Electronic Colloquium on Computational Complexity, Report No. 134 (2017)

Fast Reed-Solomon Interactive Oracle Proofs of Proximity

Eli Ben-Sasson* Iddo Bentov† Ynon Horesh* Michael Riabzev*

September 7, 2017

Abstract

The family of Reed-Solomon (RS) codes plays a prominent role in the construction of quasi-linear probabilistically checkable proofs (PCPs) and interactive oracle proofs (IOPs) with perfect zero knowledge and polylogarithmic verifiers. The large concrete computational complexity required to prove membership in RS codes is one of the biggest obstacles to deploying such PCP/IOP systems in practice.

To advance on this problem we present a new interactive oracle proof of proximity (IOPP) for RS codes; we call it the *Fast RS IOPP* (FRI) because (i) it resembles the ubiquitous Fast Fourier Transform (FFT) and (ii) the arithmetic complexity of its prover is strictly linear and that of the verifier is strictly logarithmic (in comparison, FFT arithmetic complexity is quasi-linear but not strictly linear). Prior RS IOPPs and PCPs of proximity (PCPPs) required super-linear proving time even for polynomially large query complexity.

For codes of block-length n, the arithmetic complexity of the (interactive) FRI prover is less than $6 \cdot n$, while the (interactive) FRI verifier has arithmetic complexity $\leq 21 \cdot \log n$, query complexity $2 \cdot \log n$ and constant soundness — words that are δ-far from the code are rejected with probability $\min\{\delta \cdot (1 - o(1)), \delta_0\}$ where δ_0 is a positive constant that depends only on the code rate. The particular combination of query complexity and soundness obtained by FRI is better than that of the quasilinear PCPP of [Ben-Sasson and Sudan, SICOMP 2008], even with the tighter soundness analysis of [Ben-Sasson et al., STOC 2013; ECCC 2016]; consequently, FRI is likely to facilitate better concretely efficient zero knowledge proof and argument systems.

Previous concretely efficient PCPPs and IOPPs suffered a constant *multiplicative* factor loss in soundness with each round of "proof composition" and thus used at most $O(\log \log n)$ rounds. We show that when δ is smaller than the unique decoding radius of the code ($\delta < (1 - \rho)/2$), FRI suffers only a negligible *additive* loss in soundness. This observation allows us to increase the number of "proof composition" rounds to $\Theta(\log n)$ and thereby reduce prover and verifier running time for fixed soundness.

*Technion — Israel Institute of Technology, Haifa, Israel; supported by the Israel Science Foundation (grant # 1501/14) and the US–Israel Binational Science Foundation

†Cornell University, Ithaca, NY, USA

ISSN 1433-8092

1 Introduction

The family of Reed-Solomon (RS) codes is a fundamental object of study in algebraic coding theory and theoretical computer science [RS60]. For an evaluation set S of n elements in a finite field $\mathbb{F}$ and a rate parameter $\rho \in (0, 1]$, the code $\mathsf{RS}[\mathbb{F}, S, \rho]$ is the space of functions $f : S \to \mathbb{F}$ that are evaluations of polynomials of degree $d < \rho n$ [RS60]. The *RS proximity problem* assumes a verifier has oracle access to $f : S \to \mathbb{F}$, and asks that verifier to distinguish, with "large" confidence and "small" query complexity, between the case that f is a codeword of $\mathsf{RS}[\mathbb{F}, S, \rho]$ and the case that f is δ-far in relative Hamming distance from all codewords. This problem has been addressed in several different computational models (surveyed next and summarized in Table 1), and is also the focus of this paper.

RS proximity testing: When no additional data is provided to the verifier, the RS proximity problem is commonly called a *testing* problem, and has been first defined and addressed by Rubinfeld and Sudan in [RS92] (cf. [FS95]). They showed that $d + 1$ queries are necessary and sufficient to solve the problem: codewords are accepted by their tester with probability 1 whereas functions that are δ-far from the code are rejected with probability $\geq \delta$. Since no additional information is provided to the verifier in this model, we may say that a prover attempting to convince the verifier that $f \in \mathsf{RS}[\mathbb{F}, S, \rho]$ spends zero computational effort, zero rounds of interaction and produces a proof of length 0.

RS proximity verification — PCPP model: Probabilistically checkable proofs of proximity (PCPP) [BGH$^+$06, DR04] relax the testing problem to a setting in which the verifier is given oracle access also to an auxiliary proof, called a PCPP and denoted π. This PCPP is produced by the prover, which is given $f \in \mathsf{RS}[\mathbb{F}, S, \rho]$ as input. The time required to produce π is the *prover complexity* and $|\pi|$ is called the *proof length*[1]; similarly, *verifier complexity* is the total time required to generate queries and check query-answers. The techniques used to prove the celebrated PCP Theorem [ALM$^+$98, AS98] also show that the proximity problem can be solved with constant query complexity and proof length and prover complexity $n^{O(1)}$, or with proof length $n^{1+\epsilon}$ and query complexity $(\log n)^{O(1/\epsilon)}$ [BFLS91]. The current state of the art in the PCPP model gives proofs of length $\tilde{O}(n) \stackrel{\triangle}{=} n \cdot \log^{O(1)} n$ with constant query complexity [BS08, Din07] and prover complexity $\tilde{O}(n)$ [BCGT13]; verifier complexity is $\mathsf{poly} \log n$ [BGH$^+$05, Mie09].

RS proximity verification — IOPP model: Interactive oracle proofs of proximity (IOPP), formally introduced in [BCG$^+$16] and, independently, in [RRR16] (under the name "probabilistically checkable interactive proofs of proximity"), generalize IPs, PCPs and interactive PCPs (IPCP) [KR08]. As in an IP and IPCP, several rounds of interaction are used in which the prover sends messages $\pi_1, \pi_2, \ldots, \pi_{\mathsf{r}}$ in response to successive verifier messages. As in a PCP and IPCP, the verifier is not required to read prover messages in entirety but rather may query them at random locations (in an IPCP, verifier must read the full messages $\pi_2, \ldots$ but may query π_1 randomly); the query complexity is the total number of entries read from f and $\pi_1, \pi_2, \ldots, \pi_{\mathsf{r}}$. The prover is provided with $f \in \mathsf{RS}[\mathbb{F}, S, \rho]$ as input and *prover complexity* is the total time required to produce all (prover) messages[2], while *proof length* is generalized from the PCPP setting to the IOPP setting and defined as $|\pi_1| + \ldots + |\pi_{\mathsf{r}}|$. IOPPs can be used to "replace" PCPP proof composition with more rounds of interaction, and thereby reduce proof length and prover complexity without compromising soundness. In particular, the IOPP version of the aforementioned PCPP constructions reduces

[1] Typically π is a sequence of elements in $\mathbb{F}$. Therefore, proof length is measured over the alphabet $\mathbb{F}$.

[2] Notice that prover complexity does not include the time needed to produce f.

proof length to $O(n)$ with no change to soundness and/or query complexity [BBGR16a, BCG+16]. In spite of the shorter proof length, prover complexity in prior works was $\Theta(n \mathsf{poly} \log n)$ due to a limitation on the number of proof-composition rounds, explained in Section 2.1.

	prover comp.	proof length	verifier comp.	query comp.	round comp.
1. Testing [RS92]	0	0	$\tilde{O}(\rho n)$	ρn	0
2. PCP [ALM+98, AS98]	$n^{O(1)}$	$n^{O(1)}$	$n^{O(1)}$	$O\left(\frac{1}{\delta}\right)$	1
3. PCP [BFL90, BFLS91]	$n^{1+\epsilon}$	$n^{1+\epsilon}$	$\frac{1}{\delta}\log^{O(1/\epsilon)} n$	$\frac{1}{\delta}\log^{O(1/\epsilon)} n$	1
4. PCPP [BS08, BGH+06, BCGT13]	$\geq n\log^{1.5} n$	$\geq n\log^{1.5} n$	$\geq \frac{1}{\delta}\log^{5.8} n$	$\frac{1}{\delta}\log^{5.8} n$	1
5. PCPP [Din07, Mie09]	$n\log^{c} n$	$n\log^{c} n$	$\frac{1}{\delta}\log^{c} n$	$O\left(\frac{1}{\delta}\right)$	1
6. IOPP [BCF+16, BBGR16b]	$n\log^{c} n$	$> 4\cdot n$	$\frac{1}{\delta}\log^{c} n$	$O\left(\frac{1}{\delta}\right)$	$\log\log n$
7. This work	$< 6\cdot n$	$< \frac{n}{3}$	$\leq 21\cdot\log n$	$2\log n$	$\frac{\log n}{2}$

Table 1: Comparison of RS proximity protocols. For concreteness, all results stated for binary additive RS codes with rate $\rho = 1/8$ evaluated over a sufficiently large set $S, |S| = n$ satisfying $n/|\mathbb{F}| < 0.001$ with proximity parameter $\delta < \delta_0$ (cf. Theorem 1.2) and soundness at least 0.99δ; i.e., the rejection probability of δ-far words is at least 0.99δ for $\delta < \delta_0$ (in particular, smaller δ leads to smaller soundness). Exponents for the 4th row taken from [BCGT13]; the various exponents c in the 5th and 6th row have not been estimated in prior works but are greater than the respective exponents in the 4th row.

1.1 Main results

We present a new IOPP for RS codes, called the Fast RS IOPP (FRI) because of its resemblance to the Fast Fourier Transform (FFT) [CT65]; its analysis relies on the quasi-linear RS-PCPP [BS08] (see Section 2.1). FRI the first RS-IOPP to have (i) *strictly linear* arithmetic for the prover with (ii) *strictly logarithmic* arithmetic complexity for the verifier and (iii) *constant* soundness. We start by recalling IOPP systems as described in [BCF+16, Section 3.2].

IOP An *Interactive Oracle Proof (IOP)* system S is defined by a pair of interactive randomized algorithms $\mathsf{S} = (\mathsf{P}, \mathsf{V})$, where P denotes the prover and V the verifier. On input x of length n, the number of rounds of interaction is denoted by $\mathsf{r}(n)$ and called the *round complexity* of the system. The *query complexity* of the protocol, denoted $\mathsf{q}(n)$, is the number of entries read by V and the *proof length*, denoted $\ell(n)$, is the sum of lengths of all messages sent by the prover. We denote by $\langle \mathsf{P} \leftrightarrow \mathsf{V} \rangle(x)$ the output of V after interacting with P on input x; this output is either accept or reject. An IOP is said to be *transparent* (or have *public randomness*) if all messages sent from the verifier are public random coins and all queries are determined by public coins, which are broadcast to the prover (such protocols are also known as Arthur-Merlin protocols [Bab85]).

IOPP As its name suggests, an *IOP of proximity (IOPP)* is the natural generalization of a PCP of Proximity (PCPP) to the IOP model. An IOPP for a family of codes $\mathcal{C}$is a pair (P, V) of randomized algorithms, called *prover* and *verifier*, respectively. Both parties receive as common input a specification of a code $C \in \mathcal{C}$ which we view as a set of functions $C = \{f : S \to \Sigma\}$ for a finite set S and alphabet Σ. We also assume that the verifier has oracle access to a function

$f^{(0)} : S \to \Sigma$ and that the prover receives the same function as explicit input. The number of rounds of interaction, or *round complexity*, is denoted by r, *query complexity* is denoted by q.

Definition 1.1 (Interactive Oracle Proof of Proximity (IOPP)). *An* r*-round Interactive Oracle Proof of Proximity (IOPP)* $\mathsf{S} = (\mathsf{P}, \mathsf{V})$ *is a* $(\mathsf{r}+1)$*-round IOP. We say* S *is a (*r*-round) IOPP for the error correcting code* $C = \{f : S \to \Sigma\}$ *with soundness bounds* $\mathbf{s}^-, \mathbf{s}^+ : (0,1] \to [0,1]$ *with respect to distance measure* Δ*, if the following conditions hold:*

- **First message format:** *the first prover message, denoted* $f^{(0)}$*, is a purported codeword of* C*, i.e.,* $f^{(0)} : S \to \Sigma$
- **Completeness:** $\Pr\left[\langle \mathsf{P} \leftrightarrow \mathsf{V}\rangle = \mathsf{accept} | \Delta\left(f^{(0)}, C\right) = 0\right] = 1$
- **Soundness lower bound:** *For any* P^*, $\Pr\left[\langle \mathsf{P}^* \leftrightarrow \mathsf{V}\rangle = \mathsf{reject} | \Delta\left(f^{(0)}, C\right) = \delta\right] \geq \mathbf{s}^-(\delta)$
- **Soundness upper bound:** *For any* $\delta > 0$ *there exists a prover* P^*_δ *and* $f^{(0)}_\delta$ *satisfying* $\Delta\left(f^{(0)}_\delta, C\right) = \delta$ *and* $\Pr\left[\langle \mathsf{P}^*_\delta \leftrightarrow \mathsf{V}\rangle = \mathsf{reject} | f^{(0)} = f^{(0)}_\delta\right] \leq \mathbf{s}^+(\delta)$.

