Khatam: Reducing the Communication Complexity of Code-Based SNARKs

Hadas Zeilberger

Yale University

November 13, 2024

Abstract

We prove that Basefold(Crypto 2024) is secure in the *list decoding regime*, within the double Johnson bound and with error probability $\frac{O(n)}{|\mathbb{F}|}$. At the heart of this proof is a new, stronger statement for *correlated agreement*, which roughly states that if a linear combination of vectors $\pi_L + r\pi_R$ agrees with a codeword at every element in $S \subset [n]$, then so do π_L, π_R. Our result is purely combinatorial and therefore extends to any finite field and any linear code. As such, it can be applied to any coding-based multilinear Polynomial Commitment Scheme to reduce its communication complexity.

1 Introduction

In recent years, error-correcting codes have proven indispensable to the construction of efficient SNARKs. A prover of a code-based SNARK commits to its witness by encoding it with a linear error-correcting code, which uses (relatively) cheap operations such as finite-field addition and multiplication. The verifier can test the proximity of the prover's codeword (to the error-correcting code) by engaging with the prover in an equally cheap *Interactive Oracle Proof of Proximity* [22, 5]. Then, with nothing more than a collision resistant hash function, we obtain a *Polynomial Commitment Scheme* [18] (PCS), in which a prover commits to a polynomial $P \in \mathbb{F}[X]$ so that it can later prove evaluation claims of the form $P(\alpha) = \beta$. Finally, a PCS compiles a *Polynomial Interactive Oracle Proof* (PIOP) into a SNARK. We refer to the following works for more details on this transformation ([6, 24, 11, 25]).

Despite their impressive *prover* efficiency, verifier costs remain a major bottleneck in code-based SNARKs, due mainly to the *query complexity* of the underlying IOPP. IOPPs[1] have multiple rounds; in each round the prover sends an oracle to a vector in response to verifier randomness and the verifier queries and tests elements from these oracles. It would be too expensive for the verifier to query each element from

[1] It may be useful to think of an IOPP as a PCPP [8, 13] but with multiple rounds;

the oracle. Instead the verifier obtains a *probabilistic* guarantee that a *large fraction*, $\beta \in [0,1]$, of elements from each prover oracle passes its tests. We choose the number of verifier queries to be l such that $\beta^l < 2^{-\lambda}$, where λ is a security parameter of our choosing. In the Fast Reed-Solomon Interactive Oracle Proof of Proximity [3] (FRI), $\beta > \sqrt{1-\Delta_C}$, where Δ_C is the minimum distance of the code. This setting is the best proven result and is commonly referred to as the "list-decoding" regime, since by the famous Johnson Bound, there is only a small *list* of codewords that agree with any vector in more than $\sqrt{1-\Delta_C}$ fraction of locations. Alternatively, we refer to the (inferior) case when β is greater than $(1-\Delta_C/2)$ as the "unique-decoding regime", as there is only one unique codeword that agrees with a vector in that many locations (due to the distance properties of the error-correcting code).

Since the FRI IOPP is proven secure in the *list-decoding regime*, its verifier is both asymptotically and concretely efficient. However, the FRI IOPP can only be directly used[2] as a univariate PCS, and SNARKs based on univariates have a higher overhead than those based on multilinear PCS [21] (see [25, 11] for more details on this comparison). Basefold[25], introduced a technique for using FRI directly as a multilinear PCS by weaving the sum-check protocol [20] for multilinear polynomial evaluation with (a generalization of) FRI. It avoids the overhead of univariate SNARKs while maintaining polylogarithmic communication. Several works [12, 23, 2, 17] have already adapted Basefold to different settings. However, Basefold is only proven secure in the *unique decoding regime* and so it's proofs are concretely larger than FRI's. For that matter, no (native)[3] multilinear polynomial commitment scheme has been proven secure in the list-decoding regime. In particular, Brakedown [15], Ligero [1], two other state-of-the-art multilinear PCS, are only proven secure in the *unique decoding regime*, and the same is true for two recent multilinear Polynomial Commitment Schemes, WHIR [2] and Blaze [10]. The results in this paper will improve the verifier costs of all of these protocols.

1.1 Our Contributions

Basefold IOPP In the List-Decoding Regime. In this work, we introduce a proof for Basefold in the *list-decoding regime*, that is completely generic; it extends to any linear code and any finite field. At the heart of our proof is a new and stronger notion of *correlated agreement*([9, 7, 4]), a statement that shows that the linear combination of two vectors, $\pi_L + r\pi_R$ is unlikely to be close to a code unless π_L, π_R are. In this work, we strengthen that statement and prove that for all (c_L, c_R) that is close to (π_L, π_R), $\pi_L + r\pi_R$ must be close to $c_L + rc_R$, where "close" is defined as differing in no more than $n(1-\beta)$ locations. More specifically, we set $\beta > \sqrt{\sqrt{1-\Delta_C}}$ and prove that the number of $r \in \mathbb{F}$ for which this is *not true* is in $O(n)$. In comparison, the result from [17] only manages to prove this number is in $O(n^2)$, albeit with $\beta > \sqrt{1-\Delta_C}$ (i.e. they can get away with checking fewer locations of $\pi_L + r\pi_R$ but for fewer choices of r).

[2] There exists a generic transformation for univariate to multilinear PCS, but this incurs further overhead (e.g. [19])

[3] Actually, recent concurrent work [17] also proves Basefold for RS codes secure in the list decoding regime, and we discuss this in detail in Section 1.2

We now give a very high level overview of the proof. We will prove that every codeword close to $\pi_L + r\pi_R$ can be "explained" by a pair of codewords (c_L, c_R) that is close to (π_L, π_R). Let S be a subset of $[n]$. We observe first that if $(\pi_L[S], \pi_R[S])$ *does not* overlap much with any codewords from the neighborhood of (π_L, π_R), then there is *at most one element*, $r \in \mathbb{F}$ and codeword $c \in C$ such that $\pi_L[S] + r\pi_R[S] = c[S]$. For now lets refer to this event as a mapping from (bad set) S to r. Suppose two large "bad" sets S_1, S_2 map to two distinct values. Then $S_1 \cap S_2$ also maps to two distinct values (since $S_1 \cap S_2$ is a subset of both S_1 and S_2) and therefore $S_1 \cap S_2$ must be pretty small. Otherwise, if it is large, then by our observation above, $\pi_L[S_1 \cap S_2]$ has high agreement with a codeword in the neighborhood and therefore so does $\pi_L[S_1], \pi_L[S_2]$ (resp. for π_R). And so we arrive at the key insight of this work.

The number of field elements that allow a malicious prover to cheat is bounded by the number of large subsets of $[n]$ *with small pairwise intersection.*

We show that for the right set of parameters, this quantity is in $O(\sqrt{n})$. The full soundness proof has many more details and complexities, and we defer a more detailed overview and the proof itself to Section 3 and 3.1 respectively.

1.2 Related Work

In concurrent work [17], Haböck also proves that Basefold is secure in the list decoding regime and with a superior bound of $\beta > \sqrt{1 - \Delta_C}$. However, Haböck's result, like the result in "Proximity Gaps for Reed-Solomon Codes" [4], does not have acceptably high provable security for smaller fields (e.g. 2^{128}) or large (code) rates. Additionally, it only works with the Reed-Solomon code, whereas the result of our paper extends to all linear codes. In other concurrent work entitled Deep-Fold [16], Guo et. al. adapts Deep-FRI [7] to Basefold, and in that setting also proves the bound of $\beta > \sqrt{1 - \Delta_C}$, but again this only applies to Reed-Solomon codes. Finally, just this past week, "Linear Proximity Gap for Reed-Solomon Codes within the 1.5 Johnson Bound" [14] was published. This work also focuses on RS codes and is specific to the univariate FRI IOPP. However, their results, like the ones in this paper, are purely combinatorial and can (as far as we know) also be applied to Basefold over generic linear codes. A promising next step is to incorporate that work with this one to achieve Basefold soundness with $\beta > (1 - \Delta_C)^{1/3}$ rather than $(1 - \Delta_C)^{1/4}$, while maintaining security over smaller fields.

In other concurrent work, WHIR[2] presents a new efficient multilinear PCS with an extremely efficient verifier. However, their best reported results are only provably secure if Basefold is secure in the *list-decoding regime*, and they ultimately left this as a conjecture.

Additionally, there are several papers ([4, 7, 9]) that analyze the communication complexity of FRI. In "Worst Case To Average Case Reduction For Distance to a Linear Code"([9]), the authors improve upon the original FRI paper by showing that the verifier only needs to query within the $\beta = (1 - \Delta_C)^{1/4}$ radius of the code. Next, Deep-FRI ([7]) improves this by showing that the verifier only needs to query within the $\beta = (1 - \Delta_C)^{1/3}$. Deep-FRI additionally introduces a modification of the FRI protocol which further reduces the number of verifier queries, at the cost of some (slight) prover overhead. Finally, in “Proximity Gaps of Reed-Solomon Codes”([4]), the authors use a list-decoding algorithm for Reed-Solomon codes to further decrease the number of verifier queries, but (as mentioned earlier with regards to [17]), this result is only meaningful with a field that is at least quadratic in the instance size and with a suitably small rate, both of which impact prover time.

2 Preliminaries

2.1 Notation

Sets Let $n \in \mathbb{Z}$. Denote by $[n]$ the set $[0, n-1]$. $\mathsf{even}(S)$ is the set of even integers in S, and $\mathsf{odd}(S)$ is the set of odd integers in S. Let $q \in \mathbb{N}$. Then $S/q = \{s/q : s \in S\}, S + q = \{s + q : s \in S\}$, etc. 2^S is the power set of S.

