a short algebraic geometry (AG) base-code of block-length q to a lifted-code of block-length q^m, for arbitrary integer m. The construction generalizes the way degree-d, univariate polynomials evaluated over the q-element field (also known as Reed-Solomon codes) are "lifted" to degree-d, m-variate polynomials (Reed-Muller codes). Three properties are established:

Rate The rate of the degree-lifted code is approximately a $\frac{1}{m!}$-fraction of the rate of the base-code.

Distance The relative distance of the degree-lifted code is at least as large as that of the base-code. This is proved using a generalization of the Schwartz-Zippel Lemma to degree-lifted Algebraic-Geometry codes (Lemma 5.6) .

As a first concrete example, lifting the AG codes of Garcia and Stichtenoth [J. Number Theory '96] results in Reed-Muller-like codes but over constant sized alphabets, such codes are interesting in the search for probabilistically checkable proofs (PCPs) with constant rate and polynomially small query complexity.

Local correction If the base code is invariant under a group that is "close" to being doubly-transitive (in a precise manner defined later , cf. Definition 6.1) then the degree-lifted code is locally correctable with query complexity at most q^2. The automorphisms of the base-code are crucially used to generate query-sets, abstracting the use of affine-lines in the local correction procedure of Reed-Muller codes.

Taking a second concrete example, we show that degree-lifted Hermitian codes form a family of locally correctable codes over an alphabet that is significantly smaller than that obtained by Reed-Muller codes of similar constant rate, message length, and distance.

[*]Department of Computer Science, Technion, Haifa, Israel and MIT CSAIL, Cambridge, MA. eli@cs.technion.ac.il. The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement number 240258.

[†]Department of Computer Science, Technion, Haifa. ariel.gabizon@gmail.com The research leading to these results has received funding from the European Union's Seventh Framework Programme under grant agreement no. 259426 ERC Cryptography and Complexity.

[‡]Department of Computer Science, Technion, Haifa. yohayk@cs.technion.ac.il. Research supported by a grant of the US-Israel Binational Science Foundation

[§]Department of Mathematics and Computer Science, Rutgers University. swastik.kopparty@rutgers.edu

[¶]Department of Mathematics and Computer Science, Rutgers University. shubhangi.saraf@gmail.com

ISSN 1433-8092

Contents

1 Introduction

We describe a new family of error correcting codes which share many aspects of Reed-Muller codes: Rate, existence of a Schwartz-Zippel Lemma, and notions of "degree" and "restricting to a line" which have numerous applications to computational complexity. The main advantage of our constructions is that their alphabet size is significantly smaller than previously known "low-degree" codes, and this aspect is particularly relevant in the search for new (and better) constructions of PCPs (cf. Section 1.6).

1.1 Locally correctable codes (LCCs)

Error correcting codes as envisioned in the early days of information theory [Sha48, Sha53, Ham50] were mostly intended to be produced and consumed in blocks. A message m consisting of k symbols coming from a finite alphabet Σ is encoded by a codeword w of $n \gg k$ symbols — n is the *block-length* of the code, k/n is its *rate* — and sent across a noisy channel, resulting in a corrupted word w'. To access any one message symbol m_i, a decoding procedure is applied to *all* n symbols of the received word w'. Maintaining data-integrity is done block-wise in a similar way, say, by first decoding the message m from w' and then re-encoding m to recover w.

The development of new randomized and interactive proof systems in the 80's called for error-correcting codes of a different nature (see Section 2.1 and the survey [Yek11]). The message and its encoding are now assumed to be either secret, or prohibitively long, and hence decoding/correcting a full block is either forbidden (due to secrecy) or intractable (because of its large block-length). In such settings the notions of locally decodable and locally correctable codes were distilled, because often it suffices to find out the value of a single message-symbol m_i, or a single codeword symbol w_i. To better discuss LCCs we assume oracle access to entries of the (possibly corrupted) codeword w' and accordingly henceforth view a code $\mathcal{C}$ as a set of functions $\mathcal{C} = \{w : D \to \Sigma\}$ mapping a domain D of size n to the alphabet Σ, and use $w(i)$ to denote the ith entry of w.

A locally-correctable code $\mathcal{C}$ is associated with a *local corrector*. This is a randomized procedure that is given as input a pointer $i \in D$ and has oracle access to a corrupted codeword $w' : D \to \Sigma$ that is within "small" relative Hamming distance δ of a codeword w (think of $\delta = 0.01$). The local corrector queries w' in a small number q of locations (q is called the *query complexity* of the corrector) and outputs a conjectured value $\hat{w}(i)$ for the "true" value $w(i)$ of the ith entry of the uncorrupted word w. We say the corrector has *soundness error* ϵ for *distance parameter* δ if $\Pr[\hat{w}(i) = w(i)] \geq 1-\epsilon$ holds for all $i \in D$ and all functions w' that are within normalized Hamming distance δ of w. (Probability in the previous equation is over the randomness of the local corrector.) A code $\mathcal{C}$ that has such a local corrector is called a (q, ϵ, δ)-LCC, and when we want to highlight the query complexity we call $\mathcal{C}$ a q-LCC, assuming ϵ and δ are known. (See Section 2.1 for a brief survey of LCC constructions).

1.2 On local correction, code symmetries, and local views

For a code $\mathcal{C}$ to be a q-LCC, every index $i \in D$ should be associated with a *smooth* set of *q-local reconstructors* (cf. [KT00])[1]. A q-local reconstructor for i is a reconstruction function that can be computed by making q queries to a received word w'. More formally, fix $D' \subset D$ with $|D'| = q$ and

[1]The results of [KT00] formally apply to the case of locally decodable codes, but the proofs can be examined and seen to hold for the more general case of locally correctable codes.

a function $r : \Sigma^q \to \Sigma$. We say that (D', r) is a *q-local reconstructor for $i \in D$* if for all[2] codewords $w \in \mathcal{C}$, we have $r(w|_{D'}) = w(i)$ (where $w|_{D'}$ is the restriction of the function w to the domain D'). Roughly speaking, a set of q-local reconstructors $\{(D'_1, r_1), \ldots, (D'_t, r_t)\}$ for $i \in D$ is said to be *smooth* if sampling a random $j \in [t]$ and then sampling a random $\ell \in D'_j$ gives a distribution that is close (in statistical distance) to uniform over $D \setminus \{i\}$.

Requiring a smooth set of q-local reconstructors for every $i \in D$ calls for codes with quite a lot of structure. The sheer number of constraints — n/q per index, summing up to n^2/q overall — implies that picking these constraints arbitrarily, or at random, results (whp) in a code with rate 0. Indeed, all known families of LCCs — Reed-Muller (RM), multiplicity codes [KSY11], and affine-invariant codes [KS08, GS12] — can be explained by two structural properties they possess: *(i)* a *doubly-transitive automorphism group*, and *(ii)* a *large-distance local view*. We explain these two concepts next.

A code $\mathcal{C}$ induces a group of automorphisms $\mathrm{Aut}(\mathcal{C})$. This is the group of permutations π of D that keep the code invariant, i.e., $(w \circ \pi) \in \mathcal{C}$ for all $w \in \mathcal{C}$ where $(w \circ \pi)$ is the function (or codeword) defined by $w(i) \triangleq w(\pi(i))$ for all $i \in D$. A group G acting on D is *doubly-transitive* if for any two pairs of distinct elements $(i, j), (i', j') \in D^2$ there exists $\pi \in G$ mapping i to i' and j to j'.

Fix $w \in \mathcal{C}$. A *q-local view* of w is the restriction $w|_{D'}$ of w to a domain $D' \subset D$ with $|D'| = q$. Similarly, a *q-local view* of $\mathcal{C}$ is a set of the form $\mathcal{C}|_{D'} \triangleq \{w|_{D'} \mid w \in \mathcal{C}\}$ for some $D' \subset D$ with $|D'| = q$. That is, $\mathcal{C}|_{D'}$ is the projection of all codewords of $\mathcal{C}$ to a domain $D' \subset D$ of size q. We informally say the local-view has *large distance* if the relative distance of $\mathcal{C}|_{D'}$ is large. 'fi

To take a concrete example, consider $\mathrm{RM}[m, d]_q = \left\{f : \mathbb{F}_q^m \to \mathbb{F}_q \mid \deg(f) \leq d\right\}$ and suppose $d = q/2$. *(i)* This code is doubly-transitive because it is affine-invariant, i.e., it is invariant under any invertible affine transformation A of $\mathbb{F}_q^m$, as $\deg(f) \leq d$ implies $\deg(f \circ A) \leq d$. *(ii)* The code also has a large-distance q-local view, namely, the view $D' = \{(\alpha, 0, \ldots, 0) \mid \alpha \in \mathbb{F}_q\}$ obtained by fixing all but the first variable to 0. Clearly $\mathrm{RM}[m, d]_q|_{D'}$ has relative distance $d/q = 1/2$ because this view is nothing but the well-known Reed-Solomon (RS) code $\mathrm{RS}[d]_q$ consisting of univariate degree-d polynomials over $\mathbb{F}_q$. [3] The reason we mention these two properties is because *any* code that is *(i)* doubly-transitive, and *(ii)* has a q-local view D' with relative distance δ, is, in fact, a $(q, 0.1, \delta/20)$-LCC [KV10]. In order to gain a better understanding of the fundamental notion of local correctability it is helpful to explore other constructions of LCCs, that do not possess these properties. As additional motivation we point out that doubly transitive codes, and in particular affine-invariant ones, have inherent coding-related limitations, e.g., their rate is very small when alphabet size and query complexity are fixed [BS05].

1.3 The tensor-product lifting of codes

One way to go about constructing LCCs is to start with a "small" code and "lift" it. A "small" code $\mathcal{C}$ is one with small block-length q but large rate ρ and large relative distance δ. Lifting $\mathcal{C}$ means we apply a simple algebraic or combinatorial operation to $\mathcal{C}$ and obtain as a result a code $\mathcal{C}'$ with larger block-length $n \gg q$. We want to find such lifting operations with the property that certain q-local views of $\mathcal{C}'$ will be equal to $\mathcal{C}$. This will be useful for showing $\mathcal{C}'$ 'has some local structure' and is a good LCC.

[2] One can relax this requirement to hold only for *almost* all codewords, but in the context of linear error correcting codes (most LCCs are such) the two notions coincide.

[3] The other known families of LCCs, namely, affine-invariant codes [KS08, GS12] and multiplicity codes [KSY11], are also *(i)* affine-invariant and *(ii)* have a large-distance local view.

Perhaps the simplest conceivable lifting process is *tensoring* [MS78].

Definition 1.1 (Axis Parallel Views and Code Tensors)**.**

An *axis-parallel subset* $D' \subset D^m$ is a set of the form

$$D' = \{(c_1, \ldots, c_{i-1})\} \times D \times \{(c_{i+1}, \ldots, c_m)\}, \tag{1}$$

for some $i \in m$ and $c_1, \ldots, c_{i-1}, c_{i+1}, \ldots, c_m \in D$. Fix a function $w : D^m \to \Sigma$. An *axis-parallel view* of w is a restriction $w|_{D'}$ where $D' \subset D^m$ is an axis-parallel subset. Let $\mathcal{C} = \{w : D \to \Sigma\}$ be a code. The *m-wise tensor* of $\mathcal{C}$, denoted $\mathcal{C}^{\otimes m}$, is the set of functions $w : D^m \to \Sigma$ whose every axis-parallel view is in $\mathcal{C}$.

For example, it can be seen that $(\mathrm{RS}[d]_q)^{\otimes m}$ is the set of m-variate polynomials of *individual* degree at most d. Therefore, $(\mathrm{RS}[d]_q)^{\otimes m}$ is a strict subcode of $\mathrm{RM}[m, md]_q$. It can be verified that if $\mathcal{C}$ is a linear code of rate ρ and relative distance δ, then $\mathcal{C}^{\otimes m}$ is a linear code of rate ρ^m and relative distance δ^m. The large-distance axis-parallel views of $\mathcal{C}^{\otimes m}$ can be used to show that $\mathcal{C}^{\otimes m}$ is a *locally testable code* (LTC), i.e., there exists a randomized "tester" that makes $|D|^{O(1)}$ queries to $f : D^m \to \Sigma$ and distinguishes with high probability between the case that $f \in \mathcal{C}^{\otimes m}$ and the case that f is very far from $\mathcal{C}^{\otimes m}$ [BS05, Val05, CR05, DSW06, BV09a, BV09b].

Unfortunately, code-tensoring fails as a general "lifting" method for constructing LCCs. The problem is that the only large-distance local-views we can put our hands on in $\mathcal{C}^{\otimes m}$ are the axis-parallel views and these do not correspond to a smooth set of reconstructors. There are only two axis-parallel lines that pass through (i, j) in $\mathcal{C} \otimes \mathcal{C}$ and similarly, only m axis-parallel lines pass through $(i_1, \ldots, i_m)$ in $\mathcal{C}^{\otimes m}$. This is regrettable because if $\mathcal{C}$ has a rich automorphism group, then it seems reasonable to expect that $\mathcal{C}^{\otimes m}$ has many non-axis-parallel views that are large-distance local views, as explained next. We end by pointing out that in spite of the limitations of tensoring for local correctability, we shall return to it later on to show that in some cases, the m-wise tensor of a nonlinear base-code *is* locally correctable (see 6.3).

1.4 Degree-lifting of codes

Let D be some finite domain, and $\mathbb{F}$ be a finite field. Suppose we have a degree function deg assigning a non-negative integer $\deg(f)$ to functions $f : D \to \mathbb{F}$. For ease of notation, for the rest of the introduction we fix a positive integer d which will be implicit in some definitions. We denote by $\mathcal{C}_d$ the the set of functions of degree at most d. That is, $\mathcal{C}_d = \{f : D \to \mathbb{F} | \deg(f) \leq d\}$.

The point we make in this section is that if $\deg(f)$ 'behaves like the degree of univariate polynomials', we can lift $\mathcal{C}_d$ to a code $\mathcal{C}'$ of larger block-length, that will be convenient to analyze as an LCC. For this purpose the following definition will be useful

Definition 1.2. We say deg is a *curve-degree on* D if the following properties hold.

1. $\deg(f \cdot g) = \deg(f) + \deg(g)$.
2. If $\deg(f) = d$, f either vanishes on at most d points in D, or else it vanishes on all of D.
3. For any $a, b \in \mathbb{F}$ and any $f, g : D \mapsto \mathbb{F}$, $\deg(a \cdot f + b \cdot g) \leq \max\{\deg(f), \deg(g)\}$. Thus, the set of functions $f : D \mapsto \mathbb{F}$ with $\deg(f) \leq d$ is an $\mathbb{F}$-linear subspace.
4. If $\pi \in \mathrm{Aut}(\mathcal{C})$, then $\deg(f \circ \pi) \leq \deg(f)$.

It can be seen that when $D = \mathbb{F}$ and deg is the degree function of univariate polynomials, deg is a curve-degree on $\mathbb{F}$. The only property that is not immediate is property 4: In this case $\mathrm{Aut}(\mathcal{C}_d)$ consists of the affine transformations $\pi(X) = a \cdot X + b$ for some $a, b \in \mathbb{F}$ with $a \neq 0$. And indeed, if $f(X)$ is a univariate polynomial of degree d, so is $f'(X) \triangleq f(a \cdot X + b)$. The fact that deg behaves like the degree function of univariate polynomials, suggests the idea of lifting $\mathcal{C}_d$ to a code $\mathcal{C}'$ consisting of multivariate functions of a certain 'total degree'. Denote by B the set of m-variate 'monomials' of total degree at most d. Namely,

$$B \triangleq \{g_1(X_1) \cdots g_m(X_m) | g_i \in \mathcal{C}_d; \sum_{j=1}^{m} \deg g_j \leq d\}.$$

Definition 1.3 (Degree-lifted Code)**.** Using the notation above, the m-variate *degree-lifted* code $\mathcal{C}_d^m$ of $\mathcal{C}_d$ is

$$\mathcal{C}_d^m \triangleq \mathrm{span}\{B\}.$$

The properties of curve-degree suggest a way to use automorphisms of $\mathcal{C}_d$ to locally correct $\mathcal{C}_d^m$. It relies on the following fundamental definition.

Definition 1.4 ($\mathcal{C}$-permissible subsets and views)**.** Fix a code $\mathcal{C} = \{w : D \to \Sigma\}$ and integer m. A *$\mathcal{C}$-permissible subset* of D^m is a subset $D' \subset D^m$ that is of one of the following forms.

- D' is an axis-parallel subset as per Definition 1.1.
- $D' = \{(\pi_1(i), \ldots, \pi_m(i)) \mid i \in D\}$, for some $\pi_1, \ldots, \pi_m \in \mathrm{Aut}(\mathcal{C})$.