The sum of lengths of all prover messages, except for $f^{(0)}$*, is the IOPP* proof length*; the time required to generate all messages except for* $f^{(0)}$ *is the* prover complexity*. The IOPP* query complexity *is the total number of queries to all messages, including* $f^{(0)}$ *and the* decision complexity *is the time required by the verifier to reach it's verdict, once the queries and query-answers are provided as inputs.*

By definition $\mathbf{s}^-(\delta) \leq \mathbf{s}^+(\delta)$. The concrete rejection probability of $f^{(0)}$ that is δ-far from C lies between $\mathbf{s}^-(\delta)$ and $\mathbf{s}^+(\delta)$. Therefore an important research question regarding concrete efficiency is to minimize $\mathbf{s}^+(\cdot) - \mathbf{s}^-(\cdot)$ (we shall get back to this when we discuss Main Theorem 1.2).

Main Theorem The finite field of size q is denoted here by $\mathbb{F}_q$; when q is clear from context we omit it. A field is called *binary* if $q = 2^m, m \in \mathbb{N}$. A subset S of a binary field is an *additive coset* if it is a coset of a subgroup of the additive group $\mathbb{F}^+$, i.e., if S is an additive shift of an $\mathbb{F}_2$-linear space contained in $\mathbb{F}_q$. The *binary additive RS code family* is the collection of codes $\mathsf{RS}[\mathbb{F}, S, \rho]$ where $\mathbb{F}$ is a binary field and S an additive coset. This family of codes is one for which quasilinear PCPP were defined in [BS08], and our main theorem is stated for it (see Table 1).

Theorem 1.2 (Main — FRI properties). *The binary additive RS code family of rate* $\rho = 2^{-\mathcal{R}}, \mathcal{R} \geq 2, \mathcal{R} \in \mathbb{N}$ *has an IOPP (*FRI*) with the following properties, where* n *denotes blocklength (which equals Prover's input length for a fixed* $\mathsf{RS}[\mathbb{F}, H, \rho]$ *code):*

- **Prover:** *prover complexity is less than* $6n$ *arithmetic operations in* $\mathbb{F}$*; proof length is less than* $n/3$ *field elements and round complexity is at most* $\frac{\log n}{2}$*;*
- **Verifier:** *query complexity is* $2\log n$*; the verifier decision is computed using at most* $21\log n$ *arithmetic operations over* $\mathbb{F}$
- **Soundness lower bound:** *There exists* $\delta_0 \geq \frac{1}{4}(1 - 3\rho) - \frac{1}{\sqrt{n}}$ *such that every* f *that is* δ*-far in relative Hamming distance from the code, is rejected with probability at least* $\min\{\delta, \delta_0\} - \frac{3n}{|\mathbb{F}|}$

- **Soundness upper bound** *For all $\delta > 0$ there exists f that is δ-far from the code and a prover that causes the verifier to reject f with probability at most $\delta + \frac{4}{|\mathbb{F}|}$*

- **Parallelization:** *Each prover-message can be computed in $O(1)$ time on a Parallel Random Access Machine (PRAM) with common read and exclusive write (CREW), assuming a single $\mathbb{F}$ arithmetic operation takes unit time.*

Remark 1.3 (Extension to arbitrary rate)**.** *Generalizing Theorem 1.2 to arbitrary rate $\rho \in (0, 1]$ can be done as described in [BS08, Proposition 6.13] (cf. remark 6.2 there); this leads to slightly larger constants in the prover and verifier complexity. For practical applications like ZK-IOPs [BCGV16, BCF+16], rates of the form stated in the theorem above suffice.*

Remark 1.4 (Extension to smooth multiplicative subgroups)**.** *We call a multiplicative group $H \subset \mathbb{F}_q$* smooth *if its order ($|H|$) is 2^k for $k \in \mathbb{N}$. The family of* smooth *RS codes and rate ρ is the set of $\mathsf{RS}[\mathbb{F}_q, H, \rho]$ with smooth H. Theorem 1.2 holds with respect to the family of smooth RS codes (of rate as above, cf. Remark 1.3), with somewhat smaller constants than 6 and 21 for the prover and verifier arithmetic complexity (as explained in Remark 4.8); see Section 2.1 for a high-level overview of the smooth case and Remark 3.1 for more details on modifying the protocol to this case. The protocol can be further generalized to groups of order c^k for constant c (perhaps with different arithmetic complexity constants), details omitted.*

Soundness upper bounds — discussion Several notable works from the PCP literature — starting with [Hås01] — present lower bounds on soundness that nearly-match soundness upper bounds. The tightness of the Hadammard linearity test [BLR93] is studied in [BCH+96]. The study of soundness upper bounds for quasi-linear PCP/PCPP systems was initiated in [BBGR16a], motivated by the quest for concretely efficient proof systems for succinct verification of computation. As argued there, the study of soundness upper bounds leads to (i) smaller concrete communication complexity, (ii) interesting algebraic questions and (iii) new and more efficient protocols.

In fact, the RS-IOPP presented here emerged from the study of soundness upper bounds initiated in that paper, and our hope is that the soundness upper bounds presented here will advance the search for new and improved protocols.

The upper bound above implies that for low values of δ, smaller than δ_0, our soundness lower bounds are nearly tight and equal $\approx \delta$. We conjecture that the actual soundness is close to the upper bound for all values of δ.

Conjecture 1.5. *The soundness upper bound in Theorem 1.2 is nearly tight, i.e., the rejection probability of any f that is δ-far from the code is at least $\delta - (n/|\mathbb{F}|)^{O(1)}$.*

1.2 Improved concrete efficiency

Ben-Sasson et al. defined the *concrete efficiency threshold (CET)* as a way to formalize the "practicality" of a PCPP construction [BCGT13], and we use the generalization of this definition to the IOPP setting below [BBGR16a]. The CET measure assigns a number (or ∞) to each IOPP system for a family of codes. Lower thresholds are considered better. The CET of an IOPP takes into account both (i) the query complexity $\mathsf{q}_{\epsilon,\delta}$ needed to reject with probability ϵ words that are δ-far from the code, and (ii) the proof overhead, i.e., the ratio of total proof length to message-length. Thus, to improve (i.e., decrease) the concrete efficiency threshold, one should build IOPPs that simultaneously decrease both $\mathsf{q}_{\epsilon,\delta}$ and total proof length.

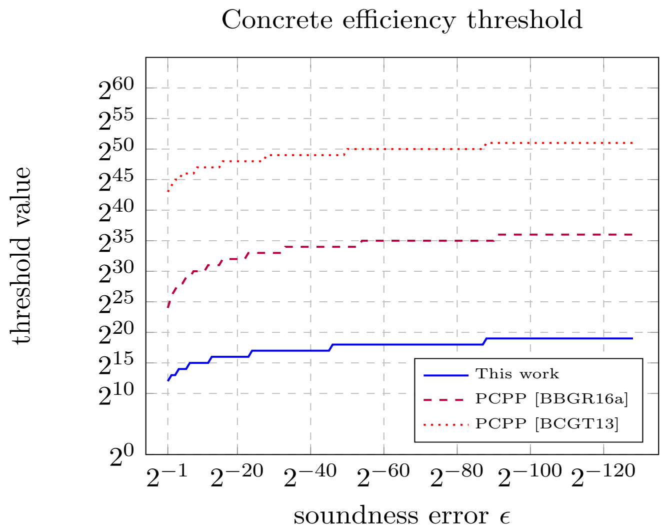

Figure 1: The concrete efficiency threshold for RS codes as a function of the soundness $\epsilon \in (2^{-1}, \ldots, 2^{-128})$. We choose the same setting as [BCGT13], namely, code rate is $\rho = 1/8$ and the proximity parameter δ is a third of the code distance, i.e., $\delta = (1 - \rho)/3 = 7/24$.

Recall that a PCPP is a 1-round IOPP, hence the following definition applies to PCPPs as a special case.

Definition 1.6 (Concrete efficiency threshold)**.** *Fix an IOPP system* $\mathsf{S} = (\mathsf{P}, \mathsf{V})$ *for a family of error correcting codes* $\mathcal{C} = \{C_k\}$ *where* C_k *has message-length* k *and block-length* $n(k)$*. Let* $\ell(k)$ *denote the IOPP proof length for* C_k *Let* $q_{\epsilon,\delta}(k)$ *denote the minimal query complexity needed to obtain soundness error* $\leq \epsilon$ *for proximity parameter* δ *for* C_k*.*

A family of error correcting codes $\mathcal{C}$ *is said to have a* concrete (soundness) efficiency threshold $\mathsf{t}_{\epsilon,\delta}$ *if for any code* $C_k \in \mathcal{C}, k \geq \mathsf{t}_{\epsilon,\delta}$ *it holds that*

$$q_{\epsilon,\delta}(k) \cdot \frac{n(k) + \ell(k)}{k} < k.$$

Figure 1 compares the concrete efficiency threshold of our system to prior published works on the subject [BCGT13, BBGR16a]. We vary the value of the soundness parameter ϵ, plotted on a double logarithmic scale. As seen there, the thresholds of our new system are significantly better (i.e., smaller) than the prior state of the art. We comment that for larger proximity parameters the soundness bounds conjectured earlier give even better (smaller) threshold values[3].

1.3 Applications to transparent zero knowledge implementations

Prover-efficient IOPPs of the kind presented here are crucially needed to facilitate *practical* ZK argument systems that are (i) *transparent* (public randomness), (ii) *universal* — apply to any computation — and (iii) *succinctly* verifiable, meaning that verification time is negligible compared to naïve execution time. In the future we hope to explore the concrete applicability of FRI to succinct ZK argument systems [BBHR17].

[3] E.g., for the maximal value of $\delta = 1 - \rho$, the threshold derived from Conjecture 1.5 ranges between 2^{10} for soundness $\epsilon = 1/2$ to 2^{16} for soundness 2^{-128}.

The seminal works of Babai et al. [BFL90, BFLS91] showed that verifying the correctness of an arbitrary nondeterministic computation running for $T(n)$ steps can be achieved by a verifier running in time $\mathsf{poly}(n, \log T(n))$ in the PCP model. Kilian's construction transforms such PCPs into a 4-round ZK argument in which the total communication complexity and verifier running time are bounded by $\mathsf{poly} \log T(n)$ [Kil92] (cf. [KPT97, IMSX15, IW14]), assuming a family of collision-resistant hash functions. Micali further compressed this system into a noninteractive computationally sound (CS) proof system, assuming both prover and verifier share access to the same random function [Mic00]; this is typically realized in practice using a hash function like SHA2 and relying on the Fiat-Shamir heuristic [FS86]. No implementation of these marvelous techniques has appeared during the quarter century that has passed since they were first published. This is explained, in part, by concerns about the efficiency of these constructions for concrete programs and run-times. Among the numerous components involved in building these systems, a significant computational bottleneck is that of computing solutions to the Reed-Muller (RM) proximity problem, also known as "low degree testing" of multivariate polynomials.

Quasilinear PCPs based on RS codes have prover complexity that is asymptotically more efficient than RM codes, and a number of works have explored the concrete efficiency of these protocols [BCGT13, BBGR16a]. Recently, Ben-Sasson et al. suggested an IOP with perfect zero knowledge (PZK) for NP [BCGV16], later extended to NEXP [BCF$^+$16], in which prover complexity is quasilinear and verifier complexity is $\mathsf{poly}(n, \log T(n))$; this PZK-IOP can be compiled, using Kilian's technique, into an interactive ZK argument with succinct[4] communication complexity, or, using Micali's technique (cf. [Val08]), into a *non-interactive random oracle proof (NIROP)* as defined in [BCS16]. In light of this, the practicality of Kilian- and Micali-type ZK argument systems with polylogarithmic verifiers should be reconsidered.

