List Decoding Reed–Solomon Codes in the Lee, Euclidean, and Other Metrics

Chris Peikert* Alexandra Veliche Hostetler†

October 13, 2025

Abstract

Reed–Solomon error-correcting codes are ubiquitous across computer science and information theory, with applications in cryptography, computational complexity, communication and storage systems, and more. Most works on efficient error correction for these codes, like the celebrated Berlekamp–Welch unique decoder and the (Guruswami–)Sudan list decoders, are focused on measuring error in the Hamming metric, which simply counts the number of corrupted codeword symbols. However, for some applications, other metrics that depend on the specific values of the errors may be more appropriate.

This work gives a polynomial-time algorithm that list decodes (generalized) Reed–Solomon codes over prime fields in ℓ_p (semi)metrics, for any $0 < p \leq 2$. Compared to prior algorithms for the Lee (ℓ_1) and Euclidean (ℓ_2) metrics, ours decodes to arbitrarily large distances (for correspondingly small rates), and has better distance-rate tradeoffs for all decoding distances above some moderate thresholds. We also prove lower bounds on the ℓ_1 and ℓ_2 minimum distances of a certain natural subclass of GRS codes, which establishes that our list decoder is actually a *unique* decoder for many parameters of interest. Finally, we analyze our algorithm's performance under *random* Laplacian and Gaussian errors, and show that it supports even larger rates than for corresponding amounts of worst-case error in ℓ_1 and ℓ_2 (respectively).

1 Introduction

Reed–Solomon codes [RS60] are among the most widely used families of error-correcting codes, with applications across computer and communication sciences. Their many virtues include: a very simple definition; the largest possible minimum distance as a function of rate; and efficient decodability from errors, via either *unique* decoding up to half the minimum distance (see, e.g., [GRS19, Section 12.1]), or *list* decoding up to the larger Johnson bound, via the celebrated works of Sudan [Sud97] and Gururswami–Sudan [GS98] (see also [GRS19, Section 12.2]).

List decoding [Eli58, Woz58] is the task of finding all codewords that are within some desired distance of a (potentially corrupted) received word. When this radius is more than half the code's minimum distance, there can potentially be more than one codeword within range (hence the name "list decoding"). Despite this non-uniqueness, list decoding can suffice for many purposes (e.g., finding a nearest codeword within range), and indeed, it has found numerous applications.

Most work on decoding Reed–Solomon codes has measured errors in the *Hamming* metric, which simply counts the *number* of corrupted codeword symbols (regardless of how they are corrupted). However, there

*University of Michigan, cpeikert@umich.edu.
†University of Michigan, aveliche@umich.edu.

are many other natural metrics that depend on the specific *values* of the errors. Such metrics can be more appropriate for settings where introducing a "large" error at a coordinate is more costly than a "small" error, or where the communication channel might add some nonzero error to every coordinate. An example is a channel that adds error according to a Gaussian or other fairly concentrated distribution. When the code alphabet is $\mathbb{Z}_q$ (the integers modulo q)—in particular, a prime field $\mathbb{F}_q$—one metric of frequent study is the Lee metric, which is merely the ℓ_1 norm $\|\mathbf{x}\|_1 = \sum_i |x_i|$ after lifting $\mathbb{Z}_q$ to its distinguished representatives in $[-q/2, q/2)$. Other natural, analogously defined choices include the Euclidean (ℓ_2) or other ℓ_p metrics.

We know of only a few prior works on efficiently decoding Reed–Solomon codes in metrics others than Hamming. For the Lee (ℓ_1) metric, Roth and Siegel [RS94] gave an algorithm that uniquely decodes up to half of (a lower bound on) the minimum distance; their algorithm works for certain subclasses of (generalized) Reed-Solomon and BCH codes. In addition, Wu, Kuijper, and Udaya [WKU03] gave a list-decoding algorithm for ℓ_1, built around Guruswami–Sudan [GS98], that decodes to larger distances than in [RS94] for all small enough rates. Finally, for the Euclidean (ℓ_2) metric, Mook and Peikert [MP22] recently gave a list-decoding algorithm that also uses [GS98] as a black box.

1.1 Contributions

This work gives a polynomial-time algorithm that list decodes any generalized Reed–Solomon (GRS) code over a prime field in the ℓ_p (semi)metric for any $0 < p \leq 2$; in particular, this includes the Lee (ℓ_1) and Euclidean (ℓ_2) metrics.[1] Our algorithm works for a broader range of parameters, and has a better distance-rate tradeoff for all decoding distances above some moderate thresholds, than the prior algorithms for ℓ_1 and ℓ_2 [RS94, WKU03, MP22]; see below for elaboration and Figure 1 for a visual depiction. For ease of comparison across the various works and ℓ_p (semi)metrics, we use a suitably normalized version of distance: for code length n, distance d corresponds to *relative distance* $\delta := d/n^{1/p}$.

For $p = 2$, our algorithm can handle an *arbitrarily large* decoding distance, for a correspondingly small enough rate: specifically, as δ and the alphabet size grow, we can decode for rates rapidly approaching $1/(\delta\sqrt{2\pi e})$. By contrast, the prior work [MP22] applies only for relative distance $\delta < 1/\sqrt{2} \approx 0.7071$ (i.e., ℓ_2 distance less than $\sqrt{n/2}$). In addition, our algorithm works for larger rates than the one in [MP22] whenever δ exceeds about 0.51797. (See Section 5.2 for a detailed comparison.) This is particularly interesting since the rates obtained in [MP22] were shown to be *optimal* (in a certain sense) for $\delta < 1/2$, but not for larger values.

For $p = 1$, again our algorithm (like the one from [WKU03]) can handle an arbitrarily large decoding distance, whereas [RS94] is limited to relative distance $\delta < 1$ (i.e., ℓ_1 distance less than n). In addition, our algorithm works for larger rates than those of [RS94, WKU03] whenever the relative decoding distance exceeds about 0.78988, and in general, as δ and the alphabet size grow, we can decode for rates rapidly approaching $1/(2e\delta)$. (See Section 6.2 for details.) Our algorithm is also qualitatively broader: it decodes from *continuous* (real-valued) error, whereas the ones from [RS94, WKU03] require *discrete* (integer) error. While continuous error can be discretized by rounding, this can increase the relative distance from the codeword by up to $1/2$ in ℓ_1, which significantly degrades the distance-rate tradeoffs of the prior works, making them worse than ours for all distances.

We also give several useful supplementary results. By adapting an argument of [RS94], we prove lower bounds on the ℓ_1 and ℓ_2 minimum distances for a certain natural subclass of GRS codes. These imply that for many parameters of interest, our list-decoding algorithm outputs at most one codeword, i.e., it is actually a *unique* decoder. (See Lemmas 6.4 and 5.7 and the discussions thereafter.) And in addition to *worst-case* errors

[1] A semimetric is just a metric that does not necessarily satisfy the triangle inequality (which we will not need).

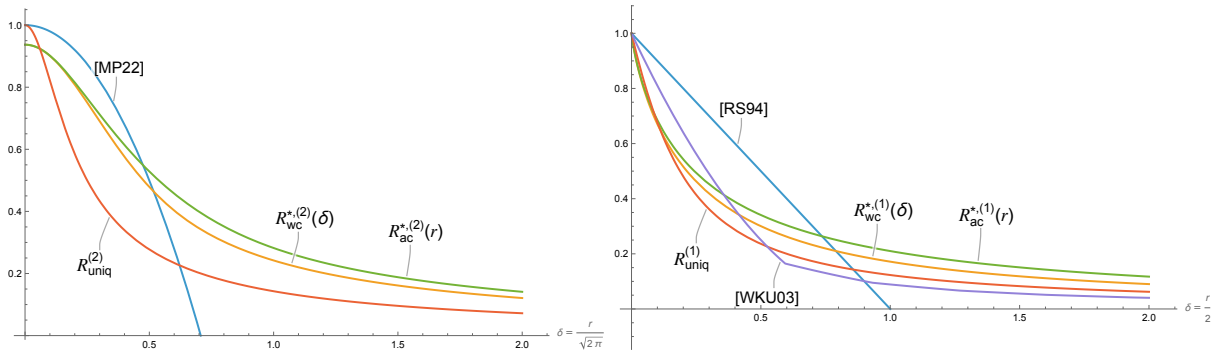

Figure 1: Plots of the adjusted rate $R^{*,(p)}$, as a function of the ℓ_p relative decoding distance $\delta = d/n^{1/p}$ or corresponding channel error width $r = p^{1/p} \cdot c_p \cdot \delta$, for which our algorithm can list decode prime-field GRS codes in the worst case (wc) or average case (ac), respectively, for $p = 2$ (left) and $p = 1$ (right). (For simplicity, these plots assume a field size $q \gg \delta, r$.) For comparison, also shown are the corresponding functions from the prior work on decoding GRS codes in these metrics: [MP22] is for list decoding in ℓ_2, and [RS94, WKU03] are respectively for *unique* and *list* decoding in the ℓ_1 (Lee) metric, but only for *discrete* (integer) error. Also shown are rate bounds $R^{(p)}_{\text{uniq}}$ for which decoding to ℓ_p relative distance δ is guaranteed to yield a *unique* codeword, for a certain natural subclass of GRS codes. (See Lemmas 6.4 and 5.7 and the discussions thereafter.)

added by an adversarial channel, we also consider our algorithm's performance under *average-case* errors produced by "memoryless additive" channels. Such channels add independent identically distributed error, drawn from some specified distribution, to each coordinate of the transmitted codeword. For Laplacian and Gaussian errors (which roughly correspond to ℓ_1 and ℓ_2, respectively), we show that our algorithm supports even larger rates than what we would get by merely applying concentration bounds on the error vector and invoking our worst-case results.

1.2 Technical Overview

At the highest level, our algorithm for list decoding prime-field GRS codes in ℓ_p follows the basic approach of [MP22] for list decoding (G)RS codes in ℓ_2: we first translate the received word into a suitable *weight* (or *reliability*) vector, then invoke a *soft-decision* list-decoding algorithm [GS98, Gur01, KV03] for GRS codes. Informally, a weight vector specifies, for each coordinate of the received word and each symbol in the code alphabet, a "confidence level" that the transmitted codeword had that symbol at that coordinate. Given such a weight vector, a soft-decision decoding algorithm then finds all codewords that are sufficiently *correlated* with it, as determined by the code rate. (For the formal definitions and theorem statement, see Definitions 3.1 and 3.2 and Theorem 3.3.)

For our purposes, the principal challenge is in mapping a received word to an appropriate weight vector so that any codeword that is close enough to the received word (in the ℓ_p metric) has sufficient correlation with the weight vector. The prior work [MP22] uses very simple weights: given a real received value r, only its floor $\lfloor r \rfloor$ and ceiling $\lceil r \rceil$ receive positive weights, of $1 - (r - \lfloor r \rfloor)$ and $1 - (\lceil r \rceil - r)$, respectively. It was shown that with these weights, the cited soft-decision algorithms decode up to any ℓ_2 relative distance $\delta < 1/\sqrt{2}$ for any code rate up to $1 - 2\delta^2$. Moreover, for $\delta \leq 1/2$ it was shown that this rate is *optimal* for those algorithms, i.e., no other weight assignment can work for a larger rate. However, [MP22] did not consider larger decoding distances than these, nor other metrics.

In this work, to handle large decoding distances and other metrics, we use "smoother" weights, which typically assign a positive weight to *every* alphabet symbol. Our overall approach (see Section 3) is quite general, and is parameterized by a function $f\colon \mathbb{R} \to [0,1]$ satisfying mild hypotheses, primarily that its Fourier transform $\widehat{f}$ is non-negative (see Assumption 2.12). This function can be seen as defining a weight—or a relative "likelihood," in the case of a random channel—for every potential real-valued error.[2] For a prime-order code alphabet $\mathbb{F}_q \cong \mathbb{Z}_q := \mathbb{Z}/q\mathbb{Z}$, and a received real value $r \in \mathbb{R}/q\mathbb{Z}$, we assign weight $f(r - w + q\mathbb{Z}) = \sum_{e = r-w \pmod q} f(e)$ to each alphabet symbol $w \in \mathbb{Z}_q$. Since the elements of the coset $r - w + q\mathbb{Z}$ are exactly those errors that would convert a transmitted symbol w to the received value r, this assignment captures the total weight of all such errors.

Our first main result, given in Theorem 3.5, lower bounds the correlation between the weight vector for a received word $\mathbf{r}$ over $\mathbb{R}/q\mathbb{Z}$ and any (code)word $\mathbf{c}$ over $\mathbb{Z}_q$, by the ratio of two quantities determined by f: its (arithmetic or geometric) mean over the coordinates of the error vector $\mathbf{r} - \mathbf{c}$, and (the square root of) its sum over a certain two-dimensional integer lattice $\mathcal{L}_q$. So, for a particular decoding distance in a metric of interest (or channel distribution, in the average case), the goal becomes to choose a suitable function f that nearly maximizes this ratio. The proof of the theorem uses a mild generalization of Fourier-analytic results on lattices from [Ban93, MR04], and is the source of our requirement that $\widehat{f}$ is non-negative, and ultimately the restriction that $0 < p \le 2$ for ℓ_p (semi)metrics.