For a code $\mathcal{C}' = \{w : D^m \to \Sigma\}$ a *$\mathcal{C}$-permissible view* of $\mathcal{C}'$ is a view of the form $\mathcal{C}'|_{D'}$ for a *$\mathcal{C}$-permissible subset* $D' \subset D^m$.

The usefulness of this definition for local correction stems from the following claim.

Claim 1.5. *Let* deg *be a curve-degree on D. Then the $\mathcal{C}_d$-permissible views of $\mathcal{C}_d^m$ are subsets of $\mathcal{C}_d$.*

Proof. Let $D' \subset D^m$ be a $\mathcal{C}_d$-permissible subset. Recall that $\mathcal{C}_d^m$ is the span of functions $w : D^m \to \mathbb{F}$ of the form $w = g_1(X_1) \cdots g_m(X_m)$, with $\sum_{j=1}^{m} deg(g_j) \leq d$. Thus, it suffices to show that for w of this form, $w|_{D'} \in \mathcal{C}_d$. Fix such w, and denote $w' = w|_{D'}$. Suppose D' is an axis-parallel subset. Then,

$$w'(X) = c_1 \cdots c_{j-1} \cdot g_j(X) \cdot c_{j+1} \cdots c_m = c \cdot g_j(X),$$

for some $c \in \mathbb{F}$. Thus, as a function from D to $\mathbb{F}$, $\deg(w') \leq d$ and $w' \in \mathcal{C}_d$. If D' is not an axis-parallel subset then w' is of the form

$$w'(X) = g_1(\pi_1(X_1)) \cdots g_m(\pi_m(X_1))$$

for some $\pi_1, \ldots, \pi_m \in \mathrm{Aut}(\mathcal{C}_d)$. From Property 4 of curve-degree, $\deg(g_j \circ \pi_j) \leq \deg(g_j)$ for every $j \in [m]$. It follows that $deg(w') \leq d$ and $w' \in \mathcal{C}_d$. □

The above claim suggests the following reconstruction procedure for $\mathcal{C}_d^m$: Suppose we need to locally correct $i \in D^m$ and have oracle access to a corrupted function $w' : D^m \to \Sigma$. We pick a random $\mathcal{C}_d$-permissible subset D' in D^m that contains i, correct $w'|_{D'}$ to the closest word $v \in \mathcal{C}_d$ and output the value assigned to i by v.

Loosely speaking, when the $\mathcal{C}_d$-permissible subsets are 'well distributed' in D^m it can be shown that this correction procedure succeeds with high probability . It can be seen that this will be the case, for instance, when $\mathcal{C}_d$ is doubly-transitive. I.e., when $\mathrm{Aut}(\mathcal{C}_d)$ is a 2-transitive group (see Section 6.2) .

1.5 Lifting of affine-invariant codes

A form of code lifting that is very similar to ours (but is subtly different) was introduced for affine-invariant codes in [BMSS10], and used there to prove that affine-invariant low-density-parity-check (LDPC) codes are not necessarily locally testable (cf. Definition 4.1 there). Recently, and independently of this work, this form of "affine lifting" was shown in [GS12] to lead to new constructions of codes that are simultaneously locally testable and locally correctable, and have parameters that essentially match those of multiplicity codes [KSY11].

In both kinds of lifting — "degree" and "affine" — one starts with a code $\mathcal{C} = \{f : D \to \mathbb{F}\}$ and ends with a code $\mathcal{C}' = \{f : D^m \to \mathbb{F}\}$ that has the property that every $\mathcal{C}$-permissible view of $\mathcal{C}'$ is a subset of $\mathcal{C}$ but there are important differences between the two. Some of the differences are syntactic: "affine lifting" assumes an affine-invariant code, and is defined using the "degree-set characterization" of affine invariant properties (cf. [GS12, Definition 2.1]) whereas "degree lifting" assumes an algebraic degree function. Certainly some codes that posses a curve-degree are not affine invariant (e.g., the Hermitian code described later), and in the other direction it is not clear that all affine invariant codes can be defined as a space of low curve-degree for some curve-degree function (cf. Definition 1.2).

The difference between the two notions of lifting runs deeper. For the case of affine invariant code $\mathcal{C}$, a function $f : D^m \to \mathbb{F}$ belongs to the affine lifting of $\mathcal{C}$ if and only if every $\mathcal{C}$-permissible view of it — i.e., every restriction of it to a 1-dimensional affine space — is a codeword of $\mathcal{C}$ ([GS12, Proposition 2.5]). In contrast, we know of algebraic geometry codes $\mathcal{C}$ for which there exist functions $f : D^m \to \mathbb{F}$ that do not belong to $\mathcal{C}'$ because their curve-degree is too large, yet every $\mathcal{C}$-permissible view of them belongs to $\mathcal{C}$ (see Section 3.7) , and in some cases even the lifting of RS codes will exhibit this behavior[FS95].

One final and crucial difference that we point out is that in the case of degree lifted codes, we do not assume that $\mathcal{C}$ is doubly transitive, and this complicates the process of locally correcting $\mathcal{C}$-lifted codes, as explained next.

1.6 AG codes and degree-lifted AG codes

The previous three subsections motivate the following questions: Besides $D = \mathbb{F}$ and deg being the standard degree of univariate polynomials, are there domains D for which we can define a curve-degree? Are there such cases where the resulting code $\mathcal{C}_d$ will be doubly transitive? We have seen that a positive answer to these questions will give us new families of locally correctable codes.

We now address the first question. It turns out that algebraic geometry gives us a way to define a curve-degree on D, when D is 'the set of rational points of an irreducible curve'. [4] Indeed, for

[4] See the next two sections for more precise definitions.

these degree functions the resulting codes $\mathcal{C}_d$ are known as algebraic-geometry (AG) codes [5], and have been extensively studied (see Section 4 for details) . In particular, the following results of [MTZ82, GS96] are well-known because they spectacularly beat the Gilbert-Varshamov bound for alphabet size $q \geq 49$. In what follows we say q is an even power of a prime if $q = p^{2\ell}$ for prime p and integer ℓ.

Theorem 1.6 (Asymptotically good AG codes)**.** *For every q that is an even power of a prime and positive constants $\delta, \rho > 0$ satisfying $\delta - \rho < \frac{1}{\sqrt{q}}$ there exists an infinite family of codes over $\mathbb{F}_q$ of increasing block-length $n_1 < n_2 < \ldots$, and the ith member $\mathcal{C}_{d_i}$ of the family (which is of blocklength n_i) contains codewords of curve-degree $d_i \leq \delta n_i$ and has rate at least ρ and relative distance at least $1 - \delta$.*

We use the properties of curve-degree to analyze the basic parameters of $\mathcal{C}_d^m$: We show that if the rate of $\mathcal{C}_d$ is ρ then the rate of $\mathcal{C}_d^m$ is roughly $\rho/m!$. Turning to relative distance, the properties of curve-degree lead to a generalization of the ubiquitous Schwartz-Zippel Lemma for degree-lifted AG codes. This generalization implies that the relative distance of $\mathcal{C}_d^m$ is at least the relative distance of $\mathcal{C}_d$ (Lemma 5.6) . Given the pivotal role of the Schwartz-Zippel Lemma in the study of RM-codes, we hope this generalization will be used to find other applications of degree-lifted AG codes. We stress that both the rate and relative distance of degree-lifted codes depend only on the rate and distance of the base-code, hence can be applied to any AG code with an appropriate degree function.

Taking a concrete example of our results, applying degree-lifting to the codes specified in Theorem 1.6 gives the following result

Corollary 1.7 (lifted codes, constant alphabet)**.** *Taking $\mathcal{C}_{d_i}$ to be a member of the family specified in Theorem 1.6 (and using the notation there), for any integer m the code $\mathcal{C}_{d_i}^m$ from Definition 1.3 has relative distance at least $1 - \delta$ and rate at least $\frac{(\rho - m/\sqrt{q}) \cdot (1 - o(1))}{m!}$.*

Proof. The relative distance follows from Theorem 5.6. The rate follows from Theorem 5.5, using the fact that the genus of the curve over which $\mathcal{C}_{d_i}$ is defined approaches $n_i/\sqrt{q}$ as i goes to infinity. □

Such codes — "low-degree", constant rate and constant alphabet — are interesting in the search for constant-rate PCPs with polynomially small query complexity, i.e., PCPs with parameters similar to those recently obtained for locally decodable codes by [KSY11, GS12] and for locally testable codes by [Vid10]. As a concrete example consider the problem of obtaining a reduction from a circuit-SAT instance ϕ of size n to a PCP instance of size $O(n)$ which has query complexity $O(\sqrt{n})$, perfect completeness and soundness $\frac{1}{2}$. Constructing LCCs with similar parameters (rate and query complexity) is trivial — take, e.g., $\mathcal{C} = \mathrm{RM}[2, \sqrt{2n}]_q$ for prime-power $q \approx 10\sqrt{n}$. But if such a code were to be used in a PCP reduction to encode an assignment to ϕ, which is n *bits* long, the resulting rate would be $O(1/\log n)$ because the alphabet of $\mathcal{C}$ is of size $\approx \sqrt{n}$ (hence a codeword is $\Omega(n \log n)$ bits long). This logarithmic rate-loss does not appear in the standard coding setting, where we can assume our message is over the alphabet $\mathbb{F}_q$. But packing an n-bit assignment to ϕ into $O(n/\log n)$ symbols of $\mathbb{F}_q$ would cause severe problems further down the line of a PCP reduction. Looking for a different solution, e.g., by taking a non-algebraic code instead of $\mathrm{RM}[2, \sqrt{2n}]_q$ — like a two-wise tensor of a high-rate binary code — would not be helpful because in PCP reductions one

[5] Specifically, one-point AG codes

needs "codes with multiplication" (cf. [Mei10, Section 3]). To conclude, constant-rate AG-codes over constant-size alphabets like those in Theorem 1.6, when lifted to constant m (like $m = 2$) as in Corollary 1.7, result in codes that are naturally equipped with multiplication — because of property 1 in Definition 1.2 — and hence are a good starting point for construction constant rate PCPs with polynomially small query complexity.

1.7 Local correction of lifted AG codes with a rich automorphism group

We now address the second question — are there other "curve-degree" codes with a rich automorphism group? We are not aware of any doubly-transitive AG-code, except RS. However, certain AG-codes — for example, Hermitian, Suzuki, and Ree codes — have a rather rich automorphism group that is not quite doubly-transitive. In what follows we describe a number of methods for locally correcting degree-lifted AG codes that have a rich automorphism group, albeit one that is not doubly-transitive. The parameters obtained by them for degree-lifted Hermitian codes are listed in theorem 1.8. Below, we denote by $\mathcal{C}$ a code $\mathcal{C}_d$ for some curve-degree function deg.

Fractal correctors To compensate for the code automorphism group not being exactly 2 transitive, we perform correction in several steps. On input i, w' as above, the fractal corrector first picks a random $\mathcal{C}$-permissible subset D' that contains i. Next, for each $j \in D'$, pick a random $\mathcal{C}$-permissible subset D'_j that passes through j and locally-correct j as explained above. Finally, use the corrected values of all $j \in D'$ to locally correct i as above. While we only performed 2 steps in this example, and we only require 2 steps to locally correct degree lifted Hermitian codes, this procedure can be generalized to any number of steps. The name we chose — "fractal corrector" — is explained by the set of queries this corrector makes, which has the size of a c-dimensional surface in D^m but resembles more a collection of 1-dimensional curves. (Details appear in Section 7 .)

High-degree correctors for degree-lifted codes For certain AG-base codes with a nearly-doubly transitive automorphism group, the query complexity of local-correction can be reduced from $|D|^2$ to $|D|$. The main challenge here is that the $\mathcal{C}$-permissible subsets containing a given $i \in D^m$ do not form a smooth set. The key to our solution is that $\mathcal{C}^m$ contains views that, although not $\mathcal{C}$-permissible, have large distance. For instance, in the case of RM-codes, consider the set of parameterized quadratic curves in $\mathbb{F}_q^m$ of the form

$$D' = \{(P_1(x), \ldots, P_m(x)) \mid x \in \mathbb{F}_q\} \quad \deg(P_i) = 2.$$

These views are clearly not $\mathrm{RS}[d]_q$-permissible, but nevertheless the relative distance of the corresponding view of $\mathrm{RS}[m, d]_q$ is still pretty good (it is $1 - 2\delta$ where $1 - \delta$ is the distance of the base-RS-code). It turns out that for certain AG-base codes, like the Hermitian code, a similar notion of high-degree local views makes sense. The restriction of $\mathcal{C}^m$ to such a high-degree view has relative distance that is smaller than that of $\mathcal{C}$, but is nevertheless sufficiently large to be of use in correction. And the main benefit of using high-degree views is that they can form, in certain cases, a smooth set and hence can be used to locally correct $\mathcal{C}^m$ (details appear in Section 8).

Locally correctable Tensors of AG-base codes We now return to the most basic "lifting" operation — code-tensoring. In Section 6.3 we give a concrete example of a code based on a slight generalization of the standard definition of AG codes. This code is doubly-transitive and its m-wise tensor $\mathcal{C}^{\otimes m}$ is locally correctable. To correct $i \in D^m$ in a corrupted word w' we pick a view D' as

defined in (1.4) that passes through i. Given that $\mathcal{C}$ is doubly-transitive we show that the set of views is smooth, and the only thing left to argue is that the view has good distance.

1.8 Explicit constructions and parameters

In theorem 1.8 we give a number of concrete constructions. These are obtained by taking $\mathcal{C}$ to be the Hermitian code defined in the next section. Notice that the rate and query complexity in all constructions are very close to that of the RM-code, and the block-length is significantly larger. (A different way to phrase this comparison is to say that for a given block-length and rate, the new AG-based LCCs have a much smaller alphabet). . All constructions are over a field of size q that is an even power of a prime (i.e., $q = p^{2\ell}$ for integer ℓ and prime p). For the sake of comparison, m-variate RM-codes similar rate relative distance and blocklength require a much larger alphabet, of size $q^{3/2}$.

Theorem 1.8 (Hermitian constructions)**.** *For any $\delta > 0$, sufficiently large q that is an even power of a prime, and integer m, there exist lifted hermitian-codes of relative distance δ and block-length $q^{3m/2}$ over the alphabet $\mathbb{F}_q$ that are*

1. *Locally correctable at rate $\frac{1}{m!}\left(1-\delta-\frac{m}{\sqrt{q}}\right)$ with query complexity q^3 via fractal correction (cf. Section 3.5).*
2. *Locally correctable at rate $\frac{1}{tm!}\left(1-\delta-\frac{1}{\sqrt{q}}-\frac{m}{t}\right)$ for any $t > 0$ at query complexity $q^{3/2}$ via high-degree correction (cf. Section 3.6).*
3. *Locally correctable at rate $\frac{1}{m^m}\left(1-\delta-\frac{1}{\sqrt{q}}\right)$ with query complexity $q^{3/2}$, based on tensored AG-codes (see full version).*

Using Ree and Suzuki curves [HS90, Ped92] instead of hermitian ones in the theorem above, we can square the blocklength to q^{3m}, leaving other parameters as stated in the theorem above (see future work). Furthermore, if codes similar to the tower-based AG-codes of [GS96] exist, and these codes are also close to doubly transitive, one immediately obtains the following result. (To state it formally we need basic concepts from Algebraic Geometry, these are are explained in Section 4).

Theorem 1.9. *If there exists a tower of curves over $\mathbb{F}_q$, such that the automorphism group of the associated function field is doubly-transitive on the set of rational points of the curve, then there are locally correctable codes of relative distance δ, rate $\frac{1}{m^m}(1-\delta-\gamma)$ over $\mathbb{F}_q$, where γ is a constant dependent on the curve.*

γ is the ratio of curve-genus to number of rational points, it is at least $1/\sqrt{q}$ and for instance for the family of codes of Theorem 1.6 it approaches $1/\sqrt{q}$.

1.9 Future work and open problems

Multiplicity AG codes It is possible to get LCCs of rate approaching 1 by generalizing [KSY11] to this setting, i.e. introducing a multiplicity version of AG and degree lifted AG codes. We will explore this in future work.

Tensored AG codes as locally testable codes It seems reasonable to expect degree lifted AG codes to be locally testable, in the same way that RM codes are. In future work we will present a "plane vs. plane"-like test for these codes, generalizing the works of [RS97, MR08]. It would also be interesting to generalize the "line V. line" tests of [PS94, AS03] to this setting.