To add motivation, a number of interesting practical succinct argument systems (with and without zero-knowledge) have been reported recently (see [WB15] for an excellent updated survey of the subject and [BBC$^+$16] for a comparison of PCP/IOP-based solutions to other approaches). A particular system based on the *quadratic span programs* (QSP) of Gennaro et al. [GGPR13] (cf. [BCG$^+$13]) has been used by Ben-Sasson et al. to build a decentralized anonymous payment (DAP) system called "Zerocash" [BCG$^+$14], later deployed as a practical commercial crypto-currency called "ZCash" [Pec16, HBHW17]. However, the QSP based ZK system used in Zerocash/Zcash, called a "preprocessing SNARK" [BCCT12], requires a setup phase that involves *private* randomness; additionally, quantum computers can create pseudo-proofs of falsities in polynomial time for this particular system [Sho94] (cf. [PZ03]). In contrast, the aforementioned succinct interactive and non-interactive (NIROP) systems based on quasilinear PZK-IOPs require only public randomness for their setup, and are not known to be breakable by quantum computers in polynomial time. Therefore, there is great interest in understanding whether succinct (interactive and non-interactive) ZK argument systems which require only public randomness (and resistant to known polynomial time quantum algorithms) can be *practically* built and used, say, by ZCash. Ben-Sasson et al. [BBC$^+$16] describe such an implemented system, called "succinct computational integrity (SCI)" which is not ZK and has comparatively large communication complexity[5]. As mentioned above, we hope to incorporate the RS proximity solution described in Theorem 1.2 within practical ZK systems [BBHR17].

[4]Here, as in past works, "succinct" is synonymous to "polylogarithmic".

[5]Communication complexity in SCI is on the order of tens of megabytes long, compared with QSP based zk-SNARKs that are shorter than 300 Bytes.

1.4 Related works

High-rate LTCs Locally testable codes (LTCs) are error correcting codes for which — by definition — prover complexity and proof length equal 0 (as stated for the case of RS codes by Rubinfeld and Sudan [RS92]); in other words, when focusing solely on prover complexity, LTCs offer an optimal solution (zero complexity). Nevertheless, as discussed in Section 1.3, the specific question of small prover complexity for RS codes is highly relevant because of the its applications to practical ZK-IOPs.

Classical "direct" constructions of LTCs, such as the Hadamard code studied by Blum, Luby and Rubinfeld [BLR93] and the $\log n$-variate RM codes used early PCP constructions [ALM+92, BFLS91] have sub-constant rate, thus lead to long proofs and large PCP prover complexity.

More recently, there has been remarkable progress on constructing locally testable codes (LTCs) with small query complexity and large soundness. Kopparty et al. obtained such codes with rate approaching 1 [KMRS16] and Gopi et al. presented LTCs that reach the Gilbert Varshamov bound [GKdO+17]. These LTCs have super-polylogarithmic query complexity. Additionally, in contrast to RS codes, we are not aware of PCP constructions with similar parameters nor do we know how to convert these LTCs into PCPs.

PCPs and IOPs: A number of recent works have considered PCP constructions with small proof length and query complexity. In addition to the aforementioned works on quasilinear PCPs, Moshkovitz and Raz constructed PCPs with optimally small query complexity (measured in bits) and proofs of length $n^{1+o(1)}$ [MR10], where n denotes the length of the NP statement (like a 3CNF) for which the PCP is constructed, achieving better soundness than Håstad's result [Hås01]. A different line of works attempts to optimize the *bit-length* of PCP proofs; the state of the art, due to Ben-Sasson et al., achieves PCPs of bitlength $O(n)$ and query complexity n^ϵ [BKK+16]. In the IOP model, which generalizes PCPs by allowing more rounds of interaction, Ben-Sasson et al. presented a 2-round IOP with bit-length $O(n)$, constant query complexity (measured in bits) and constant soundness [BCG+16]. (Prover arithmetic complexity in all of these systems is super-linear.)

Soundness amplification: A number of results in the PCP literature have suggested techniques for improving soundness of general PCP constructions, including the parallel repetition theorem of Raz [Raz95], the gap amplification technique of Dinur [Din07] and direct-product testing, introduced by Goldreich and Safra [GS00] (cf. [DG08, IKW12]). These techniques lead to excellent soundness bounds with small query complexity. The concrete prover complexity of PCPs and PCPPs associated with these methods has not been studied in prior works but prover complexity is at least super-linear, and often polynomially large.

Doubly-efficient "proofs for muggles": A recent line of works, initiated by Goldwasser, Kalai and Rothblum [GKR08], revisits the IP model which is equivalent to PSPACE [LFKN92, Sha92], focusing on *doubly efficient* systems in which the prover runs in polynomial time (as opposed to polynomial space, as in the aforementioned results) and verifier runs in nearly linear time. The state of the art along this line is due to Reingold et al. [RRR16], they construct doubly-efficient IP protocols with a constant number of rounds for a family of languages in P. Prover complexity in this line of works is at least super-linear, and typically polynomially large and verifier complexity is super-polylogarithmic, and often super-linear as well (cf. [CMT12, RRR16]).

2 Overview of the FRI IOPP and its soundness

2.1 Analogy with the inverse Fast Fourier Transform

Our protocol is best explained in similarity to the Inverse Fast Fourier Transform (IFFT) [CT65]. It is also similar to (and its soundness analysis relies on) the quasilinear RS-PCPP [BS08] as discussed in Section 2.2 below. Let $\omega^{(0)}$ generate a *smooth* multiplicative group of order 2^n (see Remark 1.4), denoted $L^{(0)}$, that is contained in a field $\mathbb{F}$; in signal processing applications $\omega^{(0)}$ is a complex root of unity of order 2^n and $\mathbb{F}$ is the field of complex numbers (we shall use a different setting). Assume the prover claims that $f^{(0)} : L^{(0)} \to \mathbb{F}$ is a member of $\mathsf{RS}[\mathbb{F}, L^{(0)}, \rho]$, i.e., $f^{(0)}$ is the evaluation of an unknown polynomial $P^{(0)}(X) \in \mathbb{F}[X], \deg(P) < \rho 2^n$; for simplicity we assume $\rho = 2^{-\mathcal{R}}$ and $\mathcal{R}$ is a positive integer. The task of the verifier is to distinguish between truisms ($f^{(0)} \equiv P^{(0)}$ for some low degree $P^{(0)}$) and cases where $f^{(0)}$ is far from all polynomials of degree $< \rho 2^n$. Recalling the IFFT, if $f^{(0)} \equiv P^{(0)}$ there exist polynomials $P_0^{(1)}, P_1^{(1)} \in \mathbb{F}[Y], \deg\left(P_0^{(1)}, P_1^{(1)}\right) < \frac{1}{2}\rho 2^n$ such that

$$\forall x \in L^{(0)} \ f^{(0)}(x) = P^{(0)}(x) = P_0^{(1)}(x^2) + x \cdot P_1^{(1)}(x^2),$$

or, letting $Q^{(1)}(X,Y) \triangleq P_0^{(1)}(Y) + X \cdot P_1^{(1)}(Y)$ and defining $q^{(0)}(X) \triangleq X^2$, we have

$$P^{(0)}(X) \equiv Q^{(1)}(X,Y) \mod Y - q^{(0)}(X) \tag{1}$$

where $\deg_X\left(Q^{(1)}\right) < 2$ and $\deg_Y\left(Q^{(1)}\right) < \frac{1}{2}\rho 2^n$. The map $x \mapsto q^{(0)}(x)$ is 2-to-1 on $L^{(0)}$ and the output of this map is a multiplicative group of order 2^{n-1} that we shall denote by $L^{(1)}$. Moreover, for every $x^{(0)} \in \mathbb{F}$ and $y \in L^{(1)}$, the value of $Q(x^{(0)}, y)$ can be computed by querying two entries of $f^{(0)}$ because $\deg_X(Q) < 2$ (the two entries are the two roots of the polynomial $y - q^{(0)}(X)$).

Our verifier thus samples $x^{(0)} \in \mathbb{F}$ uniformly at random and requests the prover to send as its first oracle a function $f^{(1)} : L^{(1)} \to \mathbb{F}$ that is supposedly the evaluation of $Q^{(1)}\left(x^{(0)}, Y\right)$ on $L^{(1)}$. Assuming $f^{(0)} \in \mathsf{RS}[\mathbb{F}, L^{(0)}, \rho]$, the discussion above shows that $f^{(1)} \in \mathsf{RS}[\mathbb{F}, L^{(1)}, \rho]$. Notice that there exists a 3-query test for the consistency of $f^{(0)}$ and $f^{(1)}$, we call it the *round consistency test*:

1. sample a pair of distinct elements $s_0, s_1 \in L^{(0)}$ such that $s_0^2 = s_1^2 = y$; in other words, sample a uniform $y \in L^{(1)}$ and let s_0, s_1 be the two roots of the polynomial $y - X^2$;
2. query $f^{(0)}(s_0), f^{(0)}(s_1)$ and $f^{(1)}(y)$, denote the query answers by α_0, α_1 and β, respectively;
3. interpolate the "line" through (s_0, α_0) and (s_1, α_1), i.e., find the polynomial $p(X)$ of degree at most 1 that satisfies $p(s_0) = \alpha_0$ and $p(s_1) = \alpha_1$; notice p is unique and well-defined because $s_0 \neq s_1$;
4. accept if and only if $p\left(x^{(0)}\right) = \beta$ and otherwise reject;

Tallying the costs of the first round, the verifier sends a single field element ($x^{(0)}$) and the prover responds with a message (oracle) $f^{(1)} : L^{(1)} \to \mathbb{F}$ evaluated on a domain that is half the size of $L^{(0)}$; testing the consistency of $f^{(0)}$ and $f^{(1)}$ requires three field elements per test (repeating the test boosts soundness). We thus reduced a single proximity problem of size 2^n and rate ρ to a single analogous problem of size 2^{n-1} and same rate. Repeating the process for $\mathsf{r} = n - \mathcal{R}$ rounds leads to a function $f^{(\mathsf{r})}$ that is supposedly of constant degree and evaluated over a domain of constant size

$2^{\mathcal{R}}$, so at this point the prover sends the single constant that describes the function, and verifier uses it as $f^{(\mathsf{r})}$ in the last round consistency test, the one that tests consistency of $f^{(\mathsf{r}-1)}$ and $f^{(\mathsf{r})}$.

Applying inductive analysis to all r rounds, if $f^{(0)} \in \mathsf{RS}[\mathbb{F}, L^{(0)}, \rho]$ (and the prover is honest) then all r round consistency tests pass with probability 1 and $f^{(\mathsf{r})}$ is indeed a constant function. In other words, the protocol we described has perfect completeness. Soundness analysis requires understanding the rejection probability when $f^{(0)}$ is far from $\mathsf{RS}[\mathbb{F}, L^{(0)}, \rho]$ and this is the most challenging aspect of our work (as is the case for all prior PCPP/IOPP works); it is described below in Section 2.2.

Differences between informal and actual protocol The differences between the informal description above and the actual protocol are mostly technical; we list them now. The field $\mathbb{F}$ is finite and *binary*, i.e., of characteristic 2; nevertheless the construction and analysis can be immediately applied to RS codes evaluated over smooth multiplicative groups (of order 2^n), as explained informally above (cf. Remarks 1.4 and 3.1). In binary fields, the natural choice of evaluation domains (like $L^{(0)}, L^{(1)}$ above) are cosets of *additive* groups (not multiplicative ones), i.e, $L^{(i)}$ is an affine shift of a linear space over $\mathbb{F}_2$. The map $q^{(0)}(X) = X^2$ is *not* 2-to-1 on $L^{(0)}$ (over binary fields it is 1-to-1) so we use a different polynomial $q^{(0)}(X)$ that is many-to-one on $L^{(0)}$ and such that the set $L^{(1)} = \left\{y = q^{(0)}(x) \mid x \in L^{(0)}\right\}$ is a coset of an *additive* group, like $L^{(0)}$, but of smaller size ($|L^{(1)}| \ll |L^{(0)}|$); the polynomial $q^{(0)}$ is known as an *affine subspace* polynomial, belonging to the class of *linearized* polynomials (cf. Section 3.1). We use $q^{(0)}$ of degree 4 instead of 2 because this reduces the number of rounds from n to $n/2$ with no increase in total query complexity; notice that a similar reduction could be applied in the multiplicative setting by using $q^{(0)} = X^4$ (but we preferred simplicity to efficiency in the informal exposition above). Finally, the actual protocol performs all queries only after the prover has sent all of $f^{(1)}, \ldots, f^{(\mathsf{r})}$. Thus, we construct a protocol with two phases. The first phase, called the COMMIT phase, involves r *rounds*. At the beginning of the ith round the prover has sent oracles $f^{(0)}, \ldots, f^{(i-1)}$, and during this ($i$th) round the verifier samples and sends $x^{(i)}$ and the prover responds by sending the next oracle $f^{(i)}$. During the second phase, called the QUERY phase, the verifier applies the *round consistency test* to all r rounds. To save query complexity *and boost soundness*, the query $s^{(i)} \in L^{(i)}$ is used to test *both* consistency of $f^{(i-1)}$ vs. $f^{(i)}$ *and* consistency of $f^{(i)}$ vs. $f^{(i+1)}$.