The bulk of the remaining work is then devoted to making a suitable choice of function f for the ℓ_p (semi)metric (and corresponding channel distributions), and analyzing its summation over $\mathcal{L}_q$. In Section 4 we consider scalings of the function $f^{(p)}(x) := \exp(-|x|^p)$, which is known to have non-negative Fourier coefficients for $0 < p \le 2$ (but not for any other p). Then in Sections 5 and 6 we specialize to $p = 2$ and $p = 1$, respectively, and give fairly tight upper bounds on $f(\mathcal{L}_q)$ using Fourier-analytic techniques or direct analysis. Finally, we use these bounds to optimize the distance-rate tradeoffs for which we can list decode GRS codes in these ℓ_p metrics, and for Gaussian and Laplacian random channels as well.

2 Preliminaries

For a positive integer n, let $[n] := \{1, \ldots, n\}$. For a positive integer q, define the quotient ring $\mathbb{Z}_q := \mathbb{Z}/q\mathbb{Z}$ and the additive quotient group $\mathbb{R}_q := \mathbb{R}/q\mathbb{Z}$. For a prime power q, let $\mathbb{F}_q$ denote the finite field of size q. When q is prime, we identify $\mathbb{Z}_q$ with the finite field $\mathbb{F}_q$ in the natural way.

For any $x \in \mathbb{R}_q$ (which is a coset of $q\mathbb{Z}$), define its "lift" $\overline{x} \in [-q/2, q/2)$ to be the unique real number such that $\overline{x} = x \pmod q$, i.e., the "zero-centered" distinguished representative of x. We also apply this notation entry-wise to vectors over $\mathbb{R}_q$.

For any $p > 0$, define the ℓ_p (quasi)norm on $\mathbb{R}^n$ as $\|\mathbf{x}\|_p := \left(\sum_{i=1}^n |x_i|^p\right)^{1/p}$. It is well known that this is a norm if and only if $p \ge 1$, and is a *quasinorm* for any $p > 0$.[3] Similarly, we define the ℓ_p (semi)metric on $\mathbb{R}_q^n$ by lifting, i.e., via $\|\mathbf{x}\|_p := \|\overline{\mathbf{x}}\|_p$.[4] For $p = 1$, this generalizes the Lee metric over $\mathbb{Z}_q$ to $\mathbb{R}_q$.

For two groups X, Y, their *direct sum* group $X \oplus Y$ is their Cartesian product with the group operation defined component-wise. This notation extends to the direct sum of group *cosets*, which is a coset of the direct sum of the groups.

[2] For example, we take f to be a Gaussian function for decoding in the ℓ_2 metric, or under a Gaussian channel.

[3] A quasinorm relaxes the triangle inequality axiom to require only that $\|\mathbf{x} + \mathbf{y}\| \le K(\|\mathbf{x}\| + \|\mathbf{y}\|)$ for some fixed K. We do not use the triangle inequality, or even this relaxation, so we can consider $p < 1$.

[4] Formally, this is not a *norm* because it is not defined on a vector space (since $\mathbb{R}_q$ is not a field), and it does not satisfy homogeneity due to the mod-q reduction. However, it does define a *(semi)metric* (where "semi" does not require the triangle inequality), with distance function $d(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|_p$.

For any function $f\colon D \to \mathbb{C}$ and countable subset $X \subseteq D$, we define $f(X) := \sum_{\mathbf{x}\in X} f(\mathbf{x})$. We extend the domain to D^k multiplicatively, as

$$f^k(\mathbf{x}) := \prod_{i=1}^{k} f(x_i)\ , \tag{2.1}$$

often omitting the superscript k when it is clear from context. When $D = \mathbb{R}^n$, for any real $s \neq 0$ we define $f_s(\mathbf{x}) := f(\mathbf{x}/s)$.

For any two vectors $\mathbf{x} = (x_1, \ldots, x_n)$ and $\mathbf{y} = (y_1, \ldots, y_n)$ of the same dimension, their coordinate-wise (or *Hadamard*) product is denoted by $\mathbf{x} \odot \mathbf{y} := (x_1 \cdot y_1, \ldots, x_n \cdot y_n)$.

For a finite sequence $X_1, \ldots, X_n$ of real values, we denote their average by $\mathrm{Avg}_i[X_i] := \frac{1}{n}\sum_{i=1}^{n} X_i$. We use the following special case of the well known Hoeffding (lower-)tail bound.

Lemma 2.1 (Hoeffding's Inequality). *Let $X_1, \ldots, X_n$ be independent identically distributed random variables in $[0,1]$ with common expectation $\mu = \mathbb{E}[X_i]$. Then for any $\gamma \geq 0$,*

$$\Pr\Big[\operatorname*{Avg}_i[X_i] \leq \mu - \gamma\Big] < \exp(-2\gamma^2 n)\ .$$

2.1 Linear Codes

A *linear (error-correcting) code* of *(block) length* n over the *alphabet* $\mathbb{F}_q$ is a linear subspace of $\mathbb{F}_q^n$. As a subspace, it has a *dimension*. In this paper, we consider the following family of codes.

Definition 2.2 ((Generalized) Reed–Solomon code). Let $n \leq q$ be positive integers, with q a prime power. For a non-negative integer k, a vector $\boldsymbol{\alpha} \in \mathbb{F}_q^n$ with distinct entries, and a vector $\mathbf{t} \in (\mathbb{F}_q \setminus \{0\})^n$ with (not necessarily distinct) non-zero entries, the *Generalized Reed–Solomon* (GRS) code of dimension k with evaluation points $\boldsymbol{\alpha}$ and twist factors $\mathbf{t}$ is defined as

$$\mathrm{GRS}_{q,k}(\boldsymbol{\alpha}, \mathbf{t}) := \{\mathbf{t} \odot f(\boldsymbol{\alpha}) = (t_1 \cdot f(\alpha_1), \ldots, t_n \cdot f(\alpha_n)) : f \in \mathbb{F}_q[x],\ \deg(f) < k\}\ .^{5}$$

A special case is a *Reed–Solomon* (RS) code, which is obtained by using trivial twist factors $\mathbf{t} = (1, \ldots, 1)$.

2.2 Lattices

Definition 2.3 (Lattice, Basis). An *(n-dimensional, full-rank) lattice* $\mathcal{L} \subset \mathbb{R}^n$ is the set of all integer linear combinations of some n linearly independent *basis* vectors $\mathbf{B} = \{\mathbf{b}_1, \ldots, \mathbf{b}_n\} \subset \mathbb{R}^n$:

$$\mathcal{L} = \mathcal{L}(\mathbf{B}) := \Big\{\sum_{i=1}^{n} z_i \mathbf{b}_i : z_i \in \mathbb{Z}\Big\}\ .$$

Equivalently, it is a discrete additive subgroup of $\mathbb{R}^n$ whose $\mathbb{R}$-span is $\mathbb{R}^n$; as such, it defines the quotient group $\mathbb{R}^n/\mathcal{L}$ of *lattice cosets* $\mathbf{x} + \mathcal{L}$ for $\mathbf{x} \in \mathbb{R}^n$. A sublattice of $\mathbb{Z}^n$ is called an *integer lattice*.

In this work, all lattices are implicitly full rank. A lattice basis can equivalently be seen as an invertible matrix $\mathbf{B} \in \mathbb{R}^{n\times n}$ whose columns are the vectors $\mathbf{b}_1, \ldots, \mathbf{b}_n$. Note that a given lattice has multiple different bases, which are all related by right-multiplication by *unimodular* matrices in $\mathbb{Z}^{n\times n}$.

[5] By convention, the zero polynomial has degree $-\infty$.

Definition 2.4 (Determinant). The *determinant* of a lattice $\mathcal{L}$ generated by basis $\mathbf{B}$ is $\det(\mathcal{L}) := |\det(\mathbf{B})|$.

Note that the determinant of a lattice is invariant under the choice of basis, by the above-mentioned relationship between the bases of a lattice.

Definition 2.5 (Dual lattice). The *dual lattice* of a lattice $\mathcal{L} \subset \mathbb{R}^n$ is

$$\mathcal{L}^* := \{\mathbf{x} \in \mathbb{R}^n : \forall\, \mathbf{v} \in \mathcal{L},\ \langle \mathbf{v}, \mathbf{x} \rangle \in \mathbb{Z}\}\ .$$

If $\mathbf{B}$ is a basis of $\mathcal{L}$, then its *dual basis* $\mathbf{B}^* := \mathbf{B}^{-t}$ is a basis of $\mathcal{L}^*$, and hence $\det(\mathcal{L}^*) = \det(\mathcal{L})^{-1}$.

Lemma 2.6. *Let $f\colon D \to \mathbb{R}$ and $X, Y \subseteq D$ be countable subsets of its domain (e.g., lattice cosets). Then*

$$f(X \oplus Y) = f(X) \cdot f(Y).$$

Proof. This follows directly from the definition of direct sum and multiplicativity (Equation (2.1)):

$$f(X \oplus Y) = \sum_{\substack{\mathbf{x} \in X \\ \mathbf{y} \in Y}} f(\mathbf{x} \oplus \mathbf{y}) = \sum_{\mathbf{x},\mathbf{y}} f(\mathbf{x}) \cdot f(\mathbf{y}) = \Big(\sum_{\mathbf{x}} f(\mathbf{x})\Big)\Big(\sum_{\mathbf{y}} f(\mathbf{y})\Big) = f(X) \cdot f(Y). \qquad \square$$

2.3 Fourier Analysis

Let $f\colon \mathbb{R}^n \to \mathbb{C}$ be a (Borel) measurable function that satisfies $\int_{\mathbb{R}^n} |f(\mathbf{x})|\, d\mathbf{x} < \infty$. Its *Fourier transform* $\widehat{f}\colon \mathbb{R}^n \to \mathbb{C}$ is defined as

$$\widehat{f}(\mathbf{w}) := \int_{\mathbb{R}^n} f(\mathbf{x}) \cdot \exp(-2\pi i \langle \mathbf{x}, \mathbf{w} \rangle)\, d\mathbf{x}\ .$$

It satisfies the following standard properties, which follow by routine calculations.

Lemma 2.7 (Multiplicativity). *For any function f as above, $\widehat{f^k} = \widehat{f}^{\,k}$ (where the exponent notation is as defined in Equation (2.1)).*

Lemma 2.8 (Time-scaling property). *For any function f as above and real $s \neq 0$, $\widehat{f_s}(\mathbf{w}) = s^n \cdot \widehat{f}_{1/s}(\mathbf{w})$.*

Lemma 2.9 (Time-shift property). *For any function f as above and $\mathbf{c} \in \mathbb{R}^n$, let $g(\mathbf{x}) = f(\mathbf{x} - \mathbf{c})$. Then $\widehat{g}(\mathbf{w}) = \widehat{f}(\mathbf{w}) \cdot \exp(-2\pi i \langle \mathbf{w}, \mathbf{c} \rangle)$.*

We say that f is *nice* if it satisfies conditions that are sufficient for the following formula to hold, e.g., those given in [Ser73, pages 106–107]. All of the specific functions f we use in this work are easily seen to be nice.

Lemma 2.10 (Poisson Summation Formula (PSF)). *For any lattice $\mathcal{L}$ and nice function f,*

$$f(\mathcal{L}) = \det(\mathcal{L}^*) \cdot \widehat{f}(\mathcal{L}^*)\ .$$

We will use a more general version of the PSF for lattice *cosets*.

Lemma 2.11 (Generalized PSF). *For any lattice $\mathcal{L} \subset \mathbb{R}^n$, nice function f, and $\mathbf{y} \in \mathbb{R}^n$,*

$$f(\mathbf{y} + \mathcal{L}) = \det(\mathcal{L}^*) \cdot \sum_{\mathbf{w} \in \mathcal{L}^*} \widehat{f}(\mathbf{w}) \cdot \exp(2\pi i \langle \mathbf{w}, \mathbf{y} \rangle)\ .$$

Proof. Define the function $g(\mathbf{x}) := f(\mathbf{x} + \mathbf{y})$. By Lemmas 2.9 and 2.10,

$$\begin{aligned} f(\mathbf{y} + \mathcal{L}) &= g(\mathcal{L}) \\ &= \det(\mathcal{L}^*) \cdot \widehat{g}(\mathcal{L}^*) \\ &= \det(\mathcal{L}^*) \cdot \sum_{\mathbf{w} \in \mathcal{L}^*} \widehat{g}(\mathbf{w}) \\ &= \det(\mathcal{L}^*) \cdot \sum_{\mathbf{w} \in \mathcal{L}^*} \widehat{f}(\mathbf{w}) \cdot \exp(2\pi i \langle \mathbf{w}, \mathbf{y} \rangle) \ . \end{aligned}$$

□

2.4 Lattice Roughness

Continuing from Section 2.3, for the rest of this work we require the following properties of f.

Assumption 2.12. The function f has range $[0, 1]$ and is nice, and $\widehat{f}$ is *non-negative real* with $\widehat{f}(0) > 0$.