Organization of this paper

In the next section we discuss previous research in areas related to our work. Inspired by the exposition in [BATS09], in Section 3 we begin by showing a simple analysis of degree-lifted AG codes based on the family of Hermitian codes, this analysis gives an example of the type of techniques and results featured in this paper, free of sophisticated algebraic-geometry. Of possible independent interest in this section is an elementary proof of a weaker variant of our Schwartz-Zippel type lemma. In Section 4 we provide a brief reminder of the relevant terms regarding AG codes in general, and Hermitian codes in particular. This section is by no means a comprehensive introduction to AG-codes, the uninitiated reader is referred to [Sti93] for such an introduction. In Section 5 we introduce a RM-like lifting of AG-codes and prove the full version of our Schwartz-Zippel type lemma to establish the distance of lifted AG-codes. In Section 6 we study the so-called single-step correction of AG codes. Section 7 discusses constructions based on fractal correctors, and Section 8 studies high-degree correction procedures.

2 Context and history

2.1 Previous work on locally correctable codes

The LCCs introduced here are, in particular, also locally decodable codes (LDCs). (In fact, any linear LCC is an LDC.) While mentioned as early as [Ree54], LDCs became widely studied (though only implicitly) during the 90s, as part of the drive towards constructions of PCPs [BF90, Lip90, LFKN92, Sha92, BFLS91, BFL92, AS98]. Their explicit definition was given in [KT00], and they have remained an object of intense, explicit study in their own right, while also being used in other results in computational complexity [BFNW93, IW97, AS03, STV99, SU05] and cryptography ([CGKS98], for instance). See [Yek11] for a recent survey on locally decodable codes.

The most intense study has been in the area of constant query LDCs. For many years, it was widely believed that constant query LDCs required exponential length ($n = \Theta\left(\exp\left(k^{\alpha}\right)\right)$ for some constant α). A recent series of results [Yek08, Rag07, Efr09, DGY10, BAETS10, IS10, CFL+10] showed this belief to be false by constructing constant query LDCs which have $n = O\left(\exp\left(\exp\left(\log^{\alpha} k\right)\right)\right)$ for a constant α (i.e. sub-exponential length).

Of equal interest is a lower bound on the length of such codes. This area has also seen considerable work [KdW04, GKST02, WdW05, Woo07, DJK+02, Oba02], and already in [KT00] it was shown that no constant query, constant rate LDCs exist. The best known lower bound on the length of constant query LDCs is $\tilde{\Omega}\left(k^{1-\delta(r)}\right)$ (where r is the number of queries and $\delta\left(r\right) < 1$). It is a major open problem to decrease the gap between these lower and upper bounds.

Flipping the question on its head, and looking only at constant rate LDCs, the only known lower bound on the query complexity is $\Omega\left(\log k\right)$ [KT00], while all the actual constructions known use a polynomial ($\Theta\left(k^{\epsilon}\right)$) number of queries.

For many years the only known LDCs with constant rate were Reed-Muller codes, which were

limited to rate $\frac{1}{2}$. This limit was inherent in the fact that Reed-Muller codes must only use polynomials whose degree was smaller than the field size, or the basic property of unique decoding of a codeword would be lost (over $\mathbb{F}_q$, $x \equiv x^q$) . Recently, [KSY11] showed that by evaluating both a polynomial and some of its derivatives, one can start with polynomials of degree greater than field size, thereby increasing the rate while at the same time, keeping the property of local correctability using an algorithm similar to the Reed-Muller correction algorithm. This enables codes of rate approaching 1, that are locally decodable with $O\left(k^{\epsilon}\right)$ queries in the presence of a constant δ (which is determined by the rate and by ϵ) fraction of errors.

2.2 Previous work studying code invariance with respect to "local" properties

The study of the automorphism group of various error correcting codes is an important and well-established branch of classical coding theory, and this is particularly true for the study of AG-codes [Gop88, Sti]. See also the recent works [KW10, KL12] which study LDPC codes with relatively high rate and a rich invariance group.

In the context of "local" codes, most of the work focused on understanding locally testable codes in terms of their automorphism group. Random low-density-parity-check (LDPC) codes are not locally testable [BHR05] and furthermore a rather large set of local views is required for local testability [BGK+09]. [BSS03] showed that cyclic-LTCs cannot have constant rate. [AKK+05] asked whether all doubly-transitive codes with a local constraint are LTCs. [KS08] initiated a study of this question in the context of affine-invariant linear codes and a large body of work has accumulated around this question [GKS09, BS10, BMSS10, BGM+11, BRS12, GS12] (see [Sud10] and references therein).

Turning to locally decodable and correctable codes, we have already pointed out that doubly-transitive codes with a local-view are locally correctable, hence also locally decodable. Works studying the connection between group representation theory and LDC constructions includes [RY07] and [Efr12].

2.3 Previous work on codes over algebraic surfaces

AG codes were introduced by Goppa [Gop82] and famously used to break the Gilbert-Varshamov bound [MTZ82]. Intuitively, AG codes involve the evaluation of appropriately chosen function spaces over the points of a 1-dimensional object in m-dimensional space. In the wake of Goppa's work, there have been several works on codes over various high dimensional algebraic surfaces ([TS91, Lac93, Han01, rod03, LGS05] for example. See [Lit08] for a survey). To the best of our knowledge, this is the first application of such high-dimensional AG-codes to theoretical computer science.

3 Elementary Construction

In this section we survey the constructions and results featured in the rest of the paper, using only elementary terms and techniques. In particular, we give an elementary statement and proof of a special case of our Schwartz-Zippel type lemma, which we use here to show that a specific instance of degree-lifted AG codes has good distance, and an elementary description of our correcting methods and their relation to the traditional local correction of RM codes.

We wish to emulate the local correction of RM codes and assume the reader has familiarity with the way it is done. Thus we describe our construction hand-in-hand with an informal description of RM-correction. We use a somewhat non-standard definition of RS- and RM-codes in order to expose their similarities with AG-codes. Since we are going to keep the description on an elementary level (without assuming any background in algebraic geometry) certain statements and definitions regarding AG-codes may seem a bit arbitrary. The reader interested in getting to the bottom of these peculiarities is encouraged to study the algebraic geometry necessary for it (cf. [Sti93]).

3.1 Reed-Solomon vs. Hermitian codes

The Reed-Solomon code $\mathrm{RS}[r]_q$ is a $[q, r+1, q-(r+1)]_q$-linear code, i.e., it is a $(r+1)$-dimensional subspace of $\mathbb{F}_q^q$. To define a basis for this code/subspace let $f_i : \mathbb{F}_q \to \mathbb{F}_q$ be the "raising-to-power-i" function, i.e., $f_i(a) = a^i$. The standard basis for $\mathrm{RS}[r]_q$ is then $\{f_i \mid i = 0, 1, \ldots, r\}$.

The Hermitian code $\mathrm{herm}[r]_{q^2}$ can be similarly described as a linear code over $\mathbb{F}_{q^2}$ whose codewords are evaluations of certain functions over a certain set of points. The set of points is different, it is the following subset of $\mathbb{F}_{q^2} \times \mathbb{F}_{q^2}$ which is known as — borrowing Algebraic-Geometry terminology — the set of $\mathbb{F}_{q^2}$-rational points of the Hermitian curve:

$$N_H := \left\{(a, b) \in \mathbb{F}_{q^2} \times \mathbb{F}_{q^2} \mid a^{q+1} = b^q + b\right\}. \tag{2}$$

To define a basis for the code/subspace $\mathrm{herm}[r]_{q^2}$ let g_{ij} be the "raising-X-to-power-i-and-Y-to-power-j" function, i.e., $g_{ij}(a, b) = a^i b^j$. We define the *curve degree* of g_{ij} to be $i \cdot q + j \cdot (q+1)$ and define the following basis for $\mathrm{herm}[r]_{q^2}$:

$$\{g_{ij} \mid qi + (q+1)j \leq r \text{ and } j < q\}.$$

The explanation for the peculiar definition of curve-degree relies on algebraic geometry and is explained later on in Section 4.2 . It suffices to say here that the curve-degree of a function shares a similarity with the standard degree for univariate polynomials — it bounds the number of zeros a function has when evaluated on the domain of interest, which in our case is N_H. To see why this particular choice of the degree-curve makes some sense consider $g_{1,0}(X, Y) = X$, its curve-degree is q and it indeed vanishes on the set $\{(0, b) \mid b^q + b = 0\} \subset N_H$ of size q. So a polynomial of degree 1 in X should have curve-degree at least q. Similarly, the function $(g_{0,1} - 1)(X, Y) = Y - 1$ vanishes on $\left\{(a, 1) \mid a^{q+1} = 1\right\} \subset N_H$ which has size $q + 1$, the same size as the curve-degree of $(g_{0,1} - 1)$, implying that a polynomial of degree 1 in Y should have curve-degree at least $q + 1$. The coding properties of the Hermitian code are recorded next without proof (for a partial elementary proof see [BATS09]).

Theorem 3.1 (Hermitian Codes)**.** *The Hermitian code $\mathrm{herm}[r]_{q^2}$, where $r \geq q^2$, is an $[q^3, r - \frac{q(q-1)}{2}, q^3 - r]_{q^2}$-linear code.*

Comparing the Reed-Solomon and Hermitian codes of block-length q^3 and rate $\frac{1}{2}$ — these are $\mathrm{RS}[\frac{q^3}{2}]_{q^3}$ and $\mathrm{herm}[\frac{q^3+q(q-1)}{2}]_{q^2}$ — we see that the Hermitian code has a smaller alphabet of size q^2 (compared to size q^3 for the RS-code) and its relative distance is also slightly smaller at $\frac{1}{2} - \frac{1}{q}$ (compared with $\frac{1}{2}$ for the RS-code).

3.2 Automorphisms of Reed-Solomon and Hermitian codes

A code $\mathcal{C}$ induces a group of *code-automorphisms* which is instrumental for obtaining locally correctable codes. The automorphism group of $\mathcal{C}$, denoted $\mathrm{Aut}(\mathcal{C})$, is the group of permutations under which the code is invariant. Viewing a codeword w as a function $w : D \to \mathbb{F}_q$ where $|D| = n$ is the block-length of $\mathcal{C}$, and letting S_n be the symmetric group over the domain D, define

$$\mathrm{Aut}(\mathcal{C}) = \{\pi \in S_n \mid \pi(\mathcal{C}) \subseteq \mathcal{C}\}$$

where $\pi(\mathcal{C}) \triangleq \{\pi(w) \mid w \in \mathcal{C}\}$ and $(\pi(w))(x) \triangleq w(\pi(x))$ for all $x \in D$.

It can be verified that $\mathrm{Aut}(\mathrm{RS}[r]_q)$ contains the set of affine transformations over $\mathbb{F}_q$ because if $\deg(f(X)) \leq d$ then also $\deg(f(aX+b)) \leq d$ for any affine map $X \mapsto aX+b$ where $a, b \in \mathbb{F}_q, a \neq 0$. The automorphism group of the Hermitian code is a bit harder to analyse [Xin95]. But it will suffice for our purposes to notice that it contains the following set of permutations:

$$\Phi \triangleq \left\{\phi_{a,b,c}(X,Y) \triangleq \left(aX+b, a^{q+1}Y + b^q aX + c\right) \mid a \in \mathbb{F}_{q^2}^*, (b,c) \in N_H\right\}. \tag{3}$$

We stress a difference between the two automorphism-groups that will pose a challenge when constructing local correctors for AG-LDCs later on: The affine-group is doubly-transitive but Φ is not. (A group of permutations is doubly transitive if for any $i \neq j$ and $i' \neq j'$ in the group there exists a permutation sending i to i' and j to j'.) To see that Φ is not doubly transitive notice its size is $\approx q^5$ whereas the number of pairs of N_H is $\approx q^6$, so Φ is simply too small to be doubly-transitive.

3.3 Reed-Muller vs. degree lifting of Hermitian codes of bounded curve-degree

The m-variate, total-degree r, Reed-Muller code over the field $\mathbb{F}_q$ of size q:

$$\mathrm{RM}[m,r]_q \triangleq \left\{f : \mathbb{F}_q^m \to \mathbb{F}_q \mid \deg(f) \leq r\right\}$$

is the set of evaluations of polynomials of total-degree $\leq r$ over $\mathbb{F}_q^m$. The relative distance of the code is established by the Schwartz-Zippel lemma to be $\delta_0 = \left(1 - \frac{r}{q}\right)$. The standard basis for this linear code is the set of monomials of degree at most r, letting $f_{i_1,\ldots,i_m}(a_1,\ldots,a_m) = a_1^{i_1}\cdots a_m^{i_m}$ this basis is

$$\left\{f_{i_1,\ldots,i_m} \mid \sum_{\ell=1}^{m} i_\ell \leq r\right\}.$$

The following new family of codes is a natural generalization of Reed-Muller codes.

Definition 3.2. [m-wise degree-lifted Hermitian code of curve-degree r]

The code $\mathrm{herm}[m,r]_{q^2}$ is the space of functions that are evaluated on N_H^m and that have total curve-degree at most r. To get a basis for this code let

$$g_{i_1,j_1,\ldots,i_m,j_m}(x_1,y_1,\ldots,x_m,y_m) = x_1^{i_1}y_1^{j_1}\cdots x_m^{i_m}y_m^{j_m}.$$

The basis for $\mathrm{herm}[m,r]_{q^2}$ is

$$\left\{g_{i_1,j_1,\ldots,i_m,j_m} \mid \sum_{\ell=1}^{m} qi_\ell + (q+1)j_\ell \leq r \text{ and } \forall l : j_l < q\right\}$$

To establish the distance of these codes we use the following generalization of the Schwartz-Zippel lemma for our case.

Lemma 3.3 (Schwartz-Zippel for degree lifted Hermitian codes)**.** *For non-zero $g \in herm\,[m,r]_{q^2}$, the probability of g being zero on a random point in N_H^m is at most $\frac{r}{|N_H|}$.*

Proof. By induction on m. The base case is given by Lemma 3.1. For the inductive step pick a random point $\{x_1, y_1, x_2, y_2 \ldots x_m, y_m\} \in N_H^m$. To show the step, let $g = \sum_{i=0}^{r} c_i \cdot g_i$ where g_i is the monomial of curve-degree i in the variables x_m, y_m (i.e. we don't count the other variables for this purpose), and c_i is its coefficient in this representation, this coefficient is a polynomial in $x_1, y_1, \ldots, x_{m-1}, y_{m-1}$. Let j be the highest index for which c_j is non-zero. c_j is a member of $\text{herm}[m-1, r-j]_{q^2}$ so by the inductive hypothesis the probability of it being zero is at most $\frac{r-j}{|N_H|}$. If c_j is non-zero then we are left with a polynomial in x_m, y_m of curve-degree j so the probability of it being zero on a random point in N_H is at most $\frac{j}{|N_H|}$. Summing up, the probability of g being zero is bounded by $\frac{r-j}{|N_H|} + \frac{j}{|N_H|} = \frac{r}{|N_H|}$ as claimed. □

This gives the distance of our codes. Calculating the dimension can be accomplished using elementary combinatorics; it can be shown to be at least $\frac{\left(r-q^2\right)^2}{2}$ as long as $r \geq q^2$.

A more powerful statement of this lemma is inSection 5 . Part of what it shows is that we can replace H by any absolutely irreducible polynomial F, N_H by the solutions to $F = 0$ in the base field and use an appropriately defined degree function and still get that the number of zeroes of a member of the lifted code is bounded by its degree.

To compare the basic error-correcting-code parameters of $\text{RM}[m,r]_{q^3}$ and $\text{herm}\,[m,r]_{q^2}$ of block-length q^{3m} and rate $\approx \frac{1}{m!}$ we see that the former has larger alphabet q^3 and both have distance $q^{3m} - q^{3m-3} \cdot r$. (The distance of the Hermitian code will be slightly worse by an additive factor of about q^{3m-1}).

3.4 Automorphism-based correction of Reed-Muller and degree lifted Hermitian codes

The local-correction procedure for $\text{RM}[m,r]_q$ goes as follows. We are given a point $x \in \mathbb{F}_q^m$ and have oracle access to a function $f : \mathbb{F}_q^m \to \mathbb{F}_q$ which is δ-close to a multivariate polynomial of degree r. Pick a uniformly random line passing through x, read the entries on this line, decode to the closest codeword in $\text{RS}[r]_q$ and output the value assigned to x by this univariate polynomial. Picking a random line through x can also be described by the following process that will better illustrate what happens in the Hermitian case: Pick $m-1$ affine permutations $\sigma_2, \ldots, \sigma_m$ from the affine group of $\mathbb{F}_q$. Project f onto the set of points

$$\ell_\sigma = \{\sigma(\alpha) = (\alpha, \sigma_2(\alpha), \ldots, \sigma_m(\alpha)) \mid \alpha \in \mathbb{F}_q\}$$

noticing ℓ_σ is a line in $\mathbb{F}_q^m$.