2.2 Soundness analysis — overview

Proof composition is a technique introduced by Arora and Safra [AS98] in the context of PCPs, adapted to PCPPs in [BGH$^+$06, DR04] and optimized for the special case of the RS code in [BS08]. Informally, it reduces proximity testing problems over a large domain to similar proximity testing problems over significantly smaller domains. The process reducing $f^{(0)}$ to $f^{(1)}$ above is a special case of proof composition, and each invocation of it incurs two costs on behalf of the verifier. The first is the *query complexity* needed to check consistency of $f^{(0)}$ and $f^{(1)}$ (the "round consistency test") and the second is the reduction in *distance*, which affects the *soundness* of the protocol. Assuming $f^{(0)}$ is $\delta^{(0)}$-far from all codewords in relative Hamming distance, for proof composition to work one should prove that with high probability $f^{(1)}$ is $\delta^{(1)}$-far from all codewords where $\delta^{(1)}$ depends on $\delta^{(0)}$; larger values of $\delta^{(1)}$ imply higher (better) soundness and smaller communication complexity. A benefit of the FRI protocol is that with high probability $\delta^{(1)} \geq (1 - o(1))\delta^{(0)}$, i.e., the reduction in distance in our protocol is *negligible*. In contrast, prior RS proximity PCPP and IOPP solutions follow the construction and analysis of [BS08] which in turn is based on the bivariate testing Theorem of Polischuk and Spielman [PS94] and incur a *constant multiplicative loss* in distance per round of proof composition ($\delta^{(1)} \leq \delta^{(0)}/2$). This loss limited the number of proof composition rounds to $\leq \log n$ and thus required replacing $q^{(0)}(X) = X^2$ with a higher degree polynomial, like $q^{(0)}(X) = X^{2^{n/2}}$. The higher degree of $q^{(0)}$ results in $Q^{(1)}(X, Y)$ having *balanced* X- and Y-degrees, namely

$$\deg_X \left(Q^{(1)}\right) \approx \deg_Y \left(Q^{(1)}\right) \approx 2^{n/2}.$$

Moving to $q^{(0)}(X)$ of *constant* degree as in FRI gives a *biased RS-IOPP* (because $\deg_X(Q^{(1)}) \ll \deg_Y(Q^{(1)})$). The main benefit of this bias is that one side of the recursive process (that of X) terminates immediately and consequently *removes* the constant multiplicative soundness loss incurred in prior works, replacing it with a negligible additive loss. More to the point, we show that for $\delta^{(0)}$ less than the unique decoding radius of the code ($\delta^{(0)} < (1 - \rho)/2$), with high probability (namely, $1 - \frac{O(1)}{|\mathbb{F}|}$) the sum of (i) the round consistency error and (ii) the "new" distance $\delta^{(1)}$ is at least as large as the "old" distance $\delta^{(0)}$. This statement is relatively straightforward to prove in case the prover is *honest*, i.e., when $f^{(1)}(y) = Q^{(1)}\left(x^{(0)}, y\right)$ for all $y \in L^{(1)}$ (in this case there is no round consistency error). The challenging part of the proof is to show this also holds for non-honest provers and arbitrary $f^{(1)}$; see Lemma 4.4 and Section 4.2 for more details.

2.3 Discussion and open problems

The IOP model has been used to obtain several results that go beyond what is known for the PCP model (which is equivalent to a 1-round IOP). Examples include perfect ZK with quasilinear provers [BCGV16, BCF+16], proofs with linear bit-length and constant soundness and query complexity [BCG+16] and constant-round interactive proofs for delegating computation [RRR16]. The IOP model is more efficient not only asymptotically but also for concrete input lengths, as shown in this work as well as in [BBGR16b], and has even been implemented to prove correctness of computations [BBC+16]. Examining Table 1 and Figure 1 raises the following question.

- Can one decrease the round complexity to $O(\log \log n)$ (or less) while maintaining linear prover arithmetic complexity and polylogarithmic verifier complexity with constant soundness?
- Can query complexity be further reduced for the same soundness with linear prover complexity?
- Can the soundness amplification techniques (cf. Section 1.4) be used to further reduce the concrete efficiency threshold for RS codes, related codes and/or PCP constructions?

A positive answer to these questions for concrete input lengths could drastically increase the range of practical potential applications for succinct PZK IOP systems, advancing the goal of "polylogarithmic verification" with public randomness closer to reality.

Acknowledgements

We thank Peter Manohar and Nicholas Spooner for helping clarify the presentation and for pointing out and correcting errors in an earlier manuscript.

3 FRI— detailed description and main properties

In this section we give a formal and detailed description of the FRI protocol, expanding on what was explained in the previous section. We start by providing additional needed definitions, followed by the description of the COMMIT and QUERY phases; we continue by listing the properties obtained by the protocol (Theorem 3.3), and conclude the section with a proof of Main Theorem 1.2. The next section is then devoted to the proof of Theorem 3.3.

3.1 Definitions and notation

Interpolant For a function $f : S \to \mathbb{F}, S \subset \mathbb{F}$, let $\mathsf{interpolant}^f$ denote the *interpolant* of f, defined as the unique polynomial $P(X) = \sum_{i=0}^{|S|-1} a_i X^i$ of degree less than $|S|$ whose evaluation on S equals $f|_S$, i.e., $\forall x \in S\ f(x) = P(x)$. We assume the interpolant $P(X)$ is represented as a formal sum, i.e., by the sequence of monomial coefficients $a_0, \ldots, a_{|S|-1}$.

Subspace polynomials Given a set $L_0 \subset \mathbb{F}$ let $\mathsf{Zero}_{L_0} \stackrel{\triangle}{=} \prod_{x \in L_0}(X - x)$ be the unique non-zero monic polynomial of degree $|L_0|$ that vanishes on L_0. When L_0 is an additive coset contained in a binary field, the polynomial $\mathsf{Zero}_{L_0}(X)$ is an *affine subspace polynomial*, a special type of a *linearized polynomial* [Ore33, Ore34]. We shall use the following properties of such polynomials, referring the interested reader to [LN97, Chapter 3.4] for proofs and additional background:

1. The map $x \mapsto \mathsf{Zero}_{L_0}(x)$ maps each additive coset S of L_0 to a single field element, which will be denoted by y_S
2. If $L \supset L_0$ is an additive coset, then $\mathsf{Zero}_{L_0}(L) \stackrel{\triangle}{=} \{\mathsf{Zero}_{L_0}(z) \mid z \in L_0\}$ is an additive coset and $\dim(\mathsf{Zero}_{L_0}(L)) = \dim(L) - \dim(L_0)$.

Subspace specification Henceforth, the letter L always denotes an additive coset in a binary field $\mathbb{F}$, we assume all mentioned additive cosets are specified by an additive shift $\alpha \in \mathbb{F}$ and a basis $\beta_1, \ldots, \beta_k \in \mathbb{F}^k$ so that $L = \left\{\alpha + \sum_{i=1}^k b_i \beta_i \mid b_1, \ldots, b_k \in \mathbb{F}_2\right\}$; we assume α and $\vec{\beta} = (\beta_1, \ldots, \beta_k)$ are agreed upon by prover and verifier.

3.2 The COMMIT phase

The protocol is parameterized by an integer $\eta \ll k^{(0)}$; to prove Theorem 1.2 we set $\eta = 2$ but in other settings a different value may be more beneficial. The number of rounds is $\mathsf{r} \stackrel{\triangle}{=} \lfloor \frac{k^{(0)} - \mathcal{R}}{\eta} \rfloor$ (recall that $\mathcal{R} = \log(1/\rho)$, where ρ is the rate). During the ith round of the COMMIT phase, $i \in \{0, \ldots, \mathsf{r} - 1\}$, the verifier has oracle access to a function $f^{(i)} : L^{(i)} \to \mathbb{F}$, where $\dim(L^{(i)}) = k^{(i)} = k^{(0)} - \eta \cdot i$ submitted by the prover and the spaces $L^{(i)}$ are fixed in advance and, in particular, do not depend on verifier messages.

A single COMMIT round We assume verifier and prover have also agreed upon a fixed "small" $L_0^{(i)} \subset L^{(i)}, \dim\left(L_0^{(i)}\right) = \eta$. Let $\mathcal{S}^{(i)}$ denote all cosets of $L_0^{(i)}$ in $L^{(i)}$. Let

$$q^{(i)}(X) \stackrel{\triangle}{=} \mathsf{Zero}_{L_0^{(i)}}(X)$$

be the subspace polynomial vanishing on $L_0^{(i)}$. Let $L^{(i+1)} \triangleq q^{(i)}\left(L^{(i)}\right)$. The verifier's ith message is a uniformly random $x^{(i)} \in \mathbb{F}$. The Prover's next message (or oracle) is $f^{(i+1)} : L^{(i+1)} \to \mathbb{F}$ computed for each $y_S \in L^{(i+1)}, y_S = q^{(i)}(S), S \in \mathcal{S}^{(i)}$, by interpolating the function $f^{(i)}|_S$ to obtain a polynomial $P_S^{(i)}(X), \deg\left(P_S^{(i)}\right) < 2^\eta$ and then setting $f^{(i+1)}(y_S) \triangleq P_S^{(i)}\left(x^{(i)}\right)$.

Termination — COMMIT During the last round ($i = \mathsf{r}$), the prover sends the interpolant $P^{(\mathsf{r})}(X) = \mathsf{interpolant}^{f^{(\mathsf{r})}}$ of $f^{(\mathsf{r})}$ rather than $f^{(\mathsf{r})}$ itself. By this point $\deg\left(P^{(\mathsf{r})}(X)\right) < \rho \cdot |L^{(\mathsf{r})}| \leq 2^\eta$ (recall $\eta \in \mathbb{N}$ is a constant).

FRI-COMMIT:

Common input:

- Parameters $\mathcal{R}, \eta, i$, all are positive integers:
  - rate parameter $\mathcal{R}$: logarithm of RS code rate ($\rho = 2^{-\mathcal{R}}$)
  - localization parameter η: dimension of $L_0^{(i)}$ (i.e., $|L_0^{(i)}| = 2^{\eta}$); let $\mathsf{r} \triangleq \lfloor \frac{k^{(0)} - \mathcal{R}}{\eta} \rfloor$ denote round complexity
  - $i \in \{0, \ldots, \mathsf{r}\}$: round counter
- A parametrization of $\mathsf{RS}^{(i)} \triangleq \mathsf{RS}\left[\mathbb{F}, L^{(i)}, \rho = 2^{-\mathcal{R}}\right]$, denote $k^{(i)} = \log_2 \left|L^{(i)}\right|$ (notice $k^{(i)} = \dim\left(L^{(i)}\right)$);
- $L_0^{(i)} \subset L^{(i)}$, $\dim\left(L_0^{(i)}\right) = \eta$; Let $q^{(i)}(X) = \mathsf{Zero}_{L_0^{(i)}}(X)$ and denote $L^{(i+1)} = q^{(i)}\left(L^{(i)}\right)$

Prover input: $f^{(i)} : L^{(i)} \to \mathbb{F}$, a purported codeword of $\mathsf{RS}^{(i)}$

Loop: While $i \leq \mathsf{r}$:

1. Verifier sends a uniformly random $x^{(i)} \in \mathbb{F}$
2. Prover defines the function $f^{(i+1)}_{f^{(i)}, x^{(i)}}$ with domain $L^{(i+1)}$ thus, for each $y \in L^{(i+1)}$:
   - Let $S_y = \left\{x \in L^{(i)} \mid q^{(i)}(x) = y\right\}$ be the coset of $L_0^{(i)}$ mapped by $q^{(i)}$ to $\{y\}$;
   - $P_y^{(i)}(X) \triangleq \mathsf{interpolant}^{f^{(i)}|_{S_y}}$;
   - $f^{(i+1)}_{f^{(i)}, x^{(i)}}(y) \triangleq P_y^{(i)}\left(x^{(i)}\right)$;
3. If $i = \mathsf{r}$ then:
   - let $f^{(\mathsf{r})} = f^{(\mathsf{r})}_{f^{(\mathsf{r}-1)}, x^{(\mathsf{r}-1)}}$ for $f^{(\mathsf{r})}_{f^{(\mathsf{r}-1)}, x^{(\mathsf{r}-1)}}$ defined in step 2 above;
   - let $P^{(\mathsf{r})}(X) = \sum_{j \geq 0} a_j^{(\mathsf{r})} X^j \triangleq \mathsf{interpolant}^{f^{(\mathsf{r})}}(X)$;
   - let $d = \rho \cdot |L^{(\mathsf{r})}| - 1$;
   - prover commits to first $d+1$ coefficients of $P^{(\mathsf{r})}(X)$, namely, to $\langle a_0^{(\mathsf{r})}, \ldots, a_d^{(\mathsf{r})} \rangle$
   - COMMIT phase terminates;
4. Else ($i < \mathsf{r}$):
   - let $f^{(i+1)} = f^{(i+1)}_{f^{(i)}, x^{(i)}}$ for $f^{(i+1)}_{f^{(i)}, x^{(i)}}$ defined in step 2 above;
   - prover commits to oracle $f^{(i+1)}$
   - both parties repeat the COMMIT protocol with common input
     - parameters $(\mathcal{R}, \eta, i+1)$
     - a paramterization of $\mathsf{RS}^{(i+1)} \triangleq \mathsf{RS}\left[\mathbb{F}, L^{(i+1)}, \rho = 2^{-\mathcal{R}}\right]$ and $L_0^{(i+1)} \subset L^{(i+1)}, \dim(L_0^{(i+1)}) = \eta$

     and prover input $f^{(i+1)}$ defined at the beginning of this step;