Because f is real, its Fourier transform is conjugate symmetric, i.e., $\widehat{f}(-w) = \widehat{f}(w)^*$ for all w, where the star denotes complex conjugation. Since $\widehat{f}$ is also real, this implies that it is symmetric, i.e., $\widehat{f}(-w) = \widehat{f}(w)$. Finally, note that if f satisfies this assumption, then so does its multiplicative extension f^k.

We now define an important Fourier-analytic quantity that plays an important role in our analysis. We adopt the name "roughness" because it is the functional inverse of the "smoothing parameter" from [MR04], which is the smallest s that makes the function $f_s(\mathbf{y} + \mathcal{L})$ sufficiently "smooth" as a function of $\mathbf{y}$.

Definition 2.13. For a function f, lattice $\mathcal{L} \subset \mathbb{R}^n$, and real $s > 0$, the *roughness* is defined as

$$\varepsilon_{\mathcal{L},s} := \frac{\widehat{f}_s(\mathcal{L}^* \setminus \{\mathbf{0}\})}{\widehat{f}_s(\mathbf{0})} = \frac{\widehat{f}_s(\mathcal{L}^*)}{\widehat{f}_s(\mathbf{0})} - 1 \geq 0.$$

More generally, for a (linear) subspace H of $\mathbb{R}^n$, the *H-roughness* is defined as

$$\varepsilon_{\mathcal{L},s}(H) := \frac{\widehat{f}_s(\mathcal{L}^* \setminus H^\perp)}{\widehat{f}_s(\mathcal{L}^* \cap H^\perp)} = \frac{\widehat{f}_s(\mathcal{L}^*)}{\widehat{f}_s(\mathcal{L}^* \cap H^\perp)} - 1 \leq \varepsilon_{\mathcal{L},s}(\mathbb{R}^n) = \varepsilon_{\mathcal{L},s}.$$

(Both inequalities follow from the non-negativity of $\widehat{f}_s$.)

Lemma 2.14 (adapted from [MR04, Lemmas 2.9 and 4.1]). *For any lattice $\mathcal{L} \subset \mathbb{R}^n$, real $s > 0$, and subspace H of $\mathbb{R}^n$ defining roughness $\varepsilon := \varepsilon_{\mathcal{L},s}(H)$, and any $\mathbf{y} \in H$,*

$$f_s(\mathbf{y} + \mathcal{L}) \in \det(\mathcal{L}^*) \cdot \widehat{f}_s(\mathcal{L}^* \cap H^\perp) \cdot [1 - \varepsilon, 1 + \varepsilon] \ ,$$

with equality against the upper bound when $\mathbf{y} = \mathbf{0}$. In particular, $f_s(\mathbf{y} + \mathcal{L}) \in f_s(\mathcal{L}) \cdot [\frac{1-\varepsilon}{1+\varepsilon}, 1]$.

Proof. By the generalized PSF (Lemma 2.11),

$$\begin{aligned} f_s(\mathbf{y} + \mathcal{L}) &= \det(\mathcal{L}^*) \cdot \sum_{\mathbf{w} \in \mathcal{L}^*} \widehat{f}_s(\mathbf{w}) \cdot \exp(2\pi i \langle \mathbf{w}, \mathbf{y} \rangle) \\ &= \det(\mathcal{L}^*) \cdot \Big(\widehat{f}_s(\mathcal{L}^* \cap H^\perp) + \sum_{\mathbf{w} \in \mathcal{L}^* \setminus H^\perp} \widehat{f}_s(\mathbf{w}) \cdot \exp(2\pi i \langle \mathbf{w}, \mathbf{y} \rangle \Big) \\ &= \det(\mathcal{L}^*) \cdot \Big(\widehat{f}_s(\mathcal{L}^* \cap H^\perp) + \sum_{\mathbf{w} \in \mathcal{L}^* \setminus H^\perp} \widehat{f}_s(\mathbf{w}) \cdot \cos(2\pi \langle \mathbf{w}, \mathbf{y} \rangle \Big) \ . \end{aligned}$$

The last equation follows from the symmetry of $\widehat{f}_s$ and by pairing each (non-zero) element of $\mathcal{L}^* \setminus H^\perp$ with its negation, which cancels out the imaginary part of $\exp(2\pi i\langle \mathbf{w}, \mathbf{y}\rangle)$.

Now observe that $\widehat{f}_s(\mathbf{w}) \cdot \cos(2\pi\langle \mathbf{w}, \mathbf{y}\rangle) \in [-\widehat{f}_s(\mathbf{w}), \widehat{f}_s(\mathbf{w})]$, with equality against the upper bound for $\mathbf{y} = \mathbf{0}$, because $\widehat{f}_s$ is non-negative. The claim then follows by the definition of roughness $\varepsilon_{\mathcal{L},s}(H)$. □

3 List-Decoding Reed–Solomon Codes

3.1 Soft-Decision Decoding

To list-decode Reed–Solomon codes under various norms and probabilistic channel models, we use the "weighted," or *soft-decision*, list decoder of Guruswami and Sudan (hereafter GS) [GS98], as elaborated upon in Guruswami's thesis [Gur01, Section 6.2.10] and the work of Koetter and Vardy [KV03]. A soft-decision decoder takes a "weight vector" as input, and outputs a set of codewords.

Definition 3.1. A *weight vector* for a length-n code over $\mathbb{F}_q$ is some $W := (W_1, \ldots, W_n) \in [0,1]^{qn}$ where each block $W_i \in [0,1]^q$ is indexed by $\mathbb{F}_q$; equivalently, each block is a function $W_i \colon \mathbb{F}_q \to [0,1]$.

Conceptually, each block W_i of a weight vector may be thought of as specifying a (posterior) probability distribution Π_i over $\mathbb{F}_q$, where $\Pi_i(x)$ is proportional to the probability that the ith transmitted symbol was $x \in \mathbb{F}_q$, given what was received from the channel (which need not be an element of $\mathbb{F}_q$). At a formal level, this interpretation makes sense only when the channel is *probabilistic* (for average-case decoding), but it still serves as useful intuition when the channel is *adversarial* (for worst-case decoding). We consider both types of channels in our results below.

For $c \in \mathbb{F}_q$, define $[c] \in [0,1]^q$ to be the binary indicator vector indexed by $\mathbb{F}_q$ that has a 1 in coordinate c and 0s elsewhere. Similarly, for any vector $\mathbf{c} = (c_1, \ldots, c_n) \in \mathbb{F}_q^n$, define the weight vector $[\mathbf{c}] := ([c_1], \ldots, [c_n]) \in [0,1]^{qn}$. Observe that its Euclidean norm is $\|[\mathbf{c}]\| = \sqrt{n}$.

Definition 3.2. The *correlation* between a weight vector $W \in [0,1]^{qn}$ and a word $\mathbf{c} \in \mathbb{F}_q^n$ is defined as their length-normalized inner product (or the cosine of the angle between them):

$$\operatorname{corr}(W, \mathbf{c}) := \frac{\langle W, [\mathbf{c}]\rangle}{\|W\| \cdot \sqrt{n}} \,.$$

Theorem 3.3 (adapted from [GS98, Theorem 18] and [Gur01, Theorem 6.21]). *For a prime power q, let $\mathcal{C} \subseteq \mathbb{F}_q^n$ be a Generalized Reed–Solomon code of dimension k and adjusted rate $R^* := (k-1)/n$. There is a deterministic algorithm that, given a weight vector W and a "tolerance" $\tau > 0$, outputs in time $\operatorname{poly}(n, q, 1/(\tau\|W\|))$ the set of all codewords $\mathbf{c} \in \mathcal{C}$ that satisfy*

$$\operatorname{corr}(W, \mathbf{c}) \geq \sqrt{R^*} + \tau \,.$$

We remark that the above theorem is originally stated for *rational* weights, but the supporting argument (from [Gur01, Lemma 6.20]) easily adapts to handle *real*-valued weights that can be lower bounded to any needed precision in polynomial time, as all of ours can be.

3.2 From Received Words to Weight Vectors

Here we describe a general approach for translating a received word to a weight vector. This translation is parameterized by a function that, conceptually, can be viewed as (proportional to) the channel's probability density function, even if the channel is not actually probabilistic.

Let $f\colon \mathbb{R} \to [0,1]$ be a function that satisfies Assumption 2.12, extended multiplicatively to $\mathbb{R}^n$ as in Equation (2.1), and recall that $f_s(x) := f(x/s)$ for any constant $s > 0$. Next let q be a positive integer, and recall that we identify $\mathbb{Z}_q := \mathbb{Z}/q\mathbb{Z}$ with $\mathbb{F}_q$ in the natural way when q is prime. Let the set of possible received values be $\mathbb{R}_q = \mathbb{R}/q\mathbb{Z}$, and for any such value $y \in \mathbb{R}_q$, define the weight function $W_{s,y}\colon \mathbb{Z}_q \to [0,1]$ by

$$W_{s,y}(x) := f_s(y - x + q\mathbb{Z}) \,.$$

Notice that here f_s is applied to a *coset* of $q\mathbb{Z}$, which represents an infinite series; for all our concrete choices, these series converge and so the function $W_{s,y}$ is well defined. This function can also be seen as the vector $W_{s,y} := (W_{s,y}(x))_{x\in\mathbb{Z}_q} \in [0,1]^q$, indexed by $\mathbb{Z}_q$.

In line with the probabilistic conception of weight vectors from Section 3.1 above, the function $W_{s,y}$ can be seen as follows. Suppose that a uniformly random symbol in $\mathbb{Z}_q$ is sent over a channel, which adds (modulo q) noise drawn from a distribution over $\mathbb{R}$ whose probability density function is proportional to f_s. Then the probability that the sent symbol was $x \in \mathbb{Z}_q$, conditioned on receiving y, is proportional to $W_{s,y}(x)$. This is because the coset $y - x \in \mathbb{R}_q$ is the set of all noise values that yield y if x is sent. Note that in the definition of $W_{s,y}$ we do *not* normalize by the total weight $W_{s,y}(\mathbb{Z}_q) = f_s(y + \mathbb{Z})$ (which may vary based on the received value y); this turns out to yield simpler analyses and tighter results in the end.

Definition 3.4. For a function f_s as above and any received vector $\mathbf{y} = (y_1, \ldots, y_n) \in \mathbb{R}_q^n$, define the corresponding weight vector as

$$W_{s,\mathbf{y}} := (W_{s,y_1}, \ldots, W_{s,y_n}) \in [0,1]^{nq} \,.$$

In order to use the soft-decision algorithm (Theorem 3.3) for decoding under an adversarial channel, it suffices to show that we can choose a suitable s so that for any received word $\mathbf{y}$ and any sufficiently close codeword $\mathbf{c}$ (in the norm of interest), the correlation $\mathrm{corr}(W_{s,\mathbf{y}}, \mathbf{c})$ satisfies (3.3). Similarly, for decoding under a probabilistic channel, it suffices to show that with high probability over the channel noise $\mathbf{e}$, the transmitted codeword $\mathbf{c}$ has large enough correlation with the weight vector $W_{s,\mathbf{y}}$ of the received word $\mathbf{y} = \mathbf{c} + \mathbf{e}$ (again, for some suitably chosen s). To this end, in what follows we give a lower bound on $\langle W_{s,\mathbf{y}}, [\mathbf{c}] \rangle$ and an upper bound on $\|W_{s,\mathbf{y}}\|$, in terms of f_s and the difference $\mathbf{y} - \mathbf{c}$ between the received word and the codeword of interest.

3.3 Main Theorem

Here we state and prove the main result of this section. For this we define the two-dimensional integer lattice $\mathcal{L}_q$ that consists of all shifts of the lattice $q\mathbb{Z}^2$ by (z,z) for an integer z, i.e.,

$$\mathcal{L}_q := \bigcup_{x\in\mathbb{Z}_q} (x \oplus x) = \bigcup_{z\in\mathbb{Z}} ((z,z) + q\mathbb{Z}^2) \supset q\mathbb{Z}^2.$$

We have that $\det(\mathcal{L}_q) = q$, and so $\det(\mathcal{L}_q^*) = 1/q$. We sometimes omit the q subscript when it is clear from context or its value is unimportant.

