

Reed-Solomon Codes at Capacity: Algorithmic List-Decoding and Proximity Gaps

Prahladh Harsha*

Mrinal Kumar*

Ramprasad Saptharishi*

Abstract

Understanding the limits of list-decodability of Reed-Solomon codes has been one of the most important open problems in algebraic coding theory. Recently, Brakensiek, Chen, Putterman, Zhang, and Zheng, in a remarkable breakthrough, showed that Reed-Solomon (RS) codes over fields of large characteristic are algorithmically list-decodable all the way up to capacity. Building on this result, Jeronimo subsequently extended these techniques to solve the proximity-gaps question for RS codes.

In this article, we give a unified and transparent exposition of these results.

1 Introduction

Reed-Solomon codes [RS60] are one of the most well-studied families of error-correcting codes. Given a finite field $\mathbb{F}_q$ of size $q = p^r$ and characteristic p and a set $E = \{\alpha_1, \dots, \alpha_n\} \subseteq \mathbb{F}_q$ of n distinct evaluation points, the code $\text{RS}_q[E, k]$ is the map $\mathbb{F}_q^k \rightarrow \mathbb{F}_q^n$ which maps a message $f \in \mathbb{F}_q^k$, viewed as a polynomial $f(X) \in \mathbb{F}_q[X]$ of degree less than k , to the evaluation vector $(f(\alpha_1), \dots, f(\alpha_n)) \in \mathbb{F}_q^n$. This is a linear code with a message space of dimension k and block length n , thereby having rate $\rho := k/n$. The code is sometimes identified with the linear space comprising the set of all evaluation vectors. Sometimes we drop the set E and instead write $\text{RS}_q[n, k]$.

The decoding problem associated with this code is as follows: given a received word $\mathbf{y} = (y_1, \dots, y_n) \in \mathbb{F}_q^n$ and an agreement parameter A , find *all* polynomials $f \in \mathbb{F}_q[X]$ of degree less than k whose evaluation vectors agree with the received word $\mathbf{y}$ on at least A locations. Determining the agreement threshold at which the list size is bounded, and designing an efficient algorithm to find the resulting list, have been long-standing problems in algebraic coding theory. A related open problem is the proximity-gaps question of RS codes, first studied by Ben-Sasson, Carmon, Ishai, Kopparty and Saraf [BCIKS23]. Like several other open problems in mathematics and theoretical

*Tata Institute of Fundamental Research, Mumbai, India. {prahladh, mrinal, ramprasad}@tifr.res.in. Research supported by the Department of Atomic Energy, Government of India, under project number RTI400112, and in part by research grants from Google and Premji Invest.

computer science that have recently met their fate, thanks to some heavy lifting by AI tools, both of these problems were resolved in the last few weeks with AI assistance.

The Proximity Prize (<https://proximityprize.org/>) initiated by the Ethereum foundation led to a large crowd-sourced effort to resolve these conjectures. A culmination of these efforts led to the following breakthrough results by Brakensiek, Chen, Putterman, Zhang and Zheng [BCPZZ26] and Jeronimo [Jer26]. First, Brakensiek, Chen, Putterman, Zhang and Zheng established that for all fields of large characteristic, we can in fact decode all the way "up to capacity", i.e., for A as small as $k(1 + \varepsilon)$ for any $\varepsilon > 0$. More precisely:

Theorem 1.1 ([BCPZZ26]). *Let $\rho, \varepsilon > 0$, and let k, n be large enough positive integers with $k/n \geq \rho$. Let $\mathbb{F}_q$ be any finite field of size $q = p^r$ for prime p satisfying $q \geq n$ and $p > k$. Then, there is an algorithm $\mathcal{A}(\varepsilon)$ that on input a list $(\alpha_1, \alpha_2, \dots, \alpha_n)$ of n distinct elements of $\mathbb{F}_q$ and a received word $\mathbf{y} = (y_1, \dots, y_n) \in \mathbb{F}_q^n$, outputs all polynomials $f \in \mathbb{F}_q[X]$ of degree less than k satisfying $|\text{agr}(f, \mathbf{y})| \geq k(1 + \varepsilon)$ where*

$$\text{agr}(f, \mathbf{y}) := \{i \in [n] : f(\alpha_i) = y_i\}$$

in time $q^{C(\varepsilon, \rho)}$ where C is some function of ε and the rate k/n ¹.

Almost immediately after this result was posted, the techniques developed in this paper were extended by Jeronimo [Jer26] to solve the proximity-gaps problem for RS codes. We state the result for lines for simplicity (although it readily extends to the more general statement).

Theorem 1.2 ([Jer26]). *Let $\rho, \varepsilon > 0$, and let k, n be large enough positive integers with $k/n \geq \rho$. Let $\text{RS}_q[n, k]$ be a Reed-Solomon code over a finite field $\mathbb{F}_q$ of characteristic p and rate at least ρ for any set $E = \{\alpha_1, \dots, \alpha_n\} \subseteq \mathbb{F}_q$ of evaluation points satisfying $q \geq n$ and $p > k$. There is a bound $L = n^{C(\varepsilon, \rho)}$, where C is some function of ε and ρ , such that the following holds for every pair of functions $u, v : E \rightarrow \mathbb{F}_q$:*

Suppose for at least L values of $\zeta \in \mathbb{F}_q$ there exists some polynomial f_ζ of degree less than k such that $|\text{agr}(u + \zeta v, f_\zeta)| \geq k(1 + \varepsilon)$. Then, there are polynomials g, h of degree less than k such that

$$|\{i \in [n] : u(\alpha_i) = g(\alpha_i) \text{ and } v(\alpha_i) = h(\alpha_i)\}| \geq k(1 + \varepsilon).$$

I.e., there is a set of size at least $k(1 + \varepsilon)$ of correlated agreements of u and v with $\text{RS}_q[n, k]$, thereby showing that $u + \zeta v$ has $k(1 + \varepsilon)$ agreements with $\text{RS}_q[n, k]$ for every $\zeta \in \mathbb{F}_q$.

¹The original writeup of [BCPZZ26] proved this result only for low rates and this was then extended to all rates in a subsequent revision using a reduction due to Alrabiah, Goyal and Guruswami. The proof presented in this exposition directly works for all rates and all sets of evaluation points and avoids the use of the Alrabiah-Goyal-Guruswami reduction (we note the revised version of [BCPZZ26] already mentions, without proof, that such a simplification is feasible).

Thus, if the field size q is a large enough polynomial in n (more precisely, $q \gg n^{C(\epsilon, \rho)}$), a very small fraction of ζ 's that witness closeness of $u + \zeta v$ with $\text{RS}_q[n, k]$ is sufficient to conclude that $u + \zeta v$ is close to $\text{RS}_q[n, k]$ for *every* choice of ζ .

In this note, we give an exposition of the proofs of both these results. On the way, we highlight the key ideas and make slight improvements in some of the parameters involved. The exposition is largely self-contained, barring a couple of standard facts from algebraic geometry and coding theory literature, which we state explicitly.

1.1 A birds-eye view of the proofs

For the benefit of readers familiar with the existing algebraic coding theory literature, we summarize below the high-level ideas that go into the proof of these statements. Others may prefer to skip this overview and proceed directly to the subsequent sections, which explain the proofs from first principles.

Algorithmic list-decoding ([BCPZZ26])

The [BCPZZ26] algorithm follows the by now standard template of Sudan and Guruswami-Sudan [Sud97, GS99] for list-decoding: explain the received using a non-zero polynomial Q and extract all close-enough codewords from this polynomial. The existence of such a non-zero polynomial Q is shown by setting a linear system and showing that the number of variables in this system exceeds the number of linear constraints.

The [BCPZZ26] algorithm deviates from the Guruswami-Sudan algorithm in multiple important aspects. Some of these new technical ingredients in the algorithm in [BCPZZ26] can be traced back to the algorithm of Kopparty [Kop15] for list-decoding univariate multiplicity codes, in both implicit and explicit ways. We start by recalling the outline of Kopparty's algorithm and then discuss how some of these ideas are adapted to the setting of Reed-Solomon codes in [BCPZZ26]. The technical insights in [BCPZZ26] that make this adaptation feasible are perhaps of independent interest in the setting of the polynomial method, beyond this specific proof.

Kopparty's univariate multiplicity code decoder: For list-decoding univariate multiplicity codes (over fields of sufficiently large characteristic), Kopparty suggested obtaining a non-zero multivariate polynomial $Q(X, Y_0, \dots, Y_d)$ of low enough degree such that for every message polynomial f , if f agrees with the received word at evaluation point α_i , then

$$Q(\alpha_i + T, f(\alpha_i + T), f^{(1)}(\alpha_i + T), \dots, f^{(d)}(\alpha_i + T)) = 0 \bmod T^m.$$

This ensures that if a low degree f has sufficiently large agreement with the received word, then f satisfies the differential equation

$$Q(X, f(X), f^{(1)}(X), \dots, f^{(d)}(X)) \equiv 0.$$

Thus, the task of list-decoding has reduced to solving the above differential equation. The final ingredient of Kopparty's decoder is an efficient algorithm that solves the above differential equation to obtain a list of degree k univariates of size at most $q^{O(d)}$ containing all close enough message polynomials, where q is the size of the underlying finite field. For a constant d , this gives a polynomial size list for decoding univariate multiplicity codes up to capacity over finite fields of sufficiently large characteristic.

We now discuss the main steps in [BCPZZ26].

Step 1 (multivariate interpolation): The first idea in [BCPZZ26] is to construct a low degree *multivariate* polynomial Q in the interpolation step. The goal again is to obtain a non-zero $Q(X, Y_0, \dots, Y_d)$ such that if $f(X)$ and the received word agree at coordinate α_i , then

$$Q(\alpha_i + T, f(\alpha_i + T), f^{(1)}(\alpha_i + T), \dots, f^{(d)}(\alpha_i + T)) = 0 \bmod T^m.$$

Note that this is equivalent to the polynomial $Q(X, f(X), \dots, f^{(d)}(X))$ having a zero of multiplicity at least m at α_i . While similar to the interpolation step in [Kop15], the absence of high multiplicity evaluations of the message polynomial in the setting of Reed-Solomon codes leads to obvious technical issues. In the absence of access to high multiplicity evaluations, [BCPZZ26] use formal indeterminates $Y_1, Y_2, \dots, Y_d$ for these high-order derivatives. However, it can be checked (with some effort) that treating the high multiplicity information as independent formal variables, with no further apparent algebraic relationship between them and the received word, is in itself insufficient to list decode beyond the Johnson radius.

Step 2 (an algebraic relation between derivatives): The second ingredient of [BCPZZ26] that gives a non-trivial way to use the first idea discussed above, is that the high-order derivatives and the evaluation indeed satisfy a useful algebraic relationship. This is given by the "reverse" Taylor expansion:

$$f(X + T) = f(X) + T f^{(1)}(X + T) + \dots + (-1)^{d+1} T^d f^{(d)}(X + T) + E$$

where the error term $E = E(X, T)$ is divisible by $T^{(d+1)}$. This relationship is then incorporated to bound the number of variables and constraints as follows:

$$\# \text{ variables} \geq c_\epsilon \cdot \frac{m^2 A^2}{k} \cdot |\mathcal{B}|, \quad \# \text{ constraints} \leq n \cdot \frac{m^2}{d} \cdot |\mathcal{B} + \Lambda(m)|$$

Here, n is the blocklength, ε measures how close we are to capacity (i.e., we are decoding from agreements $A \geq k(1 + \varepsilon)$), m is the multiplicity parameter (as in the Guruswami-Sudan algorithm), d the number of high-order derivatives used and c_ε is a constant only dependent on ε . The only new terms here are the sets $\mathcal{B}$ and $\Lambda(m)$. The set $\mathcal{B}$ refers to the set of exponent vectors $(b_1, b_2, \dots, b_d)$ allowed for monomials of the form $Y_1^{b_1} Y_2^{b_2} \dots Y_d^{b_d}$ in the polynomial Q while $\Lambda(m)$ is the set of non-negative integer vectors $(b'_1, b'_2, \dots, b'_d)$ satisfying $\sum i b'_i \leq m$.