Moving to the case of $\text{herm}\,[m,r]_{q^2}$ we employ a similar strategy. Given $f : N_H^m \to \mathbb{F}_{q^2}$ and a point $(x_1, y_1, \ldots, x_m, y_m) \in N_H^m$ that we wish to correct, pick for each $i = 2, \ldots, m$ a random automorphism of $\text{herm}\,[r]_{q^2}$ among all automorphisms that send (x_1, y_1) to (x_i, y_i). Now look at the restriction of the domain N_H^m to the set of points

$$\mathcal{C}_\sigma = \{\sigma(\alpha, \beta) = (\alpha, \beta, \sigma_2(\alpha, \beta), \ldots, \sigma_m(\alpha, \beta)) \mid (\alpha, \beta) \in N_H\}\,.$$

The automorphisms of herm $[r]_{q^2}$ are degree preserving, so the restriction to $\mathcal{C}_\sigma$ is a codeword of herm $[r]_{q^2}$.What is left is to analyse how well using the automorphisms samples the space N_H^m. We perform this analysis in Section 6.1, and get that we can construct tensored Hermitian codes of rate $\frac{1}{m!}\left(1-\delta q^{m-1}-\frac{1}{q}\right)$ that are locally correctable from an δ fraction of errors, while RM codes are locally correctable from an δ fraction of errors at rate $\frac{1}{m!}(1-\delta)$. Again the degree lifted Hermitian codes achieve this with a smaller alphabet but the need for a sub-constant error is undesirable, we have several correction techniques of slightly greater complexity that avoid this.

All of our correction methods are general, in the sense that it is possible to plug in any base AG code, provided its automorphism group satisfies certain properties. We will show that its degree lifting will be locally correctable using our methods. For instance, the general statement of the type of correcting shown in this subsection is (stated slightly informally):

Definition. [(ϵ,α)-doubly transitive groups] The group H acting on the set S is (ϵ,α)-***doubly transitive*** if for every $P_1,P_2\in S$ and at least a $1-\epsilon$ fraction of $P_3\in S$ the variable $X=\sigma(P_3)$ is distributed uniformly on an $1-\alpha$ fraction of S when σ is chosen uniformly at random from the set $\{\sigma|\sigma(P_1)=P_2\}$.

Theorem. *[Single step correction of degree lifted AG codes (informal statement)] Let $\mathcal{C}$ be an $[n,k,d]_q$ AG code such that Aut$(\mathcal{C})$ is (ϵ,α)-*doubly transitive*. Then the procedure described above, when applied to $\mathcal{C}^2$ with a δ-fraction of errors succeeds with probability at least $1-\frac{2}{d}\left(\frac{\delta}{1-\alpha}+\epsilon\right)$.*

So the closer a code's automorphism group is to being two transitive, the better it's degree lifted version does as an LDC. In particular, when an AG code has a 2 transitive automorphism group (as RS codes do) then the simple procedure of passing a random curve through the point of correction works well as a local corrector for any m. For some AG codes, the Hermitian codes among them, we can add some redundancy to the code and get one that is doubly transitive. The particulars of this are beyond our ability to elementarily describe (see sec 6.3) but it gives us the first of three methods we employ to get good LDC's for large m. The other 2 are described below.

3.5 Fractal correction of degree lifted Hermitian codes

In the following procedure we perform two-step correction, or fractal correction, in which we pick a random sampler through the point we wish to correct, and first correct each point in that sampler using "standard" correction (as in the previous section), before using the corrected values to calculate the correct value at the original point. This allows us to better sample the space, at the cost of increasing the query complexity.

Procedure 2:

In order to correct the point $(x_1,y_1,x_2,y_2\dots x_m,y_m)$ do the following:

1. Choose $m-1$ automorphisms such that $\varphi_i(x_1,y_1)=(x_i,y_i)$.
   (a) Let $C=\{(x,y,\varphi_2(x,y)\dots\varphi_m(x,y))\,|\,(x,y)\in D\}$ be the embedding of the Hermitian curve generated by these automorphisms.
   (b) For every point in C, apply standard correction.
   (c) Use the values returned from step 3, apply standard Hermitian decoding to get the restriction of f to C, and calculate it at the point $(x_1,y_1,x_2,y_2\dots x_m,y_m)$.

Theorem. *[Hermitian codes are fractaly correctable]The code defined by evaluating* $herm\left[m, q^3 - c\left(\delta + \frac{m}{q}\right)q^3\right]_{q^2}$ *on* N_H^m *(for* $m > 2$*) is a locally correctable from a* δ*-fraction of errors. It has rate* $\frac{k}{n} \approx \frac{1}{m!}\left(1 - c\delta - \frac{m}{q}\right)$*, query complexity* $\sqrt[m]{n^2}$ *and alphabet size of* $\sqrt[\frac{3m}{2}]{n}$*.*

We note that the Reed-Muller code with the similar rate of $\frac{1}{m!}(1-\delta)$ has an alphabet size and query complexity of $\sqrt[m]{n}$.

The main difficulty in proving this result is in showing that these tests cover the space in a near-uniform manner. We will show this in Section 7 .

3.6 High degree correction of degree lifted Hermitian codes

A different means to overcome the lack of a doubly transitive group is to use "high degree" curves to locally correct a degree lifted Hermitian codeword. The RM-analogue of the process described next would be to locally correct a RM-codeword at point $x \in \mathbb{F}_q^m$ by passing a parametrized low-degree curve through x, of the form

$$\{P_1(\alpha), \ldots, P_m(\alpha) \mid \alpha \in \mathbb{F}_q\}$$

where $P_1, \ldots, P_m$ are low degree polynomials.

In the context of Hermitian codes, we can replace the set Φ (equation 3) with a more general set of functions. This more general set will have the benefit of being 2-transitive (or sufficiently close for our purposes), the downside will be that the restrictions of codewords to such functions will have incur some blow-up in degree. In compensating for this, we lose something in the rate of the codes we get. The details of this construction are presented in Section 8.

3.7 "Affine" lifting vs degree lifting of Hermitian codes

As detailed in Section 1.5 it is possible to consider the notion of "affine" lifting of codes [BMSS10, GS12]. We examine this notion in the context of Hermitian codes. We first note that the $\mathcal{C}$-permissible views for Hermitian codes aren't actually affine but are those generated by the automorphism group of the Hermitian code, so the name "affine" lifting is misleading in this case. We can instead consider the "automorphic" lifting of codes, where we demand that the restriction to each automorphism of the base code is a codeword in the base code. We next give a concrete example that shows that it is very easy to find examples of codewords that would be in the automorphic lifting of the hermitian code but not in its degree lifting. This is in contrast to affine lifting of RS codes which is often (though not always) equal to its degree lifting.

Consider the function $f = x_1 y_1^{q-1} x_2 - x_1 y_2^{q-1} x_2$. Its curve degree is $q^2 + 2q - 1$, and we can see that every axis parallel view of it has lower curve degree. Now consider view of the form $f(x_1, y_1, \varphi(x_1, y_1))$ we want to show that any such view has smaller curve degree than f, so the only terms we care about in $f(x_1, y_1, \varphi(x_1, y_1))$ are the terms of curve degree $\deg_{\mathcal{C}} f$. They are $x_1 y_1^{q-1} a x_1 - x_1 a^{(q+1)(q-1)} y_1^{q-1} a x_1 = a x_1^2 y_1^{q-1} - a^{q^2} x_1^2 y_1^{q-1} = 0$. so any restriction of f is of degree at most $q^2 + q$ (gotten by the second highest degree terms in φ), which means that while f isn't in the degree lifting of $herm[q^2 + q]_{q^2}$ it is in its "affine" lifting.

4 Algebraic Function Fields And Codes

In this section we'll review the relevant terms and notation from the theory of AG codes (Section 3.1) and present the important facts about the Hermitian curve (Section 3.2), in particular we'll study with some thoroughness the structure of the automorphism group of Hermitian codes. We must stress that this section is only a reminder and will not serve as an introduction to AG codes for readers who aren't already familiar with them.

We will (mostly) follow the terminology of [Sti93]. The only non-standard terminology are definitions 4.10 and 4.13.

4.1 AG function fields and codes

We begin with a brief reminder of the important terms relating to AG-codes.

Let $\mathbb{F}_q$ be the field of size q, and $K(x)$ be the **rational function field** in x over K. F/K is an **algebraic function field** F over the **base field** K if F is a finite extension of $K(x)$.

A **valuation ring** $\mathcal{O}_P$ of F is a ring such that $K \subsetneq \mathcal{O}_P \subsetneq F$ and $\forall z \in F : z \in \mathcal{O}_P$ or $z^{-1} \in \mathcal{O}_P$. A **Place** P of a function field is the maximal ideal of a valuation ring. Since P is maximal, $F_P := {}^{\mathcal{O}_P}/_P$ is a field. The **degree** of a place is defined $\deg P := [F_P : K]$, a place of degree 1 is called a **rational** place.

For a rational place we define the following map $F \to K \cup \{\infty\}$:

$$\forall z \in F, z(P) := \begin{cases} z \bmod P & z \in \mathcal{O}_P \\ \infty & z \notin \mathcal{O}_P \end{cases}$$

We denote by v_P the **discrete valuation** associated with the valuation ring $\mathcal{O}_P$.

We note that $v_P(z) > 0$ means that z is 0 at P and we say that z has a **zero** of multiplicity $v_P(z)$ at P. If $v_P(z) < 0$ then z is infinity at P and we say that z has a **pole** of multiplicity $-v_P(z)$ at P.

$\mathcal{D}_F$ is the free abelian group generated by the places of F/K. A **divisor** is a member of this group.

We denote the coefficient of P in D by $v_P(D)$ and define the relation:

$$D_1 \geq D_2 \iff \forall P \in \boldsymbol{P}_F, v_P(D_1) \geq v_P(D_2)$$

A divisor of the form $G = r \cdot P$ for some rational place P is called a **one point** divisor.

The next definition makes sense because every member of F has a finite number of zeroes and poles.

Definition 4.1 (Divisors associated with a function field member)**.** For each $z \in F$ we define:

1. a **principal divisor:**
$$(z) = \sum_{P \in P_F} v_P(z) P$$
2. a **pole divisor:**
$$(z)_\infty = \sum_{P \in P_F, v_P(z) < 0} - v_P(z) P$$

3. a **zero divisor:**

$$(z)_0 = \sum_{P \in P_F, v_P(z) > 0} v_P(z)\, P$$

The **degree** of a divisor D is: $\deg D = \sum_{P \in P_F} n_P \cdot \deg P_i$

Definition 4.2 (Riemann-Roch spaces)**.** The **Riemann-Roch Space** of a divisor D is defined:

$$\mathcal{L}(D) = \{z \in F | (z) \geq -D\} \cup \{0\}$$

This is a finite dimensional K-vector space, the **dimension** of a divisor D is the dimension of its associated Riemann-Roch space and is denoted $l(D)$.

Theorem 4.3 (Riemann-Roch)**.** *For every function field, there is a positive constant g called the **genus** of the function field, for which:*

$$\begin{aligned} \forall A \quad \in \quad & \mathcal{D}_F, \deg A - l(A) \leq g - 1 \\ & if \quad \deg A \geq 2g - 1, \deg A - l(A) = g - 1 \end{aligned}$$

Definition 4.4 (Function field automorphism)**.** A field isomorphism $\phi : F \to F$ is an **automorphism** of the function field F/K if $\forall z \in K : \phi(z) = z$.

Theorem 4.5 (Automorphisms permute places)**.** *[Sti93, Sec 8.1] Let P be a place of the function field F and ϕ an automorphism of it. Then $P^* := \{\phi(z) | z \in P\}$ is a place of F and $\deg P = \deg P^*$.*

This allows us to extend the action of automorphisms to places in the natural way. Which, in turn, allows us to define the action of automorphisms on divisors:

Let $D = \sum_{P \in P_F} n_P P$, then $\phi(D) = \sum_{P \in P_F} n_P \phi(P)$.

Reminder: An $[n, k, d]_q$ *linear code* is a k-dimensional subspace of $\mathbb{F}_q^n$ such that the hamming distance between any two words in the code is at least d.

Theorem 4.6 (AG codes)**.** *[Sti93, Theorem 2.2.2] Let F be a function field with a base field $K = \mathbb{F}_q$. Let $N = P_1, P_2 \ldots P_n$ be a set of rational places, and D be a divisor such that $\forall P_i \in N, v_{P_i}(D) = 0$ and $\deg D < n$. Then*

$$\mathcal{C}_{\mathcal{L}}(N, D) := \{(z(P_i))_{i=1}^n | z \in \mathcal{L}(D)\}$$

is an $[n, k, d]_q$ code with $k = l(D), d \geq n - \deg D$.

Definition 4.7 (One point codes)**.** $C_{\mathcal{L}}(G, D)$ is called a **one point AG code** if G is a one point divisor.

Definition 4.8. A **code automorphism** is a permutation of the coordinates of a code, σ such that $\sigma(\mathcal{C}) = \mathcal{C}$.

Theorem 4.9 (Function field automorphisms are code automorphisms)**.** *[Sti93, Sec 8.2] Let $\mathcal{C}_{\mathcal{L}}(D, G)$ be an AG code, and σ an automorphism of the function field. If $\sigma(G) = G$ and $\sigma(D) = D$ then σ is an automorphism of $\mathcal{C}_{\mathcal{L}}(D, G)$.*

We end this subsection with the only non-standard definition in it:

Definition 4.10. The **curve degree** of $z \in F$ is: $\deg_{\mathcal{C}} z := \deg(z)_\infty = \deg(z)_0$ (this equality is always true).

We note that this means that the curve degree of $z \in F$ is the number of zeros (accounting for multiplicities, and points in the algebraic closure) it has on the curve. This matches with the degree of a polynomial, which is also equal to the number of zeroes it has (again, when accounting for multiplicities and points in the algebraic closure). Unfortunately, the degree of a polynomial can be calculated by looking at its highest power, while the curve degree of $z \in F$ can have a much more mysterious behaviour. However, in the Hermitian function field, which is the concrete example of a function field that we will be working with there is a simple way of calculating the curve degree of a polynomial.

4.2 Hermitian function field

This function field will be of particular importance in our work.

Definition 4.11 (Hermitian function field)**.** Let q be some prime power, $K = \mathbb{F}_{q^2}$, then $\mathbb{H}$, the **Hermitian function field** is the field created by taking $k(x)[y] \bmod y^q + y = x^{q+1}$

The next two theorems will state the important properties of this function field.

Theorem 4.12 (Structure of the Hermitian function field)**.** *[Sti93, theroem 2.3.2] The following properties hold for $\mathbb{H}$:*

1. *The genus of $\mathbb{H}$ is $\frac{q^2-q}{2}$.*
2. *There are q^3 pairs $(\alpha, \beta) \in \mathbb{F}_{q^2}^2$ such that $\beta^q + \beta = \alpha^{q+1}$.*
3. *For any $(\alpha, \beta) \in \mathbb{F}_{q^2}^2$ such that $\beta^q + \beta = \alpha^{q+1}$ there is a unique rational place $P_{\alpha,\beta}$ such that $(x-\alpha)(P_{\alpha,\beta}) = (y-\beta)(P_{\alpha,\beta}) = 0$.*
4. *There is a rational place Q_∞ , which is the only pole of both x and y.*
5. *The places described in 2,3 are the only rational places of $\mathbb{H}$ (for a total of q^3+1)*
6. *$(x)_\infty = q \cdot Q_\infty$, $(y)_\infty = (q+1) \cdot Q_\infty$*
7. *For any r, the set $\left\{x^i y^j | i \cdot q + j \cdot (q+1) \leq r, j < q\right\}$ is a basis for $\mathcal{L}(r \cdot Q_\infty)$*

Definition 4.13. Let N_H be the set of all rational places of the Hermitian function field of the form $P_{\alpha,\beta}$. (i.e. all the rational places of H except Q_∞)

The automorphisms of $\mathbb{H}$ will also be of importance:

Theorem 4.14 (Structure of the Hermitian automorphisms)**.** *[Sti93, Ex 6.10]*

1. *For any place $P_{\alpha,\beta} \in N_H$ there is an automorphism $\sigma_{\alpha,\beta}$ such that $\sigma_{\alpha,\beta}(x) = x + \alpha, \sigma_{\alpha,\beta}(y) = y + \alpha^q x + \beta$. These automorphisms form a group V of size q^3. Note that:*
   (a) *$\sigma_{\alpha,\beta}(P_{0,0}) = P_{\alpha,\beta}$.*

(b) $\sigma_{\alpha,\beta}^{-1} = \sigma_{-\alpha,\alpha^{q+1}-\beta}$ (it can be verified that $\left(-\alpha, \alpha^{q+1} - \beta\right)$ is, indeed, a rational point)

2. *For any* $c \in \mathbb{F}_{q^2}^*$ *there is an automorphism* τ_c *such that* $\tau_c(x) = cx, \tau_c(y) = c^{q+1}y$. *These automorphisms form a cyclic group* W *of size* q^2-1 *which stabilizes* $P_{0,0}$ *(i.e.* $\tau_c(P_{0,0}) = P_{0,0}$*)*

3. *The group* U *generated by* V *and* W *is of size* $q^3\left(q^2-1\right)$ *and stabilizes* Q_∞.