Theorem 3.5. *For any $s > 0$ and $\mathbf{y} \in \mathbb{R}_q^n$ defining $W = W_{s,\mathbf{y}}$, and any $\mathbf{c} \in \mathbb{Z}_q^n$,*

$$\mathrm{corr}(W, \mathbf{c}) \geq \frac{\mathrm{Avg}_{i\in[n]}[f_s(y_i - c_i)]}{\sqrt{f_s(\mathcal{L}_q)}} \geq \frac{f_s(\mathbf{y} - \mathbf{c})^{1/n}}{\sqrt{f_s(\mathcal{L}_q)}} \,.$$

Proof. This follows immediately from the following lower and upper bounds on the numerator and denominator of $\mathrm{corr}(W, \mathbf{c}) = \frac{\langle W, [\mathbf{c}] \rangle / n}{\|W\|/\sqrt{n}}$. For the numerator, by the definitions of W and $[\mathbf{c}]$,

$$\langle W, [\mathbf{c}] \rangle / n = \operatorname*{Avg}_{i \in [n]}[W_{s,y_i}(c_i)] = \operatorname*{Avg}_{i \in [n]}[f_s(y_i - c_i)] \geq f_s(\mathbf{y} - \mathbf{c})^{1/n} ,$$

where the last step follows by the inequality of arithmetic and geometric means, and the non-negativity and multiplicativity of f_s over direct sums of cosets (Lemma 2.6). For the denominator, the upper bound $\|W\|/\sqrt{n} \leq \sqrt{f_s(\mathcal{L}_q)}$ is proved in Lemma 3.7 below. □

Remark 3.6. If certain coordinates of $\mathbf{y}$ are "very far" from the corresponding entries of $\mathbf{c}$, we may get a better lower bound on $\mathrm{corr}(W, \mathbf{c})$ by *restricting* to "good" coordinates. Specifically, for any nonempty $G \subseteq [n]$ of cardinality $g = |G|$, by the non-negativity of f_s,

$$\operatorname*{Avg}_{i \in [n]}[f_s(y_i - c_i)] \geq \frac{g}{n} \cdot \operatorname*{Avg}_{i \in G}[f_s(y_i - c_i)] \geq \frac{g}{n} \cdot f_s(\mathbf{y}_G - \mathbf{c}_G)^{1/g} ,$$

where $\mathbf{x}_G$ is the vector obtained by restricting $\mathbf{x}$ to the coordinates in G.

Lemma 3.7. *Adopting the notation from Theorem 3.5, and letting $\tilde{\varepsilon} = \varepsilon_{\mathcal{L}_q,s}(H)$ where $H = \mathrm{span}(1, 1)$,*

$$\|W\|^2/n \in f_s(\mathcal{L}_q) \cdot \left[\frac{1 - \tilde{\varepsilon}}{1 + \tilde{\varepsilon}}, 1\right] .$$

Proof. By definition of W,

$$\|W\|^2/n = \operatorname*{Avg}_{i \in [n]}\Big[\sum_{x \in \mathbb{Z}_q} f_s(y_i - x)^2\Big] .$$

To bound this, let $y \in \mathbb{R}_q$ be arbitrary. By Lemma 2.6,

$$\begin{aligned}
\sum_{x \in \mathbb{Z}_q} f_s(y - x)^2 &= \sum_{x \in \mathbb{Z}_q} f_s((y - x) \oplus (y - x)) \\
&= \sum_{x \in \mathbb{Z}_q} f_s((y \oplus y) - (x \oplus x)) \\
&= f_s((\overline{y}, \overline{y}) + \mathcal{L}_q) \\
&\in f_s(\mathcal{L}_q) \cdot \left[\frac{1 - \tilde{\varepsilon}}{1 + \tilde{\varepsilon}}, 1\right] ,
\end{aligned}$$

where the last step follows by the latter part of Lemma 2.14 on the lattice $\mathcal{L}_q$ with subspace H, and noting that $(\overline{y}, \overline{y}) \in H$. The claim follows by averaging over $i \in [n]$. □

3.4 Average-Case Decoding

Here we consider list-decoding in the *average case*, where the channel is probabilistic (not worst case) and the goal is to output a list of codewords that includes the transmitted one. We consider channels that add independent, identically distributed random error (drawn from some specified distribution) to each coordinate of the transmitted codeword; this is often known as a *memoryless additive channel*. Specifically, we assume

that the channel's error distribution (for each coordinate) is proportional to f_r for some $r > 0$, i.e., it has probability density function

$$D_r(x) := \frac{f_r(x)}{\widehat{f_r}(0)} .$$

For example, if f_r is a Gaussian function, this is known as the *additive white Gaussian noise* (AWGN) channel model. In some settings one may also consider a *discrete* channel distribution, e.g., over $\mathbb{Z}$, in which case its probability mass function is $D_r(x) := f_r(x)/f_r(\mathbb{Z})$. For any $s > 0$ (which may differ from r), define

$$\mu_{r,s} := \mathop{\mathbb{E}}_{e \leftarrow D_r}[f_s(e)] .$$

In Section 4 we will use the following bound for a specific family of functions f to show that the transmitted codeword is recovered with high probability over the channel error.

Lemma 3.8. *For any $r, s > 0$ and T defining $\gamma := \mu_{r,s} - T \cdot \sqrt{f_s(\mathcal{L}_q)} \geq 0$, and any $\mathbf{c} \in \mathbb{Z}_q^n$,*

$$\Pr_{\mathbf{e} \leftarrow D_r^n}\left[\mathrm{corr}(W_{s,\mathbf{c}+\mathbf{e}}, \mathbf{c}) \leq T\right] < \exp(-2\gamma^2 n) .$$

Proof. The error coordinates e_i are drawn independently from D_r, so by Assumption 2.12 the values $f_s(e_i)$ are independent and identically distributed random variables in $[0, 1]$, with expected value $\mu_{r,s}$. Then by Theorem 3.5 and Hoeffding's inequality (Lemma 2.1),

$$\begin{aligned}\Pr_{\mathbf{e}}[\mathrm{corr}(W_{s,\mathbf{c}+\mathbf{e}}, \mathbf{c}) \leq T] &\leq \Pr_{\mathbf{e}}\left[\mathop{\mathrm{Avg}}_{i \in [n]}[f_s(e_i)] \leq T \cdot \sqrt{f_s(\mathcal{L}_q)}\right] \\ &< \exp(-2\gamma^2 n) .\end{aligned}$$

□

4 General ℓ_p (Semi)Metrics

In this section we define weight vectors via Definition 3.4 using the function $f : \mathbb{R} \to [0, 1]$ defined as

$$\begin{aligned} f(x) = f^{(p)}(x) &:= \exp\left(-(c_p|x|)^p\right) \\ \text{where } c_p &:= 2 \cdot \Gamma(1 + 1/p) , \end{aligned} \tag{4.1}$$

where the gamma function $\Gamma(z) = \int_0^\infty u^{z-1} \exp(-u)\, du$ for $z > 0$, and satisfies $\Gamma(1) = 1$ and $\Gamma(1 + z) = z \cdot \Gamma(z)$. As two important examples, $c_1 = 2$ and $c_2 = \sqrt{\pi}$.

Note that by multiplicativity (Equation (2.1)),

$$f(\mathbf{x}) = \prod_{i=1}^n f(x_i) = \exp\Big(-\sum_{i=1}^n (c_p|x_i|)^p\Big) = \exp\left(-(c_p\|\mathbf{x}\|_p)^p\right) = f(\|\mathbf{x}\|_p) .$$

Regarding the Fourier transform of f, the "normalizing constant" c_p has been defined to make $\widehat{f}(0) = 1$:

$$\widehat{f}(0) = \int_{-\infty}^{\infty} f(x)\, dx = 2\int_0^\infty \exp\left(-(c_p x)^p\right) dx = \frac{2}{p \cdot c_p}\int_0^\infty u^{1/p-1} \exp(-u)\, du = \frac{2 \cdot \Gamma(1/p)}{p \cdot c_p} = 1 ,$$

by the change of variable $u = (c_p x)^p$. It is also known that $\widehat{f}$ is non-negative for $0 < p \leq 2$; this follows immediately from an elegant lemma and proof due to Logan, given in [EOR91, Lemma 5].[6] So, f satisfies Assumption 2.12 for such p. Another immediate consequence of Logan's lemma is that as s grows, $\widehat{f_s}(w)/s$ strictly decreases and approaches zero for every $w \neq 0$.[7]

We will need the following simple lemma.

Lemma 4.1. *For any $s > 0$ and $\mathbf{y} \in \mathbb{R}_q^n$ defining $W = W_{s,\mathbf{y}}$, we have that $\|W\|_2 \geq \sqrt{n}/\exp(c_p^p/(2s)^p)$.*

Proof. Observe that each of the n blocks of W has an entry indexed by some $x \in \mathbb{Z}_q$ for which $|\overline{y_i - x}| \leq 1/2$. This entry's contribution to $\|W\|^2$ is at least $f_s(1/2)^2 = \exp(-2c_p^p/(2s)^p)$. □

4.1 Worst-Case Decoding

We now address list-decoding in the ℓ_p (semi)metric for $0 < p \leq 2$, under worst-case error. Consider decoding distance $d = \delta \cdot n^{1/p}$, where n is the code length, and δ can be seen as the *relative* decoding distance (relative to $n^{1/p}$, which is the most natural normalization factor for ℓ_p). For $s > 0$, relative distance $\delta \geq 0$, and positive integer modulus q, define

$$W_{q,\delta}^{(p)}(s) := \frac{f_s(\delta)}{\sqrt{f_s(\mathcal{L}_q)}} = \frac{\exp\big(-(c_p \cdot \delta/s)^p\big)}{\sqrt{f_s(\mathcal{L}_q)}} \geq 0\,. \tag{4.2}$$

By Theorems 3.3 and 3.5, to decode a GRS code of adjusted rate R^* over a prime field $\mathbb{F}_q$ to within ℓ_p distance $\delta \cdot n^{1/p}$ using the GS algorithm, it suffices to set $s > 0$ so that $W_{q,\delta}^{(p)}(s) > \sqrt{R^*}$. In other words, we can decode under relative distance δ for any R^* less than

$$R_{\mathrm{wc},q}^{*,(p)}(\delta) := \sup_{s>0} W_{q,\delta}^{(p)}(s)^2\,. \tag{4.3}$$

The following theorem makes this formal.

Theorem 4.2. *For any $0 < p \leq 2$, $\delta \geq 0$, and prime q, the GS soft-decision algorithm using weight vector given by $f_s^{(p)}$ for any $s > 0$ list-decodes, up to ℓ_p distance $d = \delta \cdot n^{1/p}$, any GRS code $\mathcal{C} \subseteq \mathbb{F}_q^n$ with adjusted rate $R^* < W_{q,\delta}^{(p)}(s)^2$, in time polynomial in n, q, and $\exp(1/s^p)/(W_{q,\delta}^{(p)}(s) - \sqrt{R^*})$.*[8]

Proof. We invoke the GS algorithm on the weight vector $W = W_{s,\mathbf{y}}$ given by the choice of s and the received word $\mathbf{y}$, and tolerance $\tau = W_{q,\delta}^{(p)}(s) - \sqrt{R^*} > 0$.[9] The running time is polynomial in n, q, and $1/(\tau\|W\|_2) \leq \exp(c_p^p/(2s)^p)/(\tau\sqrt{n})$, by Lemma 4.1.

[6] For $p > 2$, by contrast, $\widehat{f}$ can have negative values, which prevents our framework from supporting ℓ_p metrics for such p.

[7] Let $0 < p \leq 2$ and $h(x) = \exp(-|x|^p)$; since $f(x) = h(c_p x)$, it suffices to prove the claimed property for h_s. Logan shows that $h_s(x)$ can be expressed as a non-negative linear combination of Gaussians, as $h_s(x) = \int_0^\infty \exp(-t(x/s)^2)\,\mathrm{d}\alpha(t)$, where $\alpha(t)$ is bounded and non-decreasing. For every $t > 0$, the Fourier transform (with respect to x) of $\exp(-t(x/s)^2)/s$ is $\exp(-(\pi s w)^2/t) \cdot \sqrt{\pi/t}$. As s increases, this strictly decreases and approaches zero for any $w \neq 0$. Since $\alpha(t)$ is bounded and non-decreasing, the same goes for $\widehat{h_s}(w)/s$.

[8] We remark that in many cases, the bound on the polynomial running time can be improved using a better lower bound for $\|W\|_2$, such as the one given by Lemma 3.7.

[9] To be more precise, we can invoke GS on any approximation of τ in $[\tau/2, \tau]$, say. This can be computed by approximating $f_s(\mathcal{L}_q)$ from above to the needed precision, by enumerating sufficiently many points of $\mathcal{L}_q$ near the origin, and upper-bounding the contribution of the remaining points in the "tails" using, e.g., Lemma 5.3.

Now let $\mathbf{c} \in \mathcal{C}$ be a codeword within distance d of $\mathbf{y}$, i.e., $\|\overline{\mathbf{y}-\mathbf{c}}\|_p \le d$. By Theorem 3.5, Assumption 2.12, and Equation (4.2),

$$\mathrm{corr}(W, \mathbf{c}) \ge \frac{f_s(\mathbf{y}-\mathbf{c})^{1/n}}{\sqrt{f_s(\mathcal{L}_q)}} \ge \frac{f_s(d)^{1/n}}{\sqrt{f_s(\mathcal{L}_q)}} = W^{(p)}_{q,\delta}(s) = \sqrt{R^*} + \tau \ .$$

So, by Theorem 3.3, the output of the GS algorithm includes $\mathbf{c}$, as needed. □

Remark 4.3. Following Remark 3.6, suppose that the received word $\mathbf{y}$ is within relative distance δ of a codeword $\mathbf{c}$ on some subset $G \subseteq [n]$ of $g = |G|$ coordinates, i.e., $\|\overline{\mathbf{y}_G - \mathbf{c}_G}\| \le \delta \cdot g^{1/p}$, and the remaining coordinates of $\mathbf{y}$ may be *arbitrary*. Then $\mathrm{corr}(W_{s,\mathbf{y}}, \mathbf{c}) \ge (g/n) \cdot W^{(p)}_{q,\delta}(s)$. So, we can list-decode all such codewords as long as the adjusted rate $R^* < (g/n)^2 \cdot R^{*,(p)}_{\mathrm{wc},q}(\delta)$.