Strings and Functions Let $x \in \mathbb{F}, r \in \mathbb{N}$, then $x^{||r}$ is the string obtained by concatenating x to itself r times. Let $f : S \to S$ be a function and $n \in \mathbb{N}$. Then $f^{\circ n}$ denotes function composition of f with itself n times.

Error-Correcting Codes We will use C to denote a linear $[n, k, d]$ code, which is a subspace C of $\mathbb{F}^n$ with an encoding algorithm $\mathsf{Enc}_C : \mathbb{F}^k \to C$ (Definition 1). Δ_C is the minimum relative distance of the code C. Let $n \in \mathbb{N}$ and $S \subset [n]$. For a vector $\mathbf{x} \in \mathbb{F}^n$, $\mathbf{x}[S] = \{x[i] : i \in S\}$. Let $\mathbf{v} \in \mathbb{F}^n$, let C be a linear code, and let $S \subset [n]$. Then we say that $\mathbf{v}[S] \in C[S]$ if there exists a codeword $c \in C$ such that $\mathbf{v}[S] = c[S]$.

2.2 Definitions

We present a standard definition of a *linear error-correcting code.*

Definition 1 (Linear Code)**.** *A linear error-correcting code with message length k and codeword length n is an injective mapping from $\mathbb{F}^k$ to a linear subspace $C \subseteq \mathbb{F}^n$. C is associated with a generator matrix, $G \in \mathbb{F}^{k \times n}$ such that the rows of G are a basis of C and the encoding of a vector $\mathbf{v} \in \mathbb{F}^k$ is $\mathbf{v} \cdot G$. The minimum Hamming distance of a code is the minimum on the number of different entries between any two different codewords $c_1, c_2 \in C$. If C has a minimum distance $d \in [n]$, we say that C is an $[n, k, d]$ code and use Δ_C to denote d/n—the relative minimum distance.*

3 A Stronger Notion of Correlated Agreement

In this section, we state and prove our main result and compute concrete bounds in subsection 3.2. Our main result is a stronger version of the correlated agreement Theorem from “Proximity Gaps of Reed Solomon Codes” [4] (specifically Theorem 1.4 (Correlated Agreement Over Lines)).

Lemma 1 (Strong Correlated Agreement A)**.** *Let C be a linear error-correcting code with $n \in \mathbb{N}$, and $\pi_L, \pi_R \in \mathbb{F}^n$. Let $\alpha > \sqrt{1 - \Delta_C}$ and $\beta > \sqrt{\alpha}$. If*

$$\Pr_{r \in \mathbb{F}}[\Delta(\pi_L + r\pi_R) \leq 1 - \beta] > \frac{O(n)}{|\mathbb{F}|},$$

then there exists $S \subset [n]$ and $c_L, c_R \in C$ satisfying

- ***Density:*** $|S|/n \geq \beta$
- ***Agreement:*** $\pi_L[S] = c_L[S], \pi_R[S] = c_R[S]$ *and* $\forall r \in \mathbb{F}, \pi_L[S] + r\pi_R[S] = c_L[S] + rc_R[S]$

Actually, we will find it more useful to prove the following stronger statement, from which the previous statement easily follows.

Lemma 2 (Strong Correlated Agreement B)**.** *Let $\pi_L, \pi_R \in \mathbb{F}^n$ and $\pi = (\pi_L, \pi_R)$. Let $\alpha > \sqrt{1 - \Delta_C}$ and let $\beta > \sqrt{\alpha}$. Let $A_\pi \subset \mathbb{F}$ satisfy*

$$A_\pi = \left| \left\{ r \in \mathbb{F} : \exists S \subset [n], c \in C \textit{such that} \quad \begin{array}{l} |S| > \beta n, \\ (\pi_L[S] + r\pi_R[S]) = c[S], \\ \textit{but } \{\pi_R[S], \pi_L[S]\} \not\subset C[S] \end{array} \right\} \right|.$$

Then $\forall \pi_L, \pi_R \in \mathbb{F}^n$ with $\pi = (\pi_L, \pi_R)$

$$|A_\pi| \in O(n) \tag{1}$$

We defer the full proof of Lemma 2 to Section 3.1. In this section, we instead focus on building intuition and we give an informal proof that Lemma 2 is true for the looser bound of $|A_\pi| \in O(n^2)$. As a first step, we split A_π into two sets; $A_{\pi,\geq\alpha}$ and $A_{\pi,<\alpha}$. One set, $A_{\pi,\geq\alpha}$, we define as follows (modifications from the main set are in bold):

$$A_{\pi,\geq\alpha} = \left| \left\{ r \in \mathbb{F} : \exists S \subset [n], c_L, c_R, c \in C \text{ st} \quad \begin{array}{l} |S| > \beta n, \\ (\pi_L[S] + r\pi_R[S]) = c[S], \\ \mathbf{c_L + rc_R = c} \\ |\{\mathbf{i \in [n] : c_L[i] = \pi_L[i] \wedge c_R[i] = \pi_R[i]}\}| \geq \alpha \\ \text{but } \{\pi_R[S], \pi_L[S]\} \not\subset C[S] \end{array} \right\} \right|.$$

We define the other set, $A_{\pi,<\alpha}$ to be equal $A_\pi \setminus A_{\pi,\geq\alpha}$. Clearly, $A_\pi = A_{\pi,\geq\alpha} \cup A_{\pi,<\alpha}$ and so

$$|A_\pi| \leq |A_{\pi,\geq\alpha}| + |A_{\pi,<\alpha}|. \tag{2}$$

Thus our task reduces to bounding the size of each of these individual sets. First, to bound the size of $A_{\pi,\geq\alpha}$, we decompose it further into a union of even smaller sets. Define,

$$\mathcal{L}_{\pi,\alpha} = \{(c_L, c_R) \in C \times C : |\{i \in [n] : c_L[i] = \pi_L[i] \wedge c_R[i] = \pi_R[i]\}| \geq \alpha\},$$

and for each $(c_L, c_R) \in \mathcal{L}_{\pi,\alpha}$, define

$$A_{\pi,\geq\alpha,\mathbf{c_L},\mathbf{c_R}} = \left| \left\{ r \in \mathbb{F} : \exists S \subset [n_i], c \in C \text{ st} \quad \begin{array}{l} |S| > \beta n, \\ (\pi_L[S] + r\pi_R[S]) = c[S], \\ \mathbf{c_L + rc_R = c} \\ \text{but } \{\pi_R[S], \pi_L[S]\} \not\subset C[S] \end{array} \right\} \right|.$$

Clearly,

$$A_{\pi,\geq\alpha} = \bigcup_{(c_L,c_R)\in\mathcal{L}} A_{\pi,\geq\alpha,(c_L,c_R)}. \tag{3}$$

Putting together Equations 2 and 3, we have

$$|A_\pi| \leq \left(\sum_{(c_L,c_R)\in\mathcal{L}_{\pi,\alpha}} A_{\pi,\geq\alpha,(c_L,c_R)} \right) + |A_{\pi,<\alpha}| \tag{4}$$

Thus, our task reduces to bounding the size of the following three quantities:

1. $|\mathcal{L}_{\pi,\alpha}|$
2. $|A_{\pi,\geq\alpha,(c_L,c_R)}|$ for all $(c_L,c_R) \in \mathcal{L}_{\pi,\alpha}$
3. $|A_{\pi,<\alpha}|$

To bound the size of Items (1) and (3), we will make use of the following important Lemma (stated informally here but discussed in more detail in Section 3.1) and Appendix A.

Lemma 3 (Informal)**.** *The number of large subsets of $[n]$ with small pairwise intersection is in $O(n)$.*

Our strategy in all three of these proofs is to build a set $\mathcal{S} \subset 2^{[n]}$ that has a one-to-one correspondence with the set we are trying to bound. For Items (1) and (3), we will prove that this set $\mathcal{S}$ is a set of *large* sets with *small* pairwise intersection. Then, we use Lemma 3 to bound $\mathcal{S}$ which in turn, bounds the set in question. For Item (2) we will instead show that $\mathcal{S}$ is a set of *disjoint* sets, which automatically implies $|\mathcal{S}| \leq n$. We begin by bounding Item (1).

Lemma 4.

$$|\mathcal{L}_{\pi,\alpha}| \in O(n)$$

Proof. For each $(c_L,c_R) \in \mathcal{L}$, let $S_{c_L,c_R} = \{i \in [n] : \pi_L[i] = c_L \wedge \pi_R[i] = c_R\}$. Then we define,

$$\mathcal{S} = \{S_{c_L,c_R} : (c_L,c_R) \in \mathcal{L}_{\alpha,\pi}\}.$$

By definition of $\mathcal{L}_{\alpha,\pi}$, each set in $\mathcal{S}$ is larger than α. Next, we show that any two sets in $\mathcal{S}$ must have pairwise intersection *smaller* than $1 - \Delta_C$. Let (c'_L, c'_R) be an element of $\mathcal{L}_{\pi,\alpha}$ *distinct* from (c_L, c_R). Then,

$$\begin{aligned}
&|S_{c_L,c_R} \cap S_{c'_L,c'_R}| \\
&= |\{i \in [n] : i \in S_{c_L,c_R} \wedge i \in S_{c'_L,c'_R}\}| \\
&= |\{i \in [n] : \pi_L[i] = c_L[i] \wedge \pi_R[i] = c_R[i] \wedge \pi_L[i] = c'_L[i] \wedge \pi_R[i] = c'_R[i]\}| \\
&\leq |\{i \in [n] : \pi_L[i] = c_L[i] \wedge \pi_L[i] = c'_L[i]\}| \\
&\leq (1 - \Delta_C)
\end{aligned}$$

Therefore, by Lemma 3, $|\mathcal{S}| \in O(n)$. Next, it is clear by the definition of $\mathcal{S}$ that $|\mathcal{S}| = |\mathcal{L}_{\pi,\alpha}|$, which completes the proof. □

Next, we use Lemma 3 to prove Item 3.