Corollary 4.15 (Representations of Hermitian automorphisms)**.** *We can conclude the following from Theorem 4.14:*

1. *For any* $P_{\alpha,\beta} \in N_H$ *and* $c \in \mathbb{F}_{q^2}^*$ *there is an automorphism* $\varphi_{\alpha,\beta,c}$ *such that* $\varphi_{\alpha,\beta,c}(x) = cx + \alpha$ *and* $\varphi_{\alpha,\beta,c}(y) = c^{q+1}y + \alpha^q cx + \beta$. *This set of automorphisms is exactly the group* U.

2. *For any* $P_{\alpha_1,\beta_1}, P_{\alpha_2,\beta_2} \in N_H$ *there are exactly* q^2-1 *automorphisms taking* P_{α_1,β_1} *to* P_{α_2,β_2}, *these are* $\sigma_{\alpha_2,\beta_2}\tau_c\sigma_{\alpha_1,\beta_1}^{-1}$ *for any* $c \in \mathbb{F}_{q^2}^*$. *We can also write these explicitly as:*

$$\begin{aligned}\phi_c(x) &= (x-\alpha_1)\cdot c + \alpha_2\\ \phi_c(y) &= \left(y - \alpha_1^q x + \alpha_1^{q+1} - \beta_1\right)\cdot c^{q+1} + \alpha_2^q (x-\alpha_1)\cdot c + \beta_2\end{aligned}$$

and note that these automorphisms depend only on c.

Since for any $\sigma \in U$, $\sigma(Q_\infty) = Q_\infty$ and σ permutes the other rational places of N, it is a subgroup of the group of automorphisms of the AG code $\mathcal{C}_{\mathcal{L}}(r \cdot Q_\infty, N_H)$ (Theorem 4.9). In fact, for all interesting values of r, it is exactly the group of code automorphisms [Xin95].

5 Definition and fundamental coding parameters of degree lifted AG codes

In this section, we define the notion of degree-lifted AG codes and establish their fundamental coding parameters — dimension (related to code-rate) and relative distance. To prove a lower bound on distance we present a generalization of the Schwartz-Zippel Lemma (lemma 5.6), bounding the number of zeroes of polynomials on the rational points of a curve.

5.1 Definition of degree-lifted AG codes

Definition 5.1 (Degree-lifted AG codes)**.** Let $\mathcal{C}_{\mathcal{L}}(G, D)$ be a one point AG code (see def 4.7). Define:

$$\mathcal{L}^m(G) = \text{sp}\left\{f_1(X_1)\cdot f_2(X_2)\ldots f_m(X_m) \,|\forall i : f_i \in \mathcal{L}(G), \sum(f_i) \geq -G\right\}$$

Then we define the code $\mathcal{C}_{\mathcal{L}}^m(G, D)$ to be:

$$\{f(\mathcal{P}) \,|\mathcal{P} \in D^m, f \in \mathcal{L}^m(G)\}$$

Where, if $\mathcal{P} = (P_1, \ldots, P_m)$ and $f = \sum_{\forall i: f_i \in \mathcal{L}(G), \sum(f_i) \geq -G} a_i f_{i_1}(X_1)\cdot f_{i_2}(X_2)\cdot\cdots\cdot f_{i_m}(X_m)$ (where the coefficients a are in the base field), Then $f(\mathcal{P}) := \sum a_i f_{i_1}(P_1)\cdot f_{i_2}(P_2)\cdot\cdots\cdot f_{i_m}(P_m)$.

Remark 5.2. We note the difference between these codes and those obtained by taking the tensor product of $\mathcal{C}_L(G,D)$ (by definition 1.1). Here we have as a condition on the monomials spanning $L^m(G)$: $\sum\left(f_{i_j}\right) \geq -G$. This is similar to looking at the total degree of monomials in RM codes. Taking the tensor product of $\mathcal{C}_L(G,D)$ we would get that the condition would be: $\forall i : \left(f_{i_j}\right) \geq -G$, which is similar to looking at the individual degrees. So $\mathcal{C}_L^m(G,D)$ is a sub-code of the m-th tensor of $\mathcal{C}_L(G,D)$. Lemma 5.6 will show that this sub-code has good distance.

Definition 5.3 (Canonical basis)**.** Let G be a one point divisor, i.e. $G = r \cdot P$ for some rational place P. A canonical basis for $\mathcal{L}(G)$ is constructed by considering the series of divisors $0, P, 2P \ldots r \cdot P$. Whenever $l(i \cdot P) > l((i-1) \cdot P)$, pick a function $\varphi \in \mathcal{L}(i \cdot P)$ $\varphi \notin \mathcal{L}((i-1) \cdot P)$ and add it to the basis.

To study degree lifted codes we introduce the following definition which extends the definition of curve degree (Definition 4.10) to functions in $\mathcal{L}^m(G)$:

Definition 5.4 (Curve degree of tensored functions)**.** The curve degree of $\varphi_{i_1}(X_1) \cdot \ldots \varphi_{i_m}(X_m)$ is the sum $\sum_{j=1}^{m} \deg_{\mathcal{C}} \varphi_{i_j}$. The curve degree of $f \in \mathcal{L}^m(G)$ is the maximal curve degree amongst its monomials.

5.2 Dimension of degree-lifted codes

The dimension of $\mathcal{C}_{\mathcal{L}}^m(G,D)$ is equal to the number of m-tuples of basis functions for $\mathcal{L}(G)$ such that $\sum_{i=1}^{m}(\varphi_i) \leq -G$. So the exact dimension of such a code would depend upon the gap sequence of G. We can, however establish some bounds:

Theorem 5.5 (Dimension of degree-lifted AG codes)**.** *Let $2g-1 \leq \deg G \leq n$. Let k be the dimension of $\mathcal{C}_{\mathcal{L}}^m(G,D)$, then: $\binom{\deg G - m\cdot(g-2)}{m} \leq k \leq \min\left\{(\deg G + 1 - g)^m, \binom{\deg G + m}{m}\right\}$*

Proof. The first upper bound is the dimension of the tensored AG code (i.e. the mth power of the dimesnion of the base code). The second upper bound is what would happen if $l(G) = \deg G$. The lower bound is the number of m-tuples if the minimal curve degree of a basis function is g (the worst case scenario). □

5.3 A Schwartz-Zippel Lemma for degree-lifted AG codes

To be able to claim that the codes defined above have good distance we prove one of the main results of this paper, a Schwartz-Zippel type lemma for polynomials over algebraic curves:

Lemma 5.6 (Schwartz-Zippel for degree-lifted AG codes)**.** *Let $G \geq 0$ be a one point divisor, D a set of rational places disjoint from the support of G. Let $f \in \mathcal{L}^m(G) \setminus \{0\}$. Let $\mathcal{P} = \{P\}_{i=1}^m$ be an m-tuple of randomly selected places out of D. Then the probability of $f(\mathcal{P})$ being zero is at most $\frac{\deg_{\mathcal{C}} f}{|D|}$.*

Proof. Fix a canonical basis $\{\varphi_i\}_{i=1}^{l(G)}$ for $\mathcal{L}(G)$. We prove by induction on m. Let $m = 1$, then f is a member of $\mathcal{L}(G)$ and therefore a member of the function field. It can only be zero at $\deg_{\mathcal{C}} f$ places (cf. definition 4.10).

Now assume the proposition for $m-1$, and consider the case of m. Take $f = \sum_{i=1}^{k} \prod_{j=1}^{m} f_{i,j}(X_j)$ and represent each $f_{i,j}$ in the canonical basis. Note that any basis function with a non-zero coefficient in this representation cannot have a curve degree higher than $f_{i,j}$'s. So we get a representation of f as a linear combination of products of basis functions. Where each product has a curve degree smaller than $\deg G$. Randomly select a tuple $\mathcal{P} \in D^m$, let $\mathcal{P}^*$ be the first $m-1$ places in $\mathcal{P}$. Denote $f(\mathcal{P}*) = \sum_{i_j \in [l(G)] \text{ and } \sum(\varphi_{i_j}) \geq -G} a_{i_1,i_2 \ldots i_m} \varphi_{i_1}(P_1) \cdot \varphi_{i_2}(P_2) \cdot \cdots \cdot \varphi_{i_{m-1}}(P_{m-1}) \cdot \varphi_{i_m}(X_m)$ and let j be the largest index of a basis function which appears in f with X_m. Then (by induction assumption) with probability at least $1 - \frac{\deg_{\mathcal{C}} f - \deg_{\mathcal{C}} \varphi_j}{|D|}$, φ_j has a non-zero coefficient in $f(\mathcal{P}^*)$. This means that $f(\mathcal{P}^*)$ is a linear combination of $\{\varphi_i | 1 \leq i \leq j\}$, with a non-zero coefficient at φ_j, so it can't be identically zero (because the basis functions are independent) and, by definition 5.3, $\deg_{\mathcal{C}} f(\mathcal{P}^*) \leq \deg_{\mathcal{C}} \varphi_j$ so the probability of $f(\mathcal{P}^*)$ being zero at a random place of D is at most $\frac{\deg_{\mathcal{C}} \varphi_j}{D}$. In summary:

$$\Pr[f(\mathcal{P}) = 0] = \Pr[f(\mathcal{P}^*) = 0] + \Pr_{P \in D}[f(\mathcal{P}^*)(P) = 0 | f(\mathcal{P}^*) \neq 0] \leq$$
$$\frac{\deg_{\mathcal{C}} f - \deg_{\mathcal{C}} \varphi_j}{|D|} + \frac{\deg_{\mathcal{C}} \varphi_j}{|D|} = \frac{\deg_{\mathcal{C}} f}{|D|}$$

□

We conclude that the distance of $\mathcal{C}^m_{\mathcal{L}}(G, D)$ when $G \geq 0$ and D is rational is at least $|D|^m - |D|^{m-1} \cdot \deg G$.

It is natural to ask whether the requirement that G be a 1-point divisor is necessary; unfortunately, it is. As a counter example, consider the rational function field and the Riemann-Roch space $\mathcal{L}(Q_{\inf} + P_0)$. This is the space spanned by $\frac{1}{x}, 1, x$. Now consider $\mathcal{L}^2(Q_{\inf} + P_0)$ which is the space spanned by $\frac{1}{x}, \frac{1}{y}, 1, x, y, \frac{y}{x}, \frac{x}{y}$. We can evaluate functions in this space on points in $(\mathbb{F}^*_q)^2$ and were the SZ lemma to apply here, we would expect at most $2q-2$ zero. Now consider the function $f = \frac{(x-1)(y-2)}{x} - \frac{(x-1)(y-2)}{y}$ which is in $\mathcal{L}^2(Q_{\inf} + P_0)$. It is zero whenever $y = 2$,$x = 1$ or $y = x$ for a total of $3q-6$ zeros, disproving a more general application of the SZ lemma.

6 Single-step correction of degree lifted AG codes

In this section we examine the most basic correction algorithm for degree lifted AG codes, in which we correct a point by looking at the restriction of the code to a random automorphism passing through it. We show that the closer a code's automorphism group is to being two transitive (in the sense of definition 6.1) the better an LCC it makes (Theorem 6.2) . We will then examine how this test works for degree lifted Hermitian codes (proving corollary 6.4 in Section 6.1). In Section 6.2 we show that this test works well when the underlying automorphism group is 2-transitive, and in Section 6.3 we study a variant of Hermitian codes which is 2-transitive and has a locally decodable tensor.

Let $\mathcal{C}^2_{\mathcal{L}}(G, D)$ be a code, and $\mathrm{Aut}(G, D)$ the group of function field automorphisms stabilizing G and D. We study the following procedure:

Procedure 1

In order to correct the point $(P_1, P_2, \ldots, P_m)$ in the received message f, do the following:

1. Pick a set of random automorphisms $\sigma_2, \ldots, \sigma_m \in \mathrm{Aut}(G, D)$ such that $\sigma_i(P_1) = (P_i)$.
2. Read the values of the message at the points $C = \{(P, \sigma_2(P), \ldots, \sigma_m(P)) | P \in D\}$.
3. Use a decoding algorithm for AG codes on these values (treating the value at $(P, \sigma_2(P), \ldots, \sigma_m(P))$ as the value at P) and get a function $f' \in \mathcal{L}(G)$.
4. Return $f'(P_1)$.

Remark. This is exactly the test used to correct RM codes, where the base code is a RS code and its automorphism group is the affine group.

What do we need for this test to work? At the least, we need $\mathrm{Aut}(G, D)$ to be transitive or there wouldn't be any σ to choose in step 1. In order to prove that this test works, we need that for every point (P_1, P_2), the possible samplers for this point cover a significant part of the space nearly uniformly. Formally:

Definition 6.1 ((ϵ, α)-doubly transitive groups)**.** The group H acting on the set S is (ϵ, α)-***doubly transitive*** if for every $P_1, P_2 \in S$ and at least a $1 - \epsilon$ fraction of $P_3 \in S$ the variable $X = \sigma(P_3)$ is distributed uniformly on an $1 - \alpha$ fraction of S when σ is chosen uniformly at random from the set $\{\sigma | \sigma(P_1) = P_2\}$.

Theorem 6.2 (Single step correction of degree lifted AG codes)**.** *Let $\mathcal{C}_{\mathcal{L}}(G, D)$ be an $[n, k, d]_q$ code such that $Aut(G, D)$ is (ϵ, α)-*doubly transitive*. For any $f \in \mathcal{C}^m_{\mathcal{L}}(G, D)$, point of correction $(P_1, P_2, \ldots, P_m) \in D^m$ and any δ-fraction of errors, procedure 1 succeeds with probability at least $1 - \frac{2}{d}\left(\frac{\delta}{(1-\alpha)^{m-1}} + \epsilon\right)$*

Proof. For any z in the function field and function field automorphism φ, $(\varphi(z)) = \varphi((z))$ so the restriction of a codeword f to $(P, \sigma_2(P) \ldots \sigma_m(P))$ is a codeword of $\mathcal{C}_{\mathcal{L}}(G, D)$. $\mathcal{C}_{\mathcal{L}}(G, D)$ has distance d and so we can handle $\frac{d}{2}$ errors. An adversary can concentrate errors in the parts of the space sampled, so the total number of errors in the sampled space is at most $\delta |D|^m$. We also automatically assume that the ϵ fraction of points that don't behave well are errors. So the expected fraction of errors for a test is at most $\frac{\delta}{(1-\alpha)^{m-1}} + \epsilon$. By Markov, the probability of having more than $\frac{d}{2}$ errors is at most $\frac{2}{d}\left(\frac{\delta}{1-\alpha} + \epsilon\right)$. □

In Section 6.1 we'll show that this test works on Hermitian codes, by proving the following theorem:

Theorem 6.3 (Closeness of Hermitian automorphisms to doubly transitive)**.** *The automorphism group of Hermitian codes is $\left(\frac{1}{q^2}, \left(1 - \frac{1}{q}\right)\right)$-*doubly transitive.

From Theorems 6.2,6.3 we derive the following corollary:

Corollary 6.4 (Single step correction of degree lifted Hermitian codes)**.** *Procedure 1 works on $\mathcal{C}^m_{\mathcal{L}}(r \cdot Q_\infty, N_H)$ with probability at least $\frac{2}{q^3 - r}\left(q^{m-1}\delta + \frac{1}{q^2}\right)$.*

We can see that if ($\alpha > 0$) δ must decrease as m increases. This isn't satisfactory and we'll solve this problem in several ways in the next sections.

What happens if $\alpha = 0$ (i.e. the base code is 2-transitive)? In that case we get local correction for any m, which we'll prove in Section 6.2.

By adding some redundancy to the base code we can increase α, we examine a method for this in Section 6.3 and get a version of Hermitian codes which is locally correctable for larger m's.

6.1 Single-step correction of Hermitian codes

We want to prove Theorem 6.3:

Theorem (Closeness of Hermitian automorphisms to doubly transitive)**.** *The automorphism group of Hermitian codes is* $\left(\frac{1}{q^2}, \left(1-\frac{1}{q}\right)\right)$-doubly transitive.

This will imply (corollary 6.4) that the degree lifting of Hermitian codes is locally correctable from some small fraction of errors.

We need to examine the automorphisms taking (x_1, y_1) to (x_2, y_2) and how they act on some place (x_3, y_3). The next two lemmas establish the needed properties.