The key point to note is the following: for the number of variables to be larger than the number of constraints, it suffices to have

$$A > \left(\frac{1}{c_\varepsilon} \cdot \frac{nk}{d} \cdot \frac{|\mathcal{B} + \Lambda(m)|}{|\mathcal{B}|} \right)^{1/2} = k \left(\frac{1}{c_\varepsilon} \cdot \frac{1}{\rho d} \cdot \frac{|\mathcal{B} + \Lambda(m)|}{|\mathcal{B}|} \right)^{1/2},$$

where the second expression is obtained by setting $n = k/\rho$, where ρ is the rate of the code.

Thus, to minimize A , ideally to $A = (1 + \varepsilon)k$, parameters need to be set so that the expression $\left(\frac{1}{c_\varepsilon} \cdot \frac{1}{\rho d} \cdot \frac{|\mathcal{B} + \Lambda(m)|}{|\mathcal{B}|} \right)^{1/2}$ is less than $(1 + \varepsilon)$. It turns out that this requires a careful choice of $\mathcal{B}$ as we discuss below.

Step 3 (careful choice of a monomial basis for Q): A natural choice for $\mathcal{B}$ is the set $(b_1, b_2, \dots, b_d)$ in $\mathbb{Z}_{\geq 0}^d$ of vectors satisfying $\sum b_i \leq D$. This corresponds to interpolating a Q whose total degree in the variables $Y_1, \dots, Y_d$ is at most D and is the typical space of polynomials that one works with in many applications of the polynomial method, including those in algebraic coding theory. Unfortunately this does not appear to work here, as for this choice of $\mathcal{B}$, the corresponding ratio $\frac{|\mathcal{B} + \Lambda(m)|}{|\mathcal{B}|}$ is $\Omega(d)$, and the lower bound on A obtained is much larger than the desired bound of $(1 + \varepsilon)k$.

The careful choice of $\mathcal{B}$ is the third main ingredient of the proof in [BCPZZ26]. This choice ensures that the ratio $\frac{|\mathcal{B} + \Lambda(m)|}{|\mathcal{B}|}$ is less than $d^{1-\eta}$ for some constant $\eta \in (0, 1)$. By a careful setting d , depending on the rate ρ and ε , the bound on A obtained at the end ² indeed looks like $(1 + \varepsilon)k$. Choosing $\mathcal{B} = \Lambda(W)$ for an appropriate integer W works (the choice $W \approx md(1 + \varepsilon/2)/\ln d$ suffices) since in this case $\mathcal{B} + \Lambda(m) = \Lambda(W + m)$ and we can exactly estimate the ratio $\frac{|\Lambda(W+m)|}{|\Lambda(W)|}$ (for m sufficiently large compared to d).

Step 4 (differential equation solver): The non-zero polynomial Q obtained at the end of the interpolation step is then shown to contain all close enough message polynomials f in its belly in the same sense as that in [Kop15], and Kopparty's differential equation solver is then used off-the-shelf to recover these messages. The parameter d has to be set to a constant so that the list size is polynomially bounded and the differential equation solver runs in polynomial time.

²We also remark that the bound obtained on A can never be less than k , as one would expect. This is evident from the details in the proof.

Proximity gaps for Reed-Solomon codes [Jer26]

The proximity-gaps problem was first studied by Ben-Sasson, Carmon, Ishai, Kopparty, and Saraf [BCIKS23], who proved results when the number of agreements A is at least the unique-decoding radius $(n+k)/2$ or the Johnson radius $\sqrt{nk}$. In both these settings, [BCIKS23] used the corresponding decoding algorithms, in other words the Welch-Berlekamp [WB86]-algorithm (as abstracted by Gemmell and Sudan [GS92]) in the unique-decoding setting and Guruswami-Sudan [GS99]-algorithm in the list-decoding up to Johnson radius setting. Given that we now have a list-decoding algorithm all the way up to capacity, à la [BCPZZ26], we can naturally ask if we can further reduce the number of agreements in the proximity gap question to $k(1 + \varepsilon)$. And this is precisely what [Jer26]’s result does. However, [Jer26]’s result does not follow the approach of [BCIKS23]. It instead proceeds as follows. It observes that a pruning algorithm similar to that of Kopparty, Ron Zewi, Saraf and Wootters [KRSW23] and Tamo [Tam24] can be used to bound the number of polynomials in the list by a polynomial in n , whose exponent depends only on the rate ρ and ε and is independent of the underlying field size. The pruning here is more involved compared to [KRSW23, Tam24] as the objects involved are varieties instead of affine subspaces and this is where we need some basic algebraic geometry facts. Given such a pruning method, one can bound the number of values of ζ for which there is no coordinated agreement.

AI disclosure: The technical details in both the papers [BCPZZ26, Jer26] (especially the latter) are very dense and understanding these papers (for us) required several extended conversations with Claude (Opus and Sonnet). The proof here follows the same strategy as the original papers, but the details are (in our opinion) much easier to follow. This exposition is entirely written by the human authors.

Human disclosure: Many thanks to Irit Dinur, Swastik Kopparty and Madhu Sudan for substantial and extremely helpful comments on various parts of the paper. These comments helped us improve the presentation significantly.

A special thanks to Swastik for a super careful reading of the technical sections and pointing us to some references that we had missed.

2 Notation and preliminaries

We will use boldface letters such as $\mathbf{y}$ to denote a tuple $(y_1, \dots, y_k)$; the number of coordinates would usually be clear from context.

Hasse Derivatives: For a polynomial $f(X)$ of degree at most d , we define the *Hasse derivative* of f , denoted by $f^{(0)} = f, f^{(1)}$, etc. as the terms of the Taylor expansion of $f(X + T)$ for a formal

variable T :

$$f(X + T) = f^{(0)}(X) + f^{(1)}(X) \cdot T + f^{(2)}(X) \cdot T^2 + \dots + f^{(d)}(X) \cdot T^d.$$

While working with finite fields, we follow the customary procedure of using Hasse derivatives instead of standard derivatives. However, since the results presented in the article apply only to fields of large characteristic, we might have as well used standard derivatives. In this setting, the Hasse derivatives $f^{(i)}$ are related to the standard derivatives $\frac{d^i f}{dX^i}$ via the formula $f^{(i)}(X) = \frac{1}{i!} \cdot \frac{d^i f}{dX^i}$.

A polynomial $f(X)$ is said to have a root α of multiplicity m if $(X - \alpha)^m$ divides f . Equivalently, $f^{(i)}(\alpha) = 0$ for all $0 \leq i < m$.

Here is a standard combinatorial fact that will be used in the algorithm.

Fact 2.1 (number of bivariate monomials with weighted-degree bound). *For positive integers k, D , if $N(k, D)$ refers to the number of monomials of the form $X^a Y^b$ with $a + kb < D$, then*

$$\frac{D(D+k)}{2k} \leq N(k, D) \leq \frac{D(D+k)}{2k} + \frac{k}{8}$$

For a received word $\mathbf{y} \in \mathbb{F}_q^n$ and polynomial $f(x)$, define $\text{agr}(f, \mathbf{y})$ to be

$$\text{agr}(f, \mathbf{y}) := \{i \in [n] : f(\alpha_i) = y_i\}.$$

3 Sudan's template for algebraic decoding algorithms

The history of decoding algorithms for Reed-Solomon codes is quite rich with algorithms known since as far back as 1960s. We only mention a subset of these results, with the focus being on results that are most relevant to the recent progress on this problem.

Berlekamp-Welch [WB86] (as abstracted by Gemmell and Sudan [GS92]) gave an efficient *unique-decoding* algorithm where the number of agreements $A \geq \frac{n+k}{2}$. Sudan [Sud97] gave an efficient *list-decoding* algorithm with $A \gtrsim \sqrt{2nk}$, and Guruswami and Sudan [GS99] further improved this to $A \gtrsim \sqrt{nk}$ by adding the *method of multiplicities*. This bound of $\sqrt{nk}$ agreements is also known as the Johnson radius, and *any* code has small list size when dealing with $\gtrsim \sqrt{nk}$ agreements (note this is just a combinatorial guarantee about *any* code while the Guruswami-Sudan algorithm yields *efficient* list-decoding algorithms up to this radius for RS codes).

To go beyond the Johnson radius, variants of Reed-Solomon codes were devised (such as multiplicity codes, or Folded Reed-Solomon codes [PV05, GR08, GW13, Kop15]) that are efficiently list-decodable up to capacity (i.e., $A \geq k(1 + \epsilon)$ for any $\epsilon > 0$).

All the above algorithms proceed via a very similar template, first formalized for unique-decoding by Gemmell and Sudan [GS92] and for list-decoding by Sudan [Sud97]: Given a received

word $\mathbf{y} = (y_1, \dots, y_n)$,

- Build an “explainer polynomial” Q based on the data $(y_1, \dots, y_n)$.
- Extract from Q all f such that $\text{agr}(f, \mathbf{y}) \geq A$.

The algorithm of [BCPZZ26] proceeds along the same template, but with an additional ingredient of *hidden derivatives*. In its final form, this idea lets them adapt the techniques of Kopparty’s univariate multiplicity code decoder in [Kop15] to the setting of Reed-Solomon codes. To illustrate the techniques in [BCPZZ26] better, it would be useful to first revisit the method of multiplicities used in the Guruswami-Sudan [GS99] algorithm.

3.1 Guruswami-Sudan’s list-decoding algorithm

Let us fix a parameter m (for the multiplicity).

Given a received word $\mathbf{y} = (y_1, \dots, y_n)$, we try to find a nonzero bivariate polynomial $Q(X, Y)$ with the following properties:

Monomial space: Every monomial $X^a Y^r$ in Q satisfies $a + kr < mA$. In other words, the $(1, k)$ -degree of Q is strictly smaller than mA .

Constraints: For every $i \in [n]$, consider the polynomial

$$Q(\alpha_i + T, y_i + U) = \sum_{a,r} q_{i,a,r} T^a U^r$$

and add the constraint that $q_{i,a,r} = 0$ for all a, r satisfying $a + r < m$.

Claim 3.1. Suppose such a nonzero $Q(X, Y)$ exists. If f is a polynomial of degree less than k such that $|\text{agr}(f, \mathbf{y})| \geq A$, then $R_f(X) := Q(X, f(X)) \equiv 0$ or equivalently $Y - f(X)$ is a factor of Q .

Proof. Since Q has $(1, k)$ -degree less than mA , the polynomial $R_f(X) = Q(X, f(X))$ has degree strictly smaller than mA .

Suppose $f(\alpha_i) = y_i$ for some i , then $f(\alpha_i + T) = y_i + U_{f,i}(T)$ for some polynomial $U_{f,i}$ that is divisible by T . Therefore,

$$\begin{aligned} R_f(\alpha_i + T) &= Q(\alpha_i + T, f(\alpha_i + T)) \\ &= Q(\alpha_i + T, y_i + U_{f,i}(T)) \\ &= 0 \pmod{T^m} \quad (\text{from our constraints}) \end{aligned}$$

Thus, R_f has α_i as a root of multiplicity at least m . But if $\text{agr}(f, \mathbf{y}) \geq A$, then get that R_f has A roots of multiplicity at least m each but $\deg R_f < mA$. Hence, $R_f(X) \equiv 0$. $\square$

Once we find such a nonzero Q , the list-decoding algorithm is now immediate; just factor $Q(X, Y)$ and any f in the list would correspond to a factor of the form $Y - f(X)$ of $Q(X, Y)$.

Thus, the parameter choices are basically to ensure that such a nonzero polynomial exists. To this end, it suffices to show that the number of constraints is less than the number of indeterminates in Q .

Counting indeterminates in Q : We need to count the number of monomials $X^a Y^r$ such that $a + kr < mA$. When m is large enough, it can be shown (Fact 2.1) that this is roughly

$$\frac{m^2 A^2}{2k}.$$

Counting constraints: The number of constraints per $i \in [n]$ is the number of monomials of the form $T^a U^r$ with $a + r < m$, and this is about $\binom{m}{2}$.