Remark 3.1 (Adapting FRI to the family of smooth RS codes)**.** *If* $\mathsf{RS}^{(0)}$ *is a smooth code of blocklength* $2^{k^{(0)}}$ *(i.e.,* $L^{(0)}$ *is a multiplicative group of order* $2^{k^{(0)}}$*), the* FRI *protocol for* $\mathsf{RS}^{(0)}$ *is obtained from the protocol above by applying the following modifications to the* COMMIT *and* QUERY *phases:*

- *define* $L_0^{(i)}$ *to be the multiplicative subgroup of* $L^{(i)}$ *of size* 2^η*, i.e., the set of roots of the polynomial* $X^{2^\eta} - 1$
- *define* $q^{(i)}(X) = X^{2^\eta}$ *(notice* $q^{(i)}$ *does not depend on* i*); observe* $x \mapsto q^{(i)}(x)$ *is a* 2^η*-to-*1 *map on* $L^{(i)}$ *and its image is a (smooth) multiplicative group;*
- *let* $L^{(i+1)} = q^{(i)}\left(L^{(i)}\right)$ *(exactly as described in the protocol above), noticing* $L^{(i+1)}$ *is the (smooth) multiplicative group of order* $2^{k^{(i+1)}} = 2^{k^{(i)}-\eta}$*;*
- terminology: *interpret the words "coset" to mean "multiplicative coset" and the term "dimension" to mean "base-*2 *logarithm of group order" (e.g.,* $\dim\left(L^{(i)}\right) \triangleq \log_2\left|L^{(i)}\right|$*); replace the term "affine space" with "smooth group";*

3.3 The QUERY phase

During this phase the prover does not participate and the verifier merely checks that the prover operated as specified above. Concretely, a single test of the verifier consists of sampling a uniformly random $s^{(0)} \in L^{(0)}$, and computing iteratively $s^{(i+1)} = q^{(i)}\left(s^{(i)}\right)$, notice $s^{(i)} \in L^{(i)}$; let $S^{(i)}$ denote the unique coset of $L_0^{(i)}$ in which $s^{(i)}$ is contained. (Using the notation above, we have $s^{(i+1)} = y_{S^{(i)}}$.) The verifier now accepts if and only if for all $i < \mathsf{r}$ it holds that $f^{(i+1)}\left(s^{(i+1)}\right) = \mathsf{interpolant}^{f^{(i)}|_{S^{(i)}}}\left(x^{(i)}\right)$. When $i < \mathsf{r}$ the values $f^{(i)}(z)$ are queried directly by the verifier, and in the terminal case ($i = \mathsf{r}$) the verifier queries all coefficients of $P^{(\mathsf{r})}(X)$ from the last prover message and interpolates this polynomial to reconstruct $f^{(\mathsf{r})}$.

FRI-QUERY:
verifier input:

- parameters $\mathcal{R}, \eta$ as defined in the COMMIT phase
- repetition parameter ℓ
- sequence of rate-ρ RS-codes $\mathsf{RS}^{(0)}, \ldots, \mathsf{RS}^{(\mathsf{r})}$, where $\mathsf{RS}^{(i)} = \mathsf{RS}\left[\mathbb{F}, L^{(i)}, \rho\right]$ and $\log_2 \left|L^{(i)}\right| = k^{(i)} = k^{(0)} - i \cdot \eta$; (notice $k^{(i)} = \dim\left(L^{(i)}\right)$);
- sequence of affine spaces $L_0^{(0)}, \ldots, L_0^{(\mathsf{r}-1)}$, each $L_0^{(i)}$ is of dimension η and contained in $L^{(i)}$;
- transcript of verifier messages $x^{(0)}, \ldots, x^{(\mathsf{r}-1)} \in \mathbb{F}$
- access to oracles $f^{(0)}, \ldots, f^{(\mathsf{r}-1)}$
- access to last oracle $P^{(\mathsf{r})}(X) = \sum_{j=0}^{d} a_j^{(\mathsf{r})} X^j$ for $d = \rho \cdot |L^{(\mathsf{r})}| - 1$;

Terminal function reconstruction:

- query $a_0^{(\mathsf{r})}, \ldots, a_d^{(\mathsf{r})}$; (a total of $d + 1 \leq 2^\eta$ queries)
- let $P'(X) \triangleq \sum_{j \leq d} a_j^{(\mathsf{r})} X^j$;
- let $f^{(\mathsf{r})}$ be the evaluation of $P'(X)$ on $L^{(\mathsf{r})}$; (notice $f^{(\mathsf{r})} \in \mathsf{RS}^{(\mathsf{r})}$)

Repeat ℓ times: **{**

1. Sample uniformly random $s^{(0)} \in L^{(0)}$ and for $i = 0, \ldots, \mathsf{r} - 1$ let
   - $s^{(i+1)} = q^{(i)}\left(s^{(i)}\right)$
   - $S^{(i)}$ be the coset of $L_0^{(i)}$ in $L^{(i)}$ that contains $s^{(i)}$
2. For $i = 0, \ldots, \mathsf{r} - 1$,
   - query $f^{(i)}$ on all of $S^{(i)}$; (a total of 2^η queries)
   - compute $P^{(i)}(X) \triangleq \mathsf{interpolant}^{f^{(i)}|_{S^{(i)}}}$; (notice $\deg\left(P^{(i)}\right) < 2^\eta$)
3. **round consistency:** If for some $i \in \{0, \ldots, \mathsf{r} - 1\}$ it holds that

$$f^{(i+1)}\left(s^{(i+1)}\right) \neq P^{(i)}\left(x^{(i)}\right) \tag{2}$$

   then reject and abort;

}

Return accept

3.4 Main properties of the FRI protocol

The following distance measure will be used in our soundness analysis. It is similar to the relative Hamming distance, only measured on blocks of symbols. Given a function $f : S \to \Sigma$ and $S' \subset S$ we denote by $f|_{S'}$ the *restriction* of f to domain S'. Given $g : S \to \Sigma$, let $f|_{S'} = g|_{S'}$ denote equality in the space $\Sigma^{S'}$, i.e., this equality holds iff for each $x \in S'$ we have $f(x) = g(x)$.

Definition 3.2 (Blockwise distance measure)**.** *Let $\mathcal{S} = \{S_1, \ldots, S_m\}$ be a partition of a set S and Σ be an alphabet. The* relative $\mathcal{S}$-Hamming distance measure *on Σ^S is defined for $f, g \in \Sigma^S$ as the relative Hamming distance over $\Sigma^{S_1} \times \ldots \times \Sigma^{S_m}$,*

$$\Delta^{\mathcal{S}}(f,g) \triangleq \Pr_{i \in [m]}[f|_{S_i} \neq g|_{S_i}] = \frac{|\{i \in [m] \mid f|_{S_i} \neq g|_{S_i}\}|}{m}. \tag{3}$$

Thus, for $\mathcal{F} \subset \Sigma^S$ let $\Delta^{\mathcal{S}}(g, \mathcal{F}) = \min\left\{\Delta^{\mathcal{S}}(g,f) \mid f \in \mathcal{F}\right\}$.

In our soundness analysis of the FRI protocol we use the blockwise distance on $\mathbb{F}^{L^{(i)}}$ corresponding to the partition of $L^{(i)}$ to cosets of $L_0^{(i)}$; recall both $L^{(i)}, L_0^{(i)}$ are affine spaces with $L_0^{(i)} \subset L^{(i)}$, and $\mathcal{S}^{(i)}$ denotes the set of cosets of $L_0^{(i)}$ in $L^{(i)}$. To simplify notation we denote

$$\Delta^{(i)}(f,g) \triangleq \Delta^{\mathcal{S}^{(i)}}(f,g) \tag{4}$$

In words, $\Delta^{(i)}(\cdot,\cdot)$ measures the fraction of cosets of $L_0^{(i)}$ in $L^{(i)}$ on which f and g do not agree completely. Recalling $\mathsf{RS}^{(i)}$ is a code of rate ρ we have

$$1 - \rho \geq \Delta^{(i)}\left(f^{(i)}, \mathsf{RS}^{(i)}\right) \geq \Delta_{\mathsf{H}}\left(f^{(i)}, \mathsf{RS}^{(i)}\right) \tag{5}$$

where $\Delta_{\mathsf{H}}(\cdot,\cdot)$ denotes relative Hamming distance. The first inequality holds because there always exists a polynomial of degree $< \rho|L^{(i)}|$ that agrees completely with $f^{(i)}$ on a ρ-fraction of cosets of $L_0^{(i)}$. The second inequality holds because if $f^{(i)}$ differs from $g \in \mathsf{RS}^{(i)}$ on a δ-fraction of cosets in $\mathcal{S}^{(i)}$ then f and g differ on at most a δ-fraction of their entries because all cosets in $\mathcal{S}^{(i)}$ are of equal size.

The following theorem is a more detailed and precise version of Theorem 1.2.

Theorem 3.3 (Main properties of the FRI protocol)**.** *The following properties hold when the* FRI *protocol is invoked on oracle $f^{(0)} : L^{(0)} \to \mathbb{F}$ with rate parameter $\mathcal{R}$ and localization parameter η:*

1. **Completeness** *If $f^{(0)} \in \mathsf{RS}^{(0)} \triangleq \mathsf{RS}[\mathbb{F}, L^{(0)}, \rho = 2^{-\mathcal{R}}]$ and $f^{(1)}, \ldots, f^{(\mathsf{r})}$ are computed by the prover specified in the* COMMIT *phase, then the* FRI *verifier outputs* accept *with probability* 1.

2. **Soundness lower bound** *Suppose $\delta^{(0)} \triangleq \Delta^{(0)}\left(f^{(0)}, \mathsf{RS}^{(0)}\right) > 0$. Then with probability at least*

$$1 - \frac{3|L^{(0)}|}{|\mathbb{F}|} \tag{6}$$

over the randomness of the verifier during the COMMIT *phase, and for any (adaptively chosen) prover oracles $f^{(1)}, \ldots, f^{(\mathsf{r})}$, the* QUERY *protocol with repetition parameter ℓ outputs* accept *with probability at most*

$$\left(1 - \min\left\{\delta^{(0)}, \frac{1 - 3\rho - 2^{\eta}/\sqrt{|L^{(0)}|}}{4}\right\}\right)^{\ell} \tag{7}$$

Consequently, the soundness of FRI *is at least*

$$\mathbf{s}^{-}\left(\delta^{(0)}\right) \triangleq 1-\left(\frac{3|L^{(0)}|}{|\mathbb{F}|}+\left(1-\min\left\{\delta^{(0)}, \frac{1-3\rho-2^{\eta}/\sqrt{|L^{(0)}|}}{4}\right\}\right)^{\ell}\right). \tag{8}$$

3. **Soundness upper bound** *There exists a polynomial time algorithm P^* that, given $f^{(0)}$ and $w \in \mathsf{RS}^{(0)}$ with $\Delta^{(0)}\left(f^{(0)}, w^{(0)}\right) = \delta^{(0)}$, produces interactively a sequence $f^{(1)}, \ldots, f^{(\mathsf{r})}$ such that*

   - *with probability 1 over the verifier randomness during the* COMMIT *phase, the verifier rejects $f^{(1)}, \ldots, f^{(\mathsf{r})}$ during the* QUERY *phase with probability at most $\delta^{(0)}$,*
   - *moreover, with probability at least $(2^{\eta}-1)/|\mathbb{F}|$ over the randomness of the verifier during the* COMMIT *phase, the verifier accepts $f^{(1)}, \ldots, f^{(\mathsf{r})}$ with probability 1 during the* QUERY *phase*

   Consequently, the soundness of FRI *is at most*

$$\mathbf{s}^{+}\left(\delta^{(0)}\right) \triangleq 1-\left(\frac{2^{\eta}-1}{|\mathbb{F}|}+\left(1-\delta^{(0)}\right)^{\ell}\right) \tag{9}$$

4. **Prover complexity** *The ith step of commit phase can be computed by a parallel random access machine (PRAM) with concurrent read and exclusive write (CREW) in $2\eta+3$ cycles — each cycle involves a single arithmetic operation in $\mathbb{F}$ — using $2|L^{(i)}|+\eta$ processors and a total of $4|L^{(i)}|$ arithmetic operations over $\mathbb{F}$.*

   Consequently, the total prover complexity is at most $6|L^{(0)}|$ arithmetic operations, which can be carried out in at most $4\log|L^{(0)}|$ cycles on a PRAM-CREW with $2n+3$ processors.