Lemma 6.5 (Automorphisms from (x_1, y_1) to (x_2, y_2))**.** *There are* $q^2 - 1$ *automorphisms taking* x_1, y_1 *to* x_2, y_2*.*

Proof. Let $\sigma_{x,y}$ be the automorphism taking $(0,0)$ to (x,y) and τ a generator for the automorphism group that stabilizes $(0,0)$ then the set $\left\{\sigma_{x_2,y_2}\tau^i\sigma_{x_1,y_1}^{-1} | 1 \leq j \leq q^2-1\right\}$ is a set of q^2-1 automorphism taking (x_1,y_1) to (x_2,y_2) and since $|\mathrm{Aut}\left(rQ_\infty, N_H\right)| = q^3\left(q^2-1\right)$ and $|D| = q^3$ these are the only such automorphisms. □

Lemma 6.6 (Automorphisms rarely intersect more than once)**.** *Let* φ_1, φ_2 *be two different automorphisms taking* (x_1, y_1) *to* (x_2, y_2)*, if* $\varphi_1(x,y) = \varphi_2(x,y)$ *then* $x = x_1$*.*

Proof. Let τ be a generator of the automorphism group W (see theorem 4.14), $\varphi_1 = \sigma_{x_2,y_2}\tau^i\sigma_{x_1,y_1}^{-1}$ and $\varphi_2 = \sigma_{x_2,y_2}\tau^j\sigma_{x_1,y_1}^{-1}$. Then $\sigma_{x_2,y_2}\tau^i\sigma_{x_1,y_1}^{-1}(x,y) = \sigma_{x_2,y_2}\tau^i\sigma_{x_1,y_1}^{-1}(x,y)$. If $x \neq x_1$ this means that there is a point $(x',y'), x' \neq 0$ such that $\tau^i(x',y') = \tau^j(x',y')$ w.l.g let $j > i$ then $(x',y') = \tau^{j-i}(x',y')$ so τ^{j-i} stabilizes a non-zero point. But then, if we denote $\tau^{j-1}(x) = \tau_c(x)$, $x' = cx'$ so $c = 1$ (since $x' \neq 0$) and τ^{j-1} is the identity mapping in contradiction to τ being of order q^2-1. □

So we have that for all but q places (those having $x_3 = x_1$), the image of (x_3, y_3) under automorphisms taking (x_1, y_1) to (x_2, y_2) is of size $q^2 - 1$ (so uniformity is immediate) this proves Theorem 6.3.

Explicitly, this gives us that for $r = q^3 - c\epsilon q^4$, procedure 1 succeeds on the code $\mathcal{C}_{\mathcal{L}}^2(r \cdot Q_\infty, N_H)$ with probability at least $1 - \frac{2}{c}$.

So,

Corollary 6.7 (Degree lifted Hermitian codes are single step correctable)**.**

$$\mathcal{C}_L^2\left(\left(q^3 - c\epsilon q^4\right) \cdot Q_\infty, D\right)$$

is a locally correctable code with rate

$$\frac{k}{n} \approx \frac{(q^3 - c\epsilon q^4)^2 - 2(q^3 - c\epsilon q^4)q^2}{2q^6} \approx \frac{1}{2} - c\epsilon q - \frac{1}{q},$$

query complexity $\sqrt{n}$ *and alphabet size of* $\sqrt[3]{n}$ *.*

We note that the Reed-Muller code with the similar rate of $\frac{1}{2} - \epsilon$ has an alphabet size of $\sqrt{n}$ (the query complexity is the same).

6.2 Degree lifting of 2-transitive AG codes are LDCs

If our base code is 2-transitive then procedure 1 samples the whole space. What is left is to establish the uniformity of this sampling. In this section we show that if a group is $(\epsilon, 0)$-doubly transitive (i.e. 2-transitive) then $\epsilon = 0$ which will imply that the degree lifting of doubly transitive AG codes is locally correctable (Theorem 6.10).

Remark. Technically, a 2-transitive group won't sample the whole space since, for instance, when correcting (P_1, P_2) we will never sample (P_1, P_3). We can overcome this by also considering the parallel lines (i.e. $\{(P_1, P) \,|P \in D\}$ etc) as views.

In the next two lemmas, we show that uniformity is a property of any 2-transitive group.

Lemma 6.8 (Uniformity of transitive groups)**.** *Let A be a set with a group Φ acting on it transitively. Let $\Phi_{a,b} \subseteq \Phi$ be the set of automorphisms taking a to b, then $|\Phi_{a,b}|$ is independent of the choice of a, b.*

Proof. Assume by contradiction that $|\Phi_{a,b}| > |\Phi_{a,c}|$. Let $\sigma \in \Phi$ be such that $\sigma(b) = c$ then $\sigma(\Phi_{a,b}) \subseteq \Phi_{a,c}$ so there are two different $\varphi_1, \varphi_2 \in \Phi_{a,b}$ such that $\sigma \circ \varphi_1 = \sigma \circ \varphi_2 \Rightarrow \varphi_1 = \varphi_2$, a contradiction. □

Lemma 6.9 (Uniformity of doubly transitive groups)**.** *Let A be a set with a group Φ acting on it 2-transitively. Let $\Phi_{a,b,c,d} \subseteq \Phi, a \neq c, b \neq d$ be the set of automorphisms taking a to b and c to d, then $|\Phi_{a,b,c,d}|$ is independent of the choice of a, b, c, d.*

Proof. From lemma 6.8 we know that $|\Phi_{a,b}|$ is independent of a, b , so assume by contradiction that $|\Phi_{a,b,c,d}| > \left|\Phi_{a,b,c,d'}\right|$. Let $\sigma \in \Phi$ be such that $\sigma(b) = b, \sigma(d) = d'$ then $\sigma(\Phi_{a,b,c,d}) \subseteq \Phi_{a,b,c,d'}$ so there are two different $\varphi_1, \varphi_2 \in \Phi_{a,b,c,d}$ such that $\sigma \circ \varphi_1 = \sigma \circ \varphi_2 \Rightarrow \varphi_1 = \varphi_2$, a contradiction. □

We can now prove the main result of this section.

Theorem 6.10 (Doubly Transitive AG codes are locally correctable)**.** *Let $\mathcal{C}_{\mathcal{L}}(G, D)$ be an $[n, k, d]_q$ code with a 2 transitive group of function field automorphisms, then $\mathcal{C}^m_{\mathcal{L}}(G, D)$ is locally correctable from a $\delta < \frac{d}{2n}$ fraction of errors.*

Proof. set $\alpha = 0$ and $\epsilon = 0$ in Theorem 6.2. □

We are unaware of any 2-transitive AG codes (other than Reed-Solomon codes). Some immediate places to look for them are the Hermitian, Suzuki and Ree curves, as all of them have 2-transitive groups acting on their rational points ([Sti93], [HS90] and [Ped92] respectively). However, for these to give rise to 2-transitive codes there must also be a non-rational divisor of sufficiently low degree (smaller than the number of rational points) which is stabilized by all these automorphisms.

Would this kind of test work for the tensor of any 2-transitive code? The uniformity of the sampling has nothing to do with the base code being an AG code, the only property of AG codes we use is that the restriction of the tensor to the sampler is a codeword of the base code. We do not know whether this is true for the tensor (or some appropriate generalization of the degree lifted subcode of it) of some general 2-transitive code.

6.3 Increasing transitivity via redundancy

To get a 2-transitive AG code, we need an automorphism group that acts 2-transitively on some set of places, while stabilizing another (that can't be of too high a degree). While we have several examples with a 2-transitive action, we don't know of any (except for RM codes) where it also stabilizes another divisor. In particular, the Hermitian function field has a 2-transitive action on its set of rational places (including infinity), but we don't know of any low degree (degree lower than q^3) divisor that is stabilized by it.

In this section we show that a 2 transitive action is sufficient to show local correctability of the tensored code (though with a smaller rate than we would get from degree lifting), this gives us a construction of an LCC based on tensored Hermitian codes.

We first need to slightly extend our notion of AG codes.

Let $S = \{P\}_i^n$ be a set of places and $\mathrm{Aut}\{S\}$ be a group of function field automorphisms acting on S (implied here is that S is closed under the action of function from $\mathrm{Aut}\{S\}$). We define the evaluation of a function at a place it has a pole in as 0.

Definition 6.11. The *extended evaluation* of a function $z \in F/K$ is denoted $\bar{z}$ and defined

$$\bar{z}(P) = \begin{cases} z(P) & v_P(Z) \geq 0 \\ 0 & otherwise \end{cases}$$

This allows us to consider AG codes $\mathcal{C}_{\mathcal{L}}(G, D)$ when $SUPP(G) \cap SUPP(D) \neq \emptyset$.

Definition 6.12. The code $\bar{\mathcal{C}}_{\mathcal{L}}(N, D) := \{(\bar{z}(P_i))_{i=1}^n \,|z \in \mathcal{L}(D)\}$

We can now prove the main result of this section:

Theorem 6.13 (Local correction of increased transitiveness AG codes)**.** *Let S be a set of place such that $Aut\{S\}$ acts 2-transitively on S, pick $Q \in S$. Then $\bar{\mathcal{C}}_{\mathcal{L}}^m(r \cdot Q, S)$ is locally correctable from a $\delta \leq \frac{n - r \cdot m}{2n}$ fraction of errors.*

Proof. Mark $|S| = n$. Let δn^m be the number of errors allowed. Say we wish to correct the codeword w at the point $(P_1, P_2 \dots P_m)$. We randomly pick automorphisms $\varphi_2 \dots \varphi_m$ such that $\varphi_i(P_1) = P_i$, and consider the values received at the points $C = \{(P, \varphi_2(P) \dots \varphi_m(P)) \,|P \in S\}$. For all z in the function field: $(\varphi(z)) = \varphi((z))$ so $\deg_{\mathcal{C}} z = \deg_{\mathcal{C}} \varphi(z)$. Which means that the restriction of w to C is a codeword of $\mathcal{C}_{\mathcal{L}}\left(r \cdot Q + r \cdot \sum_{i=2}^{m} \varphi_i(Q), S\right)$, this code has distance of at least $d = n - m - r \cdot m$ and we can use standard algorithms to decode it. For any point $(P_1' \dots P_m')$, $P_i' \neq P_i$ the number of automorphisms selections which would lead to it being in $\mathcal{C}$ is independent of the identity of the P's (lemma 6.9) so the expected fraction of errors in C is δ.

Let X_C be a random variable denoting the fraction of errors on C. Using markov $\Pr\left[X_C > \frac{d}{2}\right] < \frac{2\delta n}{d}$, so if δ is smaller than half the relative distance of $\mathcal{C}_{\mathcal{L}}\left(r \cdot Q + r \cdot \sum_{i=2}^{m} \varphi_i(Q), S\right)$ the correction succeeds with high probability. □

The disadvantage of this method is the presence of r in the distance of the restrictions. This means that $r < \frac{n}{m}$. We also gain no benefit from bounding the total curve degree of monomials in this construction. So we bound the individual curve degrees of of members in monomials instead. i.e. we look at the full tensored code and not at a sub-code of it. This gives us a dimension of $(r - g)^m$.

Corollary 6.14 (Locally correctable tensors of Hermitian codes)**.** *The m-th tensor of $\bar{\mathcal{C}}_{\mathcal{L}}(r \cdot Q_\infty, N_H \cup Q_\infty)$ is locally correctable from a $\delta \leq \frac{q^3 - r \cdot m}{2q^3}$ fraction of errors. The rate of these codes will be $\frac{1}{m^m}\left(1 - c\delta - \frac{1}{q}\right)$.*

Proof. The set of rational points of the Hermitian curve has a 2-transitive group of automorphisms acting on it [Sti93]. Now apply Theorem 6.13. □

Remark. Theorem 6.13 can be extended for (ϵ, α)-doubly transitive groups.

7 Fractal correction of degree lifted AG codes

In this section we will examine a fractal correction algorithm, define the required properties for a degree lifted code to be locally correctable using this algorithm (Theorem 7.2), show that Hermitian codes posses these properties and conclude that fractal correction succeeds on degree lifted Hermitian codes (corollary 7.5).

Formally, fractal correction is the following procedure:

Procedure 2:

In order to correct the point $(P_1, \ldots, P_m)$ do the following:

1. Choose $m - 1$ automorphisms such that $\varphi_i(P_1) = (P_i)$.
2. Let $C = \{(P, \varphi_2(P) \ldots \varphi_m(P)) \,|\, (P) \in D\}$ be the embedding of the curve generated by these automorphisms.
3. For every point in C, apply procedure 1.
4. Use the values returned from step 3, apply standard AG decoding to get the restriction of f to C, and calculate it at the point P_1.

So in this procedure, we first correct each point on C and then use the corrected values to correct the value at the original point we wanted to correct. By iterating steps 1,2 we can increase the "depth" of the recursion in this procedure. So as described this procedure is a 2-step correction procedure, but this can be increased to c-steps. The following definition captures the requirements for this test to work.

Definition 7.1 ((α, ϵ)-closeness to c-steps uniformity)**.** The group H acting on the set S is ***α, ϵ-close to c-steps uniform*** if for every $P_1, P_2 \in S$, random series of objects $Q_1, Q_2, \ldots Q_{c-1} \in S$ and a random series of c automorphisms such that $\sigma_1(P_1) = P_2, \sigma_i(Q_{i-1}) = \sigma_{i-1}(Q_{i-1})$. For at least an $(1 - \epsilon)$ fraction of objects in S , the variable $\sigma_c(P)$ is α-close to the uniform distribution on S.

Theorem 7.2 (Fractal correction of degree lifted AG codes)**.** *Let $\mathcal{C}_{\mathcal{L}}(G, D)$ be an $[n, k, d]_q$ code such that $Aut(G, D)$ is α, ϵ-close to 2-steps uniform. For any $f \in \mathcal{C}^m_{\mathcal{L}}(G, D)$, point of correction $(P_1, P_2 \ldots P_m) \in D^m$ and any δ-fraction of errors. Procedure 2 succeeds with probability at least $1 - \left(\frac{2n}{d}\right)^c (\delta + m \cdot \alpha + \epsilon)$*

Proof. In the bottom most step of the recursion, points are sampled such that each coordinate is α-close to uniform, so the points themselves are sampled $m \cdot \alpha$-close to uniformly (relying on lemma

7.7). So the expected number of errors along a curve is at most $(\delta + m \cdot \alpha + \epsilon) n$. Using Markov, this means that each correction in the bottom of the recursion succeeds with probability at least $1 - \frac{2}{d}(\delta + m \cdot \alpha + \epsilon) n$. So in the next step of the recursion, the expected number of errors is at most $\frac{2}{d}(\delta + m \cdot \alpha + \epsilon) n^2$ and each correction succeeds with probability at least $1 - \frac{4}{d^2}(\delta + m \cdot \alpha + \epsilon) n^2$. Continuing in this way, we get that the topmost step in the recursion succeeds with probability at least $1 - \left(\frac{2n}{d}\right)^c (\delta + m \cdot \alpha + \epsilon)$. □

This does give us a way to locally correct codes with $m > 2$. We now return to the automorphisms of the Hermitian code, the remainder of this section will be concerned with proving the following:

Theorem 7.3 (2-steps uniformity of Hermitian automorphisms)**.**

$$Aut(r \cdot Q_\infty, N_H)$$

is $\frac{1}{q}, \frac{1}{q}$*-close to 2-steps uniform.*

Corollary 7.4 (Fractal correction of degree lifted Hermitian codes)**.** *Consider*

$$\mathcal{C}_L^m \left(\left(q^3 - c\left(\delta + \frac{m}{q}\right) q^3 \right) \cdot Q_\infty, D \right)$$

, for some constant c, if the fraction of errors in the received codeword is smaller than δ *then procedure 2 succeeds with probability at least* $1 - \frac{4}{\left(\delta + \frac{m}{q}\right) c^2}$.

Corollary 7.5 (Fractal correction degree lifted Hermitian codes)**.**

$$\mathcal{C}_L^m \left(\left(q^3 - c\left(\delta + \frac{m}{q}\right) q^3 \right) \cdot Q_\infty, D \right)$$

is a locally correctable code with rate

$$\frac{k}{n} \approx \frac{(q^3 - c\left(\delta + \frac{m}{q}\right) q^3)^m - m(q^3 - c\left(\delta + \frac{m}{q}\right) q^3)^{m-1} q^2}{m! \cdot q^{3m}} \approx \frac{1}{m!}\left(1 - c\delta - \frac{m}{q}\right)$$

query complexity $\sqrt[m]{n^2}$ *and alphabet size of* $\sqrt[1.5m]{n}$.

We note that the Reed-Muller code with the similar rate of $\frac{1}{m!}(1 - \delta)$ has a query complexity of $\sqrt[m]{n}$ and an alphabet size of $\sqrt[m]{n}$.