Hence, to ensure that there is a nonzero Q , it suffices to ensure that

$$\frac{m^2 A^2}{2k} > n \cdot \frac{m^2}{2}$$

and this is satisfied if $A \gtrsim \sqrt{nk}$. This completes the description of the Guruswami-Sudan algorithm.

3.2 Using additional derivatives

A very natural idea is to use some of the techniques that were useful in the setting of multiplicity codes or Folded Reed-Solomon codes, namely to use higher derivative information. Suppose we could, as first suggested by Kopparty [Kop15] while decoding univariate multiplicity codes, find a nonzero polynomial $Q(X, Y_0, \dots, Y_d)$ with $(1, k, k, \dots, k)$ -degree³ less than mA with the property that whenever $f(\alpha_i) = y_i$ we have

$$Q(\alpha_i + T, f(\alpha_i + T), f^{(1)}(\alpha_i + T), \dots, f^{(d)}(\alpha_i + T)) = 0 \mod T^m \quad (3.2)$$

where $f^{(i)}(X)$ is the i -th Hasse derivative of f .

Then, if f has at least A agreements, then we would again have that

$$R_f(X) := Q(X, f(X), f^{(1)}(X), \dots, f^{(d)}(X)) = 0.$$

That is, f satisfies the “differential equation” given by Q . The intuition behind working with a Q that is multivariate (and not just bivariate) is that the dimension of the space of such polynomials at any fixed weighted degree grows as the number of variables increases, and hence there is a chance that there would be a non-zero Q of lower overall degree that satisfies the interpolation constraints.

³Technically, one should work with $(1, k, k-1, \dots, k-d)$ -degree, but the simpler $(1, k, k, \dots, k)$ -degree bound suffices.

Fortunately, once we have such a Q , we also know how to find all such f 's that satisfy $R_f(X) \equiv 0$ via a differential equation solver in Kopparty's [Kop15] decoding algorithm for univariate multiplicity codes.

Theorem 3.3 ([Kop15]). *Suppose $Q(X, Y_0, \dots, Y_d) \in \mathbb{F}_q[X, Y_0, \dots, Y_d]$ is a nonzero polynomial with $\text{char}(\mathbb{F}_q) = p > k, \deg_{Y_i}(Q)$. Then the set of all polynomials $f(X)$ of degree less than k that satisfy*

$$Q(X, f(X), f^{(1)}(X), \dots, f^{(d)}(X)) = 0$$

can be enumerated in time $q^{O(d)} \cdot \text{poly}(k, d_Q)$, where d_Q is the $(1, k, k, \dots, k)$ -degree of Q .

Remark. *The conditions above are slightly different from [Kop15] but follow from the proof.* $\diamond$

The main question is how we should be setting up our constraints to ensure (3.2).

Naïve attempt

Let us try to follow the Guruswami-Sudan outline. We know that if $f(\alpha_i) = y_i$, then $f(\alpha_i + T) = y_i + U$ for some polynomial U divisible by T . However, we have no information about the higher order derivatives. Thus, we may attempt to leave them as indeterminates.

Constraints: For each $i \in [n]$, consider the polynomial

$$Q(\alpha_i + T, y_i + U, Y_1, \dots, Y_d) = \sum q_{i,a,r,(b_1,\dots,b_d)} T^a U^r Y_1^{b_1} \dots Y_d^{b_d} = \sum q_{i,a,r,\mathbf{b}} T^a U^r \mathbf{Y}^{\mathbf{b}}$$

and insist that $q_{i,a,r,\mathbf{b}} = 0$ whenever $a + r < m$.

Thus, if $\mathcal{B}$ refers to the space of "valid" $\mathbf{b} = (b_1, \dots, b_d)$, then this amounts to roughly $n \cdot \binom{m+1}{2} \cdot |\mathcal{B}|$ constraints overall.

Number of indeterminates: We wish to get an estimate of the dimension of the interpolation space, namely the number of monomials we could have in Q . Thus, we want to count $X^a Y_0^{b_0} Y_1^{b_1} \dots Y_d^{b_d}$. To make it easier to relate this with $\mathcal{B}$, let us assume that we $|\mathbf{b}| = b_1 + \dots + b_d < 0.99 \cdot \frac{mA}{k}$, so that $a + kb_0$ could be as large as $0.01 \cdot mA$ no matter what $\mathbf{b} = (b_1, \dots, b_d)$ is. Combining with Fact 2.1, we get that the number of indeterminates in Q is at least

$$\frac{(0.01 \cdot mA)^2}{2k} \cdot |\mathcal{B}|.$$

Observe that we actually made things worse by looking at higher derivatives: our interpolation space seemed to have shrunk (superficially at least from the above bound) and we didn't get any useful gain from the constraints. They both got scaled by a factor of $|\mathcal{B}|$ and feels like we might as well have forced $\mathbf{b} = (b_1, \dots, b_d) = \mathbf{0}$ (thus reducing to Guruswami-Sudan).

3.3 The key insight: hidden derivatives

The key insight of [BCPZZ26] is that there is a relation between the evaluation y_i and the higher derivatives given by the following "reverse" Taylor expansion:

$$\begin{aligned} f(X) &= f((X+T) - T) \\ &= f(X+T) + (-T) \cdot f^{(1)}(X+T) + \cdots + (-T)^d f^{(d)}(X+T) \bmod T^{d+1} \\ \therefore f(X+T) &= f(X) + Tf^{(1)}(X+T) + \cdots + (-1)^{d+1} T^d f^{(d)}(X+T) + E \end{aligned}$$

where the error term $E = E(X, T) = 0 \bmod T^{d+1}$.

To illustrate the usefulness of this, let us focus on the case when $d = 1$, namely we introduce one hidden derivative. In this scenario,

$$f(\alpha_i + T) = y_i + Tf^{(1)}(\alpha_i + T) + E$$

where $E = 0 \bmod T^2$. We will see that this change is already enough to go past the Johnson radius.

3.3.1 Breaching the Johnson radius

Let A be the number of agreements we are targeting (and we just want to go below $\sqrt{nk}$ for now). Gabizon [Gab26] studied the $d = 1$ setting of the [BCPZZ26] where one can go past the Johnson radius with a single hidden derivative, and also provided a linear list size in this setting (using ideas from [Jer26]).

Interpolation space: Fix a large constant L . Let us find a polynomial $Q(X, Y_0, Y_1) = \sum q_{a,r,b} X^a Y_0^r Y_1^b$ such that

- $a + kr + kb < mA$,
- $b < Lm$.

The dimension of this space is at least

$$\sum_{b=0}^{Lm-1} \sum_{\substack{a,r \\ a+kr < ma-kb}} 1 \geq Lm \cdot \frac{(mA - kmL)^2}{2k} = \frac{Lm^3}{2k} \cdot (A - kL)^2$$

Constraints: For each $i \in [n]$, consider the polynomial

$$Q_i(T, E, Y_1) = Q(\alpha_i + T, y_i + TY_1 + E, Y_1) = \sum_{a,r,b} q_{i,a,r,b} T^a E^r Y_1^b.$$

We will add a constraint forcing $q_{i,a,r,c} = 0$ for every (a, r, c) satisfying $a + 2r < m$.

Note that for monomials $T^a E^b Y_1^c$ with $a < m$, we must have that $c < Lm + m$ since any contribution of Y_1^ℓ from $(y_i + TY_1 + E)^r$ comes accompanied by T^ℓ as well and the Y_1 degree of Q itself is at most Lm . Thus, the total number of constraints (again using [Fact 2.1](#)) is at most

$$n \cdot \sum_{c=0}^{Lm+m-1} \sum_{a+2r < m} 1 \leq n \cdot m(L+1) \cdot \frac{m^2}{4}$$

Thus, we would have a nonzero Q if the number of constraints is smaller than the number of indeterminates, namely

$$\frac{Lm^3}{2k} \cdot (A - kL)^2 > n \cdot m(L+1) \cdot \frac{m^2}{4}.$$

Thus, suffices to have $(A - kL)^2 > \frac{nk}{2} \cdot (1 + \frac{1}{L})$. Or, equivalently,

$$A > kL + \left(\frac{nk}{2} \cdot \left(1 + \frac{1}{L} \right) \right)^{1/2}.$$

We already see that we are asymptotically approaching $A \gtrsim \sqrt{nk/2}$ by setting L to be a large constant (at least when the rate is small), even with a single hidden derivative. By setting appropriately large number of derivatives, we will be able to further reduce the number of agreements $A = k(1 + \varepsilon)$ for an arbitrarily small ε .

3.4 High-level picture for the general case

Like in the illustrated case above, we begin with $Q(X, Y_0, \dots, Y_d)$ and add linear constraints to try and ensure that

$$Q(\alpha_i + T, y_i + TY_1 - T^2Y_2 + \dots (-1)^{d+1}T^dY_d + E, Y_1, \dots, Y_d) = 0 \bmod T^m$$

while keeping in mind that we expect E to be $0 \bmod T^{d+1}$. In other words, if we expand the above into monomials of the form $T^a E^r Y_1^{c_1} \dots Y_d^{c_d}$, we will expect the coefficient of all monomials with $a + (d+1)r < m$ to be zero. Thus, if $f(y_i) = \alpha_i$ and

$$R_f(\alpha_i + T) = Q(\alpha_i + T, f(\alpha_i + T), \dots, f^{(d)}(\alpha_i + T)),$$

we observe that this is divisible by T^m by substituting $Y_j = f^{(j)}(\alpha_i + T)$ and

$$E = f(\alpha_i + T) - y_i - \sum_{j=1}^d (-1)^{d+1} T^j f^{(j)}(\alpha_i + T)$$

To count the constraints, we need to count the number of monomials of the form $T^a E^r Y_1^{b_1} \dots Y_d^{b_d}$ with $a + (d+1)r < m$. Focusing on the number of possibilities for (a, r) , we get $\frac{m^2}{2(d+1)}$ instead of the naïve bound of $\frac{m^2}{2}$. Hence, we have somehow gotten a saving of a factor of $d+1$.

But we have lost something as well. The set of exponents $(c_1, \dots, c_d)$ for $Y_1, \dots, Y_d$ is now no longer just an element of $\mathcal{B}$ as we have a contribution from the substitution we made. In fact, any $Y_1^{c_1} \dots Y_d^{c_d}$ that we obtain can be written as $Y_1^{b_1} \dots Y_d^{b_d} \cdot Y_1^{b'_1} \dots Y_d^{b'_d}$ where $(b_1, \dots, b_d) \in \mathcal{B}$ and $Y_1^{b'_1} \dots Y_d^{b'_d}$ is the additional degree we obtain from the substitution. However, every monomial of the form $Y_1^{b'_1} \dots Y_d^{b'_d}$ from $(y_i + TY_1 - T^2Y_2 + \dots \pm T^dY_d + E)^r$ comes accompanied by a $T^{\sum ib'_i}$. Since the constraints we are imposing are only on monomials with T -degree less than m , we have that $(c_1, \dots, c_d) \in \mathcal{B} + \Lambda(m)$ where

$$\Lambda(m) := \left\{ (b'_1, \dots, b'_d) \in \mathbb{Z}_{\geq 0}^d : \sum ib'_i \leq m \right\}.$$

Thus, the number of constraints is at most $n \cdot \frac{m^2}{2(d+1)} \cdot |\mathcal{B} + \Lambda(m)|$.

From the counting done in the naive attempt in the previous section, the number of coefficients in Q on the other hand is at least $(0.01mA)^2 |\mathcal{B}| / 2k$, which, for $A = k(1 + \varepsilon)$ is at least $(0.00005(1 + \varepsilon)^2 m^2 k |\mathcal{B}|) = c_\varepsilon m^2 k |\mathcal{B}|$ for some constant c_ε .

Hence, if the factor of $(d+1) = \Omega(d)$ can offset $\frac{1}{c_\varepsilon} \cdot \frac{n|\mathcal{B} + \Lambda(m)|}{k|\mathcal{B}|}$, then, we would have fewer constraints than indeterminates and we can find our polynomial Q .