Lemma 5. $|A_{\pi,<\alpha}| \in O(n)$

Proof. We follow the structure of the previous proof. We build a set $\mathcal{S} \subset 2^n$ as follows. For each $r \in A_{\pi,<\alpha}$, let $S_{\pi,r} \subset [n]$ be the maximal set such that $|S_{\pi,r}| > \beta$ and $\pi_L[S_{\pi,r}] + r\pi_R[S_{\pi,r}] \in C[S]$. Define

$$\mathcal{S} = \{S_{\pi,r} : r \in A_{\pi,<\alpha}\}$$

By Definition of $S_{\pi,r}$, $|S_{\pi,r}| > \beta$. Next, we show that any two sets in $\mathcal{S}$ must have (small) pairwise intersection smaller than α. Suppose otherwise. Then there exists $r, r' \in \mathbb{F}$ such that $r \neq r'$, $c, c' \in C$, and $S' \subset [n]$ such that $|S'| > \alpha$ (i.e. $S' = S_{\pi,r} \cap S_{\pi,r'}$) where

$$\pi_L[S'] + r\pi_R[S'] = c[S'] \wedge \pi_L[S'] + r'\pi_R[S'] = c'[S']$$

By linearity of the code, we can subtract the left equation from the right equation to obtain

$$\pi_R[S'](r - r') = (c - c')[S'].$$

Since $r \neq r'$, this implies that

$$\pi_R[S'] = c^*_R[S'],$$

where $c^*_R = \frac{(c-c')}{r-r'}$. Then, we can plug this back into the original equations to obtain,

$$\pi_L[S'] = c^*_L[S'],$$

where $c^*_L = c - rc^*_R$ (and by linearity of the code, $c^*_L \in C$). Therefore,

$$|\{i \in [n] : \pi_L[i] = c^*_L[i] \wedge \pi_R[i] = c^*_R[i]\}| > \alpha.$$

But since $c^*_L + rc^*_R = (c - rc^*_R) + rc^*_R = c$, it follows that $r \in A_{\pi,\geq\alpha,(c^*_L,c^*_R)}$, which contradicts our assumption that $r \in A_{\pi,<\alpha}$. Therefore, $\mathcal{S}$ is a set of large ($> \beta n$) sets with smaller ($\leq \alpha n$) pairwise intersection and therefore $|\mathcal{S}| \in O(n)$. Finally, we complete the proof by observing that there is a one-to-one correspondance between $\mathcal{S}$ and $A_{\pi,\leq\alpha}$. □

Finally, we bound the size of $A_{\pi,\geq\alpha,(c_L,c_R)}$ for each $(c_L, c_R) \in \mathcal{L}$.

Lemma 6. *For all* $(c_L, c_R) \in \mathcal{L}$*,*

$$|A_{\pi,\geq\alpha,(c_L,c_R)}| \in O(n)$$

Proof. To prove this Lemma, we will make use of the following Claim.

Claim 1. *For each* $i \in [n]$ *where* $\pi_L[i] \neq c_L[i], \pi_R[i] \neq c_R[i]$*, there is exactly one* $r \in \mathbb{F}$ *satisfying*

$$\pi_L[i] + r\pi_R[i] = c_L[i] + rc_R[i]. \quad (5)$$

Proof. Let $P(X) = (\pi_L[i] - \mathbf{c}_L[i]) + X(\pi_R[i] - \mathbf{c}_R[i])$. Then r is a zero of $P(X)$ if and only if it satisfies Equation 5. By the Schwart-Zippel Lemma, $P(X)$ has only one zero in $\mathbb{F}$. This completes the proof of the Claim. □

Proceeding with the proof of the Lemma, define $S'_{c_L,c_R} \subset [n]$ as

$$S'_{c_L,c_R} = \{i \in [n] : c_L[i] \neq \pi_L[i] \wedge c_R[i] \neq \pi_R[i]\}.$$

For each $r \in A_{\pi,\geq\alpha,(c_L,c_R)}$, let S_r be the maximal subset of S'_{c_L,c_R} such that $\pi_L[S_r] + r\pi_R[S_r] = c_L[S_r] + rc_R[S_r]$. Define $\mathcal{S} \subset 2^{[n]}$ as,

$$\mathcal{S} = \{S_r : r \in A_{\pi,\geq\alpha,(c_L,c_R)}\}.$$

By Claim 1, every two sets in $\mathcal{S}$ are *disjoint*. Therefore

$$|\bigcup_{S \in \mathcal{S}} \mathcal{S}| = \sum_{S \in \mathcal{S}} |S| \leq n.$$

For each $r \in \mathbb{F}$, $S_r \in \mathcal{S}$ is non-empty, because otherwise π_L, π_R both agree with codewords at $> \beta$ locations and so $r \notin A_\pi$. Therefore,

$$|\mathcal{S}| \leq |\sum_{S \in \mathcal{S}} S| \leq n.$$

Since there is a one to one relationship between $A_{\pi,\geq\alpha,(c_L,c_R)}$ and $\mathcal{S}$, we have

$$|A_{\pi,\geq,\alpha,(c_L,c_R)}| = |\mathcal{S}| = n,$$

which completes the proof. □

Remark 1. *It has been proven in a blog post*[4] *that the bound of $\beta > 1 - \Delta_C/3$ is tight for general linear codes, which seems to contradict the results in this paper. However, upon closer inspection, the codes they consider are ones with minimum distance Δ_C, such that $1 - \Delta_C/3$ is always less than $\sqrt{1 - \Delta_C}$. Thus, there is no contradiction after all, as we do not expect the Correlated Agreement/Proximity Gaps statement to hold for $\beta < \sqrt{\sqrt{1 - \Delta_C}}$, let alone $\beta < \sqrt{1 - \Delta_C}$.*

3.1 Strong Correlated Agreement With Better Bounds

Based on the results in the prior section, the final size of A_π is in $O(n^2)$. In this section, we show that we can do better, achieving a bound of $O(n)$. We also provide a more formal and detailed treatment of our main result.

Lemma 7 (Strong Correlated Agreement C)**.** *Let $\pi_L, \pi_R \in \mathbb{F}^n$ with $\pi = (\pi_L, \pi_R)$ and $\boldsymbol{\epsilon} \in [\frac{2}{\sqrt{n}}]^3$. Define $\alpha_{\boldsymbol{\epsilon}} := \sqrt{1 - \Delta_C + \boldsymbol{\epsilon}[0]}$, and $\beta_{\boldsymbol{\epsilon}} > \sqrt{\alpha_{\boldsymbol{\epsilon}} + \boldsymbol{\epsilon}[1]} + \boldsymbol{\epsilon}_2$. Define the set $A_\pi(\boldsymbol{\epsilon})$ as follows.*

$$A_\pi(\boldsymbol{\epsilon}) := \left|\left\{ r \in \mathbb{F} : \exists S \subset [n], c \in C \;\; st \quad \begin{array}{l} |S| > \beta_{\boldsymbol{\epsilon}} n, \\ (\pi_L[S] + r\pi_R[S]) = c[S], \\ \;\textit{but}\ \forall (\mathbf{c_L}, \mathbf{c_R}) \in \mathbf{C} \times \mathbf{C}, \\ |\{\mathbf{i} \in [\mathbf{n}] : \pi_{\mathbf{L}}[\mathbf{i}] = \mathbf{c_L}[\mathbf{i}] \wedge \pi_{\mathbf{R}}[\mathbf{i}] = \mathbf{c_R}[\mathbf{i}]\}| < \beta_{\boldsymbol{\epsilon}} - \boldsymbol{\epsilon}[\mathbf{2}] \end{array} \right\}\right|.$$

[4] https://notes.0xparc.org/results/counterexample-proximity-gap/

Then $\forall \pi_L, \pi_R \in \mathbb{F}^n$ *with* $\pi = (\pi_L, \pi_R)$ *and for all* $\boldsymbol{\epsilon} \in [\frac{2}{\sqrt{n}}]^3$

$$|A_\pi(\boldsymbol{\epsilon})| \in O(n) \tag{6}$$

Proof. We follow the structure of the previous, (informal) proof of Lemma 2. As before, define $A_{\pi,\geq\alpha_{\boldsymbol{\epsilon}}}(\boldsymbol{\epsilon})$ as follows.

$$A_{\pi,\geq\alpha_{\boldsymbol{\epsilon}}}(\boldsymbol{\epsilon}) := \left| \left\{ r \in \mathbb{F} : \exists S \subset [n], \mathbf{c_L^*}, \mathbf{c_R^*}, c \in C \text{ st} \quad \begin{array}{l} |S| > \beta_{\boldsymbol{\epsilon}} n, \\ (\pi_L[S] + r\pi_R[S]) = c[S], \\ \mathbf{c_L^*} + \mathbf{r c_R^*} = \mathbf{c} \\ |\{\mathbf{i} \in [\mathbf{n}] : \mathbf{c_L^*}[\mathbf{i}] = \pi_{\mathbf{L}}[\mathbf{i}] \wedge \mathbf{c_R^*}[\mathbf{i}] = \pi_{\mathbf{R}}[\mathbf{i}]\}| \geq \alpha_{\boldsymbol{\epsilon}} \\ \text{ but } \forall (c_L, c_R) \in C \times C, \\ |\{i \in [n] : \pi_L[i] = c_L[i] \wedge \pi_R[i] = c_R[i]\}| < \beta_\epsilon - \epsilon[2] \end{array} \right\} \right|,$$

and define $A_{\pi,<\alpha_{\boldsymbol{\epsilon}}}(\boldsymbol{\epsilon}) := A_\pi(\boldsymbol{\epsilon}) \setminus A_{\pi,\geq\alpha}(\boldsymbol{\epsilon})$ (i.e. if $r \in A_{\pi,<\alpha_{\boldsymbol{\epsilon}}}$, then there is no nearby c_L^*, c_R^* that "explains" c). By definition of set compliment,

$$|A_{\pi,\boldsymbol{\epsilon}}| = |A_{\pi,\geq\alpha_{\boldsymbol{\epsilon}}}(\boldsymbol{\epsilon})| + |A_{\pi,<\alpha_{\boldsymbol{\epsilon}}}(\boldsymbol{\epsilon})|. \tag{7}$$