Proof. (of Theorem 7.3) This is an immediate consequence of lemma 7.8. But first we will need some preparatory lemmas:

Lemma 7.6 (Image of bounded polynomial is uniform)**.** *A degree* $q + 1$ *polynomial* $f : \mathbb{F}_{q^2} \to D$, $|D| = q$ *has a statistical difference of at most* $\frac{1}{q} - \frac{1}{q^2}$ *from the uniform distribution on* D.

Proof. Let Q be the distribution induced by f, assume that $\delta(Q, U) > \frac{1}{q} - \frac{1}{q^2}$. Then there is a set $A \subseteq D$ such that $\left| \Pr_{x \in \mathbb{F}_{q^2}} [f(x) \in A] - \frac{|A|}{q} \right| > \frac{1}{q} - \frac{1}{q^2}$. This means that one of the following 2 things must happen:

1. f takes more values to A then the uniform distribution does, in that case $\left| f^{-1}(A) \right| \geq |A| q + q$ but $\left| f^{-1}(x) \right| \leq q + 1$ (because of its degree) so $|A| = q$ which is ludicrous.

2. f takes less values to A then the uniform distribution does. Apply 1. to $\bar{A}$ to get a contradiction.

□

Lemma 7.7 (Statistical distance is additive)**.** *Let Q be a distribution on D and P a distribution on D' which is independent of Q. P and Q are μ close to uniform. Then $Q \times P$ is at least 2μ-close to uniform.*

Proof. Statistical distance obeys the triangle inequality so

$$d\left(Q \times P, U_D \times U_{D'}\right) \leq d\left(Q \times P, Q \times U_{D'}\right) + d\left(Q \times U_{D'}, U_D \times U_{D'}\right) = 2\mu$$

□

Now consider the following probabilistic procedure as part of correcting the point $(x_1, y_1, x_2, y_2 \ldots x_m, y_m)$:

Procedure 3:

1. Pick an index $2 \leq i \leq m$.
2. Choose $m-2$ automorphisms $\{\varphi_j\}$with the indices $X = \{2, 3 \ldots, i-1, i+1 \ldots m\}$) such that $\forall j \in X : \varphi_j\left(x_1, y_1\right) = \left(x_j, y_j\right)$.
3. Choose a rational point $(x', y') \in N_H$ (this is a particular point along C we now wish to correct).
4. Choose $m-2$ automorphisms $\left\{\varphi_j^{'}\right\}$with the indices $X = \{2, 3 \ldots, i-1, i+1 \ldots m\}$) such that $\forall j \in X : \varphi_j^{'}\left(x', y'\right) = \varphi_j\left(x', y'\right)$ (this is [almost] picking a random curve along which to correct the point $\left(x', y', \varphi_2\left(x', y'\right) \ldots \varphi_m\left(x', y'\right)\right)$).
5. Pick a random point $(x, y) \in N_H$.

   Lemma 7.8 (Uniformity of single coordinates)**.** *after following procedure 3, when selecting random $\varphi_i, \varphi_i^{'}$ such that $\varphi_i\left(x_1, y_1\right) = \left(x_i, y_i\right)$ and $\varphi_i^{'}\left(x', y'\right) = \varphi_i\left(x', y'\right)$. For all but q^2 values of (x, y), $\varphi_i^{'}(x, y)$ is distributed at least $\frac{1}{q}$-close to the uniform distribution on N_H.*

   Proof. Using the representation of automorphisms described in Corollary 4.15, we can state the explicit requirements for $\varphi_i, \varphi_i^{'}$ as:

$$\begin{aligned} \varphi_i(x) &= (x - x_1) \cdot c + x_i \\ \varphi_i(y) &= \left(y - x_1^q x + x_1^{q+1} - y_1\right) \cdot c^{q+1} + x_i^q (x - x_1) \cdot c + y_i \end{aligned}$$

$$\varphi_i^{'}(x) = \left(x - x'\right) \cdot c' + \varphi_i\left(x'\right)$$

$$\varphi_i'(y) = \left(y - x'^q x + x'^{q+1} - y'\right) \cdot c'^{q+1} + \varphi_i\left(x'\right)^q \left(x - x'\right) \cdot c' + \varphi_i\left(y'\right)$$

combining them we get:

$$\begin{aligned} \varphi_i'(x) &= \left(x - x'\right) \cdot c' + \left(x' - x_1\right) \cdot c + x_i \\ \varphi_i'(y) &= \left(y - x'^q x + x'^{q+1} - y'\right) \cdot c'^{q+1} + \left((x - x_1) \cdot c + x_i\right)^q (x - x') \cdot c' \\ &\quad + \left(y' - x_1^q x' + x_1^{q+1} - y_1\right) \cdot c^{q+1} + x_i^q (x' - x_1) \cdot c + y_i \end{aligned}$$

Let $(x_T, y_T) \in N_H$, for all but q values of (x, y), $x \neq x'$ so when $c' = \frac{-(x'-x_1)\cdot c - x_i + x_T}{(x-x')}$, $\varphi_i'(x) = x_T$ and so w.h.p. $\varphi_i'(x)$ is uniformly distributed on F_{q^2}. If we set $c' = \frac{-(x'-x_1)\cdot c - x_i + x_T}{(x-x')}$ in the second equation we get that $\varphi_i'(y)$ is a degree (at most) $q+1$ polynomial in c. The coefficient of c^{q+1} is $\left(y' - x_1^q x' + x_1^{q+1} - y_1\right) + \left(y - x'^q x + x'^{q+1} - y'\right)\left(\frac{(x'-x_1)}{(x-x')}\right)^{q+1}$. Fixing x_1, y_1, x', y' this coefficient has most q^2 zeros on the Hermitian curve so for all but q^2 values of (x, y) it is not zero and $\varphi_i'(y)$ is a degree $q+1$ polynomial in c whose image is of at most size q (since $(x_T, \varphi_i'(y))$ is a rational point on the Hermitian curve) . So the farthest statistical difference it can have from a uniform distribution is $\frac{1}{q}$ (lemma 7.6). □

The coefficient of c^{q+1} is independent of (x_i, y_i) so for all but q^2 of the points on each curve in step 3 of procedure 2, each of the coordinates of $\left((x, y), \varphi_2'(x, y) \ldots \varphi_m'(x, y)\right)$ is $\frac{1}{q}$ close to uniform. □

8 Correction via high-degree samplers

In this section we examine correction via high-degree samplers. We define the required properties of a base AG code for its degree lifting to be locally correctable in this manner (Theorem 8.3) and construct an explicit set of high-degree samplers for degree lifted Hermitian codes which allows us to locally correct them (Theorem 8.5).
So far we have used automorphisms in our correcting, this had the benefit of having the restriction of the degree lifted code be a word in the base code. The downside of this is that the number of automorphisms isn't as large as we would like it to be. In the section we replace the automorphisms with a larger class of functions that will allow us to easily sample the whole space but with some increase in the curve degree of the restrictions of a code-word to a sampler.

Definition 8.1. A **generalized automorphism** of a function field is a function $f : F/K \to F/K$ such that for all $P \in \mathcal{P}_F$ the set $f(P)$ is contained in only one place of F/K. We define $f(P)$ to be that unique place.

Definition 8.2 ((l,t)-samplers)**.** The set of generalized automorphisms Φ is an (l,t)**-sampler** for the code $\mathcal{C}_{\mathcal{L}}(G, D)$ if there is a subset $A \subseteq L(G)$ of dimension $\frac{l(G)}{t}$ such that for all $f \in \Phi$ and $g \in A$, $\deg_{\mathcal{C}} g + l \geq \deg_{\mathcal{C}} f(g)$ and Φ has a $(0,0)$-doubly transitive action on D.

Theorem 8.3 (High degree correction of degree lifted AG codes)**.** *If $\mathcal{C}_{\mathcal{L}}(G, D)$ is an $[n, k, d]_q$ code which has an (l,t)-sampler Φ then for any $f \in A^m$, point of correction $(P_1, P_2 \ldots P_m) \in D^m$ and any δ-fraction of errors. Procedure 1 (when choosing functions from Φ) succeeds with probability at least $1 - \frac{2\delta}{d-(m-1)l}$*

Proof. For any $z \in A$ and function $\varphi \in \Phi$, $\deg_{\mathcal{C}} \varphi(z) \leq \deg_{\mathcal{C}} z + l$ so the restriction of a codeword f to $(P, \sigma_2(P) \dots \sigma_m(P))$ is a codeword of $\mathcal{C}_{\mathcal{L}}(G', D)$ for some G' such that $\deg G' \leq (m-1)l + \deg G$. $\mathcal{C}_{\mathcal{L}}(G', D)$ has distance $d - (m-1)l$ and so we can handle $\frac{d-(m-1)l}{2}$ errors. By Markov, the probability of having more than $\frac{d-(m-1)l}{2}$ errors is at most $\frac{2\delta}{d-(m-1)l}$. □

We will spend the remainder of this section proving the following theorem:

Theorem 8.4 (Samplers for Hermitian codes)**.** *Hermitian codes have a* $\left(\frac{q^3}{t}, t\right)$ *sampler for any* $0 < t \leq 1$

The corollary of which will be:

Theorem 8.5 (High degree correction of degree lifted Hermitian codes)**.**

$$\mathcal{C}_L^m\left(\left(\left(1 - \frac{m}{t}\right)q^3 - c\delta q^3\right) \cdot Q_\infty, D\right)$$

has a locally correctable subcode of rate:

$$\frac{k}{n} \approx \frac{\left(\left(1 - \frac{m}{t}\right)q^3 - c\delta q^3\right)^m - m\left(\left(1 - \frac{m}{t}\right)q^3 - c\delta q^3\right)^{m-1} q^2}{tm! q^{3m}} = \frac{1}{tm!}\left(1 - \frac{m}{t} - c\delta - \frac{1}{q}\right)$$

, query complexity $\sqrt[m]{n}$ *and alphabet size of* $\sqrt[1.5m]{n}$ *.*

Our set of functions will be:

$$\varphi(x) = ax + b$$

$$\varphi(y) = a^{q+1}y + b^q ax + c + T^*(\alpha x + \beta y)$$

Where $a \in \mathbb{F}_{q^2}^*, (b, c) \in N_H, \alpha, \beta \in \mathbb{F}_{q^2}$ and $T^*(z) = z^q - z$.

We note that $Tr(T^*(z)) \equiv 0$ so these are indeed generalized automorphisms.

We define $\mathcal{C}_{\mathcal{L}}^m(r \cdot Q_\infty, N_H)_{\frac{1}{t}}$ to be the sub-code of $\mathcal{C}_{\mathcal{L}}^m(r \cdot Q_\infty, N_H)$ in which the y-degrees of the evaluated functions are bound by $\frac{q}{t}$. This bounds the potential increase in the curve degree of the functions by an additive factor of $\frac{mq^3}{t}$, and decreases the dimension by a multiplicative factor of at most $\frac{1}{t}$. All that remains is to show that these functions cover the whole space uniformly. This will be proven in lemma 8.9 but we need to do a little work before-hand.

First we show that these function are actually different from one another.

Lemma 8.6 (High degree testers are different)**.** *If* $\forall (x, y) \in N_H : \varphi_1(x, y) = \varphi_2(x, y)$ *then* $\varphi_1 \equiv \varphi_2$

Proof. we can immediately derive that $a_1 = a_2$ and $b_1 = b_2$, we are left with the equation $T^*((\alpha_1 - \alpha_2)x + (\beta_1 - \beta_2)y) + c_1 - c_2 = 0$, pick $y_0 \in \mathbb{F}_{q^2}$ s.t. $y_0^q + y = 1$ and set $y = y_0$. This equation then become a degree q polynomial in x, but there are $q+1$ x values for which (x, y_0) is a Hermitian rational point. So this polynomial must have $q+1$ zeroes, so we get that $\alpha_1 = \alpha_2$ and $c_1 = c_2$. We can then conclude that $\beta_1 = \beta_2$ and $\varphi_1 \equiv \varphi_2$. □

Now we show that to correct a particular point we have many possible tests.

Lemma 8.7 (Equal number of functions through each point)**.** *Let* $(x_1, y_1), (x_2, y_2) \in N_H$*, the number of functions for which* $\varphi(x_1, y_1) = (x_2, y_2)$ *is* $q^4\left(q^2-1\right)$*.*

Proof. Pick any a $\left(q^2-1 \text{ options}\right)$, there is a single solution to $ax_1 + b = x_2$ so b is set.

We need $y_2 = b^q a x_1 + a^{q+1} y_1 + c + T^*(\alpha x + \beta y)$ pick any c such that $b^{q+1} = c^q + c$ (q options), and let $b^q a x_1 + a^{q+1} y_1 + c = \lambda$ then we need $y_2 - \lambda = T^*(\alpha x_1 + \beta y_1)$, note that $Tr(\lambda) = N(x_2) = Tr(y_2)$ so $y_2 - \lambda$ is a trace zero element, denote it by λ'.

There are q elements z in $\mathbb{F}_{q^2}$ such that $z^q - z = \lambda'$ ($T^*$is an additive homomorphism from $\mathbb{F}_{q^2}$ onto the trace zero elements). So for any choice of α there are q options of β such that $T^*(\alpha x_1 + \beta y_1) = \lambda'$. (the only issue is if $x_1, y_1 = (0,0)$. However, in that case, $\varphi(0,0) = (b,c)$ regardless of a, α, β, so we just set $(b,c) = (x_2, y_2)$ and still get the same number of φ's). So the number of $\varphi' s$ going through (x_1, y_1, x_2, y_2) is $\left(q^2-1\right) \cdot q \cdot q^2 \cdot q$. □

So when looking to correct a particular point, we can choose between $q^4\left(q^2-1\right)$ different testers. And now we can show that these tests cover nearly the whole space.

Lemma 8.8 (Equal number of functions through each 2 points)**.** *Let* $\left(x_1, y_1, x_1^{'}, y_1^{'}\right), \left(x_2, y_2, x_2^{'}, y_2^{'}\right)$ *be two points on the Hermitian plane such that* $x_1 \neq x_2, x_1^{'} \neq x_2^{'}$ *then there are* q^3 *functions such that* $\varphi(x_1, y_1) = \left(x_1^{'}, y_1^{'}\right), \varphi(x_2, y_2) = \left(x_2^{'}, y_2^{'}\right)$

Proof. The conditions on the x's yield the equations $ax_1 + b = x_1^{'}, ax_2 + b = x_2^{'}$ which have a single solution when $x_1 \neq x_2$.

Pick any c such that $Tr(c) = N(b)$ (q options) and we now get the equations:

$$b^q a x_1 + a^{q+1} y_1 + c + T^*(\alpha x_1 + \beta y_1) = y_1^{'}$$

$$b^q a x_2 + a^{q+1} y_2 + c + T^*(\alpha x_2 + \beta y_2) = y_2^{'}$$

let $b^q a x_1 + a^{q+1} y_1 + c = \lambda_1, b^q a x_2 + a^{q+1} y_2 + c = \lambda_2$ and note that $Tr(\lambda_i) = Tr\left(y_i^{'}\right)$ so $y_1^{'} - \lambda_1 = \lambda_1^{'}$ and $y_2^{'} - \lambda_2 = \lambda_2^{'}$ are both trace zero elements. Pick z_1, z_2 such that $T^*(z_i) = \lambda_i^{'}$ (q^2 options) , The set of equations $\alpha x_i + \beta y_i = z_i$ has a solution unless $\exists \gamma : (x_1, y_1) = (\gamma x_2, \gamma y_2)$ this however implies that $Tr(\gamma y_2) = N(\gamma x_2) \implies \left(\gamma^{q+1} - \gamma^q\right) = 0$ and so can only happen if $\gamma \in \{0,1\}$, $\gamma = 1$ would mean that both points are equal which is is not an option. If $\gamma = 0$ then $x_1 = y_1 = 0$. In that case, instead of picking a random c we pick one where $c = y_1^{'}$. We get that $\varphi(0,0) = x_1^{'}, y_1^{'}$ for any values of α, β. But we still need to get $T^*(\alpha x_2 + \beta y_2) = \lambda_2^{'}$, for every choice of α there are q options for β for a total of q^3 total options. □

Lemma 8.9 (Uniform coverage by high degree functions)**.** *The set of functions used to correct any point cover nearly the whole space uniformly.*

Proof. for any point $\left(x_1, y_1, x_1^{'}, y_1^{'}\right)$ there are $q^4\left(q^2-1\right)$ functions passing through it, pick (x_2, y_2) such that $x_2 \neq x_1$, For any choice of $x_2^{'}, y_2^{'}$ such that $x_2^{'} \neq x_1^{'}$ (there are $\left(q^2-1\right)q$) such points), there are q^3 functions passing through $\left(x_1, y_1, x_1^1, y_1^{'}\right)$ and $\left(x_2, y_2, x_2^{'}, y_2^{'}\right)$ so going through all the options for $x_2^{'}, y_2^{'}$ we cycle through all the functions passing through $\left(x_1, y_1, x_1^{'}, y_1^{'}\right)$, which

shows that the functions passing through a single point cover almost all of the space in a uniform manner. □

The fraction of points unsampled by the functions through a particular point is so small as to not matter. We can then conclude that
$\mathcal{C}_L^m\left(\left(\left(1-\frac{m}{t}\right)q^3 - c\delta q^3\right)\cdot Q_\infty, D\right)_{\frac{q}{t}}$ is the subcode for which Theorem 8.5 applies.