Getting a sense of the sizes: We are hoping to work with $A = k(1 + \varepsilon)$ for some small constant $\varepsilon > 0$ and hence $mA/k = m(1 + \varepsilon)$. If $\mathcal{B}$ is chosen to be $\mathcal{B}_d := \{(b_1, \dots, b_d) : \sum b_i \leq D := m(1 + \varepsilon)\}$, we have that $|\mathcal{B}_d| \approx \frac{D^d}{d!}$. However, note that $\mathcal{B} + \Lambda(m)$ certainly contains points of the form $\{(a, b_2, \dots, b_d) : 0 \leq a \leq m, b_2 + \dots + b_d \leq D\}$ and this already accounts for about

$$m \cdot \frac{D^{d-1}}{(d-1)!} = \Omega(d \cdot |\mathcal{B}_d|)$$

since $D \approx m$. Therefore, this completely offsets the gain we obtained earlier.

Hence, we need to choose a suitable $\mathcal{B}$ so that $\mathcal{B} + \Lambda(m)$ is not that much larger than $\mathcal{B}$ itself, and turns out a convenient choice is $\mathcal{B} = \Lambda(W)$ for a suitably chosen W . Then, $\mathcal{B} + \Lambda(W) = \Lambda(W) + \Lambda(m) = \Lambda(W + m)$ and

$$\frac{|\mathcal{B} + \Lambda(m)|}{|\mathcal{B}|} = \frac{|\Lambda(W + m)|}{|\Lambda(W)|} \approx \frac{(W + m)^d}{W^d}$$

If $W = \frac{md(1+\delta)}{\ln d}$ for some suitable $0 < \delta < \varepsilon$, the above ratio becomes $d^{1/(1+\delta)}$ which can be offset by the factor d we had obtained above. While lower bounding the dimension of the interpolation space, we had used the rough estimate that $|\mathbf{b}| = \sum b_i$ is bounded from above by $0.99 \cdot \frac{mA}{k}$. This

argument can be made precise if we had a bound on $\mathbb{E}_{\mathbf{b} \in \Lambda(W)}[\sum b_i]$ instead.

Thus, much of the care in the proof is in this choice of W , understanding the size of $\Lambda(W)$ and $\mathbb{E}_{\mathbf{b} \in \Lambda(W)}[\sum b_i]$. We do that in the next section and defer the detailed calculations to the subsequent section.

4 The simplex $\Lambda(W)$

We will mostly be focusing on the discrete set of points $\Lambda(W)$ defined earlier, and its continuous analogue:

$$\begin{aligned}\Lambda(W) &:= \left\{ (b_1, \dots, b_d) \in \mathbb{Z}_{\geq 0}^d : b_1 + 2b_2 + \dots + db_d \leq W \right\}, \\ \lambda(W) &:= \left\{ (b_1, \dots, b_d) \in \mathbb{R}_{\geq 0}^d : b_1 + 2b_2 + \dots + db_d \leq W \right\}.\end{aligned}$$

4.1 Counting points in $\Lambda(W)$

Note that $\lambda(W)$ is just the standard simplex $S := \{(b_1, \dots, b_d) \in \mathbb{R}_{\geq 0}^d : \sum b_i \leq 1\}$ with the i -th direction scaled by $\frac{W}{i}$. It is well-known that $\text{vol}(S) = \frac{1}{d!}$ and this immediately yields that

$$\text{vol}(\lambda(W)) = \frac{W^d}{d!^2}.$$

Turns out, this is also a very good approximation of the number of points in $\Lambda(W)$ and the following lemma makes this precise.

Lemma 4.1 (relating discrete and continuous simplex).

$$\text{vol}(\lambda(W)) \leq |\Lambda(W)| \leq \text{vol}(\lambda(W + d^2)).$$

Proof. Take every point of $\mathbf{b} \in \Lambda(W)$ and consider the unit cube $\mathbf{b} + [0, 1)^d$ (i.e., you may add any real number less than 1 to each of the coordinates). Clearly $\lambda(W)$ is completely contained in the union of these cubes and that gives the inequality on the left. On the other hand, any point in such a cell is a vector of the form $(b'_1, \dots, b'_d)$ with $b'_i < b_i + 1$. Hence, $\sum ib'_i \leq \sum ib_i + \sum i \leq W + d^2$. Hence, the union of all these cells is contained within $\lambda(W + d^2)$. $\square$

Corollary 4.2. If $W = \Omega(d^4)$, then $|\Lambda(W)| = \text{vol}(\lambda(W)) \cdot (1 \pm \frac{d^3}{W})$.

Proof. Follows from the fact that the two sides of the inequality in the previous lemma are within a factor of $\left(1 + \frac{d^2}{W}\right)^d \approx 1 + \frac{d^3}{W}$. $\square$

4.2 Expected value of a coordinate in $\Lambda(W)$

We'll state the following fact we would need (with the proof provided in [Section B](#)) that relates the expected ℓ_1 norm of a vector in $\Lambda(W)$.

Fact 4.3. For $W = \Omega(d^4)$, if $\mathcal{D}$ is the uniform distribution on $\Lambda(W)$, then

$$\mathbb{E}_{\mathbf{b} \sim \mathcal{D}} [b_i] = \frac{W}{i(d+1)} + O_d(1).$$

Consequently, $\mathbb{E}_{\mathbf{b} \sim \mathcal{D}} [\sum b_i] = \frac{W \cdot H_d}{d+1} + O_d(1)$ where $H_d = \sum_{i=1}^d \frac{1}{i} \approx \ln d + \gamma + o(1)$ ($\gamma \approx 0.577..$ is the Euler-Mascheroni constant).

5 Algorithmic list-decoding: completing the proof

We have seen the rough setup, so we will now proceed with the actual calculations. Let $A = k(1 + \varepsilon)$, the number of agreements we are targeting. We will choose a multiplicity parameter $m = d^4$.

Interpolation space ($\mathcal{M}$): We will search for a nonzero polynomial $Q(X, Y_0, \dots, Y_d)$ with support $X^a Y_0^r Y_1^{b_1} \dots Y_d^{b_d}$ with

- $(1, k, k, \dots, k)$ -degree of Q is less than $mA = mk(1 + \varepsilon)$,
- $(b_1, \dots, b_d) \in \Lambda(W)$ where $W = \frac{md(1+\frac{\varepsilon}{2})}{H_d}$

By [Fact 4.3](#), note that the expected value of $b_1 + \dots + b_d$ for this choice of W is at most $m(1 + \frac{\varepsilon}{2})$, which is smaller than $mA/k = m(1 + \varepsilon)$.

Number of indeterminates: For ease of notation, we use x_+ to denote $\max(x, 0)$. If $|\mathbf{b}| = \sum b_i$, then using [Fact 2.1](#) we get that the number of monomials that Q is supported on is at least

$$\begin{aligned} \sum_{\mathbf{b} \in \Lambda(W)} \frac{(mA - k|\mathbf{b}|)_+^2}{2k} &= |\Lambda(W)| \cdot \mathbb{E}_{\mathbf{b} \in \Lambda(W)} \frac{(mA - k|\mathbf{b}|)_+^2}{2k} \\ &\geq |\Lambda(W)| \frac{(mA - k \mathbb{E}_{\mathbf{b} \sim \Lambda(W)} [\sum b_i])_+^2}{2k} && \text{(by Jensen on } x_+^2 \text{)} \\ &\geq |\Lambda(W)| \frac{(mk(1 + \varepsilon) - mk(1 + \frac{\varepsilon}{2}))_+^2}{2k} \\ &= |\Lambda(W)| \cdot \frac{\varepsilon^2}{8} \cdot m^2 k. \end{aligned}$$

The constraints ($\mathcal{C}$): For each $i \in [n]$, consider the polynomial

$$\begin{aligned} Q_i(T, E, Y_1, \dots, Y_d) &= Q(\alpha_i + T, y_i + TY_1 - T^2Y_2 + \dots + (-1)^{d+1}T^dY_d + E, Y_1, \dots, Y_d) \\ &= \sum_{a,r,\mathbf{c}} q_{i,a,r,\mathbf{c}} \cdot T^a E^r Y_1^{c_1} \dots Y_d^{c_d} \end{aligned}$$

where each $q_{i,a,r,\mathbf{c}}$ is a homogeneous linear combination of the coefficients of Q . For each $(a, r, \mathbf{c})$ with $a + (d+1)r < m$, we add the constraint $q_{i,a,r,\mathbf{c}} = 0$.

Any monomial $Y_1^{b'_1} \dots Y_d^{b'_d}$ obtained from $(y_i + TY_1 - T^2Y_2 + \dots + (-1)^{d+1}T^dY_d + E)^r$ comes coupled with $T^{\sum ib'_i}$. The constraints above are only for monomials with T -degree bounded by m . Hence, we have that $c_1 + 2c_2 + \dots + dc_d \leq W + m$ for all such monomials. Hence, the number of constraints we are adding is at most

$$n \cdot \frac{m^2(1+o(1))}{2(d+1)} \cdot |\Lambda(W+m)| \leq n \cdot \frac{m^2}{d} \cdot |\Lambda(W+m)|$$

Therefore, a nonzero Q satisfying the above constraints will exist if the number of indeterminates in Q is more than the number of constraints we impose.

$$\begin{aligned} \frac{\# \text{ indeterminates}}{\# \text{ constraints}} &= \frac{\frac{\varepsilon^2}{8} \cdot m^2 k \cdot |\Lambda(W)|}{n \cdot \frac{m^2}{d} \cdot |\Lambda(W+m)|} \\ &= \frac{\varepsilon^2}{8} \cdot \text{rate} \cdot d \cdot \frac{|\Lambda(W)|}{|\Lambda(W+m)|} \approx \frac{\varepsilon^2}{8} \cdot \text{rate} \cdot d \cdot \frac{W^d}{(W+m)^d} \quad (\because W \gg d^3) \\ &\geq \frac{\varepsilon^2}{8} \cdot \text{rate} \cdot d \cdot \frac{1}{\exp(\frac{md}{W})} = \frac{\varepsilon^2}{8} \cdot \text{rate} \cdot d \cdot \frac{1}{\exp(H_d \cdot \frac{2}{2+\varepsilon})} \\ &\geq \frac{\varepsilon^2}{8} \cdot \text{rate} \cdot d \cdot \frac{1}{\exp((\ln d + \gamma) \cdot \frac{2}{2+\varepsilon})} \\ &\geq \frac{\varepsilon^2}{8} \cdot \text{rate} \cdot d \cdot d^{-\frac{2}{2+\varepsilon}} \cdot e^{-\gamma} \geq \frac{\varepsilon^2}{16} \cdot \text{rate} \cdot d^{\frac{\varepsilon}{2+\varepsilon}}. \end{aligned}$$

Hence, if d is set so that

$$d > \left(\frac{16}{\varepsilon^2 \cdot \text{rate}} \right)^{\frac{2+\varepsilon}{\varepsilon}},$$

we have more indeterminates than constraints and hence a nonzero Q satisfying our constraints exists. The above proof also readily leads to a list-decoding algorithm that computes a nonzero Q by finding a kernel vector of an $|\mathcal{C}| \times |\mathcal{M}|$ matrix where $|\mathcal{M}|, |\mathcal{C}| \leq n^{O_{\varepsilon, \text{rate}}(1)}$, and then solves the differential equation via [Theorem 3.3](#). This completes the proof of [Theorem 1.1](#).

Algorithm 1: List-decoding $\text{RS}_q[n, k]$

Input: Received word $\mathbf{y} = (y_1, \dots, y_n)$ for evaluation points $(\alpha_1, \dots, \alpha_n)$, and $\varepsilon > 0$

Output: All f of degree less than k such that $|\text{agr}(f, \mathbf{y})| \geq k(1 + \varepsilon)$

- 1 Choose parameters $d = \left\lceil \left(\frac{16}{\varepsilon^2 k/n} \right)^{3/\varepsilon} \right\rceil$, $m = d^4$ and $W = \left\lceil \frac{md(1+\frac{\varepsilon}{2})}{H_d} \right\rceil$.
- 2 Set the linear system to find a nonzero $Q(X, Y_0, \dots, Y_d) = \sum q_{a,r,\mathbf{b}} X^a Y_0^r Y_1^{b_1} \dots Y_d^{b_d}$ with $(1, k, k, \dots, k)$ -degree less than $mk(1 + \varepsilon)$ and $\mathbf{b} \in \Lambda(W)$ that satisfy the following constraints: for all $i \in [n]$, all monomials in

$$Q(\alpha_i + T, y_i + TY_1 - T^2 Y_2 + \dots + (-1)^{d+1} T^d Y_d + E, Y_1, \dots, Y_d)$$

of the form $T^a E^r \mathbf{Y}^{\mathbf{b}}$ with $a + (d+1)r < m$ have a coefficient of zero.