Next, define

$$\mathcal{L}_{\pi,\alpha_{\boldsymbol{\epsilon}}} := \{(c_L, c_R) \in C \times C : |\{i \in [n] : c_L[i] = \pi_L[i] \wedge c_R[i] = \pi_R[i]\}| \geq \alpha_{\boldsymbol{\epsilon}}\},$$

and for each $(c_L^*, c_R^*) \in \mathcal{L}_{\pi,\alpha_{\boldsymbol{\epsilon}}}$, define

$$A_{\pi,\geq\alpha_{\boldsymbol{\epsilon}},c_L^*,c_R^*}(\boldsymbol{\epsilon}) := \left| \left\{ r \in \mathbb{F} : \exists S \subset [n], c \in C \text{ st} \quad \begin{array}{l} |S| > \beta_{\boldsymbol{\epsilon}} n, \\ (\pi_L[S] + r\pi_R[S]) = c[S], \\ \mathbf{c_L^*} + \mathbf{r c_R^*} = \mathbf{c} \\ \text{ but } \forall (c_L, c_R) \in C \times C, \\ |\{i \in [n] : \pi_L[i] = c_L[i] \wedge \pi_R[i] = c_R[i]\}| \leq \beta_{\boldsymbol{\epsilon}} - \boldsymbol{\epsilon}[2] \end{array} \right\} \right|.$$

Then,

$$A_{\pi,\geq\alpha}(\boldsymbol{\epsilon}) = \bigcup_{(c_L^*, c_R^*) \in \mathcal{L}} A_{\pi,\geq\alpha,(c_L^*,c_R^*))}(\boldsymbol{\epsilon}), \tag{8}$$

and so, combining Equations 7 and 8, we have

$$|A_\pi|(\boldsymbol{\epsilon}) \leq \left(\sum_{(c_L,c_R) \in \mathcal{L}_{\pi,\alpha_{\boldsymbol{\epsilon}}}} |A_{\pi,\geq\alpha_{\boldsymbol{\epsilon}},(c_L,c_R)}(\boldsymbol{\epsilon}|) \right) + |A_{\pi,<\alpha_{\boldsymbol{\epsilon}}}(\boldsymbol{\epsilon})| \tag{9}$$

Thus, our task reduces to bounding the size of the following three quantities:

1. $|\mathcal{L}_{\pi,\alpha_{\boldsymbol{\epsilon}}}|$
2. $|A_{\pi,\geq\alpha_{\boldsymbol{\epsilon}},(c_L,c_R)}(\boldsymbol{\epsilon})|$ for all $(c_L, c_R) \in \mathcal{L}_{\pi,\alpha_{\boldsymbol{\epsilon}}}$
3. $|A_{\pi,<\alpha_{\boldsymbol{\epsilon}}}(\boldsymbol{\epsilon})|$

To bound the size of Items (1) and (3), we will make use of the following important Lemma (which was stated informally in Lemma 3).

Lemma 8. *Let $\alpha, \beta \in [0,1], n \in \mathbb{Z}$ and define $\mathcal{S} \subset 2^{[n]}$ as follows.*

- *If $S \in \mathcal{S}$, then $|S| > \beta n$*
- *For any two sets $S_1, S_2 \in \mathcal{S}$, $|S_1 \cap S_2| < \alpha n$.*

Let $\epsilon \in (0,1]$ such that $\alpha + \epsilon = \beta^2$. Then,

$$|\mathcal{S}| \leq \frac{\beta - \alpha}{\epsilon} = \frac{\sqrt{\alpha + \epsilon} - \alpha}{\epsilon}$$

Proof. We defer the proof to Appendix A as it follows the proof of the Johnson Bound[5] closely. □

Lemma 9.

$$|\mathcal{L}_{\pi,\alpha_{\boldsymbol{\epsilon}}}| \leq \frac{\sqrt{1 - \Delta_C + \boldsymbol{\epsilon}[0]} - (1 - \Delta_C)}{\boldsymbol{\epsilon}[0]}$$

Proof. Following the logic of Lemma 4 (and only changing the parameters), we can prove that the size of $\mathcal{L}_{\pi,\alpha_{\boldsymbol{\epsilon}}}$ is bounded by the number of sets of $[n]$ of size $> \alpha_{\boldsymbol{\epsilon}}$ whose pairwise intersection is smaller than $1 - \Delta_C$. Recall that $\alpha_{\boldsymbol{\epsilon}} = \sqrt{1 - \Delta_C + \boldsymbol{\epsilon}[0]}$. Thus, plugging in Lemma 8, where $\beta := \alpha_{\boldsymbol{\epsilon}}$ and $\alpha := 1 - \Delta_C$ completes the proof. □

Lemma 10.

$$|A_{\pi,<\alpha_{\boldsymbol{\epsilon}}}| \leq \frac{\sqrt{\sqrt{1 - \Delta_C + \boldsymbol{\epsilon}[0]} + \boldsymbol{\epsilon}[1]} - \sqrt{1 - \Delta_C + \boldsymbol{\epsilon}[0]}}{\boldsymbol{\epsilon}[1]}$$

Proof. Following the logic of Lemma 5 (and only changing the parameters), we can prove that the size of $|A_{\pi,\leq\alpha}(\boldsymbol{\epsilon})|$ is less than or equal to the number of subsets of $[n]$ that are larger than $\beta_{\boldsymbol{\epsilon}} - \boldsymbol{\epsilon}[2]$, but with pairwise intersection smaller than $\alpha_{\boldsymbol{\epsilon}}$. Thus, applying Lemma 8, with $\beta := \beta_{\boldsymbol{\epsilon}} - \boldsymbol{\epsilon}[2]$ and $\alpha := \alpha_{\boldsymbol{\epsilon}}$, completes the proof. □

Remark 2. *When we apply this Lemma to prove soundness of the multi-round Basefold IOPP in Section* **??**, *we will actually bound $A_{\pi,\leq\alpha_{\boldsymbol{\epsilon}}}$ to be the number of sets larger than $\beta - d\boldsymbol{\epsilon}[2]$ (with pairwise intersection smaller than $\alpha_{\boldsymbol{\epsilon}}$). That way, we rule out the event that an oracle in one round goes from being very far from a codeword to within $\boldsymbol{\epsilon}[2]$ distance of one, as our analysis does not include that event.*

Finally, we obtain a tighter bound for $A_{\pi,\geq\alpha_{\boldsymbol{\epsilon}},(c_L,c_R)}(\boldsymbol{\epsilon})$ (for each $c_L, c_R \in \mathcal{L}_{\pi,\alpha_1}$). We follow the proof of Lemma 6 closely, highlighting in bold the parts that differ.

Lemma 11. *For all $(c_L, c_R) \in \mathcal{L}_{\pi,\boldsymbol{\epsilon}}$,*

$$|A_{\pi,\geq\alpha,\boldsymbol{\epsilon}(c_L,c_R)}(\boldsymbol{\epsilon})| \leq \frac{1}{\boldsymbol{\epsilon}[2]}$$

[5] https://www.cs.cmu.edu/~venkatg/teaching/au18-coding-theory/lec-scribes/list-decoding.pdf

Proof. Define $S'_{c_L,c_R} \subset [n]$ as

$$S'_{c_L,c_R} = \{i \in [n] : c_L[i] \neq \pi_L[i] \wedge c_R[i] \neq \pi_R[i]\}.$$

For each $r \in A_{\pi,\geq\alpha,(c_L,c_R)}$, let S_r be the maximal subset of S'_{c_L,c_R} such that $\pi_L[S_r] + r\pi_R[S_r] = c_L[S_r] + rc_R[S_r]$. Define $\mathcal{S} \subset 2^{[n]}$ as,

$$\mathcal{S} = \{S_r : r \in A_{\pi,\geq\alpha,(c_L,c_R)}\}.$$

By Claim 1, every two sets in $\mathcal{S}$ are *disjoint*. Therefore

$$|\bigcup_{S \in \mathcal{S}} \mathcal{S}| = \sum_{S \in \mathcal{S}} |S| \leq n.$$

For each $r \in \mathbb{F}$**,** $|\mathbf{S_r}| > \epsilon[\mathbf{2}]\mathbf{n}$, because otherwise π_L, π_R both agree with codewords at $\geq \beta_{\boldsymbol{\epsilon}} - \boldsymbol{\epsilon}[2]$ locations and so $r \notin A_\pi(\boldsymbol{\epsilon})$. Therefore,

$$|\mathcal{S}| \cdot \boldsymbol{\epsilon}[2]n \leq \sum_{S \in \mathcal{S}} |S|,$$

and so, combining the previous two equations,

$$|\mathcal{S}| \leq \frac{\sum_{S \in \mathcal{S}} |S|}{\boldsymbol{\epsilon}[2]n} \leq \frac{n}{n\boldsymbol{\epsilon}[2]} = \frac{1}{\boldsymbol{\epsilon}[2]}$$