5. **Verifier complexity** *Verifier communication during the* COMMIT *phase equals* r *field elements; query complexity (during* QUERY *phase) equals $\ell 2^{\eta}\mathsf{r} = \ell 2^{\eta}(\log|L^{(0)}|-\mathcal{R})$. On a PRAM with exclusive read and exclusive write (EREW) with $\ell\mathsf{r}\cdot 2^{\eta}$ processors, the verifier's decision is obtained after $2\eta+3+\log\ell$ cycles and a total of $\ell\cdot\mathsf{r}\cdot(4\cdot 2^{\eta}+6\eta+1)$ arithmetic operations in $\mathbb{F}$.*

Remark 3.4 (Tightness of soundness upper and lower bounds)**.** *For $\delta^{(0)} \leq \frac{1-3\rho-2^{\eta}/\sqrt{|L^{(0)}|}}{4}$ the soundness upper and lower bounds nearly match,*

$$\mathbf{s}^{+}\left(\delta^{(0)}\right)-\mathbf{s}^{-}\left(\delta^{(0)}\right)<\frac{3|L^{(0)}|}{|\mathbb{F}|},$$

and both give a rejection probability of approximately $\delta^{(0)}$. Closing the gap between these two bounds for larger $\delta^{(0)}$ remains an intriguing open problem.

3.5 Proof of Main Theorem 1.2

We now show that the Theorem above indeed proves our Main Theorem 1.2 and the remainder of the paper is devoted to proving Theorem 3.3.

Proof of Main Theorem 1.2. Apply Theorem 3.3 with $n = |L^{(0)}|$ and $k = k^{(0)} = \dim\left(L^{(0)}\right)$. Fix $\eta = 2$ and $\ell = 1$. Prover complexity follows immediately from Theorem 3.3, part 4. By construction $\dim(L^{(i)}) = \dim(L^{(0)}) - i\eta$ and thus, using the geometric series formula, the total proof length is

$$|L^{(1)}| + \ldots + |L^{(\mathsf{r})}| = |L^{(0)}| \cdot \sum_{i=1}^{\mathsf{r}} \frac{1}{2^{i\eta}} = |L^{(0)}| \cdot \sum_{i=1}^{\mathsf{r}} 4^{-i} < |L^{(0)}|/3.$$

Round complexity is

$$\mathsf{r} = \lfloor (k - \mathcal{R})/\eta \rfloor \leq k/2 - 1$$

the last inequality follows because $\mathcal{R} \geq \eta$. This completes the proof of the first bullet ("prover") of Theorem 1.2.

Moving to the second bullet ("verifier"), query complexity is at most $2 \log n$ for our selection of $\eta = 2$ and the resulting value of r. The decision complexity of the verifier follows immediately from Theorem 3.3, part 5 using the setting of r, η and ℓ. This completes the proof of the second bullet.

The lower bound on soundness (third bullet) follows from (8) by setting $\ell = 1$; although (8) is stated for the blockwise distance measure, the same bound holds also with respect to the relative Hamming distance measure; this follows from (5).

Regarding the soundness upper bound (fourth bullet), it follows from (9) for any function $f^{(0)}$ that is chosen to be δ-far from $w \in \mathsf{RS}^{(0)}$ in both the Hamming distance measure and the blockwise distance measure.

The parallelization bullet follows from Theorem 3.3, part 4. This completes the proof of Main Theorem 1.2. □

4 Proof of Theorem 3.3

We prove the items of Theorem 3.3 in the order stated there. The main technical challenge is that of proving soundness lower bounds in Section 4.2.

4.1 Completeness — Part 1

The proof of the completeness claim follows from the following lemma.

Lemma 4.1 (Inductive argument)**.** *If $f^{(i)} \in \mathsf{RS}^{(i)}$ then for all $x^{(i)} \in \mathbb{F}$ it holds that $f^{(i+1)}_{f^{(i)},x^{(i)}} \in \mathsf{RS}^{(i+1)}$.*

We complete the proof of completeness assuming the lemma above, then prove the lemma.

Proof of Theorem 3.3, item 1 (perfect completeness). If one applies the prover specified in the COMMIT phase to an arbitrary function, then for any $i < \mathsf{r} - 1$ all round consistency tests pass because the equality (2) checked by the verifier is fulfilled by the construction of $f^{(i+1)}$ from $f^{(i)}$ described in step 2 of the COMMIT phase.

Thus we need only prove that the round consistency test passes also for $i = \mathsf{r} - 1$. By assumption $\delta^{(0)} = 0$ so $f^{(0)} \in \mathsf{RS}^{(0)}$. Applying Lemma 4.1 inductively shows that $f^{(\mathsf{r})} \in \mathsf{RS}^{(\mathsf{r})}$ which means that its interpolant is a polynomial of degree $< \rho \cdot |L^{(\mathsf{r})}|$. In this case the function ${f'}^{(\mathsf{r})}$ extracted from $P^{(\mathsf{r})}(X)$ in the "terminal function reconstruction" step is indeed equal to $f^{(\mathsf{r})}$ and hence all round consistency tests pass for $i = \mathsf{r} - 1$ as well. This completes the proof. □

4.1.1 Proof of Lemma 4.1

For our proof, we need the following claim from [BS08, Section 6] and repeat its proof for self-containment. We use capitalized letters like X, Y to denote formal variables and non-capitalized ones like x, y to denote field elements.

Claim 4.2. *For every $f^{(i)} : L^{(i)} \to \mathbb{F}$ there exists $Q^{(i)}(X,Y) \in \mathbb{F}[X,Y]$ satisfying*

1. $f^{(i)}(x) = Q^{(i)}\left(x, q^{(i)}(x)\right)$ *for all* $x \in L^{(i)}$
2. $\deg_X\left(Q^{(i)}\right) < |L_0^{(i)}|$
3. *If* $f^{(i)} \in \mathsf{RS}[\mathbb{F}, L^{(i)}, \rho]$ *then* $\deg_Y\left(Q^{(i)}\right) < \rho |L^{(i+1)}|$

Proof. Let $P^{(i)} = \mathsf{interpolant}^{f^{(i)}}$. Let $\mathbb{F}[X,Y]$ denote the ring of bivariate polynomials over $\mathbb{F}$; order monomials first according to total degree, then according to X-degree. Let

$$Q^{(i)}(X,Y) = P^{(i)}(X) \mod Y - q^{(i)}(X) \tag{10}$$

be the remainder from dividing $P^{(i)}(X)$ by $Y - q^{(i)}(X)$. By definition, there exists $R(X,Y) \in \mathbb{F}[X,Y]$ such that

$$P^{(i)}(X) = Q^{(i)}(X,Y) + \left(Y - q^{(i)}(X)\right) \cdot R(X,Y).$$

For $x \in L^{(i)}$ and $y = q^{(i)}(x)$ the rightmost summand above vanishes, hence $P^{(i)}(x)$ equals $Q^{(i)}(x,y) = Q^{(i)}\left(x, q^{(i)}(x)\right)$, implying item 1.

By the ordering chosen for monomials, the remainder Q defined in (10) satisfies

$$\deg_X\left(Q^{(i)}(X,Y)\right) < \deg\left(q^{(i)}\right) = |L_0^{(i)}|$$

and hence item 2 holds.

Finally, by the rules of division and the chosen monomial ordering,

$$\deg_Y\left(Q^{(i)}\right) = \lfloor \frac{\deg\left(P^{(i)}\right)}{\deg\left(q^{(i)}\right)} \rfloor = \lfloor \frac{\deg\left(P^{(i)}\right)}{|L_0^{(i)}|} \rfloor < \rho |L^{(i+1)}|$$

The inequality follows because $|L^{(i+1)}| = |L^{(i)}|/|L_0^{(i)}|$ and $f^{(i)} \in \mathsf{RS}^{(i)}$, implying $\deg\left(P^{(i)}\right) < \rho |L^{(i)}|$. We conclude item 3 holds and this proves the claim. □

Proof of Lemma 4.1. We use the notation from Claim 4.2. From item 3 of that claim it follows that for any $x^{(i)}$ we have $\deg_Y\left(Q^{(i)}\right) < \rho \cdot |L^{(i+1)}|$. We will thus prove

$$\forall y \in L^{(i+1)}, f^{(i+1)}(y) = Q^{(i)}\left(x^{(i)}, y\right) \tag{11}$$

and this implies $\deg\left(f^{(i+1)}\right) \leq \deg_Y\left(Q^{(i)}\right) < \rho \cdot |L^{(i+1)}|$, as required.

To prove (11) fix $y \in L^{(i+1)}$ and let $S_y \in \mathcal{S}^{(i)}$ satisfy $q^{(i)}\left(S_y\right) = \{y\}$. By construction of $f^{(i+1)}$ we have

$$f^{(i+1)}(y) = \mathsf{interpolant}^{f^{(i)}|_{S_y}}\left(x^{(i)}\right). \tag{12}$$

By Claim 4.2, item 1,

$$\forall x \in S_y, \quad f^{(i)}(x) = P^{(i)}(x) = Q^{(i)}(x, y) \tag{13}$$

And because $\deg_X\left(Q^{(i)}\right) < |S_y|$, due to Claim 4.2, item 2, we conclude that

$$\mathsf{interpolant}^{f^{(i)}|_{S_y}}(X) = Q^{(i)}(X, y) \tag{14}$$

as formal polynomials in X, hence evaluating both polynomials on $x^{(i)}$ gives equal values. Combining this with (12) and (13) proves that (11) holds, and this completes the proof. □

4.2 Soundness — Part 2

Soundness analysis is typically the most challenging aspect of proximity testing protocols; our case is no different. First we provide a few needed definitions, and continue in Section 4.2.2 with a statement of two main lemmas (Lemmas 4.3 and 4.4) that imply soundness. After completing the proof of soundness in that section, we prove the two main lemmas in Sections 4.2.3 and 4.2.4.

4.2.1 Definitions — round consistency error and distortion set

Given oracles $f^{(i)}$ and $f^{(i+1)}$ produced in response to verifier randomness $x^{(i)}$, we shall use the following terms and notation:

- **inner-layer distance** the ith *inner-layer distance* is the $\Delta^{(i)}$-distance of $f^{(i)}$ from $\mathsf{RS}^{(i)}$,

$$\delta^{(i)} = \Delta^{(i)}\left(f^{(i)}, \mathsf{RS}^{(i)}\right)$$

- **round error** For $i > 0$, the ith *round error set* is the subset of $L^{(i)}$ defined by

$$A_{\mathsf{err}}^{(i)}\left(f^{(i)}, f^{(i-1)}, x^{(i-1)}\right) \triangleq \bigcup \left\{ y_S^{(i)} \in L^{(i)} \mid \mathsf{interpolant}^{f^{(i-1)}|_S}\left(x^{(i-1)}\right) \neq f^{(i)}\left(y_S^{(i)}\right) \right\}$$

  and the ith *round error* $\mathsf{err}^{(i)}$ is the probability that the round consistency test rejects $f^{(i)}$ and $f^{(i-1)}$,

$$\mathsf{err}^{(i)}\left(f^{(i)}, f^{(i-1)}, x^{(i-1)}\right) \triangleq \frac{|A_{\mathsf{err}}^{(i)}|}{|L^{(i)}|}$$

- **closest codeword** Let $\bar{f}^{(i)}$ denote the $\mathsf{RS}^{(i)}$-codeword that is closest to $f^{(i)}$ in the $\Delta^{(i)}(\cdot)$-measure, breaking ties arbitrarily. Let $\mathcal{S}_B\left(f^{(i)}\right) \subset \mathcal{S}^{(i)}$ denote the set of "bad" cosets on which $f^{(i)}$ and $\bar{f}^{(i)}$ disagree,

$$\mathcal{S}_B\left(f^{(i)}\right) = \left\{ S \in \mathcal{S}^{(i)} \mid f^{(i)}|_S \neq \bar{f}^{(i)}|_S \right\}. \tag{15}$$

  Let $D^{(i)} = \bigcup_{S \in \mathcal{S}_B^{(i)}} S$ denote the subset of $L^{(i)}$ of elements that belong to some "bad" coset. Notice that $\delta^{(i)} < (1-\rho)/2$ implies uniqueness of $\bar{f}^{(i)}, \mathcal{S}_B^{(i)}$ and $D^{(i)}$.