- 3 Run the differential equation solver of [Kop15] to find all f of degree less than k such that

$$Q(X, f, f^{(1)}, \dots, f^{(d)}) = 0.$$

- 4 **return** all f 's from the previous line that agree with $\mathbf{y}$ on at least $k(1 + \varepsilon)$ locations.
-

6 Proximity gaps

An important related question is the notion of proximity gaps for these codes. Throughout this section, we shall focus on the instructive case of proximity gaps for lines (i.e., received word of the form $u + Zv$). The extension to more general statements is relatively straightforward.

Problem statement: Suppose we are given two tables $u, v: \mathbb{F}_q \rightarrow \mathbb{F}_q$ such that for “many” values $\zeta \in \mathbb{F}_q$, there table $u + \zeta v$ agrees with some codeword of $\text{RS}_q[n, k]$ on at least A locations (i.e., there is some f of degree less than k such that f and $u + \zeta v$ agree on at least A evaluations among $\{\alpha_1, \dots, \alpha_n\}$). Show that there are polynomials g, h of degree less than k and a set $S \subseteq [n]$ of size at least A such that $g|_S = u|_S$ and $h|_S = v|_S$ (and consequently, $u + \zeta v$ agrees with $g + \zeta h$ on S , for any choice of ζ).

This problem was first studied by Ben-Sasson, Carmon, Ishai, Kopparty, and Saraf [BCIKS23], who proved results when A is at least the unique-decoding radius $(n+k)/2$ or the Johnson radius $\sqrt{nk}$. The open question was then to understand the regime where A is barely above the capacity of the code, i.e., $k(1 + \varepsilon)$ for arbitrarily small ε . The main observation of Jeronimo [Jer26] is that the setup of [BCPZZ26] and the differential equation solver of [Kop15] have enough algebraic structure to derive the above statement even for such small values of A . The overall sketch is quite natural, but the final analysis requires some tools from algebraic geometry. We will first provide the

sketch, and defer the use of algebraic geometry as much as possible (a brief primer on the algebraic geometric concepts used in this exposition is provided in [Section A](#)).

6.1 Overall sketch

Let us fix some $\zeta_0 \in \mathbb{F}_q$ and a polynomial $f_0(X) \in \mathbb{F}_q[X]$ of degree less than k with the property that $|\text{agr}(u + \zeta_0 v, f_0)| \geq A$. Let $S_{\zeta_0} := \text{agr}(u + \zeta_0 v, f_0) = \{i \in [n] : (u + \zeta_0 v)(\alpha_i) = f_0(\alpha_i)\}$. Fix any subset $T \subset S_{\zeta_0}$ of size k and find the unique polynomials $g, h \in \mathbb{F}_q[X]$ of degree less than k such that $g|_T = u|_T$ and $h|_T = v|_T$. Note that this also implies that $f_0 = g + \zeta_0 h$.

Let $S := \text{agr}(g, u) \cap \text{agr}(h, v) = \{i \in [n] : g(\alpha_i) = u(\alpha_i), h(\alpha_i) = v(\alpha_i)\}$. Clearly, $T \subseteq S \subseteq S_{\zeta_0}$ as any correlated agreement of (g, h) with (u, v) will translate to agreement of $g + \zeta_0 h = f_0$ with $u + \zeta_0 v$. If $S = S_{\zeta_0}$, we have the required correlated agreement we are after. Hence, we just need to address the situation that $S \subsetneq S_{\zeta_0}$.

For any $i \in [n]$, let $H_i := \{(\zeta, f) : \deg(f) < k \text{ and } f(\alpha_i) = (u + \zeta v)(\alpha_i)\}$, which is a affine hyperplane in $\mathbb{F}_q^{k+1}$.

Lemma 6.1. *Suppose $i \in S_{\zeta_0} \setminus S$. Then, we have*

$$\bigcap_{j \in T \cup \{i\}} H_j = \{(\zeta_0, f_0)\}.$$

Proof. Note that for any ζ , from our construction of g and h , the unique polynomial of degree less than k agreeing with $u + \zeta v$ on the set T is the polynomial $g + \zeta h$. Observe that since $i \in S_{\zeta_0} \setminus S$, we get that either $u(\alpha_i) \neq g(\alpha_i)$ or $v(\alpha_i) \neq h(\alpha_i)$. Therefore, the polynomial

$$(u(\alpha_i) - g(\alpha_i)) + Z(v(\alpha_i) - h(\alpha_i))$$

is a nonzero linear function in Z and hence there is at most one value of Z that can make this equal to zero, and we already know that it is zero at ζ_0 since $f_0 = g + \zeta_0 h$. Therefore, at all $\zeta \neq \zeta_0$, we have $(u + \zeta v)(\alpha_i) \neq (g + \zeta h)(\alpha_i)$.

Since the size of T is k , for every ζ , the unique f of degree less than k that agrees with $(u + \zeta v)$ at every $\alpha \in T$ is $g + \zeta h$. But when $\zeta \neq \zeta_0$, the polynomial $g + \zeta h$ does not agree with $u + \zeta v$ at α_i . Furthermore, for $\zeta = \zeta_0$, the polynomial $u + \zeta_0 v$ agrees with the degree k polynomial $g + \zeta_0 h$ at α_j for $j \in T \cup \{i\}$, and since the size of T is k , it is the unique such polynomial. $\square$

We want to show that such ζ_0 's (where $S_{\zeta_0} \neq S$) are quite rare, and we will do that by identifying such ζ_0 through different means. This is where the setup of the interpolant system and the decoding algorithm actually enters the picture. Essentially, we can show that every $(\zeta, f) \in \mathbb{F}_q^{k+1}$ such that $|\text{agr}(u + \zeta v, f)| \geq A$ can be covered by a *small* number of algebraic varieties whose dimension is

at most $d + 1$ and degree at most $n^{\text{poly}(d)}$. This is a very general method that is applicable in such templates “algebraic decoders” such as [WB86, GS92, Sud97, GS99, BCPZZ26].

Lemma 6.2 (informal). *There is a set $B \subset \mathbb{F}_q$ of not-too-larger size, and irreducible varieties $\{V_m\}_{m \in [M]}$ (for M not too large) such that $\dim V_m \leq d + 2$ and $\deg V_m \leq \text{poly}(n)$ such that*

$$\{(\zeta, f) : |\text{agr}(u + \zeta v, f)| \geq A\} \subseteq (B \times \mathbb{F}_q^k) \cup \bigcup_{m \in M} V_m.$$

More precisely, the decoding procedure can be used to show that either ζ belongs to an ‘obviously-bad’ set B , or (ζ, f) is in the image of some rational function $\Psi_m : \mathbb{F}_q^{d+2} \rightarrow \mathbb{F}_q^{k+1}$ of not-too-large degree (and V_m is just the smallest algebraic variety containing the image, and it is irreducible).

The following is a standard fact from algebraic geometry.

Fact 6.3 ([Dvi12, Theorem 2.2]). *Let V be an irreducible variety of dimension d and degree D . If H is a hyperplane such that $\emptyset \neq H \cap V \neq V$, then if we write $H \cap V$ as a union of irreducible components $V_1 \cup \dots \cup V_t$, then*

- $\dim V_i = d - 1$ for all i ,
- $\sum \deg V_i \leq D$. In particular, the number of components of $H \cap V$ is at most D .

We are now ready to describe the *pruning process* (which is inspired by the pruning procedure of Kopparty, Ron-Zewi, Saraf and Wootters [KRSW23] and Tamo [Tam24]) to bound the number of ζ_0 such that $S_{\zeta_0} \neq S$. We may assume that $\zeta_0 \notin B$ as B is not-too-large.

The pruning process: Note that $\bigcap_{i \in S_{\zeta_0}} H_i = \{(\zeta_0, f_0)\}$. We build a sequence of irreducible varieties $C_0 \supsetneq C_1 \supsetneq \dots \supsetneq C_t = \{(\zeta_0, f_0)\}$ as follows.

- Let C_0 be the one of the V_m ’s that contain the point (ζ_0, f_0) .
- For each $i > 0$, if $\dim C_i = 0$, we stop. Else, find choose $j_i \in S_{\zeta_0}$ such that $\emptyset \neq C_i \cap H_{j_i} \neq C_i$. Such a hyperplane must exist since $\bigcap_{i \in S_{\zeta_0}} H_i = \{(\zeta_0, f_0)\}$ and we always maintain that $(\zeta_0, f_0) \in C_i$. Among the irreducible components of $C_i \cap H_{j_i}$, let C_{i+1} be one that contains the point (ζ_0, f_0) .

Since the dimension keeps decreasing at each step, we have that $t \leq d + 2$. Furthermore, the above gives a method to index into the point (ζ_0, f_0) by the tuple $(m_0, j_0, m_1, j_1, \dots, j_{t-1}, m_{t-1})$ where $m_0 \in [M]$, each j_i is the index of the hyperplane H_{j_i} chosen at step i , and m_i is the index of C_{i+1} among irreducible components of $C_i \cap H_{j_i}$. The number of such tuples is at most $M \cdot (nD)^{O(d)}$ and thus the number of such ζ_0 ’s is at most $|B| + M(nD)^{O(d)}$.

If we set $L > |B| + M(nD)^{O(d)}$, we obtain Theorem 1.2 which we restate below.

Theorem ((Theorem 1.2 restated)). Let $\rho, \varepsilon > 0$, and let k, n be large enough positive integers with $k/n \geq \rho$. Consider the Reed-Solomon code $\text{RS}_q[n, k]$ over a finite field $\mathbb{F}_q$ (of characteristic p) with $k/n \geq \rho$ and $n \geq k(1 + \varepsilon)$ and $p \geq k$.

Suppose $u, v: E \rightarrow \mathbb{F}_q$ with the property that for at least L values $\zeta \in \mathbb{F}_q$ there exists some degree $< k$ polynomial f_ζ such that $|\text{agr}(u + \zeta v, f_\zeta)| \geq A := k(1 + \varepsilon)$. Then, there are polynomial g, h of degree less than k with correlated agreement of at least A . That is, there is an $S \subseteq [n]$ of size at least A such that

$$|\{i \in [n] : u(\alpha_i) = g(\alpha_i) \text{ and } v(\alpha_i) = h(\alpha_i)\}| \geq k(1 + \varepsilon).$$

Thus, the task boils down to systematically going over the list-decoding setup and Kopparty's differential equation solver to get a bound on $|B|$, D and M .

Remark 6.4. The above pruning already provides a bound for the number of codewords that have a $k(1 + \varepsilon)$ agreement with any received word by a function that depends only on n, k, ε , and independent of p . The same pruning process allows us to bound the number of exceptional ζ 's in the proximity gap setting. $\diamond$

Recalling parameter settings: Recall from the earlier proof that we will be choosing $m = d^4$ for $d = \left(\frac{16}{\varepsilon^2 \cdot \text{rate}}\right)^{\frac{2+\varepsilon}{\varepsilon}}$. Thus, both m and d are constants independent of n .