Since there is a one-to-one relationship between $A_{\pi,\geq\alpha,(c_L,c_R)}(\boldsymbol{\epsilon})$ and $\mathcal{S}$, we have

$$|A_{\pi,\geq\alpha_{\boldsymbol{\epsilon}},(c_L,c_R)}(\boldsymbol{\epsilon})| = |\mathcal{S}| \leq \frac{1}{\boldsymbol{\epsilon}[2]},$$

which completes the proof. □

Combining these three bounds with Equation 9, we have

$$|A_\pi(\boldsymbol{\epsilon})| \leq (\frac{\sqrt{1 - \Delta_C + \boldsymbol{\epsilon}[0]} - (1 - \Delta_C)}{\boldsymbol{\epsilon}[0]}) \cdot \frac{1}{\boldsymbol{\epsilon}[2]} + \frac{\sqrt{\sqrt{1 - \Delta_C + \boldsymbol{\epsilon}[0]} + \boldsymbol{\epsilon}[1]} - \sqrt{1 - \Delta_C + \epsilon_1}}{\boldsymbol{\epsilon}[1]}.$$

Finally, since $\boldsymbol{\epsilon}[0], \boldsymbol{\epsilon}[1], \boldsymbol{\epsilon}[2] \in [\frac{2}{\sqrt{n}}]$, each of the individual terms is smaller than $\frac{\sqrt{n}}{2}$ and so

$$|A_\pi(\boldsymbol{\epsilon})| \leq (\sqrt{n}/2) \cdot \sqrt{n}/2 + \sqrt{n}/2 = n/4 + \sqrt{n}/2,$$

which completes the proof. □

3.2 Concrete Bounds and Comparison To Other Work

Concretely, Lemma 7 implies the verifier need only check π_L, π_R in

$$\beta > \beta_\epsilon > \sqrt{\sqrt{1 - \Delta_C + \boldsymbol{\epsilon}[0]} + \boldsymbol{\epsilon}[1]} + \boldsymbol{\epsilon}[2]$$

locations to ascertain that they are “close” to consistent codewords with their linear combintion, and its assessment will only be *incorrect* with probability *at most* $\frac{n/4+\sqrt{n}/2}{|\mathbb{F}|}$. On the other hand, the bound from [17] over Reed-Solomon codes (translated to our notation and for a batch of only 2 polynomials) achieves

$$|A_\pi| \leq 2\frac{(m+1/2)}{\sqrt{1-\Delta_C}} \cdot \max\left(\frac{(m+1/2)^6}{3(1-\Delta_C)} \cdot n^2, 2 \cdot (B \cdot n + 1)\right),$$

where m is a parameter larger than 3 and for verifier query complexity

$$\beta > \sqrt{(1-\Delta_C)}(\frac{1}{2m}),$$

i.e. the verifier only needs to query π_L, π_R in $\beta > \sqrt{1-\Delta}(\frac{1}{2m})$ locations (which is better than our bound), but will be *incorrect* with the probability of $\frac{O(n^2)}{|\mathbb{F}|}$ (which is worse than our bound). We show a comparison of concrete results in Figure 1. We focus on the setting of an 128-bit field, as for larger fields (e.g. 192 bit), it always makes to use the result from [17].

Result	Distance	Probability ($|A_\pi|/|\mathbb{F}|$)	β
Weighted Corr Agreement	3/4	2^{-53}	$2^{-0.77}$
This result	3/4	2^{-108}	$2^{-0.49}$
Weighted Corr Agreement	7/8	2^{-51}	$2^{-1.2}$
This result	7/8	2^{-108}	$2^{-0.746}$
Weighted Corr Agreement	15/16	2^{-49}	$2^{-1.7}$
This result	15/16	2^{-108}	$2^{-0.99}$
Weighted Corr Agreement	31/32	2^{-48}	$2^{-2.27}$
This result	31/32	2^{-108}	$2^{-1.24}$

Figure 1: We consider instance sizes of 2^{30} over finite field, $\mathbb{F}$ such that $\log_2(|\mathbb{F}|) = 128$. For our result, we set $\epsilon_1 = \epsilon_2 = \epsilon_3 = 0.0005$. For Weighted Correlated Agreement, we set $m = 3$ to minimize $|A_\pi|/|\mathbb{F}|$.

4 Improved Soundness of the Basefold Protocol

In this section, we re-prove the soundness theorem of Basefold, and show that it is sound even if the verifier only makes $l := \frac{\lambda}{\log_2(\beta)}$ queries for $\beta > (1-\Delta)^{1/4}$. Previously, we only proved this for $\beta > 1 - \Delta/3$. The Basefold IOP remains unchanged from [25][6]. We restate the IOPP in Figure ?? for completeness. We also restate the definition of a *foldable code*[7], which was introduced in Basefold [25]. A *foldable code* is a family of codes, which we will denote $(C_d, .., C_0)$, which are characterized by a set of vectors

[6] The syntax of this description is slightly different than that in [25], but the protocol itself is equivalent

[7] For ease of exposition, we define the codes according to a different ordering than the original.

$\{\mathbf{t}_i \in \mathbb{F}^{n_i} : i \in [1, d]\}$. Each codeword $c_i \in C_i$ is composed of two codewords in C_{i-1}. Additionally, the structure of a *foldable codes* enables local consistency checks between codewords in adjacent codes. These consistency tests allow the Basefold (and FRI) IOPPs to maintain logarithmic communication complexity.

Definition 2 (Foldable Code)**.** *Let $c, d \in \mathbb{N}$ and for each $i \in [d]$, define $k_i = 2^i, n_i = c \cdot k_i$. A $[n_d, k_d]$, foldable code is a family of codes $(C_0, .., C_d)$, where the base code, C_0, is equal to the repetition code, $\{m^{||c} : m \in \mathbb{F}\}$ and each $[n_i, k_i]$ code, C_i, and each $\mathbf{v} \in C_i$ satisfies the following:*

$$\begin{aligned}\mathbf{v} := Enc_{C_i}(\mathbf{m}) = Enc_{C_{i-1}}(\mathbf{m}_L) + \mathbf{t}_i \circ Enc_{C_{i-1}}(\mathbf{m}_R) \\ ||Enc_{C_{i-1}}(\mathbf{m}_L) - \mathbf{t}_i \circ Enc_{C_{i-1}}(\mathbf{m}_R)\end{aligned}$$

where $\{\mathbf{t}_i \in \mathbb{F}^{n_i : i \in [1,d]}$ is given in the description of the code, $\mathbf{m} \in \mathbb{F}^{k_i}$, $\mathbf{m} = \mathbf{m}_L || \mathbf{m}_R$ and $\circ$ denotes the Hadamard product.

Foldable codes are attractive because a codeword in C_i can be transformed into a smaller codeword in C_{i-1} using only *local* operations. More specifically, we query the *same* random point on each of the $n_i/2$ lines defined by the pairs $\{(\mathbf{v}[j], \mathbf{v}[j+1]) : j \in \mathsf{even}([n_i])$, and obtain a new codeword $\in C_{i-1}$. We describe this formally with the following definition.

Definition 3 (Fold)**.** *Define* $\mathsf{interp} : \mathbb{F}^2 \times \mathbb{F}^2 \to \mathbb{F}[X]$ *to be Lagrange Interpolation of a degree-one univariate polynomial. Let $(C_d, .., C_0)$ be a family of foldable codes characterized by $\{\mathbf{t}_i \in \mathbb{F}^{n_i} : i \in [1, d]\}$, $\mathbf{v} \in C_i$, and for each $j \in \mathsf{even}([n_i])$,*

$$(p_j, p_{j+1}) = ((\mathbf{t}[j], \mathbf{v}[j]), (-\mathbf{t}[j], \mathbf{v}[j+1])).$$

Then, the fold *of $\mathbf{v}$ with respect to $r \in \mathbb{F}$ is the vector,* $\mathsf{fold}(\mathbf{v}, r)[j]$ *satisfying (after re-ordering),*

$$\mathsf{fold}(\mathbf{v}, r)[j] = \mathsf{interp}(p_j, p_{j+1})(r).$$

At times, we will need to work with the univariate polynomials defined by $\mathsf{interp}(p_j, p_{j+1})$ directly. We call these polynomials the *unfolding* of $\mathbf{v}$. To ease exposition, we denote by $\mathbf{v}_L, \mathbf{v}_R$ the codewords that for all $j \in \mathsf{even}([n_i])$ satisfy,

$$\mathsf{interp}(p_j, p_{j+1}) = \mathbf{v}_L[j] + X\mathbf{v}_R[j]. \tag{10}$$

We remark that fold can also be defined over arbitrary vectors that are not codewords, and indeed the FRI and Basefold IOPPs rely on this fact. For a generic $\pi \in \mathbb{F}^{n_i}$, define the pair of points $(p_j, p_{j+1}) = ((\mathbf{t}[j], \pi[j]), (\mathbf{t}[j], \pi[j+1]))$. Then, as before $\mathsf{fold}(\pi, r) = \mathsf{interp}(p_j, p_{j+1})$. Finally, we will sometimes fold over entire sets $S \subset [n_i]$, and this operation is well defined as long as S contains $j+1$ whenever it contains j. We introduce additional notation for this as follows.

$$\mathsf{fold}(\pi, r)[S] = \{\mathsf{interp}((\mathbf{t}_i[j], \pi[j]), (-\mathbf{t}_i, \pi[j+1]))(r) : j \in \mathsf{even}(S)\} \tag{11}$$