- **distortion set** For $\epsilon > 0$ the *distortion set* of $f^{(i)}$ is

$$B\left[f^{(i)}; \epsilon\right] = \left\{ x^{(i)} \in \mathbb{F} \mid \Delta_{\mathsf{H}}\left(f^{(i+1)}_{f^{(i)}, x^{(i)}}, \mathsf{RS}^{(i+1)}\right) < \epsilon \right\}$$

  Notice the use of the Hamming distance measure above.

4.2.2 Proof of soundness

The following pair of lemmas will be needed to complete the analysis of soundness.

Lemma 4.3 (Soundness above unique decoding radius). *For any $\epsilon \geq \frac{2^\eta}{|\mathbb{F}|}$ and $\delta^{(i)} > 0$*

$$\Pr_{x^{(i)} \in \mathbb{F}} \left[x^{(i)} \in B\left[f^{(i)}; \frac{1}{2} \cdot \left(\delta^{(i)}(1-\epsilon) - \rho \right) \right] \right] \leq \frac{2^\eta}{\epsilon |\mathbb{F}|} \tag{16}$$

Lemma 4.4 (Soundness within unique decoding radius). *If $\delta^{(i)} < (1-\rho)/2$ then*

$$\Pr_{x^{(i)} \in \mathbb{F}} \left[x^{(i)} \in B\left[f^{(i)}, \delta^{(i)} \right] \right] \leq \frac{|L^{(i)}|}{|\mathbb{F}|}. \tag{17}$$

Moreover, suppose that for $i < \mathsf{r}$ the sequences $\vec{f} = \left(f^{(i)}, \ldots, f^{(\mathsf{r})}\right)$ and $\vec{x} = \left(x^{(i)}, \ldots, x^{(\mathsf{r}-1)}\right)$ satisfy

1. *for all $j \in \{i, \ldots, \mathsf{r}\}$ we have $\delta^{(j)} < \frac{1-\rho}{2}$*
2. *for all $j \in \{i, \ldots, \mathsf{r}-1\}$ we have $\bar{f}^{(j+1)} = f^{(j+1)}_{\bar{f}^{(j)}, x^{(j)}}$*
3. *for all $j \in \{i, \ldots, \mathsf{r}-1\}$ we have $x^{(j)} \notin B\left[f^{(j)}; \delta^{(j)}\right]$*

Then

$$\Pr_{s^{(i)} \in D^{(i)}} \left[\mathsf{QUERY}\left(\vec{f}, \vec{x}\right) = \mathsf{reject} \right] = 1 \tag{18}$$

and consequently

$$\Pr_{s^{(i)} \in L^{(i)}} \left[\mathsf{QUERY}\left(\vec{f}, \vec{x}\right) = \mathsf{reject} \right] \geq \frac{|D^{(i)}|}{|L^{(i)}|} = \delta^{(i)} \tag{19}$$

We are ready to prove the soundness of the protocol, in three steps. First, we define a sequence of "bad" events $E^{(0)}, \ldots, E^{(\mathsf{r}-1)}$ that may occur (only) during the COMMIT phase. Second, we bound from above the probability that some bad event occurs by $\frac{3|L^{(0)}|}{|\mathbb{F}|}$, as stated in (6). Third and last, assuming no bad event occurs, we bound from below the probability of the verifier rejecting during the QUERY phase, proving this rejection probability is at least as stated in (7). Details follow.

Proof of Theorem 3.3, item 2 (soundness). Set $\epsilon = \frac{2^\eta}{|L^{(\mathsf{r}/2)}|}$; for simplicity we assume r is even (using $\epsilon = \frac{2^\eta}{|L^{(\lceil \mathsf{r}/2 \rceil)}|}$ gives the same bounds but with a slightly messier analysis).

Part I — A sequence of bad events The ith bad event $E^{(i)}$ is defined thus:

- **large distance:** If $\delta^{(i)} \geq \frac{1-\rho}{2}$ then $E^{(i)}$ is the event

$$x^{(i)} \in B\left[f^{(i)}; \frac{1}{2} \cdot \left(\delta^{(i)}(1-\epsilon) - \rho \right) \right]$$

- **small distance:** If $\delta^{(i)} < \frac{1-\rho}{2}$ then $E^{(i)}$ is the event

$$x^{(i)} \in B\left[f^{(i)}, \delta^{(i)} \right]$$

Assuming that event $E^{(i)}$ does not hold implies that for $\delta^{(i)} < \frac{1-\rho}{2}$,

$$\Delta^{(i+1)}\left(f^{(i+1)}_{f^{(i)},x^{(i)}}, \mathsf{RS}^{(i+1)}\right) \geq \delta^{(i)} \tag{20}$$

and for $\delta^{(i)} \geq \frac{1-\rho}{2}$,

$$\Delta^{(i+1)}\left(f^{(i+1)}_{f^{(i)},x^{(i)}}, \mathsf{RS}^{(i+1)}\right) \geq \frac{1}{2}\cdot\left(\delta^{(i)}(1-\epsilon)-\rho\right) \geq \frac{(1-\rho)(1-\epsilon)}{4} - \frac{\rho}{2} \geq \frac{1-3\rho-\epsilon}{4}. \tag{21}$$

We use δ_0 to denote the rightmost term of (21) and summarize this by saying that when no $E^{(i)}$ holds we have

$$\Delta^{(i+1)}\left(f^{(i+1)}_{f^{(i)},x^{(i)}}, \mathsf{RS}^{(i+1)}\right) \geq \min\left\{\delta^{(i)}, \delta_0\right\} \tag{22}$$

Part II — bounding the probability of a bad event occurring By Lemmas 4.3 and 4.4, and by our choice of ϵ we have

$$\Pr\left[E^{(i)}\right] \leq \max\left\{\frac{2^\eta}{\epsilon|\mathbb{F}|}, \frac{|L^{(i)}|}{|\mathbb{F}|}\right\} \leq \begin{cases} \frac{|L^{(i)}|}{|\mathbb{F}|} & i \leq \mathsf{r}/2 \\ \frac{|L^{(\mathsf{r}/2)}|}{|\mathbb{F}|} & i > \mathsf{r}/2 \end{cases}$$

so the probability that none of $E^{(0)}, \ldots, E^{(\mathsf{r}-1)}$ hold is at least

$$1 - \sum_{i\leq \mathsf{r}/2} \frac{|L^{(i)}|}{|\mathbb{F}|} + \frac{\mathsf{r}|L^{(\mathsf{r}/2)}|}{2|\mathbb{F}|} > 1 - 3\frac{|L^{(0)}|}{|\mathbb{F}|}.$$

The inequality above follows because $\mathsf{r} \leq \log|L^{(0)}|$ and $|L^{(\mathsf{r}/2)}| = \sqrt{|L^{(0)}}$. We continue with the proof assuming no $E^{(i)}$ holds.

Part III — bounding soundness when no bad event occurred There are two cases to consider. The first and simpler case is when the sequences $\vec{f} = \left(f^{(0)}, \ldots, f^{(\mathsf{r})}\right)$ and $\vec{x} = \left(x^{(0)}, \ldots, x^{(\mathsf{r}-1)}\right)$ satisfy the three assumptions of Lemma 4.4; In this case Lemma 4.4 immediately gives the desired lower bound of $\delta^{(0)}$ on rejection probability.

The other case is when the sequences $\vec{f}$ and $\vec{x}$ do not satisfy all conditions of Lemma 4.4. It cannot be the case that both assumptions 1 and 2 hold while assumption 3 fails, because that would imply that some event $E^{(i)}$ holds, contradicting our earlier assumption. Thus, it must be the case that either assumption 1 or 2 of that lemma fails to hold for the sequences $\vec{f}, \vec{x}$, so there exists some $i \in \{0, \ldots, \mathsf{r}-1\}$ for which either

1. $\delta^{(i)} \geq \frac{1-\rho}{2}$, or
2. $\delta^{(i)} < \frac{1-\rho}{2}$ and $\bar{f}^{(i+1)} \neq f^{(i+1)}_{\bar{f}^{(i)},x^{(i)}}$

Abusing notation, let $i < \mathsf{r}$ be the largest integer satisfying either of the above conditions. Notice $D^{(i+1)}$ is uniquely defined because $\delta^{(i+1)} < \frac{1-\rho}{2}$ and hence $\bar{f}^{(i+1)}$ is unique. The following claim says that the honest prover's $(i+1)$ message is at least δ_0 from $\bar{f}^{(i+1)}$ in relative Hamming distance.

Claim 4.5.

$$\Delta_{\mathsf{H}}\left(\bar{f}^{(i+1)}, f^{(i+1)}_{f^{(i)},x^{(i)}}\right) \geq \delta_0$$

Proof. If $\delta^{(i)} \geq \frac{1-\rho}{2}$ then the assumption that $E^{(i)}$ doesn't occur means that $\Delta_{\mathsf{H}}\left(f^{(i+1)}_{f^{(i)},x^{(i)}}, \mathsf{RS}^{(i+1)}\right) \geq \delta_0$ and the claim clearly holds. Otherwise we are in the case that both $\delta^{(i)} < \frac{1-\rho}{2}$ and $\bar{f}^{(i+1)} \neq f^{(i+1)}_{\bar{f}^{(i)},x^{(i)}}$ hold. To simplify exposition denote $f^{(i+1)}_{\bar{f}^{(i)},x^{(i)}}$ by g. Both $\bar{f}^{(i+1)}$ and g belong to $\mathsf{RS}^{(i+1)}$, hence are at least $(1-\rho)$-far from each other. The triangle inequality gives

$$1-\rho \leq \Delta_{\mathsf{H}}\left(\bar{f}^{(i+1)}, g\right) \leq \Delta_{\mathsf{H}}\left(\bar{f}^{(i+1)}, f^{(i+1)}_{f^{(i)},x^{(i)}}\right) + \Delta_{\mathsf{H}}\left(f^{(i+1)}_{f^{(i)},x^{(i)}}, g\right) \tag{23}$$

By the assumption $\delta^{(i)} < \frac{1-\rho}{2}$ we have $\Delta_{\mathsf{H}}\left(f^{(i+1)}_{f^{(i)},x^{(i)}}, g\right) < \frac{1-\rho}{2}$. Rearranging (23) gives

$$\Delta_{\mathsf{H}}\left(\bar{f}^{(i+1)}, f^{(i+1)}_{f^{(i)},x^{(i)}}\right) \geq \Delta_{\mathsf{H}}\left(\bar{f}^{(i+1)}, g\right) - \Delta_{\mathsf{H}}\left(f^{(i+1)}_{f^{(i)},x^{(i)}}, g\right) > (1-\rho) - \frac{1-\rho}{2} > \frac{1-\rho}{2} > \delta_0.$$

This completes the proof. □

Our next claim is

Claim 4.6.

$$\frac{|A^{(i+1)}_{\mathsf{err}} \cup D^{(i+1)}|}{|L^{(i+1)}|} \geq \Delta_{\mathsf{H}}\left(\bar{f}^{(i+1)}, f^{(i+1)}_{f^{(i)},x^{(i)}}\right). \tag{24}$$

Proof. For all $x \notin A^{(i+1)}_{\mathsf{err}} \cup D^{(i+1)}$ we have

$$\bar{f}^{(i+1)}(x) = f^{(i+1)}(x) = f^{(i+1)}_{f^{(i)},x^{(i)}}(x)$$

because the first equality holds for $x \notin D^{(i+1)}$ and the second for $x \notin A^{(i+1)}_{\mathsf{err}}$. But

$$\Pr_{x \in L^{(i+1)}}\left[\bar{f}^{(i+1)}(x) \neq f^{(i+1)}_{f^{(i)},x^{(i)}}(x)\right] = \Delta_{\mathsf{H}}\left(f^{(i+1)}_{f^{(i)},x^{(i)}}, \bar{f}^{(i+1)}\right)$$

so the claim holds. □

Combining Claims 4.5 and 4.6 gives

$$\frac{|A^{(i+1)}_{\mathsf{err}} \cup D^{(i+1)}|}{|L^{(i+1)}|} \geq \delta_0.$$

Consider $s^{(i+1)}$ used during the QUERY phase. If $s^{(i+1)} \in A^{(i+1)}_{\mathsf{err}}$ then the QUERY test rejects by definition. If $i+1 = \mathsf{r}$ then $D^{(i+1)} = \emptyset$ by definition becuase $f^{(\mathsf{r})} \in \mathsf{RS}^{(\mathsf{r})}$ so in this case we have already shown the rejection probability is at least δ_0. Otherwise we are in the case that $i+1 < \mathsf{r}$ and by choice of i, the sequences $\left(f^{(i+1)}, \ldots, f^{(\mathsf{r})}\right)$ and $\left(x^{(i+1)}, \ldots, x^{(\mathsf{r}-1)}\right)$ which are both non-empty, satisfy all three assumptions of Lemma 4.4. By the conclusion of that lemma, if $s^{(i+1)} \in D^{(i+1)}$ then the QUERY phase rejects, cf. (18). We have shown that the probability of error is at least the probability that $s^{(i+1)}$ belongs to $A^{(i+1)}_{\mathsf{err}} \cup D^{(i+1)}$ and this probability is at least δ_0, completing the proof of soundness. □