Remark 4.4. Interestingly, as δ, q/δ, and n grow (and the other parameters remain fixed), the product of the relative distance δ and the adjusted rate R^* for which we can decode approaches the *relative radius of a unit-volume ℓ_p ball*. To see this, first observe that as q/s grows, $f_s(\mathcal{L}_q)$ approaches

$$f_s(\mathcal{L}_q \cap \mathrm{span}(1,1)) = \sum_{z \in \mathbb{Z}} f_s(z,z) = \sum_{z \in \mathbb{Z}} f_{s/2^{1/p}}(z) = f_{s/2^{1/p}}(\mathbb{Z}) = \widehat{f_{s/2^{1/p}}}(\mathbb{Z}) \ ,$$

where the last equality is by the PSF (Lemma 2.10). Then as s grows, the above approaches $s/2^{1/p}$, because $\widehat{f}(0) = 1$ and hence $\widehat{f_{s/2^{1/p}}}(0) = s/2^{1/p}$ by Lemma 2.9, and the other Fourier coefficients approach zero.

So, as s and q/s grow, the bound $W^{(p)}_{q,\delta}(s)^2$ on the adjusted rate approaches $2^{1/p} \cdot \exp\big(-(2^{1/p} c_p \cdot \delta/s)^p\big)/s$. A straightforward calculation using the change of variable $t = (2^{1/p} c_p \cdot \delta/s)^p$ shows that this is maximized for $t = 1/p$ (and hence $s = (2p)^{1/p} c_p \cdot \delta$), so we can decode to within relative ℓ_p distance δ for an adjusted rate R^* approaching

$$\frac{1}{(ep)^{1/p} \cdot c_p \cdot \delta} \ .$$

By comparison, it is known that the volume of an n-dimensional ℓ_p ball of unit relative radius (i.e., radius $n^{1/p}$) has nth root

$$\frac{2 \cdot \Gamma(1+1/p)}{\Gamma(1+n/p)^{1/n}} \cdot n^{1/p} \longrightarrow (ep)^{1/p} \cdot c_p \ ,$$

using Stirling's approximation for the denominator as n grows. So, the relative radius of a unit-volume ℓ_p ball is the reciprocal of this, which is what $R^* \cdot \delta$ approaches.

4.2 Average-Case Decoding

We now consider average-case decoding under a memoryless additive (continuous or discrete) channel whose density function is proportional to a scaling of $f = f^{(p)}$. Specifically, we consider the continuous distribution with probability density function $D_r(x) := f_r(x)/r$, and the discrete distribution over $\mathbb{Z}$ with probability mass function $D_r(x) := f_r(x)/f_r(\mathbb{Z})$. Following Section 3.4, for any $r, s > 0$ define

$$\mu^{(p)}_{r,s} := \mu_{r,s} = \mathop{\mathbb{E}}_{e \leftarrow D_r}[f_s(e)] \ .$$

For these channel distributions we derive suitable bounds on $\mu^{(p)}_{r,s}$, then reach the conclusion via Lemma 3.8 and Theorem 3.3.

Lemma 4.5. *For any $0 < p \le 2$, any $r > 0$ defining a continuous or discrete distribution D_r, and $s > 0$,*

$$\mu_{r,s}^{(p)} \ge \frac{s}{\|(r,s)\|_p} ,$$

with equality in the continuous case and strict inequality in the discrete case.

Proof. First note that for $t := (rs)/(r^p + s^p)^{1/p}$, we have $f_r(x) \cdot f_s(x) = f_t(x)$, since $1/t^p = 1/r^p + 1/s^p$. For the continuous case,

$$\mu_{r,s}^{(p)} = \int_{\mathbb{R}} D_r(e) \cdot f_s(e)\,\mathrm{d}e = \frac{1}{r}\int_{\mathbb{R}} f_t(e)\,\mathrm{d}e = \frac{t}{r} = \frac{s}{\|(r,s)\|_p} .$$

For the discrete case, since $t < r$, we have that $\widehat{f}_t(w)/t > \widehat{f}_r(w)/r$ for all $w \in \mathbb{Z}$, so by Lemma 2.10,

$$\mu_{r,s}^{(p)} = \sum_{e\in\mathbb{Z}} D_r(e) \cdot f_s(e) = \frac{f_t(\mathbb{Z})}{f_r(\mathbb{Z})} = \frac{t}{r} \cdot \frac{\widehat{f}_t(\mathbb{Z})/t}{\widehat{f}_r(\mathbb{Z})/r} > \frac{t}{r} . \qquad \square$$

Now, for any channel parameter $r > 0$ and for $s > 0$, define

$$A_{q,r}^{(p)}(s) := \frac{\mu_{r,s}^{(p)}}{\sqrt{f_s(\mathcal{L}_q)}} \ge \frac{s}{\|(r,s)\|_p \cdot \sqrt{f_s(\mathcal{L}_q)}} , \tag{4.4}$$

where the inequality is by Lemma 4.5. By Theorems 3.3 and 3.5, to decode (with high probability) a GRS code of adjusted rate R^* over a prime field $\mathbb{F}_q$ under a channel with parameter r, it suffices to set $s > 0$ so that $A_{q,r}^{(p)}(s) > \sqrt{R^*}$. In other words, we can decode under channel parameter r for any R^* less than

$$R_{\mathrm{ac},q}^{*,(p)}(r) := \sup_{s>0} A_{q,r}^{(p)}(s)^2 . \tag{4.5}$$

See Theorem 4.6 below for the formal statement.

Theorem 4.6. *Let $0 < p \le 2$, $r > 0$, $\alpha \in (0,1)$, and q be prime. Under a memoryless additive (continuous or discrete) channel with distribution D_r, the GS soft-decision algorithm, using weight vector given by $f_s^{(p)}$ for any $s > 0$, list-decodes any GRS code $\mathcal{C} \subseteq \mathbb{F}_q^n$ with adjusted rate $R^* < A_{q,r}^{(p)}(s)^2$, in time polynomial in n, q, and $\exp(1/s^p)/(A_{q,r}^{(p)}(s) - \sqrt{R^*})$, except with probability less than*

$$\exp\left(-2n \cdot f_s(\mathcal{L}_q) \cdot \alpha^2 \cdot \left(A_{q,r}^{(p)}(s) - \sqrt{R^*}\right)^2\right) .$$

Proof. Throughout the proof let $A(s) := A_{q,r}^{(p)}(s)$. We invoke the GS algorithm on the weight vector $W = W_{s,\mathbf{y}}$ given by the choice of s and the received word $\mathbf{y}$, and tolerance $\tau = T - \sqrt{R^*} > 0$, where

$$\begin{aligned} T &= A(s) - \alpha(A(s) - \sqrt{R^*}) \\ &= \sqrt{R^*} + (1-\alpha)(A(s) - \sqrt{R^*}) \in (\sqrt{R^*}, A(s)) . \end{aligned} \tag{4.6}$$

The running time is polynomial in n, q, and $1/(\tau\|W\|_2) \le \exp(c_p^p/(2s)^p)/(\tau\sqrt{n})$, by Lemma 4.1.

Now suppose that $\mathbf{y} = \mathbf{c} + \mathbf{e}$, where $\mathbf{c} \in \mathcal{C}$ is the transmitted codeword and $\mathbf{e} \leftarrow D_r^n$ is the channel error. To show that the list output by the GS algorithm contains $\mathbf{c}$ with the claimed probability, by Theorem 3.3 it suffices to show that $\mathrm{corr}(W, \mathbf{c}) \geq \sqrt{R^*} + \tau = T$. Following the setup of Lemma 3.8, let

$$\begin{aligned}
\gamma &= \mu_{r,s} - T \cdot \sqrt{f_s(\mathcal{L}_q)} \\
&= \sqrt{f_s(\mathcal{L}_q)} \cdot (A(s) - T) && \text{(Equation (4.4))} \\
&= \sqrt{f_s(\mathcal{L}_q)} \cdot \alpha(A(s) - \sqrt{R^*}) > 0 && \text{(Equation (4.6)).}
\end{aligned}$$

So, by Lemma 3.8, $\Pr_{\mathbf{e}}[\mathrm{corr}(W, \mathbf{c}) < T] < \exp(-2\gamma^2 n)$, which yields the claim. □

Remark 4.7. Theorem 4.6 outperforms Theorem 4.2 (for worst-case decoding) by a factor that approaches $(e/2)^{1/p}$ in the adjusted rate R^* it can handle, as r and q/r grow. Specifically, consider a channel with parameter r. A calculation reveals that its relative error (in ℓ_p, relative to $n^{1/p}$) is tightly concentrated around $\delta = r/(p^{1/p} \cdot c_p)$, so following the analysis in Remark 4.4, Theorem 4.2 applies for R^* that approaches $1/(r \cdot e^{1/p})$. By comparison, Theorem 4.6 applies for R^* that approaches $1/(r \cdot 2^{1/p})$.

Remark 4.8. Following Remark 3.6, and similarly to Remark 4.3, suppose that there exists a subset $G \subseteq [n]$ of $g = |G|$ "good" coordinates for which the channel generates the received word $\mathbf{y}$ from the transmitted codeword $\mathbf{c}$ by adding independent noise from distribution D_r on those coordinates, and sets the remaining coordinates *arbitrarily*. Then our lower bound on $\mathrm{corr}(W_{s,\mathbf{y}}, \mathbf{c})$ from Theorem 3.5 involves an extra g/n factor, and an average over just the coordinates in G. So, we can correctly list-decode for any adjusted rate $R^* < (g/n)^2 \cdot A_{q,r}^{(p)}(s)^2$. More precisely, the statement and proof of Theorem 4.6 hold with every occurrence of $A_{q,r}^{(p)}(s)$ having an additional g/n factor, and with an extra n/g factor inside the exp expression for the failure probability.

5 The ℓ_2 Metric and Gaussian Error

In the remainder of the paper we instantiate our general list-decoding results for ℓ_p (semi)metrics (Theorems 4.2 and 4.6) for specific metrics of interest and memoryless additive channels. In this section, we consider the ℓ_2 metric and Gaussian channels.

We specialize Equation (4.1) to $p = 2$, i.e., the Gaussian function

$$f(x) := f^{(2)}(x) = \exp(-\pi x^2) \ .$$

By a straightforward calculation it can be seen that this function is its own Fourier transform: $\widehat{f} = f$. Note that $\widehat{f_s} = s \cdot f_{1/s}$ by the time-scaling property of the Fourier transform (Lemma 2.8). Finally, recalling that $f(\mathbf{x}) = f(\|\mathbf{x}\|_2)$, we get that f is invariant under rotations.

5.1 Bounds

In this subsection we derive fairly tight bounds on the factor $f_s(\mathcal{L}_q)$ that appears in the quantities that govern the adjusted rates under which we can decode in the worst and average cases (Equations (4.2) and (4.4), respectively). For this purpose we need to define a suitable "fudge factor." For $r \geq r_0 := \sqrt{\ln(4)/\pi} \approx 0.66428$, define

$$E(r) := 1 - 2\exp(-\pi r^2/2) \in [0, 1) \ .$$

Notice that $E(r)$ is positive for $r > r_0$, is strictly increasing, and rapidly approaches 1 as r increases. Next, for real s, q such that $s \in [r_0, q/r_0]$, define

$$E_q(s) := \sqrt{E(q/s) \cdot E(s)} \in [0, 1) \ .$$

Similarly, $E_q(s)$ is positive for $s \in (r_0, q/r_0)$, and rapidly approaches 1 as both $s, q/s$ increase.

Lemma 5.1. *For any real s and positive integer q such that $s \in (r_0, q/r_0)$,*

$$\frac{1}{f_s(\mathcal{L}_q)} > \frac{\sqrt{2}}{s} \cdot E_q(s)^2 \ .$$

Proof. This follows directly from Lemmas 5.2 and 5.4 below. Specifically, let $\varepsilon' = \varepsilon_{\mathbb{Z}, q/(s\sqrt{2})}$ and $\tilde{\varepsilon} = \varepsilon_{\mathcal{L}_q, s}(H)$. By Lemma 5.4 (applied twice, with $r = q/s$ and $r = s$, which are both greater than r_0),

$$\frac{1}{(1+\varepsilon')(1+\tilde{\varepsilon})} > E_q(s)^2 \ .$$

The result then follows by Lemma 5.2. □

Lemma 5.2. *For any real $s > 0$ and positive integer q, let $\varepsilon' = \varepsilon_{\mathbb{Z}, q/(s\sqrt{2})}$ and $\tilde{\varepsilon} = \varepsilon_{\mathcal{L}_q, s}(H)$ where $H = \mathrm{span}(1, 1)$. Then*

$$f_s(\mathcal{L}_q) = \frac{s}{\sqrt{2}} \cdot (1+\varepsilon') \cdot (1+\tilde{\varepsilon}) \ .$$

Proof. Recall that $\mathcal{L} = \mathcal{L}_q$ has determinant $\det(\mathcal{L}) = q$, hence its dual has determinant $\det(\mathcal{L}^*) = 1/q$. A basis for $\mathcal{L}$ consists of the vectors $(1, 1)$ and $(q, 0)$, and its dual basis consists of the vectors $(0, 1)$ and $(1, -1)/q$.