Protocol 1 .commit

```
Input oracle: $\pi_d \in \mathbb{F}^{n_d}$
Output oracles: $(\pi_{d-1}, \dots, \pi_0) \in \mathbb{F}^{n_{d-1}} \times \dots \times \mathbb{F}^{n_0}$
  • For $i$ from $d-1$ downto 0:
      1. The verifier samples and sends $\alpha_i \xleftarrow{\$} \mathbb{F}$ to the prover
      2. For each index $j \in \mathsf{even}[0, n_{i+1} - 1]$, the prover
          (a) sets $f(X) := \mathsf{interp}(((T_i)[j], \pi_{i+1}[j]), ((-T_i)[j], \pi_{i+1}[j+1]))$
          (b) sets $\pi_i[j] = f(\alpha_i)$
      3. The prover outputs oracle $\pi_i \in \mathbb{F}^{n_i}$.
```

Protocol 2 .query

```
Oracles: $(\pi_d, .., \pi_0)$
Repetition Parameter: $\lambda \in \mathbb{N}$
  • For $j \in [0, \lambda - 1]$
      – The verifier samples an index $\mu_j \xleftarrow{\$} \mathsf{even}[1, n_d - 1]$
      – For $i$ from $d-1$ downto 0, the verifier
          1. queries oracle entries $\pi_{i+1}[\mu_j]$, $\pi_{i+1}[\mu_j + 1]$
          2. computes $p(X) := \mathsf{interpolate}(((T_i)[\mu_j], \pi_{i+1}[\mu_j]), ((-T_i)[\mu_j], \pi_{i+1}[\mu_j + 1]))$
          3. checks that $p(\alpha_i) = \pi_i[\mu_j/2]$
          4. if $i > 0$ and $\mu_j/2 \mod 2 = 0$, update $\mu_j \leftarrow \mu_j/2$, otherwise update $\mu_j \leftarrow \mu_j/2 - 1$.
      – If $\pi_0$ is a valid codeword w.r.t. generator matrix $C_0$, output accept, otherwise output reject
```

Figure 2: The IOPP protocol for foldable codes.

Theorem 1 (Soundness of Basefold IOP)**.** *Let $\lambda \in \mathbb{N}$ be a security parameter, $\pi_d \in \mathbb{F}^{n_d}$, $l \in [0,1]$, and $\boldsymbol{\epsilon} \in [2/\sqrt{n_d}]^3$, with $\alpha_{\boldsymbol{\epsilon}} = \sqrt{1 - \Delta_{C_d} + \boldsymbol{\epsilon}[0]}$, $\beta_{\boldsymbol{\epsilon}} > \sqrt{\alpha_{\boldsymbol{\epsilon}} + \boldsymbol{\epsilon}[1]} + d\boldsymbol{\epsilon}[2]$, and $\beta^l_{\boldsymbol{\epsilon}} \leq \mathsf{negl}(\lambda)$. Then, with probability greater than*

$$1 - \frac{O(dn_d)}{|\mathbb{F}|},$$

over verifier randomness $(r_d, .., r_1)$ in the `commit` *phase, and letting $\{\pi_i \in \mathbb{F}^{n_i} : i \in [d]\}$ be the corresponding oracles sent by the prover, either the verifier accepts with probability less than*

$$\beta^\lambda_{\boldsymbol{\epsilon}},$$

or $\exists P \in \mathbb{F}[X_1, .., X_d]$ *such that* $Enc_{C_0}(P(r_d, .., r_1)) = \pi_0$ *and* $\Delta(Enc_{C_d}(P), \pi_d) < (1 - (\beta_{\boldsymbol{\epsilon}} - d\boldsymbol{\epsilon}[2]))n_d$.

High Level Overview of Proof In the remainder of the paper, we assume all linear codes are *Reed-Muller codes*, that are evaluations of multilinear polynomials. Recall from Lemma 7, that if $\pi_L[S] + r\pi_R[S] = c[S]$ for some $c \in C$, then $\pi_L[S], \pi_R[S]$ differ from $c_L[S], c_R[S]$ in very few locations, where $c_L + rc_R = c$. We will show in Lemma 12, that in this case, c is the encoding of polynomial P, c_L is the encoding of polynomial P_L and c_R is the encoding of polynomial P_R where $P_L + rP_R = P$. Next, in Lemma 13, we prove soundness for just one round of the IOPP, and finally, we show how to extend this to the full, multi-round IOPP.

Proof.

Lemma 12. *Let* $n, k \in \mathbb{N}$ *and let* C *be an* $[n, k]$ *linear error-correcting code. Let* $\beta, \tau_1, \tau_2 \in [0, 1]$ *such that* $\beta - (\tau_1 + \tau_2) > 1 - \Delta_C$. *Let* $\pi_L, \pi_R \in \mathbb{F}^n$ *and* $d = \log_2(n)$. *Suppose that* $S \subset [n]$ *where* $|S| > \beta n$ *and there exists* $P_L, P_R \in \mathbb{F}[X_1, .., X_d]$ *such that*

$$|\{i \in S : \pi_L[i] = \mathsf{Enc}_C(P_L)[i] \land \pi_R[i] = \mathsf{Enc}_C(P_R)[i]\}| > (\beta - \tau_1)n.$$

Suppose further that there exists $c \in C$ *such that*

$$|\{i \in S : \pi_L[S] + r\pi_R[S] = c[S]\}| > (\beta - \tau_2)n. \tag{12}$$

Then

$$c = \mathsf{Enc}_C(P_L + rP_R). \tag{13}$$

Proof. Let $c_L := \mathsf{Enc}_C(P_L), c_R = \mathsf{Enc}_C(P_R)$, $\pi^* := \pi_L + r\pi_R$, $P^*(X_1, .., X_d) = P_L + rP_R$ and $c^* = \mathsf{Enc}_C(P^*)$. Then,

$$\begin{aligned}
&|\{i \in S : \pi^*[i] = c^*[i]\}| \\
&= |\{i \in S : \pi_L[i] + r\pi_R[i] = c_L[i] + rc_R[i]\}| \text{ (By Definition of } \pi^* \text{ and linearity of } C) \\
&\geq |\{i \in S : \pi_L[i] = c_L[i] \land \pi_R[i] = c_R[i]\}| \\
&\geq (\beta - \tau_1)n
\end{aligned}$$

.

Let $c \in C$ be the codeword satisfying Equation 12. Then by a simple counting argument,

$$|\{i \in S : c^*[i] = c[i]\}| > (\beta - (\tau_1 + \tau_2))n.$$

By assumption of the Lemma, $\beta - (\tau_1 + \tau_2) > 1 - \Delta_C$. Therefore, by minimum distance properties of C, $c^* = c$, which completes the proof. □

Next, we combine the previous Lemma (Lemma 12) with our correlated agreement Lemma (Lemma 7) to prove soundness over one single round of the IOPP.

Lemma 13 (One Round Soundness)**.** *Let $d \in \mathbb{N}$ and let C_d, C_{d-1} be a pair of codes from an $[n_d, k_d]$ foldable code family (Definition 2). Let $\pi \in \mathbb{F}^{n_d}, \boldsymbol{\epsilon} \in [2/\sqrt{n_d}]^3$ and let β_ϵ be defined as in Lemma 7. Suppose that $\beta_{\boldsymbol{\epsilon}} - 2d\boldsymbol{\epsilon}[2] > 1 - \Delta_{C_d}$, and that $\exists S \subset [n_d]$ and $c \in C_{d-1}$ such that*

$$|\{i \in S : \mathsf{fold}(\pi, r)[i] = c[i]\}| > (\beta_{\boldsymbol{\epsilon}} - (d-1)\boldsymbol{\epsilon}[2])n_{d-1}. \tag{14}$$

Then with probability greater than $1 - \frac{O(n)}{|\mathbb{F}|}$ (over verifier randomness r) there exists $P_L, P_R \in \mathbb{F}[X_1, .., X_{d-1}]$ such that $c = \mathsf{Enc}_{C_{d-1}}(P_L + rP_R)$ and

$$|\{i \in S : \pi[i] = \mathsf{Enc}_{C_i}(P_L + XP_R)[i]\}| > (\beta_{\boldsymbol{\epsilon}} - d\boldsymbol{\epsilon}[2])n_d. \tag{15}$$

Proof. By Equation 14 and by Definition of fold(Definition 2), it follows that

$$|\{i \in S : \pi_L[i] + r\pi_R[i] = c_L[i] + rc_R[i]\}| > (\beta_{\boldsymbol{\epsilon}} - (d-1)\boldsymbol{\epsilon}[2])n_{d-1},$$

where $(\pi_L, \pi_R), (c_L, c_R)$ are the *unfolding*(Definition 10) of π, c respectively. Define $A_\pi(\boldsymbol{\epsilon})$ as in Lemma 7. Then if $r \notin A_\pi(\boldsymbol{\epsilon})$, it follows from Lemma 7 that,

$$|\{i \in S : \pi_L[i] = c_L[i] \wedge \pi_R[i] = c_R[i]\}| > (\beta_{\boldsymbol{\epsilon}} - d\boldsymbol{\epsilon}[2])n_{d-1}. \tag{16}$$

Let $P_L, P_R \in \mathbb{F}[X_1, .., X_d]$ satisfy $c_L = \mathsf{Enc}_{C_{d-1}}(P_L), c_R = \mathsf{Enc}_{C_{d-1}}(P_R)$. Then, applying Lemma 12,

$$c = \mathsf{Enc}_{C_{d-1}}(P_L + rP_R).$$

By definition of *foldable code* (Definition 2), it follows from Equation 16 that.

$$|\{i \in S : \pi[i] = c[i]\}| > 2(\beta_{\boldsymbol{\epsilon}} - d\boldsymbol{\epsilon}[2])n_{i-1},$$

which implies that

$$|\{i \in S : \pi[i] = \mathsf{Enc}_{C_d}(P_L + XP_R)[i]\}| > (\beta - d\boldsymbol{\epsilon}[2])n_i.$$

By Lemma 7, $|A_\pi(\boldsymbol{\epsilon})| \in O(n)$, and therefore the probability that $r \notin A_\pi$ is greater than $1 - \frac{O(n_d)}{|\mathbb{F}|}$, which completes the proof. □

Next, we show that if the verifier accepts with probability greater than β^l, then this implies the existence of d large sets, one in each oracle, that are consistent with each other with respect to the fold operation.