References

[AKK+05] Noga Alon, Tali Kaufman, Michael Krivelevich, Simon Litsyn, and Dana Ron, *Testing Reed-Muller codes*, IEEE Transactions on Information Theory **51** (2005), no. 11, 4032–4039.

[AS98] Sanjeev Arora and Shmuel Safra, *Probabilistic checking of proofs: A new characterization of NP*, Journal of the ACM **45** (1998), no. 1, 70–122.

[AS03] Sanjeev Arora and Madhu Sudan, *Improved low-degree testing and its applications*, Combinatorica **23** (2003), no. 3, 365–426.

[BAETS10] Avraham Ben-Aroya, Klim Efremenko, and Amnon Ta-Shma, *Local list decoding with a constant number of queries*, Electronic Colloquium on Computational Complexity (ECCC) **17** (2010), 47.

[BATS09] Avraham Ben-Aroya and Amnon Ta-Shma, *Constructing small-bias sets from algebraic-geometric codes*, FOCS, IEEE Computer Society, 2009, pp. 191–197.

[BF90] Donald Beaver and Joan Feigenbaum, *Hiding instances in multioracle queries*, Proceedings of the 7th Annual Symposium on Theoretical Aspects of Computer Science, STACS '90, 1990, pp. 37–48.

[BFL92] László Babai, Lance Fortnow, and Carsten Lund, *Addendum to non-deterministic exponential time has two-prover interactive protocols*, Computational Complexity **2** (1992), 374.

[BFLS91] László Babai, Lance Fortnow, Leonid A. Levin, and Mario Szegedy, *Checking computations in polylogarithmic time*, STOC '91: Proceedings of the 23rd Annual ACM Symposium on Theory of Computing (New York, NY, USA), ACM, 1991, pp. 21–32.

[BFNW93] László Babai, Lance Fortnow, Noam Nisan, and Avi Wigderson, *Bpp has subexponential time simulations unless exptime has publishable proofs*, Computational Complexity **3** (1993), 307–318.

[BGK+09] Eli Ben-Sasson, Venkatesan Guruswami, Tali Kaufman, Madhu Sudan, and Michael Viderman, *Locally testable codes require redundant testers*, Proceedings of the 24th Annual IEEE Conference on Computational Complexity (CCC), 2009, pp. 52–61.

[BGM+11] Eli Ben-Sasson, Elena Grigorescu, Ghid Maatouk, Amir Shpilka, and Madhu Sudan, *On sums of locally testable affine invariant properties*, Electronic Colloquium on Computational Complexity (ECCC) **18** (2011), 79.

[BHR05] Eli Ben-Sasson, Prahladh Harsha, and Sofya Raskhodnikova, *Some 3CNF properties are hard to test*, SIAM J. on Computing **35** (2005), no. 1, 1–21.

[BMSS10] Eli Ben-Sasson, Ghid Maatouk, Amir Shpilka, and Madhu Sudan, *Symmetric LDPC codes are not necessarily locally testable*, In preparation, 2010.

[BRS12] Eli Ben-Sasson, Noga Ron-Zewi, and Madhu Sudan, *Sparse affine-invariant linear codes are locally testable*, Electronic Colloquium on Computational Complexity (ECCC) **19** (2012), 49.

[BS05] Eli Ben-Sasson and Madhu Sudan, *Simple PCPs with poly-log rate and query complexity*, Proceedings of the 37th Annual ACM Symposium on Theory of Computing, Baltimore, MD, USA, May 22-24, 2005 (Harold N. Gabow and Ronald Fagin, eds.), ACM, 2005, pp. 266–275.

[BS10] ———, *Limits on the rate of locally testable affine-invariant codes*, Electronic Colloquium on Computational Complexity (ECCC) **17** (2010), 108.

[BSS03] László Babai, Amir Shpilka, and Daniel Stefankovic, *Locally testable cyclic codes*, Proceedings: 44th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2003, 11–14 October 2003, Cambridge, Massachusetts (pub-IEEE:adr) (IEEE, ed.), IEEE Computer Society Press, 2003, pp. 116–125.

[BV09a] Eli Ben-Sasson and Michael Viderman, *Composition of Semi-LTCs by Two-Wise Tensor Products*, Proceedings of the Approximation, Randomization, and Combinatorial Optimization, (APPROX-RANDOM 2009), Lecture Notes in Computer Science, vol. 5687, Springer, 2009, pp. 378–391.

[BV09b] ———, *Tensor Products of Weakly Smooth Codes are Robust*, Theory of Computing **5** (2009), no. 1, 239–255.

[CFL+10] Yeow Meng Chee, Tao Feng, San Ling, Huaxiong Wang, and Liang Feng Zhang, *Query-efficient locally decodable codes of subexponential length*, CoRR **abs/1008.1617** (2010).

[CGKS98] Benny Chor, Oded Goldreich, Eyal Kushilevitz, and Madhu Sudan, *Private information retrieval*, Journal of the ACM **45** (1998), 965–981.

[CR05] Don Coppersmith and Atri Rudra, *On the Robust Testability of Product of Codes*, Electronic Colloquium on Computational Complexity (ECCC) (2005), no. 104.

[DGY10] Zeev Dvir, Parikshit Gopalan, and Sergey Yekhanin, *Matching vector codes*, Electronic Colloquium on Computational Complexity (ECCC) **17** (2010), 12.

[DJK+02] Amit Deshpande, Rahul Jain, Telikepalli Kavitha, Jaikumar Radhakrishnan, and Satyanarayana V. Lokam, *Better lower bounds for locally decodable codes*, IEEE Conference on Computational Complexity, 2002, pp. 184–193.

[DSW06] Irit Dinur, Madhu Sudan, and Avi Wigderson, *Robust Local Testability of Tensor Products of LDPC Codes*, Proceedings of the Approximation, Randomization, and Combinatorial Optimization, (APPROX-RANDOM 2006), Lecture Notes in Computer Science, vol. 4110, Springer, 2006, pp. 304–315.

[Efr09] Klim Efremenko, *3-query locally decodable codes of subexponential length*, STOC (Michael Mitzenmacher, ed.), ACM, 2009, pp. 39–44.

[Efr12] ———, *From irreducible representations to locally decodable codes*, STOC, 2012, pp. 327–338.

[FS95] Katalin Friedl and Madhu Sudan, *Some improvements to total degree tests*, ISTCS, 1995, pp. 190–198.

[GKS09] Elena Grigorescu, Tali Kaufman, and Madhu Sudan, *Succinct representation of codes with applications to testing*, APPROX-RANDOM (Irit Dinur, Klaus Jansen, Joseph Naor, and José D. P. Rolim, eds.), Lecture Notes in Computer Science, vol. 5687, Springer, 2009, pp. 534–547.

[GKST02] Oded Goldreich, Howard J. Karloff, Leonard J. Schulman, and Luca Trevisan, *Lower bounds for linear locally decodable codes and private information retrieval*, IEEE Conference on Computational Complexity, 2002, pp. 175–183.

[Gop82] Valery Denisovich Goppa, *Algebraic-geometric codes*, Izu. Akad. Nauk SSSR Ser. Mat **46** (1982), no. 4, 762–781.

[Gop88] ———, *Geometry and codes*, Springer, 1988.

[GS96] Arnaldo Garcia and Henning Stichtenoth, *On the Asymptotic Behaviour of Some Towers of Function Fields over Finite Fields*, Journal of Number Theory **61** (1996), 248–273.

[GS12] Alan Guo and Madhu Sudan, *New affine-invariant codes from lifting*, Electronic Colloquium on Computational Complexity (ECCC) **19** (2012), 106.

[Ham50] Richard W. Hamming, *Error detecting and error correcing codes*, Bell System Technical Journal **29** (1950), 147–160.

[Han01] S.H. Hansen, *Error-correcting codes from higher-dimensional varieties*, Finite Fields Appl. **7** (2001), 530–552.

[HS90] Johan P. Hansen and Henning Stichtenoth, *Group codes on certain algebraic curves with many rational points*, Appl. Algebra Eng. Commun. Comput. **1** (1990), 67–77.

[IS10] Toshiya Itoh and Yasuhiro Suzuki, *Improved constructions for query-efficient locally decodable codes of subexponential length*, IEICE Transactions **93-D** (2010), no. 2, 263–270.

[IW97] Russell Impagliazzo and Avi Wigderson, *$P = BPP$ if e requires exponential circuits: Derandomizing the xor lemma*, STOC (Frank Thomson Leighton and Peter W. Shor, eds.), ACM, 1997, pp. 220–229.

[KdW04] Iordanis Kerenidis and Ronald de Wolf, *Exponential lower bound for 2-query locally decodable codes via a quantum argument*, J. Comput. Syst. Sci. **69** (2004), no. 3, 395–420.

[KL12] Tali Kaufman and Alexander Lubotzky, *Edge transitive ramanujan graphs and symmetric ldpc good codes*, STOC (Howard J. Karloff and Toniann Pitassi, eds.), ACM, 2012, pp. 359–366.

[KS08] Tali Kaufman and Madhu Sudan, *Algebraic property testing: the role of invariance*, Proceedings of the 40th Annual ACM Symposium on Theory of Computing (STOC), 2008, pp. 403–412.

[KSY11] Swastik Kopparty, Shubhangi Saraf, and Sergey Yekhanin, *High-rate codes with sublinear-time decoding*, STOC, 2011, pp. 167–176.

[KT00] Jonathan Katz and Luca Trevisan, *On the efficiency of local decoding procedures for error-correcting codes*, Proceedings of the thirty-second annual ACM symposium on Theory of computing (New York, NY, USA), STOC '00, ACM, 2000, pp. 80–86.

[KV10] Tali Kaufman and Michael Viderman, *Locally testable vs. locally decodable codes*, APPROX-RANDOM (Maria J. Serna, Ronen Shaltiel, Klaus Jansen, and José D. P. Rolim, eds.), Lecture Notes in Computer Science, vol. 6302, Springer, 2010, pp. 670–682.

[KW10] Tali Kaufman and Avi Wigderson, *Symmetric ldpc codes and local testing*, ICS (Andrew Chi-Chih Yao, ed.), Tsinghua University Press, 2010, pp. 406–421.

[Lac93] G. Lachaud, *Number of points of plane sections and linear codes defined on algebraic varieties*, Arithmetic, Geometry and Coding Theory, Proceedings Luminy (1993), 77–104.

[LFKN92] Carsten Lund, Lance Fortnow, Howard Karloff, and Nisan Noam, *Algebraic methods for interactive proof systems*, Journal of the ACM **39** (1992), no. 4, 859–868.

[LGS05] J. Little L. Gold and H. Schenck, *Cayley-bacharach and evaluation codes on complete intersections*, J. Pure Appl. Algebra **196** (2005), 91–99.

[Lip90] Richard J. Lipton, *Efficient checking of computations*, Proceedings of the 7th Annual Symposium on Theoretical Aspects of Computer Science, STACS '90, 1990, pp. 207–215.

[Lit08] John B. Little, *Algebraic geometry codes from higher dimensional varieties*, CoRR **abs/0802.2349** (2008).

[Mei10] Or Meir, *Ip = pspace using error correcting codes*, Electronic Colloquium on Computational Complexity (ECCC) **17** (2010), 137.

[MR08] Dana Moshkovitz and Ran Raz, *Two-query PCP with subconstant error*, Journal of the ACM **57** (2008), 1–29, Preliminary version appeared in FOCS '08.

[MS78] Florence J. MacWilliams and Neil J. A. Sloane, *The theory of error-correcting codes*, North-Holland Amsterdam, 1978.

[MTZ82] S.G. Vladut M.A. Tsfasman and T. Zink, *Modular curves, shimura curves, and goppa codes, better than varshamov-gilbert bound*, math. Nachr. **109** (1982), 21–28.

[Oba02] Kenji Obata, *Optimal lower bounds for 2-query locally decodable linear codes*, RANDOM (José D. P. Rolim and Salil P. Vadhan, eds.), Lecture Notes in Computer Science, vol. 2483, Springer, 2002, pp. 39–50.

[Ped92] J.P. Pedersen, *A function field related to the ree group*, Lect. Notes Math. **1518** (1992), 122–132.

[PS94] Alexander Polishchuk and Daniel A. Spielman, *Nearly-linear size holographic proofs*, STOC '94: Proceedings of the 26th Annual ACM Symposium on Theory of Computing (New York, NY, USA), ACM, 1994, pp. 194–203.

[Rag07] Prasad Raghavendra, *A note on yekhanin's locally decodable codes*, Electronic Colloquium on Computational Complexity (ECCC) **14** (2007), no. 016.

[Ree54] Irving S. Reed, *A class of multiple-error-correcting codes and the decoding scheme*, IEEE Transactions on Information Theory (1954), no. 4, 38–49.

[rod03] F. rodier, *Codes from flag varieties over a finite field*, J. Pure Appl. Algebra **178** (2003), 203–214.

[RS97] Ran Raz and Shmuel Safra, *A sub-constant error-probability low-degree test, and a sub-constant error-probability pcp characterization of np*, Proceedings of the 29th Annual ACM Symposium on Theory of Computing, STOC '97, 1997, pp. 475–484.

[RY07] Alexander A. Razborov and Sergey Yekhanin, *An omega($n^{1/3}$) lower bound for bilinear group based private information retrieval*, Theory of Computing **3** (2007), no. 1, 221–238.

[Sha48] Claude E. Shannon, *A mathematical theory of communication*, Bell System Technical Journal **27** (1948), 379–423, 623656.

[Sha53] ———, *Communication theory - exposition of fundamentals*, IEEE Transactions on Information Theory **1** (1953), 44–47.

[Sha92] Adi Shamir, *IP = PSPACE*, Journal of the ACM **39** (1992), no. 4, 869–877.

[Sti] Henning Stichtenoth, *On automorphisms of geometric goppa codes*, Journal of Algebra, 113.

[Sti93] ———, *Algebraic function fields and codes*, Universitext, Springer, 1993.

[STV99] Madhu Sudan, Luca Trevisan, and Salil P. Vadhan, *Pseudorandom generators without the xor lemma (extended abstract)*, STOC (Jeffrey Scott Vitter, Lawrence L. Larmore, and Frank Thomson Leighton, eds.), ACM, 1999, pp. 537–546.

[SU05] Ronen Shaltiel and Christopher Umans, *Simple extractors for all min-entropies and a new pseudorandom generator*, J. ACM **52** (2005), no. 2, 172–216.

[Sud10] Madhu Sudan, *Invariance in property testing*, Electronic Colloquium on Computational Complexity (ECCC) (2010), no. 051.

[TS91] M.A. Tsfasman and S.G.Vladut, *Algebraic-geometric codes*, (Kluwer, Dordrecht, 1991.

[Val05] Paul Valiant, *The tensor product of two codes is not necessarily robustly testable*, APPROX-RANDOM (Chandra Chekuri, Klaus Jansen, José D. P. Rolim, and Luca Trevisan, eds.), Lecture Notes in Computer Science, vol. 3624, Springer, 2005, pp. 472–481.

[Vid10] M. Viderman, *A note on high-rate locally testable codes with sublinear query complexity.*

[WdW05] Stephanie Wehner and Ronald de Wolf, *Improved lower bounds for locally decodable codes and private information retrieval*, ICALP (Luís Caires, Giuseppe F. Italiano, Luís Monteiro, Catuscia Palamidessi, and Moti Yung, eds.), Lecture Notes in Computer Science, vol. 3580, Springer, 2005, pp. 1424–1436.

[Woo07] David P. Woodruff, *New lower bounds for general locally decodable codes*, Electronic Colloquium on Computational Complexity (ECCC) **14** (2007), no. 006.

[Xin95] Chaoping Xing, *On automorphism groups of the hermitian codes*, IEEE Transactions on Information Theory **41** (1995), no. 6, 1629–1635.

[Yek08] Sergey Yekhanin, *Towards 3-query locally decodable codes of subexponential length*, J. ACM **55** (2008), no. 1.

[Yek11] ———, *Locally decodable codes: A brief survey*, IWCC (Yeow Meng Chee, Zhenbo Guo, San Ling, Fengjing Shao, Yuansheng Tang, Huaxiong Wang, and Chaoping Xing, eds.), Lecture Notes in Computer Science, vol. 6639, Springer, 2011, pp. 273–282.

ECCC ISSN 1433-8092

http://eccc.hpi-web.de