Since $H^\perp = \mathrm{span}(1, -1)$, we have that $\mathcal{L}^* \cap H^\perp$ consists merely of all the integer multiples of the dual basis vector $(1, -1)/q$, which has Euclidean norm $\sqrt{2}/q$. Therefore, $\mathcal{L}^* \cap H^\perp$ is a rotation of $(\sqrt{2}/q)\mathbb{Z}$. So,

$$\begin{aligned}
f_s(\mathcal{L}) &= (1/q) \cdot \widehat{f_s^2}(\mathcal{L}^* \cap H^\perp) \cdot (1+\tilde{\varepsilon}) && \text{(Lemma 2.14 and definition of } \tilde{\varepsilon}) \\
&= (s^2/q) \cdot f_{1/s}^2(\mathcal{L}^* \cap H^\perp) \cdot (1+\tilde{\varepsilon}) && \text{(Lemma 2.8)} \\
&= (s^2/q) \cdot f_{q/(s\sqrt{2})}(\mathbb{Z}) \cdot (1+\tilde{\varepsilon}) && \text{(rotational invariance of } f^2\text{, rescaling)} \\
&= (s/\sqrt{2}) \cdot (1+\varepsilon') \cdot (1+\tilde{\varepsilon}) && \text{(Lemma 2.14 and definition of } \varepsilon'). \qquad \square
\end{aligned}$$

Next we bound the roughness quantities $\varepsilon', \tilde{\varepsilon}$ from Lemmas 5.1 and 5.2, using the following classic tail inequality.

Lemma 5.3 (adapted from [Ban95, Lemma 2.4]). *For any lattice $\mathcal{L}$, unit vector $\mathbf{u}$, and $s, t > 0$, let $T_{\mathbf{u},t} = \{\mathbf{x} : |\langle \mathbf{x}, \mathbf{u} \rangle| \geq t\}$. Then*

$$f_s(\mathcal{L} \cap T_{\mathbf{u},t}) < 2\exp(-\pi t^2/s^2) \cdot f_s(\mathcal{L}) \ .$$

Lemma 5.4. *Let $r > r_0$ and $H = \mathrm{span}(1, 1)$. Then*

$$\frac{1}{1+\varepsilon_{\mathbb{Z}, r/\sqrt{2}}} \ , \ \frac{1}{1+\varepsilon_{\mathcal{L}_q, r}(H)} > E(r) = 1 - 2\exp(-\pi r^2/2) \ .$$

Proof. We first bound $\varepsilon_{\mathbb{Z},r/\sqrt{2}}$. Let $s = r/\sqrt{2} > \sqrt{\ln(2)/\pi}$. By the definition of roughness (Definition 2.13) and the facts that $\mathbb{Z}^* = \mathbb{Z}$, $\widehat{f}_s = s \cdot f_{1/s}$ (Lemma 2.8), and $f_{1/s}(0) = 1$,

$$1 + \varepsilon_{\mathbb{Z},s} = \widehat{f}_s(\mathbb{Z})/\widehat{f}_s(0) = f_{1/s}(\mathbb{Z}) \; .$$

Now, by Lemma 5.3 and rearranging (where the denominator is positive due to the lower bound on s),

$$f_{1/s}(\mathbb{Z}) = f_{1/s}(0) + f_{1/s}(\mathbb{Z} \cap T_{1,1}) < 1 + 2\exp(-\pi s^2) \cdot f_{1/s}(\mathbb{Z}) \le \frac{1}{1 - 2\exp(-\pi s^2)} \; ,$$

which yields the claim.

For $\varepsilon_{\mathcal{L},r}(H)$ where $\mathcal{L} = \mathcal{L}_q$, again by Definition 2.13 and Lemma 2.8,

$$1 + \varepsilon_{\mathcal{L},r}(H) = \frac{\widehat{f}_r(\mathcal{L}^*)}{\widehat{f}_r(\mathcal{L}^* \cap H^\perp)} = \frac{f_{1/r}(\mathcal{L}^*)}{f_{1/r}(\mathcal{L}^* \cap H^\perp)} \; .$$

Because $H^\perp = \mathrm{span}(1,-1)$ and the vectors $(0,1), (1,-1)/q$ form a basis of $\mathcal{L}^*$, every point in $\mathcal{L}^*$ lies one of the lines (i.e., affine subspaces) $L_k = k \cdot (1,0) + H^\perp$ for some $k \in \mathbb{Z}$. The unit vector $\mathbf{u} = (1,1)/\sqrt{2}$ is orthogonal to $H^\perp$, so for any $\mathbf{x} \in L_k$, we have that $\langle \mathbf{x}, \mathbf{u} \rangle = k/\sqrt{2}$, and hence $|\langle \mathbf{x}, \mathbf{u} \rangle| \ge 1/\sqrt{2}$ if $k \ne 0$. Therefore, $\mathcal{L}^*$ can be partitioned as the disjoint union

$$\mathcal{L}^* = (\mathcal{L}^* \cap H) \cup (\mathcal{L}^* \cap T_{\mathbf{u},1/\sqrt{2}}) \; .$$

So, by Lemma 5.3 and rearranging (where again the denominator is positive due to the bound on r),

$$\begin{aligned} f_{1/r}(\mathcal{L}^*) &= f_{1/r}(\mathcal{L}^* \cap H) + f_{1/r}(\mathcal{L}^* \cap T_{\mathbf{u},1/\sqrt{2}}) \\ &< f_{1/r}(\mathcal{L}^* \cap H) + 2\exp(-\pi r^2/2) \cdot f_{1/r}(\mathcal{L}^*) \\ &\le \frac{f_{1/r}(\mathcal{L}^* \cap H)}{1 - 2\exp(-\pi r^2/2)} . \end{aligned}$$

The result follows by dividing $f_{1/r}(\mathcal{L}^* \cap H)$ by both sides. □

5.2 Worst-Case Decoding

We now address list-decoding in the ℓ_2 metric, under worst-case error of bounded distance, by specializing the material of Section 4.1 to $p = 2$ and using our bounds on $f_s(\mathcal{L}_q)$ from Section 5.1. So, we consider decoding distance $d = \delta\sqrt{n}$, where n is the code length and δ is the relative decoding distance. Then by Equations (4.2) and (4.3), we can list-decode for any R^* less than

$$R^{*,(2)}_{\mathrm{wc},q}(\delta) = \sup_{s>0} W^{(2)}_{q,\delta}(s)^2 > \sup_{s \in (r_0, q/r_0)} \frac{\sqrt{2} \cdot \exp(-2\pi\delta^2/s^2)}{s} \cdot E_q(s)^2 \; ,$$

where the inequality follows by Lemma 5.1.

Corollary 5.5 below is obtained by nearly maximizing the right-hand side of (5.2). More specifically, a standard calculation shows that taking $s = \delta\sqrt{4\pi}$ maximizes the "main term" $\sqrt{2} \cdot \exp(-2\pi\delta^2/s^2)/s$, to have value $1/(\delta\sqrt{2\pi e})$. For moderate or larger values of δ (and hence s), this very nearly maximizes the entire expression, because $E_q(s) \ge E(s)$ since $q/s \ge s$, and $E(s)$ rapidly approaches 1 as s grows. For example, $E(s)^2 \ge 1 - 10^{-8}$ for $\delta \ge 1$. So, as δ grows, the R^* for which we can list-decode rapidly approaches $1/(\delta\sqrt{2\pi e})$.

Corollary 5.5. *For any $\delta > \sqrt{\ln(4)}/(2\pi) \approx 0.1874$ and prime $q \geq 4\pi\delta^2$, the GS algorithm using weight vector given by f_s for $s = \delta\sqrt{4\pi}$ list-decodes, up to ℓ_2 distance $\delta\sqrt{n}$ in time $\mathrm{poly}(n, q, 1/(\sqrt{\widetilde{R}^{*,(2)}_{wc,q}(\delta)} - \sqrt{R^*}))$, any GRS code with adjusted rate*

$$R^* < \widetilde{R}^{*,(2)}_{wc,q}(\delta) := \frac{1}{\delta\sqrt{2\pi e}} \cdot E_q(\delta\sqrt{4\pi})^2 \, .$$

Proof. For $s = \delta\sqrt{4\pi}$, the lower bounds on δ and q imply that $s = \delta\sqrt{4\pi} \in (r_0, q/r_0)$. Then by hypothesis and Lemma 5.1 and Equation (4.2),

$$R^* < \widetilde{R}^{*,(2)}_{\mathrm{wc},q}(\delta) = \frac{1}{\delta\sqrt{2\pi e}} \cdot E_q(\delta\sqrt{4\pi})^2 < \frac{\exp(-2\pi\delta^2/s^2)}{f_s(\mathcal{L}_q)} = W^{(2)}_{q,\delta}(s)^2 \, .$$

The claim then follows directly by Theorem 4.2. □

Comparison to [MP22]. The previous best result for list-decoding (Generalized) Reed–Solomon codes in the ℓ_2 metric was given by Mook and Peikert [MP22].[10]

Proposition 5.6 ([MP22, Theorem 3.4]). *For any GRS code $\mathcal{C} \subseteq \mathbb{F}_q^n$ with any adjusted rate $R^* < 1$ and any $\varepsilon > 0$, there is a $\mathrm{poly}(n, q, 1/\varepsilon)$-time algorithm that list-decodes $\mathcal{C}$ up to ℓ_2 distance $d = \sqrt{n(1-R^*)(1-\varepsilon)/2}$.*

Equivalently, for a relative decoding distance $\delta = d/\sqrt{n} > 0$, the result from [MP22] works for adjusted rates R^* approaching $1 - 2\delta^2$, so it applies only for

$$\delta \leq \sqrt{(1-R^*)/2} \leq 1/\sqrt{2} \, .$$

By contrast, our Theorem 4.2 works for *any* (arbitrarily large) $\delta > 0$ (and Corollary 5.5 gives a simpler and more explicit rate bound for any $\delta > 0.1875$). Moreover, for those δ for which both Theorem 4.2 and Proposition 5.6 apply, our result works for a larger R^* as long as $R^{*,(2)}_{\mathrm{wc},q}(\delta) > 1 - 2\delta^2$ (see (4.3)). For typical (moderate or larger) q, this holds for all $\delta \gtrapprox 0.51797$, which corresponds to $R^* \lessapprox 0.46342$. (For tiny $\delta \approx 0$, Theorem 4.2 works for $R^* \approx 0.93700$, whereas [MP22] works for $R^* \approx 1$, so the latter is better for very small distances.)

We also point out that [MP22] proves that for any $\delta \leq 1/2$, which corresponds to $R^* \geq 1/2$, its (very simple) choice of weight vector gives an *optimal* tradeoff between δ and R^* *for the GS/KV soft-decision algorithm and analysis*. However, the optimality argument breaks down for $\delta > 1/2$ (equivalently, for $R^* < 1/2$). And indeed, as we have just seen, we obtain a better distance-rate tradeoff than [MP22] for *almost all* such δ. This highlights the interesting question of determining an optimal choice of weights for the GS soft-decision algorithm for $\delta > 1/2$ (especially at the low end of this range).

5.3 Unique Decoding for a Subclass of GRS Codes

For a certain natural subclass of GRS codes, and certain rates and decoding distances covered by our list-decoding algorithm, decoding is in fact *unique* (i.e., the list size is at most one). We show this by giving a lower bound on the ℓ_2 minimum distance of such codes, and then observing that our list-decoding algorithm can decode to beyond half this distance for all small enough rates.

[10] By a standard reduction, the result from [MP22] also applies to GRS codes, not just RS codes as was originally stated.