Lemma 14 (Verifier Queries)**.** *Let $\beta \in [0, 1]$, $l \in \mathbb{N}$. If the verifier accepts the* query *phase with probability greater than β^l then there exists d large sets $\{f_i(S) \subset [n_i] : i \in [d], |f_i(S)| > \beta n_i\}$ such that for all $i \in [d]$,*

$$\mathsf{fold}(\pi_{i+1}, \mathbf{r}[i+1])[f_{i+1}(S)] = \pi_i[\mathsf{even}(f_{i+1}(S))/2] \tag{17}$$

where fold is defined in Equation 11.

Proof. Define the function $Q : \mathsf{even}([n_d]) \to \{0,1\}$ as $Q(\mu) = 1$ if the unique verifier query beginning with $\mu \leftarrow \mathsf{even}(n_d)$ passes the verifier tests and $Q(\mu) = 0$ otherwise. Let $S = Q^{-1}(1)$, i.e S is the set of elements in $\mathsf{even}([n_d])$ that pass the verifier tests. Then, each verifier sample is a Bernoulli trial with success probability $\frac{|S|}{|\mathsf{even}([n_d])|}$. After l queries, the probability of acceptance is $(\frac{|S|}{|\mathsf{even}([n_d])|})^l$. Therefore, if the verifier accepts with probability greater than β^l, then $(\frac{|S|}{\mathsf{even}([n_d])})$ must be larger than β, and so $|S| > \beta|\mathsf{even}([n_d])| = \beta n_{d-1}$. Next, we define $f_i(S)$.

Definition 4 ($f_i(S)$)**.** *Let $d \in \mathbb{N}$ and $S \subset \mathsf{even}([n_d])$. Then,*

$$f_d(S) = S \cup (S+1).$$

For $i \in [d-1]$, $f_i(S)$ satisfies the following:

$$\mathsf{even}(f_i(S)) = \{\mathsf{even}(\{j/2, j/2-1\}) : j \in \mathsf{even}(f_{i+1}(S)\}$$

$$\mathsf{odd}(f_i(S)) = \mathsf{even}(f_i(S)) + 1.$$

To complete the Lemma, we need to prove that for each $i \in [0,d]$,

$$\mathsf{fold}(\pi_i, \mathbf{r}[i])[f_i(S)] = \pi_{i-1}[\mathsf{even}(f_i(S))/2].$$

For each $\mu \in S$, let $((\mu_d, \mu_d+1), .., (\mu_1, \mu_1+1))$ be the unique set of queries associated with μ. Then, by definition of fold (Definition 11), for each $i \in [1,d]$

$$\mathsf{fold}(\pi_i, r_i)[f_i(S)] = \{\pi_{i,L}[\mu_i] + r_i]\pi_{i,R}[\mu_i+1]\} \tag{18}$$

Furthermore,

$$\pi_{i,L}[\mu_i] + \mathbf{r}[i]\pi_{i,R}[\mu_i+1] = \pi_{i-1}[\mu_i/2]. \tag{19}$$

Combining Equations 18 and 19 gives

$$\mathsf{fold}(\pi_i, r_i)[f_i(S)] = \pi_{i-1}[\mathsf{even}(f_i(S)/2)],$$

which completes the proof of Lemma 14. □

Finally, we are ready to prove the main statement of the Theorem. Suppose by contradiction that the verifier accepts with probability $> \beta^l$ but there does not exist $P \in \mathbb{F}[X_1, .., X_d]$ such that $\mathsf{Enc}_{C_0}(P(r_d, .., r_1)) = \pi_0$ and $\Delta(\mathsf{Enc}_{C_d}(P), \pi_d) < (1 - (\beta_{\boldsymbol{\epsilon}} - d\boldsymbol{\epsilon}[2]))$. But by Lemma 14, there *does* exist a set $S \subset \mathsf{even}(n_d)$ such that for each $i \in [0,d]$,

$$\mathsf{fold}(\pi_i, \mathbf{r}[i])[f_i(S)] = \pi_{i-1}[\mathsf{even}(f_i(S))/2].$$

Therefore, since $\pi_0 \in C_0$ there must exist a round where Equation 14 holds but Equation 15 does not. By Lemma 13, this only happens with probability $\frac{O(n)}{|\mathbb{F}|}$. Taking the union bound over d rounds completes the proof. □

Acknowledgements

I would like to thank the following people: Binyi Chen for feedback on earlier drafts and ongoing helpful discussions; Ulrich Haböck for checking correctness of the proof and giving very useful feedback on how to present it; Ron Rothblum and Ben Fisch for general guidance and advice; Michael Rosenberg and Paul Grubbs for edits and feedback; and Giacomo Fenzi for early brainstorming sessions.

Additionally, I would like to thank AR for coming up with the name Khatam; Oded Goldreich for useful advice on how to write a proof; Joan Feigenbaum for gifting me stacks of old conference proceedings, from which I drew inspiration on writing; and KL for gifting me the fountain pen I used to figure out the solution.

References

[1] Scott Ames, Carmit Hazay, Yuval Ishai, and Muthuramakrishnan Venkitasubramaniam. "Ligero: Lightweight Sublinear Arguments Without a Trusted Setup". In: *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security.* CCS '17. Dallas, Texas, USA: Association for Computing Machinery, 2017, 2087–2104. ISBN: 9781450349468. DOI: 10.1145/3133956.3134104. URL: https://doi.org/10.1145/3133956.3134104.

[2] Gal Arnon, Alessandro Chiesa, Giacomo Fenzi, and Eylon Yogev. *WHIR: Reed–Solomon Proximity Testing with Super-Fast Verification.* Cryptology ePrint Archive, Paper 2024/1586. 2024. URL: https://eprint.iacr.org/2024/1586.

[3] Eli Ben-Sasson, Iddo Bentov, Yinon Horesh, and Michael Riabzev. "Fast Reed-Solomon Interactive Oracle Proofs of Proximity". In: *45th International Colloquium on Automata, Languages, and Programming (ICALP 2018).* Ed. by Ioannis Chatzigiannakis, Christos Kaklamanis, Dániel Marx, and Donald Sannella. Vol. 107. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2018, 14:1–14:17. ISBN: 978-3-95977-076-7. DOI: 10.4230/LIPIcs.ICALP.2018.14. URL: https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.ICALP.2018.14.

[4] Eli Ben-Sasson, Dan Carmon, Yuval Ishai, Swastik Kopparty, and Shubhangi Saraf. "Proximity Gaps for Reed–Solomon Codes". In: *J. ACM* 70.5 (Oct. 2023). ISSN: 0004-5411. DOI: 10.1145/3614423. URL: https://doi.org/10.1145/3614423.

[5] Eli Ben-Sasson, Alessandro Chiesa, Ariel Gabizon, Michael Riabzev, and Nicholas Spooner. *Short Interactive Oracle Proofs with Constant Query Complexity, via Composition and Sumcheck*. Cryptology ePrint Archive, Report 2016/324. 2016. URL: https://eprint.iacr.org/2016/324.

[6] Eli Ben-Sasson, Alessandro Chiesa, Michael Riabzev, Nicholas Spooner, Madars Virza, and Nicholas P. Ward. "Aurora: Transparent Succinct Arguments for R1CS". In: *Advances in Cryptology – EUROCRYPT 2019, Part I*. Ed. by Yuval Ishai and Vincent Rijmen. Vol. 11476. Lecture Notes in Computer Science. Darmstadt, Germany: Springer, Cham, Switzerland, 2019, pp. 103–128. DOI: 10.1007/978-3-030-17653-2_4.

[7] Eli Ben-sasson, Lior Goldberg, Swastik Kopparty, and Shubhangi Saraf. *DEEP-FRI: Sampling outside the box improves soundness*. Mar. 2019. DOI: 10.48550/arXiv.1903.12243.

[8] Eli Ben-Sasson, Oded Goldreich, Prahladh Harsha, Madhu Sudan, and Salil Vadhan. "Robust PCPs of proximity, shorter PCPs and applications to coding". In: *Proceedings of the thirty-sixth annual ACM symposium on Theory of computing*. 2004, pp. 1–10.

[9] Eli Ben-Sasson, Swastik Kopparty, and Shubhangi Saraf. "Worst-case to average case reductions for the distance to a code". In: *Proceedings of the 33rd Computational Complexity Conference*. CCC '18. San Diego, California: Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2018. ISBN: 9783959770699.

[10] Martijn Brehm, Binyi Chen, Ben Fisch, Nicolas Resch, Ron D. Rothblum, and Hadas Zeilberger. *Blaze: Fast SNARKs from Interleaved RAA Codes*. Cryptology ePrint Archive, Paper 2024/1609. 2024. URL: https://eprint.iacr.org/2024/1609.

[11] Binyi Chen, Benedikt Bünz, Dan Boneh, and Zhenfei Zhang. *HyperPlonk: Plonk with Linear-Time Prover and High-Degree Custom Gates*. Cryptology ePrint Archive, Report 2022/1355. 2022. URL: https://eprint.iacr.org/2022/1355.

[12] Benjamin E. Diamond and Jim Posen. *Polylogarithmic Proofs for Multilinears over Binary Towers*. Cryptology ePrint Archive, Paper 2024/504. https://eprint.iacr.org/2024/504. 2024. URL: https://eprint.iacr.org/2024/504.