6.2 Setting the interpolant system of equations over $\mathbb{F}(Z)$

Lemma 6.5 (Computing an interpolant over $\mathbb{F}(Z)$). There is a nonzero polynomial $Q_Z(X, Y_0, \dots, Y_d) \in \mathbb{F}[Z, X, Y_0, \dots, Y_d]$ with $\deg_Z(Q_Z) \leq \Delta = 2^{O(\text{poly}(d))} \cdot \text{poly}(n)$ such that for any ζ and polynomial $f(X)$ such that $|\text{agr}(u + \zeta v, f)| \geq A$, we have

$$Q_\zeta(X, f, f^{(1)}, \dots, f^{(d)}) \equiv 0.$$

Proof. Let us consider the received word as $u + Zv \in (\mathbb{F}[Z])^n$ with $y_i = u(\alpha_i) + Zv(\alpha_i)$ for $i \in [n]$. We follow the same procedure as earlier to now find a polynomial $Q_Z(X, Y_0, \dots, Y_d) = \sum_{m \in \mathcal{M}} q_m \cdot m$ where $\mathcal{M}$ is the support of monomials allowed for Q_Z , and $q_m \in \mathbb{F}[Z]$ are the indeterminates. As earlier, we add constraints expecting $q_{i,a,r,c}$ to be zero for $(i, a, r, c) \in \mathcal{C}$. Since Q_Z has $(1, k, \dots, k)$ -degree at most mA , each $q_{i,a,r,c}$ is a polynomial in Z of degree at most $mA = 2^{O(d)} \cdot n$.

Therefore, we are looking for a kernel of an $|\mathcal{C}| \times |\mathcal{M}|$ matrix, with $|\mathcal{C}| < |\mathcal{M}| = 2^{\text{poly}(d)} \cdot n$ and each of whose entries is a polynomial of degree at most $mA \leq \text{poly}(d) \cdot n$. By Cramer's rule, we can obtain a nonzero polynomial $Q_Z(X, Y_0, \dots, Y_d) \in \mathbb{F}[Z, X, Y_0, \dots, Y_d]$ that is solution to the system of constraints with $\deg_Z Q_Z \leq \Delta := 2^{\text{poly}(d)} \cdot n^2$. By a direct application of Cramer's rule the solution Q might be a rational function in Z . By clearing the denominator, if needed, we can assume without loss of generality that the resulting Q is a polynomial (and not a rational function) in Z . We may also assume without loss of generality that Q_Z is *primitive* in Z i.e., the coefficients of

Q_Z (as polynomials in Z) have no non-trivial common factor.

Now, $Q_\zeta(X, Y_0, \dots, Y_d)$ is a polynomial of $(1, k, \dots, k)$ -degree less than mA . Thus, like earlier, $R_f(X) := Q_\zeta(X, f, f^{(1)}, \dots, f^{(d)})$ has α_i as a root of multiplicity m whenever $f(\alpha_i) = y_i = (u + \zeta v)(\alpha_i)$ and this allows us to conclude that $R_f(X) = Q_\zeta(X, f, f^{(1)}, \dots, f^{(d)}) \equiv 0$. $\square$

Hence, the interpolant $Q(Z, X, Y_0, \dots, Y_d)$ is a polynomial with $\deg_Z Q \leq 2^{\text{poly}(d)} \cdot n^2$, $\deg_X Q < mA \leq \text{poly}(d) \cdot n$ and $\deg_{Y_i} Q \leq \frac{mA}{k} = m(1 + \varepsilon) = \text{poly}(d)$.

6.3 Kopparty's differential equation solver

For a polynomial $P(Z, X, Y_0, \dots, Y_d) \in \mathbb{F}[Z, X, Y_0, \dots, Y_d]$, let $\text{TopVar}(P)$ refer to the highest index $i \in \{0, \dots, d\}$ such that P depends non-trivially on Y_i .

The following lemma is implicit in Kopparty's differential equation solver but we state it formally here. The proof of this is deferred to the Appendix (Section C).

Lemma 6.6 ([Kop15]). *Let $P \in \mathbb{F}[Z, X, Y_0, \dots, Y_d]$ be a nonzero polynomial with $\text{TopVar}(P) = j \leq d$. Assume that $\text{char}(\mathbb{F}) > \max(k, \deg_{Y_i}(P))$ for all i . There are rational functions $\Phi_{P,i}(Z, X, Y_0, \dots, Y_d)$ defined for $0 \leq i < k$ as*

$$\Phi_{P,i}(Z, X, Y_0, \dots, Y_d) = \begin{cases} Y_i & \text{if } 0 \leq i \leq d, \\ \frac{N_{P,i}(Z, X, Y_0, \dots, Y_d)}{(\partial_{Y_j} P)^{2e}} & \text{if } d < i < k, \end{cases}$$

with $\deg N_{P,i} \leq 2(i - j) \cdot \deg P$ and $e \leq 2(i - j)$ with the following property:

Suppose $\zeta \in \mathbb{F}$ is such that $\text{TopVar}(P) = \text{TopVar}(P|_{Z=\zeta})$, $f(X)$ is a polynomial of degree less than k , and $\gamma \in \mathbb{F}$ with $a_i = f^{(i)}(\gamma)$ for $0 < i < k$ such that

- $P(\zeta, X, f, f^{(1)}, \dots, f^{(d)}) = 0$,
- $(\partial_{Y_j} P)(\zeta, \gamma, a_0, \dots, a_d) \neq 0$.

Then, for all $0 \leq i < k$,

$$a_i = \Phi_{P,i}(\zeta, \gamma, a_0, \dots, a_d).$$

Consequently, $f(X) = \sum_{i=0}^{k-1} \Phi_{P,i}(a_0, \dots, a_d) \cdot (X - \gamma)^i$.

Intuitively, assuming some initial conditions on ζ, γ with respect to f , the first d terms of the Taylor expansion of f around the point γ is sufficient to reconstruct f entirely, and in a very explicit sense given by the rational functions $\Phi_{P,i}$.

From the above $\Phi_{P,i}$, let us define a rational map $\Psi_{P,\gamma}$ as follows

$$\Psi_{P,\gamma}: (\zeta, a_0, \dots, a_d) \mapsto (\zeta, f_0, \dots, f_{k-1})$$

$$\text{where } f(X) = \sum_{i=0}^{k-1} f_i X^i = \sum_{i=0}^{k-1} \Phi_{P,i}(\zeta, \gamma, a_0, \dots, a_d) \cdot (X - \gamma)^i.$$

The above Ψ is defined at all points in $\overline{\mathbb{F}}^{d+2}$ that ensures that the denominator $(\partial_{Y_j} P)(\zeta, \gamma, a_0, \dots, a_d) \neq 0$.

Let us fix a polynomial $Q_Z(X, Y_0, \dots, Y_d)$ obtained from [Lemma 6.5](#), and a set $T \subseteq \mathbb{F}_p$ of at least mA elements. Let $\partial_Y^{\overline{*}}(Q_Z)$ refer to the set of all polynomials that are derivatives of Q_Z with respect to Y variables of any order. Define the set B of “obviously-bad” ζ ’s as

$$B = \left\{ \zeta : \begin{array}{l} \text{TopVar}(P(Z, X, Y_0, \dots, Y_d)) \neq \text{TopVar}(P(\zeta, X, Y_0, \dots, Y_d)) \\ \text{for some non-constant } P(Z, X, Y_0, \dots, Y_d) \in \partial_Y^{\overline{*}}(Q_Z) \end{array} \right\}.$$

Claim 6.7. Fix any set $T \subseteq \mathbb{F}$ of size at least mA . Suppose $\zeta \in \mathbb{F} \setminus B$ and $f(X)$ is a polynomial of degree less than k such that we have $Q_Z(X, f, f^{(1)}, \dots, f^{(d)}) = 0$. Then, (ζ, f) is in the image of $\Psi_{P,\gamma}$ for some $P \in \partial_Y^{\overline{*}}(Q)$ and $\gamma \in T$.

Proof. Let $P(Z, X, Y_0, \dots, Y_d)$ be the minimal degree non-zero polynomial in $\partial_Y^{\overline{*}}(Q_Z)$ with the property that $P(\zeta, X, f, f^{(1)}, \dots, f^{(d)}) = 0$. From the minimality, we must have that $P'(X) = \partial_{Y_j}(P)(\zeta, X, f, f^{(1)}, \dots, f^{(d)}) \neq 0$ where $j = \text{TopVar}(P) = \text{TopVar}(P_\zeta)$ (since $\zeta \notin B$). Since $P'(X)$ is a polynomial of degree less than $mA = |T|$, there is some $\gamma \in T$ such that $P'(\gamma) \neq 0$. Let $a_i = f^{(i)}(\gamma)$ for all $0 \leq i < k$. By [Lemma 6.6](#), we have that $\Psi_{P,\gamma}(\zeta, a_0, \dots, a_j) = (\zeta, f)$. $\square$

We need the following fact from algebraic geometry, which appears to be a standard consequence of Bézout’s theorem. We state it here without proof, but provide a rough proof-sketch in [Section A](#).

Fact 6.8 (image of a rational map on an open set (cf. [\[Jer26, Lemma 5.4\]](#))). Let $\overline{\mathbb{F}}$ be the algebraic closure of $\mathbb{F}$. Suppose $\Psi(X_1, \dots, X_d)$ be a rational map of the form

$$\Psi = \left(\frac{N_1}{G_1}, \dots, \frac{N_n}{G_n} \right)$$

where $N_i, G_i \in \overline{\mathbb{F}}[X_1, \dots, X_d]$ with $\deg N_i, \deg G_i \leq \Delta$, that is well-defined on all points of $\overline{\mathbb{F}}^d$ where $G_1 \cdots G_n$ does not vanish. Then, the Zariski closure of $\text{Image}(\Psi)$ is an algebraic variety V such that

- V is irreducible,
- $\dim V \leq d$,
- $\deg V \leq (n\Delta)^d$.

We now have the formal version of [Lemma 6.2](#)

Corollary 6.9 (formal version of [Lemma 6.2](#)). *Let B be the set defined above, and $T \subseteq \mathbb{F}$ of at least mA elements. Then,*

$$\{(\zeta, f) : |\text{agr}(u + \zeta v, f)| \geq A\} \subseteq (B \times \mathbb{F}^k) \cup \bigcup_{\substack{P \in \partial_Y^{\neq}(Q) \\ \gamma \in T}} V_{P, \gamma}$$

where $V_{P, \gamma} = \overline{\text{Image}(\Psi_{P, \gamma})}$ defined above. Furthermore, each $V_{P, \gamma}$ is a variety of dimension at most $d + 2$ and degree at most $2^{\text{poly}(d)} \cdot n^{O(d)}$.

6.4 Bounding the final bad set of ζ 's

We now just need to plug in the bounds for $|B|$, M and D in [Lemma 6.2](#) to compute L for [Theorem 1.2](#).

Recall

$$B = \left\{ \zeta : \begin{array}{l} \text{TopVar}(P(Z, X, Y_0, \dots, Y_d)) \neq \text{TopVar}(P(\zeta, X, Y_0, \dots, Y_d)) \\ \text{for some non-constant } P(Z, X, Y_0, \dots, Y_d) \in \partial_Y^{\neq}(Q_Z) \end{array} \right\}.$$

Since Q is a polynomial of Z -degree at most $n^2 \cdot 2^{O(\text{poly}(d))}$, and Y -degree at most $\text{poly}(d)$ there are at most $2^{\text{poly}(d)}$ elements in $\partial_Y^{\neq}(Q)$. Thus, the size of the set B is at most $n^2 \cdot 2^{\text{poly}(d)}$.

From [Corollary 6.9](#), each (ζ, f) we are interested in with $\zeta \notin B$ are covered by one of the varieties $V_{P, \gamma}$. Thus, we have at most $M = |\partial_Y^{\neq}(Q_Z)| \cdot |T| = n \cdot 2^{\text{poly}(d)}$ varieties of dimension $O(d)$ and degree at most $D = 2^{\text{poly}(d)} \cdot n^{O(d)}$ each. Thus, if $p \gg n^{O(d^2)} \cdot 2^{\text{poly}(d)}$, then even a small fraction of ζ 's that witness A agreements to $\text{RS}[n, k]$ result in a correlated set of A agreements. This complete the proof of [Theorem 1.2](#).