Lemma 5.7 (adapted from [RS94, Theorem 4]). *Any prime-field GRS code* $\mathrm{GRS}_{q,k}(\boldsymbol{\alpha}, \boldsymbol{\alpha}) \subseteq \mathbb{F}_q^n$ *(whose twist factors* $\mathbf{t}$ *equal the nonzero evaluation points* $\boldsymbol{\alpha}$*) of rate* $R = k/n$ *has* squared ℓ_2 *minimum distance at least*

$$\frac{(n+1)^2 - k^2}{12k^2} \cdot (n+1) > \frac{1-R^2}{12R^2} \cdot n \,.$$

Proof. Any codeword of $\mathrm{GRS}_{q,k}(\boldsymbol{\alpha}, \boldsymbol{\alpha})$ consists of the evaluations at $\boldsymbol{\alpha}$ of a polynomial of degree at most k whose constant term is zero. Consider any non-zero codeword $\mathbf{c} \in \mathrm{GRS}_{q,k}(\boldsymbol{\alpha}, \boldsymbol{\alpha})$ and let $u(x) \in \mathbb{F}_q[x]$ be its defining polynomial. Since $u(x)$ is nonzero with degree at most k, it evaluates to any particular element of $\mathbb{F}_q$ at no more than k evaluation points. Hence, any non-zero element of $\mathbb{F}_q$ appears with multiplicity at most k in $\mathbf{c}$, and zero appears with multiplicity at most $k-1$, because $u(0) = 0$ and zero is not an evaluation point. So, $\mathbf{c}$ has at most $k-1$ zeros, and at most k each of $\pm 1, \pm 2, \ldots, \pm \ell$, where $\ell := \lfloor (n+1-k)/(2k) \rfloor$; the remaining $r := n + 1 - k(2\ell+1)$ or more coordinates all have magnitudes greater than ℓ. Thus,

$$\begin{aligned} \|\mathbf{c}\|_2^2 &\geq 2k \cdot \sum_{i=1}^{\ell} i^2 + r(\ell+1)^2 \\ &= (\ell+1) \cdot (k \cdot \ell(2\ell+1)/3 + r(\ell+1)) \,. \end{aligned}$$

Now define $\beta := (n+1-k)/(2k) = \ell + \gamma$, whose fractional part is $\gamma := r/(2k) \in [0,1)$, so $r = 2k \cdot \gamma$. Then the above bound is

$$k(\beta - \gamma + 1) \cdot \big((\beta-\gamma)(2(\beta-\gamma)+1)/3 + 2\gamma(\beta-\gamma+1)\big) \geq k \cdot \beta(\beta+1)(2\beta+1)/3 \,,$$

where the inequality follows from the fact that the minimum over $\gamma \in [0,1)$ is obtained at $\gamma = 0$. (This can be seen by the closed interval method on $[0,1]$, i.e., differentiating with respect to γ, and evaluating at the critical and boundary points.) Substituting $\beta(\beta+1) = ((n+1)^2 - k^2)/(4k^2)$ and $k \cdot (2\beta+1) = n+1$, the claim follows. □

Lemma 5.7 gives a relationship between the code rate R and (a lower bound on) half the ℓ_2 minimum distance, for which decoding to that distance yields a unique solution. By taking the functional inverse of half this minimum-distance bound, we see that decoding to relative distance δ yields a unique solution as long as

$$R < R^{(2)}_{\text{uniq}}(\delta) := \frac{1}{\sqrt{48\delta^2 + 1}} \,,$$

which approaches $1/(4\sqrt{3}\delta)$ as δ grows. This curve is shown in Figure 1. Observe that for any δ for which our list-decoding algorithm outperforms the one of [MP22], we have that $R^{*,(2)}_{\text{wc}}(\delta) > R^{(2)}_{\text{uniq}}(\delta)$. In other words, we can efficiently list decode to relative distance δ for all rates up to $R^{(2)}_{\text{uniq}}(\delta)$ (and beyond), thus yielding a *unique* decoder for these parameters. Alternatively, as the rate R approaches zero, we can efficiently list decode to a multiple of the unique-decoding distance bound that approaches $4\sqrt{3}/\sqrt{2\pi e} \approx 1.6764$.

5.4 Average-Case Decoding

We now consider average-case decoding under a memoryless additive (continuous or discrete) Gaussian channel, by specializing the material of Section 4.2 to $p = 2$ and using our bounds on $f_s(\mathcal{L}_q)$ from Section 5.1.

Consider a Gaussian channel of parameter $r > 0$. Then by Equations (4.4) and (4.5), we can list-decode for any R^* less than

$$R^{*,(2)}_{\mathrm{ac},q}(r) = \sup_{s>0} A^{(2)}_{q,r}(s)^2 > \sup_{s\in(r_0,q/r_0)} \frac{s\sqrt{2}}{r^2+s^2} \cdot E_q(s)^2 \,,$$

where the inequality is by Lemma 5.1.

Corollary 5.8 below is obtained by nearly maximizing the right-hand side of (5.4). More specifically, setting $s = r$ maximizes the "main term" $s\sqrt{2}/(r^2+s^2)$, to have value $1/(r\sqrt{2})$. As above, for moderate or larger values of r (and hence s), this very nearly maximizes the entire expression, because $E_q(s)$ rapidly approaches 1 as s grows.[11] So, as r grows, the rate R^* for which we can list-decode rapidly approaches $1/(r\sqrt{2})$.

Corollary 5.8. *For any $r \in (r_0, q/r_0)$, $\alpha \in (0,1)$, and prime q, the GS algorithm using weight vector given by f_r list-decodes, in time $\mathrm{poly}(n, q, 1/(\sqrt{\widetilde{R}^{*,(2)}_{ac,q}(r)} - \sqrt{R^*}))$, any GRS code with adjusted rate*

$$R^* < \widetilde{R}^{*,(2)}_{ac,q}(r) := \frac{1}{r\sqrt{2}} \cdot E_q(r)^2 \,,$$

except with probability less than $\exp\left(-\sqrt{2}n \cdot \alpha^2 \cdot r \cdot \left(\sqrt{\widetilde{R}^{*,(2)}_{ac,q}(r)} - \sqrt{R^*}\right)^2\right)$.

Proof. By hypothesis, Lemmas 5.1 and 4.5 and Equation (4.4),

$$R^* < \frac{1}{r\sqrt{2}} \cdot E_q(r)^2 < \frac{\mu^2_{r,r}}{f_r(\mathcal{L}_q)} = A^{(2)}_{q,r}(r)^2 \,.$$

The claim then follows directly by Theorem 4.6, and the fact that $f_r(\mathcal{L}_q) > r/\sqrt{2}$ by Lemma 5.2. □

6 The ℓ_1 Metric and Laplacian Error

In this section, we consider the ℓ_1 metric and Laplacian channels. We specialize Equation (4.1) to $p = 1$, i.e., the Laplacian function

$$f(x) := f^{(1)}(x) = \exp(-2|x|) \,.$$

(The Fourier transform of this function is given by $\widehat{f}(w) = 1/(1+(\pi w)^2)$, but we will not use this; as already noted earlier, $f^{(1)}$ satisfies Assumption 2.12.)

Throughout this section we use the hyperbolic tangent function

$$\tanh(x) := \frac{e^x - e^{-x}}{e^x + e^{-x}} = \frac{1 - e^{-2x}}{1 + e^{-2x}} = \frac{e^{2x} - 1}{e^{2x} + 1} < 1$$

and its reciprocal $\coth(x) = 1/\tanh(x) > 1$. Observe that $\tanh(x)$ approaches 1 as x grows; it also satisfies $\tanh(x) < x$ for all $x > 0$, and approaches x as x approaches zero.[12]

[11] By contrast, $E_q(s) \ll 1$ for values of s very close to r_0, in which case the bound is maximized by taking s somewhat larger than r.

[12] Both facts can be seen from the Taylor series $\tanh(x) = x - x^3/3 + \cdots$, valid for $|x| < \pi/2$.

6.1 Bounds

In this subsection, we analyze the exact value of $f_s(\mathcal{L}_q)$ and derive an asymptotic bound. This appears in the quantities that govern the adjusted rates under which we can decode in the worst and average cases (Equations (4.2) and (4.4), respectively). For this purpose, we define a suitable "fudge factor". For any real $x > 0$, define

$$E(x) := \left(\coth(x) + \frac{4x \cdot e^{2x}}{(e^{2x}-1)^2}\right)^{-1} \in (0,1)\,, \tag{6.1}$$

where the upper bound comes from the fact that $\coth(x) > 1$. Note that, as x grows, the first term in the sum rapidly approaches one, and the second term rapidly approaches zero. More precisely, a brief calculation reveals that

$$E(x) = 1 - O(x \cdot e^{-2x})\,. \tag{6.2}$$

Lemma 6.1. *For any $s > 0$ and positive integer q,*

$$\frac{1}{f_s(\mathcal{L}_q)} > \tanh(2/s) \cdot E(q/s)\,.$$

Note that by Equation (6.2), for any fixed $s > 0$, as q (or equivalently, q/s) grows, $1/f_s(\mathcal{L}_q)$ rapidly approaches $\tanh(2/s)$. In turn, this approaches $2/s$ as s grows.

Proof. This follows directly from Lemma 6.2 below and Equation (6.1). By Lemma 6.2, the bound $\coth(2/s) > s/2$, and the definition of $E(x)$,

$$f_s(\mathcal{L}_q) < \coth(2/s)\left(\coth(q/s) + \frac{2q \cdot e^{2q/s}}{(e^{2q/s}-1)^2} \cdot \frac{2}{s}\right) = \frac{\coth(2/s)}{E(q/s)}\,.$$

The claim then follows by taking reciprocals. □

Lemma 6.2. *For any $s > 0$ and positive integer q,*

$$f_s(\mathcal{L}_q) = \coth(2/s) \cdot \coth(q/s) + \frac{2q \cdot e^{2q/s}}{(e^{2q/s}-1)^2}\,.$$

Proof. By Lemma 2.6, we can write

$$f_s(\mathcal{L}_q) = \sum_{\mathbf{v} \in \mathcal{L}_q} f_s(\mathbf{v}) = \sum_{x=0}^{q-1} f_s((x,x) + q\mathbb{Z}^2) = \sum_{x=0}^{q-1} f_s(x + q\mathbb{Z})^2.$$

We first rewrite $f_s(x + q\mathbb{Z})$ as a sum of two geometric series:

$$\begin{aligned}
f_s(x + q\mathbb{Z}) &= \sum_{z \in \mathbb{Z}} \exp(-2|x+qz|/s) \\
&= \sum_{z \geq 0} \exp\big(-2(x+qz)/s\big) + \sum_{z < 0} \exp\big(2(x+qz)/s\big) \\
&= \sum_{z \geq 0} \exp\big(-2(x+qz)/s\big) + \sum_{z \geq 0} \exp\big(2(x-q(z+1))/s\big) \\
&= \sum_{z \geq 0} \Big(\exp\big(-2(x+qz)/s\big) + \exp\big(-2(q(z+1)-x)/s\big)\Big) \\
&= \frac{e^{-2x/s} + e^{-2(q-x)/s}}{1 - e^{-2q/s}}\,.
\end{aligned}$$

Substituting this into the summation, we get that

$$\begin{aligned}(1-e^{-2q/s})^2 \cdot f_s(\mathcal{L}_q) &= \sum_{x=0}^{q-1}(e^{-2x/s}+e^{-2(q-x)/s})^2 \\ &= \sum_{x=0}^{q-1}\left(e^{-4x/s}+e^{-4(q-x)/s}+2e^{-2q/s}\right) \\ &= \frac{1-e^{-4q/s}}{1-e^{-4/s}}\cdot(1+e^{-4/s})+2q\cdot e^{-2q/s} \\ &= \coth(2/s)\cdot(1-e^{-4q/s})+2q\cdot e^{-2q/s} \\ &= \coth(2/s)\cdot\frac{1+e^{-2q/s}}{1-e^{-2q/s}}\cdot(1-e^{-2q/s})^2+2q\cdot e^{-2q/s} \\ &= \coth(2/s)\cdot\coth(q/s)\cdot(1-e^{-2q/s})^2+2q\cdot e^{-2q/s}\,.\end{aligned}$$

The claim follows by dividing both sides by $(1-e^{-2q/s})^2$, and multiplying both the numerator and denominator of the final term by $e^{4q/s}$. □

6.2 Worst-Case Decoding

Now we address list-decoding in the ℓ_1 metric, under worst-case error of bounded distance, by specializing the material of Section 4.1 to $p=1$ and using our bound on $f_s(\mathcal{L}_q)$ from Lemma 6.1. We consider decoding distance $d=\delta n$, where n is the code length and δ is the relative decoding distance. Then by Equations (4.2) and (4.3) and Lemma 6.1, we can list-decode for any R^* less than

$$R^{*,(1)}_{\mathrm{wc},q}(\delta) = \sup_{s>0} W^{(1)}_{q,\delta}(s)^2 > \sup_{s>0}\exp(-4\delta/s)\cdot\tanh(2/s)\cdot E(q/s)\,. \tag{6.3}$$

Corollary 6.3 below is obtained by maximizing the "main term" $\exp(-4\delta/s)\cdot\tanh(2/s)$ of the right-hand side of (6.3). By calculus, this is done by taking $s=4/\ln(D(\delta))>0$, where

$$D(\delta) := \sqrt{1+\frac{1}{\delta^2}}+\frac{1}{\delta} > 1\,.$$

Substituting, this means we can list-decode for any R^* less than

$$\widetilde{R}^{*,(1)}_{\mathrm{wc},q}(\delta) := \frac{\tanh(\ln\sqrt{D(\delta)})}{D(\delta)^\delta}\cdot E(q\ln(D(\delta))/4) = \frac{D(\delta)-1}{D(\delta)+1}\cdot\frac{E(q\ln(D(\delta))/4)}{D(\delta)^\delta}\,. \tag{6.4}$$

We consider this quantity's asymptotic behavior for large and small δ:

- As δ grows, $D(\delta)=1+1/\delta+O(1/\delta^2)$ and $D(\delta)^\delta$ approaches e, hence $\widetilde{R}^{*,(1)}_{\mathrm{wc},q}(\delta)$ approaches $1/(2e\delta)$ as q/δ also grows. This is consistent with Remark 4.4.
- As δ approaches zero, $D(\delta)$ approaches $2/\delta$ and $D(\delta)^\delta$ approaches 1, hence $\widetilde{R}^{*,(1)}_{\mathrm{wc},q}(\delta)$ approaches 1 as q/δ also grows.