[13] Irit Dinur and Omer Reingold. "Assignment testers: Towards a combinatorial proof of the PCP theorem". In: *SIAM Journal on Computing* 36.4 (2006), pp. 975–1024.

[14] Yiwen Gao, Haibin Kan, and Yuan Li. *Linear Proximity Gap for Reed-Solomon Codes within the 1.5 Johnson Bound*. Cryptology ePrint Archive, Paper 2024/1810. 2024. URL: https://eprint.iacr.org/2024/1810.

[15] Alexander Golovnev, Jonathan Lee, Srinath Setty, Justin Thaler, and Riad S. Wahby. *Brakedown: Linear-time and field-agnostic SNARKs for R1CS*. Cryptology ePrint Archive, Paper 2021/1043. 2021. URL: https://eprint.iacr.org/2021/1043.

[16] Yanpei Guo, Xuanming Liu, Kexi Huang, Wenjie Qu, Tianyang Tao, and Jiaheng Zhang. *DeepFold: Efficient Multilinear Polynomial Commitment from Reed-Solomon Code and Its Application to Zero-knowledge Proofs*. Cryptology ePrint Archive, Paper 2024/1595. 2024. URL: https://eprint.iacr.org/2024/1595.

[17] Ulrich Haböck. *Basefold in the List Decoding Regime*. Cryptology ePrint Archive, Paper 2024/1571. 2024. URL: https://eprint.iacr.org/2024/1571.

[18] Aniket Kate, Gregory M. Zaverucha, and Ian Goldberg. "Constant-Size Commitments to Polynomials and Their Applications". In: *Advances in Cryptology – ASIACRYPT 2010*. Ed. by Masayuki Abe. Vol. 6477. Lecture Notes in Computer Science. Singapore: Springer, Berlin, Heidelberg, Germany, 2010, pp. 177–194. DOI: 10.1007/978-3-642-17373-8_11.

[19] Tohru Kohrita and Patrick Towa. *Zeromorph: Zero-Knowledge Multilinear-Evaluation Proofs from Homomorphic Univariate Commitments*. Cryptology ePrint Archive, Paper 2023/917. 2023. URL: https://eprint.iacr.org/2023/917.

[20] Carsten Lund, Lance Fortnow, Howard Karloff, and Noam Nisan. "Algebraic methods for interactive proof systems". In: *J. ACM* 39.4 (1992), 859–868. ISSN: 0004-5411. DOI: 10.1145/146585.146605. URL: https://doi.org/10.1145/146585.146605.

[21] Charalampos Papamanthou, Elaine Shi, and Roberto Tamassia. "Signatures of Correct Computation". In: *TCC 2013: 10th Theory of Cryptography Conference*. Ed. by Amit Sahai. Vol. 7785. Lecture Notes in Computer Science. Tokyo, Japan: Springer, Berlin, Heidelberg, Germany, 2013, pp. 222–242. DOI: 10.1007/978-3-642-36594-2_13.

[22] Omer Reingold, Guy N. Rothblum, and Ron D. Rothblum. "Constant-round interactive proofs for delegating computation". In: *48th Annual ACM Symposium on Theory of Computing*. Ed. by Daniel Wichs and Yishay Mansour. Cambridge, MA, USA: ACM Press, 2016, pp. 49–62. DOI: 10.1145/2897518.2897652.

[23] Hang Su, Qi Yang, and Zhenfei Zhang. *Jolt-b: recursion friendly Jolt with basefold commitment*. Cryptology ePrint Archive, Paper 2024/1131. https://eprint.iacr.org/2024/1131. 2024. URL: https://eprint.iacr.org/2024/1131.

[24] Alexander Vlasov and Konstantin Panarin. *Transparent Polynomial Commitment Scheme with Polylogarithmic Communication Complexity*. Cryptology ePrint Archive, Report 2019/1020. 2019. URL: https://eprint.iacr.org/2019/1020.

[25] Hadas Zeilberger, Binyi Chen, and Ben Fisch. *BaseFold: Efficient Field-Agnostic Polynomial Commitment Schemes from Foldable Codes*. Cryptology ePrint Archive, Paper 2023/1705. https://eprint.iacr.org/2023/1705. 2023. URL: https://eprint.iacr.org/2023/1705.

Appendix

A Proofs Contd

Proof of Lemma 8. Let $\mathcal{S} \subset 2^{[n]}$ be the set of large subsets where for each $S \in \mathcal{S}$, $|S| > \beta n$ and for each $S_1, S_2 \in S, S_1 \cap S_2 < \alpha n$. Recall that our goal is bound the size of $\mathcal{S}$. We do this by constructing a bipartite graph, G, where right vertices are labeled by *elements* in $[n]$ and left vertices are labeled by *subsets* of $[n]$. We place an edge between a vertex $v \in L$ and a vertex $w \in R$ if the element associated with w is contained inside the set associated with v. This implies that that any vertex in L has more than βn neighbors but *shares* less than αn neighbors with any other vertex in L. In other words, letting $N(v)$ denotes the neighbor set of $v \in L$:

- $\forall v \in L, |N(v)| > \beta n$
- $\forall v_1, v_2 \in L, |N(v_1) \cap N(v_2)| < \alpha n$

We prove in the following Lemma, that in a bipartite graph with these two properties,

$$|L| \leq \mathsf{poly}(n)$$

□

Lemma 15. *Let $G = ((L, R), E)$ be a bipartite graph. For each vertex v, let $d(v)$ be the degree of v. Let $\alpha, \beta \in [0, 1), \beta > \sqrt{\alpha}$. For $v \in L$, let $N_v = \{w \in R : (v, w) \in E\}$ be the neighborhood of v. Then if for all $v \in L$,*

$$|N(v)| > \beta n$$

and if for each two $v_1, v_2 \in L$,

$$|N_{v_1} \cap N_{v_2}| \leq \alpha \cdot n$$

Then,

$$|L| \in O(n)$$

Proof. This proof is a generalization of the proof of the Johnson bound [8]. For $v_1, v_2 \in L$ and $w \in R$, define an "angle" as a triple (v_1, w, v_2), such that there is an edge between v_1 and w, and between w and v_2. Let $d(w)$ be the degree of node w. Then the number of angles in the graph is equal to

$$\sum_{v \in R} \binom{d(v)}{2}$$

Next, we make the following claim, which puts $\sum_{v \in R} \binom{d(v)}{2}$ in terms of $\sum_{v \in R} d(v)$.

Claim 2.

$$\sum_{v \in R} \binom{d(v)}{2} > |R| \binom{(\sum_{v \in R} d(v))/|R|}{2}.$$

Since each vertex in L has more than βn neighbors, it follows that the sum of the degrees of all vertices in R is greater than $|L| \cdot \beta |R|$, i.e. $\sum_{v \in R} d(v) \geq |L| \cdot \beta |R|$. Thus, by Claim 2,

$$\sum_{v \in R} \binom{d(v)}{2} \geq n \binom{\beta |L|}{2}$$

On the other hand, any two vertices in L can share at most $\alpha |R| = \alpha n$ vertices in R. Thus the number of angles between any two vertices is at most αn, and so the total number of angles in G is *at most*

$$\binom{|L|}{2} \alpha n$$

Combining the two inequalities, we have

$$\binom{\beta |L|}{2} \leq \binom{|L|}{2} \alpha.$$

Solving for L we have,

$$\frac{\beta |L| (\beta |L| - 1)}{2} \leq \frac{\alpha |L| (|L| - 1)}{2} \tag{20}$$

$$\beta |L| (\beta |L| - 1) \leq \alpha |L| (|L| - 1) \tag{21}$$

$$\beta (\beta |L| - 1) \leq \alpha (|L| - 1) \tag{22}$$

$$\beta^2 |L| - \beta \leq \alpha |L| - \alpha \tag{23}$$

$$|L| (\beta^2 - \alpha) \leq \beta - \alpha \tag{24}$$

$$\tag{25}$$

Therefore,

$$|L| \leq \frac{\beta - \alpha}{\beta^2 - \alpha}$$

[8] https://www.cs.cmu.edu/~venkatg/teaching/au18-coding-theory/lec-scribes/list-decoding.pdf

Let $\epsilon = \beta^2 - \alpha$. Then we can rewrite the above equation as

$$|L| \leq \frac{\sqrt{\alpha + \epsilon} - \alpha}{\epsilon}.$$

The right hand size of this equation is at its highest value when ϵ is at its lowest value. Because ϵ represents fraction of elements of a codeword of length n, its smallest value is $1/n$. In that case,

$$|L| \leq n(\sqrt{\alpha + 1/n} - \alpha),$$

and so $L \in O(n)$. □

Proof of Claim 1. By definition of factorial,

$$\begin{aligned}
|R|\binom{(\sum d(v)/|R|)}{2} =&|R|\frac{(\sum d(v)/R)((\sum d(v)/R) - 1)}{2} && (26)\\
=& \frac{(\sum d(v))((\sum d(v)/R) - 1)}{2} && (27)\\
=& \frac{(\sum d_v)^2/R - (\sum d_v)}{2} && (28)\\
=& \frac{1}{2}(\sum d_v)^2/R - (\sum d_v) && (29)\\
& && (30)
\end{aligned}$$

On the other hand

$$\sum \binom{d(v)}{2} = \frac{1}{2}\sum(d(v)^2 - d(v)) = \frac{1}{2}(\sum d(v)^2 - \sum d(v))$$

Since $R > 1$, it follows that

$$\frac{\sum d(v)^2}{|R|} < \sum d(v)^2 < (\sum d(v))^2,$$

and therefore

$$\sum_{v \in R} \binom{d(v)}{2} > |R|\binom{(\sum_{v \in R} d(v))/|R|}{2}.$$

□