References

- [BCIKS23] ELI BEN-SASSON, DAN CARMON, YUVAL ISHAI, SWASTIK KOPPARTY, and SHUBHANGI SARAF. *Proximity gaps for Reed-Solomon codes*. J. ACM, 70(5):31:1–31:57, 2023. (Preliminary version in 61st FOCS, 2020). [eccc:2020/083](#), [eprint.iacr:2020/654](#).
- [BCPZZ26] JOSHUA BRAKENSIEK, YEYUAN CHEN, AARON PUTTERMAN, ZIHAN ZHANG, and KAI ZHE ZHENG. *Algorithmic list decoding of Reed-Solomon codes up to capacity*, 2026. (manuscript). [arXiv:2609.08005](#), [eccc:2026/164](#).
- [Dvi12] ZEEV DVIR. *Extractors for varieties*. Comput. Complexity, 21(4):515–572, 2012. (Preliminary version in 24th CCC, 2009).
- [Gab26] ARIEL GABIZON. *Linear list size bounds for Reed-Solomon beyond the Johnson radius*, 2026. (manuscript). [eprint.iacr:2026/2048](#).

- [GR08] VENKATESAN GURUSWAMI and ATRI RUDRA. *Explicit codes achieving list decoding capacity: Error-correction with optimal redundancy*. IEEE Trans. Inform. Theory, 54(1):135–150, 2008. (Preliminary version in 38th STOC, 2006). [arXiv:cs/0511072](#), [eccc:2005/133](#).
- [GS92] PETER GEMMELL and MADHU SUDAN. *Highly resilient correctors for polynomials*. Inform. Process. Lett., 43(4):169–174, 1992.
- [GS99] VENKATESAN GURUSWAMI and MADHU SUDAN. *Improved decoding of Reed-Solomon and algebraic-geometry codes*. IEEE Trans. Inform. Theory, 45(6):1757–1767, 1999. (Preliminary version in 39th FOCS, 1998). [eccc:1998/043](#).
- [GW13] VENKATESAN GURUSWAMI and CAROL WANG. *Linear-algebraic list decoding for variants of Reed-Solomon codes*. IEEE Trans. Inform. Theory, 59(6):3257–3268, 2013. (Preliminary version in 26th IEEE Conference on Computational Complexity, 2011 and 15th RANDOM, 2011). [eccc:2012/073](#).
- [Jer26] FERNANDO GRANHA JERONIMO. *Algorithmic list decoding at capacity and optimal proximity gaps for Reed-Solomon codes*, 2026. (manuscript). [arXiv:2609.05870](#), [eccc:2026/169](#).
- [Kop15] SWASTIK KOPPARTY. *List-decoding multiplicity codes*. Theory of Computing, 11:149–182, 2015. [eccc:2012/044](#).
- [KRSW23] SWASTIK KOPPARTY, NOGA RON-ZEWI, SHUBHANGI SARAF, and MARY WOOTTERS. *Improved list decoding of Folded Reed-Solomon and Multiplicity codes*. SIAM J. Comput., 52(3):794–840, 2023. (Preliminary version in 59th FOCS, 2018). [arXiv:1805.01498](#), [eccc:2018/091](#).
- [PV05] FARZAD PARVARESH and ALEXANDER VARDY. *Correcting errors beyond the Guruswami-Sudan radius in polynomial time*. In ÉVA TARDOS, ed., *Proc. 46th IEEE Symp. on Foundations of Comp. Science (FOCS)*, pages 285–294. 2005.
- [RS60] IRVING S. REED and GUSTAVE SOLOMON. *Polynomial codes over certain finite fields*. J. Soc. Indust. Appl. Math., 8(2):300–304, 1960.
- [Sud97] MADHU SUDAN. *Decoding of Reed-Solomon codes beyond the error-correction bound*. J. Complexity, 13(1):180–193, 1997. (Preliminary version in 37th FOCS, 1996).
- [Tam24] ITZHAK TAMO. *Tighter list-size bounds for list-decoding and recovery of folded Reed-Solomon and multiplicity codes*. IEEE Trans. Inform. Theory, 70(12):8659–8668, 2024. [arXiv:2312.17097](#).
- [WB86] LLOYD R. WELCH and ELWYN R. BERLEKAMP. *Error correction of algebraic block codes*, December 1986. U.S. Patent 4 633 470.

A A brief primer on algebraic geometry

We use this section to provide a very basic primer on algebraic geometry. The primary goal is not to provide an exhaustive, formal treatment, but rather to communicate the intuitive meaning of the concepts that would help digest the facts from algebraic geometry used in this write-up.

Let $\mathbb{F}$ be an algebraically closed field.

Definition A.1 (Algebraic varieties: irreducibility, dimension and degree). A subset V of $\mathbb{F}^n$ is said to be an algebraic variety if there is a finite set of polynomials $f_1, \dots, f_m \in \mathbb{F}[x_1, \dots, x_n]$ such that

$$V = \mathbb{V}(f_1, \dots, f_m) := \{\mathbf{a} \in \mathbb{F}^n : f_i(\mathbf{a}) = 0 \text{ for all } i \in [m]\}.$$

A variety is said to be irreducible if it cannot be written as non-trivial finite union of two varieties. In other words, if $V = V_1 \cup V_2$ for two algebraic varieties V_1, V_2 , then either $V_1 = V$ or $V_2 = V$.

The dimension of the variety V (denoted by $\dim V$) is defined as the smallest r such that a generic affine subspace of co-dimension r (or dimension, as standard vector space dimension, $n - r$) intersects V at a finitely many points. The degree of the variety (denoted by $\deg V$) is the finite number that results from such a generic intersection of a co-dimension r affine subspace. $\diamond$

Since algebraic varieties are closed under finite unions and arbitrary intersections, we can define a topology on $\mathbb{F}^n$ by setting algebraic varieties as the *closed sets*. The set $\mathbb{F}^n$ equipped with this topology (called the Zariski topology) is often referred to as $\mathbb{A}^n$.

Under this topology, polynomial maps from $\mathbb{A}^d$ to $\mathbb{A}^n$ are continuous maps, and rational functions are continuous where defined (when the denominators don't vanish). The image of an irreducible space under a continuous map is always irreducible, and thus the image of rational maps (and thereby its closure too) are always irreducible.

Most quantitative bounds regarding degree of varieties follow from an application of Bézout's theorem. The following is the form that is most useful to us.

Theorem A.2 (Bézout's theorem). Suppose $V = \mathbb{V}(f_1, \dots, f_m) \subseteq \mathbb{F}^n$, with $\deg f_i = d_i$, then the sum of the degrees of its irreducible components is at most $\prod d_i$.

We now give an informal proof of [Fact 6.8](#) (with a minor modification of making all denominators equal).

Fact A.3 (Image of a rational map on an open set). Suppose $\Psi(X_1, \dots, X_d) : \mathbb{A}^d \setminus U \rightarrow \mathbb{A}^n$ be a rational map of the form

$$\Psi = \left(\frac{N_1}{G}, \dots, \frac{N_n}{G} \right)$$

where $N_i, G \in \mathbb{F}[X_1, \dots, X_d]$ with $\deg N_i, \deg G \leq \Delta$ and $U = \mathbb{V}(G)$. Then, the Zariski closure of $\text{Image}(\Psi)$ is an algebraic variety V such that

- V is irreducible,
- $\dim V \leq d$,
- $\deg V \leq \Delta^d$.

Proof sketch. The set $\mathbb{A}^d \setminus U$ is irreducible, and hence its image under Ψ (a continuous map) is also irreducible. Hence, the closure of the image is an irreducible algebraic variety. Since Ψ maps d coordinates into the space $\mathbb{A}^n$, we have $\dim V$ is at most d .

For the final point, let us assume for simplicity that $\dim V = d$ (otherwise, we have to work with a *generic* $(\dim V)$ -dimensional subspace of $\mathbb{A}^d$ and repeat the argument). Let us take a generic affine subspace $L = \mathbb{V}(\ell_1, \dots, \ell_d)$ of co-dimension d , where $\ell_i = a_{i0} + \sum_{j=1}^n a_{ij}y_j$, and intersect V with it. For $i = 1, \dots, d$, define

$$P_i(x_1, \dots, x_d) = c_{i0}G(\mathbf{x}) + \sum_{j=1}^n c_{ij}N_j(\mathbf{x})$$

and $Z = \mathbb{V}(P_1, \dots, P_d)$, and note $\deg P_i \leq \Delta$ for all i . Since $\dim V = d$ and the ℓ_i 's were generic, turns out it can be argued that each constraint $P_i = 0$ reduces the dimension of the space by one, and hence Z will be finite. [Theorem A.2](#) shows that $|Z| \leq \Delta^d$. Each point of $V \cap L$ is certainly in $\Psi(Z)$. Therefore,

$$\deg V = |V \cap L| \leq |Z| \leq \Delta^n$$

□

B Deferred proofs

Proof of [Fact 4.3](#). Consider the continuous relaxation $\lambda(W) = \{(b_1, \dots, b_d) \in \mathbb{R}_{\geq 0}^d : \sum ib_i \leq W\}$. Under the transformation $a_i = i \cdot b_i$, then $(a_1, \dots, a_d)$ is an element of the standard simplex $S(W) = \{(a_1, \dots, a_d) : \sum a_i \leq W\}$. Let us add a slack variable a_0 so that $a_0 + \dots + a_d = W$ and consider the set $\tilde{S}(W) = \{(a_0, \dots, a_d) : \sum a_i = W\}$.

Clearly, $\tilde{S}(W)$ is symmetric over all the coordinates and hence $\mathbb{E}_{\mathbf{a} \in \tilde{S}(W)}[a_i] = \frac{W}{d+1}$ for any i . Since the transformation from a_i 's to b_i 's is a coordinate-wise scaling, we get that

$$\mathbb{E}_{\mathbf{b} \in \lambda(W)}[b_i] = \mathbb{E}_{\mathbf{a} \in \tilde{S}(W)}\left[\frac{a_i}{i}\right] = \frac{W}{i \cdot (d+1)}.$$

Once again, we wish to relate the expectation over $\lambda(W)$ to the expectation over discrete set $\Lambda(W)$. As in [Lemma 4.1](#), let us add a half-open cube $[0, 1)^d$ to each point of $\Lambda(W)$. For the cube $(a_1, \dots, a_d) + [0, 1)^d$, the expectation of the i -th coordinate is just $a_i + \frac{1}{2}$. Hence, if $\Gamma = \bigcup_{\mathbf{b} \in \Lambda(W)} (\mathbf{b} + [0, 1)^d)$, then $\lambda(W) \subseteq \Gamma \subseteq \lambda(W + d^2)$. Hence,

$$\begin{aligned} \int_{\lambda(W)} x_i dx &\leq \int_{\Gamma} x_i dx \leq \int_{\lambda(W + d^2)} x_i dx \\ \text{i.e., } \text{vol}(\lambda(W)) \cdot \mathbb{E}_{\mathbf{b} \in \lambda(W)}[b_i] &\leq \text{vol}(\Gamma) \cdot \mathbb{E}_{\mathbf{b} \in \Gamma}[b_i] \leq \text{vol}(\lambda(W + d^2)) \cdot \mathbb{E}_{\mathbf{b} \in \lambda(W + d^2)}[b_i] \end{aligned}$$

$$\implies \text{vol}(\lambda(W)) \cdot \frac{W}{(d+1)i} \leq \left(\sum_{\mathbf{b} \in \Lambda(W)} b_i \right) + \frac{1}{2} |\Lambda(W)| \leq \text{vol}(\lambda(W + d^2)) \cdot \frac{(W + d^2)}{(d+1)i}.$$

Dividing throughout by $|\Lambda(W)|$, we get

$$\begin{aligned} \frac{W}{(d+1) \cdot i} \cdot (1 - O(\frac{d^3}{W})) &\leq \mathbb{E}_{\mathbf{b} \in \Lambda(W)} [b_i] + \frac{1}{2} \leq \frac{W + d^2}{(d+1) \cdot i} \cdot (1 + O(\frac{d^3}{W})) \\ \implies \mathbb{E}_{\mathbf{b} \in \Lambda(W)} [b_1 + \dots + b_d] &= \frac{W}{(d+1)} \cdot \sum_{i=1}^d \frac{1}{i} \pm O_d(1) = \frac{W \cdot H_d}{(d+1)} \pm O_d(1) \end{aligned}$$

where $H_d = \sum_{i=1}^d \frac{1}{i} = \ln d + \gamma + o(1)$. □

C Degree bounds for Kopparty's reconstruction algorithm

Lemma C.1 (Lemma 6.6 restated). *Let $P \in \mathbb{F}[Z, X, Y_0, \dots, Y_d]$ be a nonzero polynomial with $\text{TopVar}(P) = j \leq d$. Assume that $\text{char}(\mathbb{F}) > \max(k, \deg_{Y_i}(P))$ for all i . There are rational functions $\Phi_{P,i}(Z, X, Y_0, \dots, Y_d)$ defined for $0 \leq i < k$ as*

$$\Phi_{P,i}(Z, X, Y_0, \dots, Y_d) = \begin{cases} Y_i & \text{if } 0 \leq i \leq d, \\ \frac{N_{P,i}(Z, X, Y_0, \dots, Y_d)}{(\partial_{Y_j} P)^{2e}} & \text{if } d < i < k, \end{cases}$$

with $\deg N_{P,i} \leq 2(i - j) \cdot \deg P$ and $e \leq 2(i - j)$ with the following property:

Suppose $\zeta \in \mathbb{F}$ is such that $\text{TopVar}(P) = \text{TopVar}(P_{Z=\zeta})$, $f(X)$ is a polynomial of degree less than k , and $\gamma \in \mathbb{F}$ with $a_i = f^{(i)}(\gamma)$ for $0 < i < k$ such that

- $P(\zeta, X, f, f^{(1)}, \dots, f^{(d)}) = 0$,
- $(\partial_{Y_j} P)(\zeta, \gamma, a_0, \dots, a_d) \neq 0$.