Alternatively, we can get a simpler but cruder bound by replacing $\tanh(2/s)$ in Equation (6.3) with its upper bound of $2/s$. Then the resulting "main term" of $2\exp(-4\delta/s)/s$ is maximized at $s = 4\delta$; substituting, this means we can list decode for any R^* less than

$$e^{-1} \cdot \tanh(1/(2\delta)) \cdot E(q/(4\delta)) \; .$$

This bound approaches $1/(2e\delta)$ as δ and q/δ grow, which matches the behavior of $\widetilde{R}^{*,(1)}_{\mathrm{wc},q}(\delta)$ as described above. However, as δ approaches zero (and q/δ grows), the above bound merely approaches $1/e$, which is much worse than the limit of 1 for $\widetilde{R}^{*,(1)}_{\mathrm{wc},q}(\delta)$.

Corollary 6.3. *For any $\delta > 0$ and prime q, the GS algorithm using weight vector f_s for $s = 4/\ln(D(\delta))$ list-decodes, up to ℓ_1 distance δn in time $\mathrm{poly}(n, q, 1/(\sqrt{\widetilde{R}^{*,(1)}_{wc,q}(\delta)} - \sqrt{R^*}))$, any GRS code with adjusted rate $R^* < \widetilde{R}^{*,(1)}_{wc,q}(\delta)$ (see Equation (6.4)).*

Proof. By hypothesis and Lemma 6.1 and Equation (4.2),

$$R^* < \widetilde{R}^{*,(1)}_{\mathrm{wc},q}(\delta) = \frac{\tanh(\ln\sqrt{D(\delta)})}{D(\delta)^\delta} \cdot E(q/s) < \frac{\exp(-4\delta/s)}{f_s(\mathcal{L}_q)} = W^{(1)}_{q,\delta}(s)^2 \; .$$

The claim then follows directly by Theorem 4.2. □

Comparison to [RS94, WKU03]. To our knowledge, the only prior algorithms for (unique or list) decoding Reed–Solomon codes in the ℓ_1 (Lee) metric are [RS94, Section 5] and [WKU03]. We note that both of these require *discrete* (integer) error, whereas our algorithm works for *continuous* error.

For a certain subclass of GRS codes (and BCH codes more generally), [RS94] gives a *unique* decoding algorithm for up to half (a lower bound on) the ℓ_1 minimum distance, using Euclid's algorithm for polynomials. This algorithm decodes up to any relative distance $\delta < 1 - R < 1 - R^*$. For any prime-field GRS code, [WKU03] gives a list-decoding algorithm that uses GS as a subroutine, and has a piecewise distance-rate tradeoff due to its optimization over an integer parameter. (The algorithm works by putting equal weight on a range of alphabet symbols centered at the received symbol, optimizing over the range size for a given rate.)

By contrast with [RS94], and like [WKU03], our Corollary 6.3 works for *any* GRS code, and for *any* (arbitrarily large) relative decoding distance $\delta > 0$, for sufficiently small $R^* > 0$. Our rate-distance trade-off surpasses that of both [RS94, WKU03] for all $\delta \gtrapprox 0.78988$, which corresponds to rates $R^* \lessapprox 0.21012$; see Figure 1.

6.3 Unique Decoding for a Subclass of GRS Codes

As in Section 5.3, for the same subclass of GRS codes and certain parameters covered by our list-decoding algorithm, the decoding output is in fact *unique*. To show this, we give a lower bound on the ℓ_1 minimum distance of such codes, and then observe that our list-decoding algorithm can decode to beyond half this distance for all small enough rates.

Lemma 6.4 (adapted from [RS94, Theorem 4]). *Any prime-field GRS code $\mathrm{GRS}_{q,k}(\boldsymbol{\alpha}, \boldsymbol{\alpha}) \subseteq \mathbb{F}_q^n$ (whose twist factors $\mathbf{t}$ equal the nonzero evaluation points $\boldsymbol{\alpha}$) of rate $R = k/n$ has ℓ_1 minimum distance at least*

$$\frac{(n+1)^2 - k^2}{4k} > \frac{1 - R^2}{4R} \cdot n \; .$$

Proof. Define $\beta := (n+1-k)/(2k)$ and write it in terms of its integer and fractional parts as $\beta = \ell + \gamma$, where $\ell := \lfloor (n+1-k)/(2k) \rfloor$ and $\gamma := r/(2k)$ for $r := n+1-k(2\ell+1)$. Following the same reasoning as in the proof of Lemma 5.7, and using the fact that $\gamma < 1$, the ℓ_1 distance of any nonzero $\mathbf{c} \in \mathrm{GRS}_{q,k}(\boldsymbol{\alpha}, \boldsymbol{\alpha})$ is

$$\begin{aligned}
\|\mathbf{c}\|_1 &\geq 2k \cdot \sum_{i=1}^{\ell} i + r(\ell+1) \\
&= (k\ell + r)(\ell+1) \\
&= (k(\beta-\gamma) + 2k\gamma)(\beta-\gamma+1) \\
&= k(\beta+\gamma)(\beta-\gamma+1) \\
&= k(\beta^2 - \gamma^2 + \beta + \gamma) \\
&\geq k\beta(\beta+1) \\
&= \frac{(n+1)^2 - k^2}{4k} \,.
\end{aligned}$$

□

Lemma 6.4 gives a relationship between the code rate R and (a lower bound on) half the ℓ_1 minimum distance, for which decoding to that distance yields a unique solution. By taking the functional inverse of half this minimum-distance bound, we see that decoding to relative distance δ yields a unique solution as long as

$$R < R^{(1)}_{\text{uniq}}(\delta) := -4\delta + \sqrt{(4\delta)^2 + 1} \,,$$

which approaches $1/(8\delta)$ as δ grows. This curve is shown in Figure 1. Observe that for any δ for which our list-decoding algorithm outperforms the unique decoder of [RS94] (or for which [RS94] does not apply), we have that $R^{*,(1)}_{\text{wc}}(\delta) > R^{(1)}_{\text{uniq}}(\delta)$. In other words, we can efficiently list decode to relative distance δ for all rates up to $R^{(1)}_{\text{uniq}}(\delta)$ (and beyond), thus yielding a *unique* decoder for these parameters. Alternatively, as the rate R approaches zero, we can efficiently list decode to a multiple of the unique-decoding distance bound that approaches $8/(2e) \approx 1.4715$.

6.4 Average-Case Decoding

We now consider average-case decoding under a memoryless additive (continuous or discrete) Laplacian channel, by specializing the material of Section 4.2 to $p = 1$ and using our bound on $f_s(\mathcal{L}_q)$ from Lemma 6.1. Consider a Laplacian channel of parameter $r > 0$. Then by Equations (4.4) and (4.5), we can list-decode for any R^* less than

$$R^{*,(1)}_{\text{ac},q}(r) = \sup_{s>0} A^{(1)}_{q,r}(s)^2 > \sup_{s>0} \frac{s^2 \cdot \tanh(2/s)}{(r+s)^2} \cdot E(q/s) \,,$$

where the inequality is by Lemma 6.1.

Corollary 6.5 below is obtained by nearly maximizing the right-hand side of (6.4), at least for moderate or large values of r. Specifically, we use the bound $\tanh(2/s) < 2/s$ to approximate the "main term" of (6.4) by $2s/(r+s)^2$. This is maximized at $s = r$, which makes the original main term equal to $\tanh(2/r)/4$. Note that $R^{*,(1)}_{\text{ac},q}(r)$ does indeed approach this value as r and q/r grow, because $\tanh(2/r)$ approaches $2/r$, and $E(q/r)$ rapidly approaches 1 (see Equation (6.2)).

However, for small values of r, the expression in (6.4) is maximized for s significantly larger than r, to have value much larger than $\tanh(2/r)/4 < 1/4$. This maximization can be computed numerically, and indeed, $R^{*,(1)}_{\text{ac},q}(r)$ approaches 1 as r approaches 0; see Figure 1.

Corollary 6.5. *For any $r > 0$, $\alpha \in (0,1)$, and prime q, the GS algorithm using weight vector given by f_r list-decodes, in time* $\mathrm{poly}(n, q, 1/(\sqrt{R^{*,(1)}_{ac,q}} - \sqrt{R^*}))$*, any GRS code with adjusted rate*

$$R^* < \widetilde{R}^{*,(1)}_{ac,q}(r) := \frac{\tanh(2/r)}{4} \cdot E(q/r)\,,$$

except with probability less than $\exp\left(-n \cdot \alpha^2 \cdot r \cdot \left(\sqrt{\widetilde{R}^{*,(1)}_{ac,q}(r)} - \sqrt{R^*}\right)^2\right)$.

Proof. By hypothesis, Lemmas 6.1 and 4.5 and Equation (4.4),

$$R^* < \widetilde{R}^{*,(1)}_{\mathrm{ac},q}(r) = \frac{\tanh(2/r)}{4} \cdot E(q/r) < \frac{\mu^2_{r,r}}{f_r(\mathcal{L}_q)} = A^{(1)}_{q,r}(r)^2\,.$$

The claim then follows directly by Theorem 4.6, and (for the probability bound) the fact that $f_r(\mathcal{L}_q) > \coth(2/r) > r/2$ by Lemma 6.2. □

References

[Ban93] W. Banaszczyk. New bounds in some transference theorems in the geometry of numbers. *Mathematische Annalen*, 296(4):625–635, 1993. Page 4.

[Ban95] W. Banaszczyk. Inequalites for convex bodies and polar reciprocal lattices in $\mathbb{R}^n$. *Discrete & Computational Geometry*, 13:217–231, 1995. Page 16.

[Eli58] P. Elias. Zero error capacity under list decoding. *IEEE Transactions on Information Theory*, 34(5):1070–1074, September 1988. Originally appeared as *Quarterly Progress Report*, vol. 48, pp. 88-90, Research Laboratory of Electronics, MIT, January 1958. Page 1.

[EOR91] N. D. Elkies, A. M. Odlyzko, and J. A. Rush. On the packing densities of superballs and other bodies. *Inventiones mathematicae*, 105:613–639, December 1991. Page 12.

[GRS19] V. Guruswami, A. Rudra, and M. Sudan. Essential coding theory. March 2019. https://cse.buffalo.edu/faculty/atri/courses/coding-theory/book/web-coding-book.pdf. Page 1.

[GS98] V. Guruswami and M. Sudan. Improved decoding of Reed-Solomon and algebraic-geometry codes. *IEEE Trans. Inf. Theory*, 45(6):1757–1767, 1999. Preliminary version in FOCS 1998. Pages 1, 2, 3, and 8.

[Gur01] V. Guruswami. *List decoding of error correcting codes*. Ph.D. thesis, Massachusetts Institute of Technology, 2001. Pages 3 and 8.

[KV03] R. Koetter and A. Vardy. Algebraic soft-decision decoding of Reed-Solomon codes. *IEEE Trans. Inf. Theory*, 49(11):2809–2825, 2003. Pages 3 and 8.

[MP22] E. Mook and C. Peikert. Lattice (list) decoding near Minkowski's inequality. *IEEE Trans. Inf. Theory*, 68(2):863–870, 2022. Pages 2, 3, 18, and 19.

[MR04] D. Micciancio and O. Regev. Worst-case to average-case reductions based on Gaussian measures. *SIAM J. Comput.*, 37(1):267–302, 2007. Preliminary version in FOCS 2004. Pages 4 and 7.

[RS60] I. S. Reed and G. Solomon. Polynomial codes over certain finite fields. *Journal of the Society for Industrial and Applied Mathematics*, 8(2):300–304, 1960. Page 1.

[RS94] R. M. Roth and P. H. Siegel. Lee-metric BCH codes and their application to constrained and partial-response channels. *IEEE Trans. Inf. Theory*, 40(4):1083–1096, 1994. Pages 2, 3, 19, 23, and 24.

[Ser73] J.-P. Serre. *A Course in Arithmetic*. Springer New York, NY, 1973. ISBN 978-0-387-90040-7. doi:https://doi.org/10.1007/978-1-4684-9884-4. Page 6.

[Sud97] M. Sudan. Decoding of Reed Solomon codes beyond the error-correction bound. *J. Complex.*, 13(1):180–193, 1997. Page 1.

[WKU03] X.-W. Wu, M. Kuijper, and P. Udaya. Lee-metric decoding of BCH and Reed–Solomon codes. *Electronics Letters*, 39(21):1522–1524, October 2003. Pages 2, 3, and 23.

[Woz58] J. M. Wozencraft. List decoding. *Quarterly Progress Report, Research Laboratory of Electronics, MIT*, 48:90–95, 1958. Page 1.