Then, for all $0 \leq i < k$,

$$a_i = \Phi_{P,i}(\zeta, \gamma, a_0, \dots, a_d).$$

Consequently, $f(X) = \sum_{i=0}^{k-1} \Phi_{P,i}(a_0, \dots, a_d) \cdot (X - \gamma)^i$.

The crucial content of this lemma is in the fact that it gives a polynomial upper bound on the degrees of the numerator and denominator of $\Phi_{P,i}$. Such a bound is not entirely immediate from the reconstruction algorithm in [Kop15] since the reconstruction involves repeated composition of P with the partial solution constructed so far (monomials of f of low degree). Naively, this gives the impression that the degree of $\Phi_{P,i}$ is growing superpolynomially in i . We observe, using a slight modification of the reconstruction procedure, that this is not the case.

The rest of this section would be a proof of the above lemma. This is morally the same argument in Kopparty's paper but is a little more convenient to obtain the necessary degree bounds. For simplicity, let us assume that $\text{TopVar}(P) = \text{TopVar}(P_\zeta) = d$, where $P_\zeta = P(\zeta, X, Y_0, \dots, Y_d)$. We have fixed some $\gamma \in T$ such that $(\partial_{Y_d} P_\zeta)(\gamma, a_0, \dots, a_d) \neq 0$, where $a_i = f^{(i)}(\gamma)$.

Expressing a_{d+1} in terms of $a_0, \dots, a_d$: Note that for any i we have $\frac{d}{dX} f^{(i)} = (i+1)f^{(i+1)}$. Since $Q(X, f, f^{(1)}, \dots, f^{(d)}) = 0$, we have that

$$\frac{d}{dX} P_\zeta(X, f, f^{(1)}, \dots, f^{(d)}) = 0$$

as well. Let $\Delta(f)(X) := (f, f^{(1)}, f^{(2)}, \dots, f^{(d)})$.

$$\begin{aligned} 0 &= \frac{d}{dX} P_\zeta(X, f, f^{(1)}, \dots, f^{(d)}) \\ &= \partial_X P_\zeta(X, \Delta(f)) + \sum_{i=0}^{d-1} (i+1) \cdot \partial_{Y_i} (P_\zeta)(X, \Delta(f)) \cdot f^{(i+1)} \\ &\quad + (d+1) f^{(d+1)} \cdot \partial_{Y_d} P_\zeta(X, \Delta(f)) \\ \implies f^{(d+1)} &= \frac{-\left(\partial_X P_\zeta(X, \Delta(f)) + \sum_{i=0}^{d-1} (i+1) \cdot \partial_{Y_i} (P_\zeta)(X, \Delta(f)) \cdot f^{(i+1)}\right)}{(d+1) \cdot \partial_{Y_d} P_\zeta(X, \Delta(f))} \end{aligned}$$

Define the rational function $\Gamma_P(Z, X, Y_0, \dots, Y_d)$ as

$$\Gamma_P(Z, X, Y_0, \dots, Y_d) = \frac{-\left(\partial_X P + \sum_{i=0}^{d-1} (i+1) \cdot \partial_{Y_i} P \cdot Y_{i+1}\right)}{(d+1) \cdot \partial_{Y_d} P}$$

so that $f^{(d+1)} := \Gamma_P(Z, X, \Delta(f))$.

Recall that we are trying to recover f via its Taylor expansion around the point γ . Formally, $f(X) = \sum_i a_i (X - \gamma)^i$ and the goal is to obtain expressions for $a_{d+1}, \dots, a_{k-1}$ as functions of $a_0, a_1, \dots, a_d$ and Z . Thus given the formal expression for $f^{(d+1)}$ as above ($f^{(d+1)} = \Gamma_P(Z, X, \Delta(f))$), we can compute a_{d+1} by setting X to γ in this expression (while making sure that this is well defined) and scaling the value obtained appropriately. The idea of the recovery algorithm here is to iterate on this: given the expression for $f^{(d+1)}$, we differentiate it ℓ times with respect to X to obtain a rational expression for $f^{(d+1+\ell)}$ and then set X to γ in this expression to obtain $a_{d+1+\ell}$. The goal is to understand the bounds on the degree of the formal rational expressions for $f^{(d+1+\ell)}$ as ℓ increases, and this is done via a simple induction. If we differentiate the expression for $f^{(d+1)}$ obtained above, via an application of chain rule and product rule of derivatives, the expression on the right hand side depends upon $f, f^{(1)}, \dots, f^{(d)}$, various derivatives of P_ζ and $f^{(d+1)}$. In this expression, we substitute the rational expression for $f^{(d+1)}$ obtained here so that the resulting

expression only depends on $f, f^{(1)}, \dots, f^{(d)}$ and derivatives of P_ζ . Staring at the form of this rational expression immediately gives us a sense of the increase in degree of the rational expression as we iterate this process. The rest of this section works out the formal details of this strategy.

Building a derivation operator: We will build a suitable “derivation operator” to help capture the structure of $\frac{d}{dX}h(Z, X, \Delta(f))$ for arbitrary $h(Z, X, Y_0, \dots, Y_d)$.

$$\mathcal{D} := \partial_X + \sum_{i=0}^{d-1} (i+1) \cdot Y_{i+1} \cdot \partial_{Y_i} + (d+1) \cdot \Gamma_P \cdot \partial_{Y_d}$$

Observation C.2. For any $h(Z, X, Y_0, Y_1, \dots, Y_d)$, we have

$$\frac{d}{dX}(h(Z, X, \Delta(f))) = (\mathcal{D}h)(Z, X, \Delta(f)).$$

More generally,

$$\frac{d^t}{dX^t}(h(Z, X, \Delta(f))) = (\mathcal{D}^t h)(Z, X, \Delta(f)).$$

Proof. By the chain rule,

$$\begin{aligned} \frac{d}{dX}(h(Z, X, \Delta(f))) &= (\partial_X h)(Z, X, \Delta(f)) + \sum_{i=0}^d (\partial_{Y_i} h)(Z, X, \Delta(f)) \cdot \frac{d}{dX} f^{(i)} \\ &= (\partial_X h)(Z, X, \Delta(f)) + \sum_{i=0}^d (\partial_{Y_i} h)(Z, X, \Delta(f)) \cdot (i+1) f^{(i+1)} \\ &= (\partial_X h)(Z, X, \Delta(f)) + \sum_{i=0}^{d-1} (\partial_{Y_i} h)(Z, X, \Delta(f)) \cdot (i+1) f^{(i+1)} \\ &\quad + (\partial_{Y_d} h)(Z, X, \Delta(f)) \cdot (d+1) f^{(d+1)} \\ &= (\partial_X h)(Z, X, \Delta(f)) + \sum_{i=0}^{d-1} (\partial_{Y_i} h)(Z, X, \Delta(f)) \cdot (i+1) f^{(i+1)} \\ &\quad + (\partial_{Y_d} h)(Z, X, \Delta(f)) \cdot (d+1) \cdot \Gamma_P(Z, X, \Delta(f)) \\ &= (\mathcal{D}h)(Z, X, \Delta(f)). \end{aligned}$$

The statement for higher derivatives follow by iterating the above t times. □

Proposition C.3 (Basic properties of the operator $\mathcal{D}$). For any $g, h \in \mathbb{F}(Z, X, Y_0, \dots, Y_d)$, the following properties hold:

- $\mathcal{D}(g+h) = \mathcal{D}g + \mathcal{D}h$,
- $\mathcal{D}(gh) = h\mathcal{D}g + g\mathcal{D}h$,

- If $h \in \mathbb{F}[Z, X, Y_0, \dots, Y_d]$ of degree at most Δ , then $\mathcal{D}h$ is a rational function whose numerator has degree $\Delta + \deg P$ with denominator being a scalar multiple of $P' = \partial_{Y_d}(P)$.
- More generally, for any $h \in \mathbb{F}[Z, X, Y_0, \dots, Y_d]$, and any integer $e \geq 0$

$$\mathcal{D}\left(\frac{h}{(P')^e}\right) = \frac{g}{(P')^{e+2}}$$

for some polynomial $g \in \mathbb{F}[Z, X, Y_0, \dots, Y_d]$ with $\deg g \leq \deg h + 2 \deg P'$.

Proof. The first two points follow readily from the definition of $\mathcal{D}$. The degree bounds also hold since the only introduction of denominators is from the $\Gamma_P \cdot \partial_{Y_d}$ term in $\mathcal{D}$ and the denominator is just $(d+1)P'$. For the last point,

$$\begin{aligned} \mathcal{D}\left(\frac{h}{(P')^e}\right) &= \frac{\mathcal{D}(h)}{(P')^e} - \frac{e \cdot h \cdot \mathcal{D}(P')}{(P')^{e+1}} \\ &= \frac{P' \cdot \tilde{h} + e \cdot h \cdot \tilde{P}}{(P')^{e+2}} \\ \text{where } \mathcal{D}(P') &= \frac{\tilde{P}}{P'} \text{ and } \mathcal{D}(h) = \frac{\tilde{h}}{P'}. \end{aligned}$$

Then $g = P' \cdot \tilde{h} + e \cdot h \cdot \tilde{P}$, which is a polynomial of degree at most $\deg(h) + 2 \deg(P')$. $\square$

With the above derivation operator, we can obtain the rational functions computing a_{d+t}' 's via repeated application of $\mathcal{D}$ to the function $h(Z, X, Y_0, \dots, Y_d) := Y_d$.

$$\begin{aligned} (\mathcal{D}^t h)(Z, X, f, f^{(1)}, \dots, f^{(d)}) &= \frac{d^t}{dX^t} (h(Z, X, f, f^{(1)}, \dots, f^{(d)})) = \frac{d^t}{dX^t} f^{(d)} \\ &= \frac{(d+t)!}{d!} \cdot f^{(d+t)} \end{aligned}$$

Evaluating the above expression for $f^{(d+t)}$ at $X = \gamma$, we get

$$a_{d+t} = f^{(d+t)}(\gamma) = \frac{d!}{(d+t)!} \cdot (\mathcal{D}^t Y_d)(\zeta, \gamma, a_0, \dots, a_d).$$

The rational function $\Phi_{P,d+t}(Z, X, Y_0, \dots, Y_d)$ is just $\frac{d!}{(d+t)!} \cdot (\mathcal{D}^t Y_d)$. Since $\deg Y_d = 1$, repeated applications of [Proposition C.3](#) gives that

$$\mathcal{D}^t Y_d = \frac{N_{P,d+t}}{(P')^e}$$

where $\deg N_{P,d+t} \leq 1 + (2t-1) \deg P' \leq 2t \deg P$ and $e \leq 2t-1$