4.2.3 Unique decoding radius — Proof of Lemma 4.4

Proof of Lemma 4.4. Recall $\bar{f}^{(i)}$ and $\mathcal{S}_B\left(f^{(i)}\right)$ are uniquely defined because $\delta^{(i)} < \frac{1-\rho}{2}$. For a bad coset $S \in \mathcal{S}_B\left(f^{(i)}\right)$ let

$$X_S^{(i)} \triangleq \left\{x^{(i)} \in \mathbb{F} \mid \mathsf{interpolant}^{f^{(i)}|_S}\left(x^{(i)}\right) = \mathsf{interpolant}^{\bar{f}^{(i)}|_S}\left(x^{(i)}\right)\right\} \tag{25}$$

be the set of "misleading" values of $x^{(i)}$ on which $\mathsf{interpolant}^{f^{(i)}|_S}$ and $\mathsf{interpolant}^{\bar{f}^{(i)}|_S}$ agree, even though the two are distinct low-degree polynomials. We claim that

$$B\left[f^{(i)}, \delta^{(i)}\right] = \bigcup_{S\in\mathcal{S}_B\left(f^{(i)}\right)} X_S^{(i)} \tag{26}$$

Indeed, by Lemma 4.1 we conclude that $f^{(i+1)}_{\bar{f}^{(i)},x^{(i)}} \in \mathsf{RS}^{(i+1)}$. For all $S \notin \mathcal{S}_B\left(f^{(i)}\right)$ with $y_S = q^{(i)}(S)$ we have $f^{(i+1)}_{f^{(i)},x^{(i)}}(y_S) = f^{(i+1)}_{\bar{f}^{(i)},x^{(i)}}(y_S)$. Since $\delta^{(i)}$ is smaller than the unique decoding distance it follows that $f^{(i+1)}_{\bar{f}^{(i)},x^{(i)}}$ is the $\mathsf{RS}^{(i+1)}$-codeword closest to $f^{(i+1)}_{f^{(i)},x^{(i)}}$ in Hamming distance. Therefore, $\Delta_{\mathsf{H}}\left(f^{(i+1)}_{f^{(i)},x^{(i)}}, \mathsf{RS}^{(i+1)}\right) = \Delta_{\mathsf{H}}\left(f^{(i+1)}_{f^{(i)},x^{(i)}}, f^{(i+1)}_{\bar{f}^{(i)},x^{(i)}}\right)$ and the two functions agree on y_S if and only if either $S \notin \mathcal{S}_B\left(f^{(i)}\right)$ or $S \in \mathcal{S}_B\left(f^{(i)}\right)$ *and* $x^{(i)} \in X_S^{(i)}$. This shows that $f^{(i+1)}_{f^{(i)},x^{(i)}}$ disagrees with the (unique) closest $\mathsf{RS}^{(i+1)}$-codeword, which is $\bar{f}^{(i+1)}_{f^{(i)},x^{(i)}}$, on all of $\left\{y_S \mid S \in \mathcal{S}_B\left(f^{(i)}\right)\right\}$ if and only if $x^{(i)} \notin \bigcup_{S\in\mathcal{S}_B\left(f^{(i)}\right)} X_S^{(i)}$, proving (26).

With this equality in hand, we bound the right hand side of (26). Indeed, $\mathsf{interpolant}^{f^{(i)}|_S}$ and $\mathsf{interpolant}^{\bar{f}^{(i)}|_S}$ are distinct polynomials of degree less than $|S|$, so $|X_S| < |S|$ and hence

$$\left|B\left[f^{(i)}; \delta^{(i)}\right]\right| = \left|\bigcup_{S\in\mathcal{S}\left(f^{(i)}\right)} X_S\right| < |S| \cdot \left|\mathcal{S}_B\left(f^{(i)}\right)\right| \leq |L^{(i)}|,$$

and this proves (17).

Next, consider the sequences $\vec{f}, \vec{x}$ assumed in the Lemma. Assume for simplicity that $\bar{f}^{(i)}$ is the zero function evaluated over $L^{(i)}$, by subtracting $\bar{f}^{(i)}$ from $f^{(i)}$ if this is not the case; denote by $\mathbf{0}|_{L^{(i)}}$ this function. Then

$$f^{(i+1)}_{\bar{f}^{(i)},x^{(i)}} = f^{(i+1)}_{\mathbf{0}|_{L^{(i)}},x^{(i)}} = \mathbf{0}|_{L^{(i+1)}}$$

so by assumption 2 of Lemma 4.4 we have $\bar{f}^{(i+1)} = \mathbf{0}|_{L^{(i+1)}}$ and similarly by induction we have $\bar{f}^{(j)} = \mathbf{0}|_{L^{(j)}}$ for all $j \in \{i, \ldots, \mathsf{r}\}$. In particular, $f^{(\mathsf{r})} = \mathbf{0}|_{L^{(\mathsf{r})}}$.

Consider the sequence $\left(s^{(i)}, \ldots, s^{(\mathsf{r})}\right)$ defined in the QUERY phase, where $s^{(i)} \in D^{(i)}$. Let j denote the largest integer such that $s^{(j)} \in D^{(j)}$. This j is well defined because $s^{(i)} \in D^{(i)}$. Notice $j < \mathsf{r}$ because by definition $f^{(\mathsf{r})} = \mathbf{0}|_{L^{(\mathsf{r})}}$ so $D^{(\mathsf{r})} = \emptyset$. Assumption 3 of Lemma 4.4 together with (26) implies that $x^{(j)} \notin \cup_{S\in\mathcal{S}^{(j)}} X_S^{(j)}$, so $f^{(j+1)}_{f^{(j)},x^{(j)}}\left(s^{(j+1)}\right) \neq 0$. But by definition of j we have $f^{(j+1)}\left(s^{(j+1)}\right) = \bar{f}^{(i+1)}\left(s^{(j+1)}\right) = 0$. We conclude

$$f^{(j+1)}_{f^{(j)},x^{(j)}}\left(s^{(j+1)}\right) \neq f^{(j+1)}\left(s^{(j+1)}\right)$$

which means that QUERY rejects the sequence $\left(s^{(i)}, \ldots, s^{(\mathsf{r})}\right)$. This proves (18), and thus implies (19) because $\delta^{(i)} = \left|D^{(i)}\right| / |L^{(i)}|$. □

4.2.4 Beyond unique decoding radius — Proof of Lemma 4.3

To prove Lemma 4.3 we need the following improved version of Lemma 4.2.18 from [Spi95]. See Appendix A for a proof sketch.

Lemma 4.7. *Let $E(X,Y)$ be a polynomial of degree $(\alpha m, \delta n)$ and $P(X,Y)$ a polynomial of degree $((\alpha+\epsilon)m, (\delta+\rho)n)$. If there exist distinct $x_1,\ldots,x_m$ such that $E(x_i,Y)|P(x_i,Y)$ and $y_1,\ldots,y_n$ such that $E(X,y_i)|P(X,y_i)$ and*

$$1 > \max\left\{\delta+\rho, 2\alpha+\epsilon+\frac{\rho}{\delta}\right\} \tag{27}$$

then $E(X,Y)|P(X,Y)$.

Proof of Lemma 4.3. We shall prove the contrapositive, namely, if for some $\epsilon \geq \frac{2^\eta}{|\mathbb{F}|}$

$$\frac{\left|B\left[f^{(i)}; \frac{1}{2}\cdot(\delta(1-\epsilon)-\rho)\right]\right|}{|\mathbb{F}|} > \frac{2^\eta}{\epsilon|\mathbb{F}|} \tag{28}$$

then

$$\Delta^{(i)}\left(f, \mathsf{RS}^{(i)}\right) < \delta. \tag{29}$$

We fix a few constants: Let $n = |L^{(i+1)}|$, $\alpha = \frac{1}{2}\left(1-\epsilon-\frac{\rho}{\delta}\right)$, $\delta' = \delta\cdot\alpha$, $B = B\left[f^{(i)};\delta'\right]$, and $m = |B|$. By definition, for every $x \in B$ we have $\Delta_{\mathsf{H}}\left(f^{(i+1)}_{f^{(i)},x}, \mathsf{RS}^{(i+1)}\right) < \delta'$. Recall $\bar{f}^{(i+1)}_{f^{(i)},x} \in \mathsf{RS}^{(i+1)}$ is the codeword closest to $f^{(i+1)}_{f^{(i)},x}$, breaking ties arbitrarily.

Let $C(X,Y)$ be the polynomial with $\deg_X(C) < m, \deg_Y(C) < \rho n$ that agrees with $\bar{f}^{(i+1)}_{f^{(i)},x}$ for each $x \in B$; this polynomial exists because, by definition, $\bar{f}^{(i+1)}_{f^{(i)},x}$ is an evaluation of a polynomial of degree less than ρn. Let $Q^{(i)}$ be the polynomial corresponding to $f^{(i)}$ from Claim 4.2 as defined in (10) and recall from item 2 of that claim that $\deg_X\left(Q^{(i)}\right) < |L_0^{(i)}|$; By definition $|L_0^{(i)}| = 2^\eta$ and by assumption above $2^\eta < \epsilon m$, so $\deg_X\left(Q^{(i)}\right) < \epsilon m$. From item 1 of Claim 4.2 we deduce that for all $x \in \mathbb{F}$ and $y \in L^{(i+1)}$ we have $Q^{(i)}(x,y) = f^{(i+1)}_{f^{(i)},x}(y)$. By assumption (28),

$$\Pr_{x\in B, y\in L^{(i+1)}}\left[C(x,y) \neq Q^{(i)}(x,y)\right] \leq \delta'. \tag{30}$$

By construction $\alpha\delta \geq \delta'$, so there exists a non-zero polynomial

$$E(X,Y), \quad \deg_X(E) \leq \alpha m, \deg_Y(E) \leq \delta n$$

that vanishes on all points (x,y) where $x \in B, y \in L^{(i+1)}$ and $C(x,y) \neq Q^{(i)}(x,y)$. The polynomial E is known as the *error locator polynomial* [Sud92] because its zeros cover the set of error locations, where Q deviates from being a low-degree polynomial.

Since $\deg_Y(C) < \rho|L^{(i+1)}|$ and $\deg_X(Q^{(i)}) < 2^\eta = \epsilon m$, by [Spi95, Chapter 4] there exists a polynomial $P(X,Y)$ satisfying

$$\deg_X(P) < (\epsilon+\alpha)m \text{ and } \deg_Y(P) < (\rho+\delta)n \tag{31}$$

such that

$$\forall x \in B, y \in L^{(i+1)} P(x,y) = C(x,y) \cdot E(x,) = Q^{(i)}(x,y) \cdot E(x,y) \tag{32}$$

We conclude from (31), (32) that for every row $y \in L^{(i+1)}$ we have $E(X,y)|P(X,y)$ and similarly for every column $x \in B$ we have $E(x,Y)|P(x,Y)$. By (5) we have $\delta + \rho \leq 1$, and by definition of α we also have $2\alpha + \epsilon + \rho/\delta \leq 1$. So the assumption (27) of Lemma 4.7 holds. By the conclusion of that lemma $E(X,Y)|P(X,Y)$ as polynomials in the ring $\mathbb{F}[X,Y]$. Let $Q \equiv P/E$. We conclude Q agrees with $Q^{(i)}$ on every row $y \in L^{(i+1)}$ such that $E(X,y)$ is non-zero. By the bound on $\deg_Y(E)$, the fraction of such rows is at least $1-\delta$. In other words $f^{(i)}$ agrees with some polynomial of degree $\rho|L^{(i)}|$ on more than a $(1-\delta)$-fraction of cosets of $L_0^{(i)}$ in $L^{(i)}$, implying (29) and completing the proof of the Lemma. □

4.3 Soundness upper bound — Part 3

Given $\delta^{(0)} \in (0, 1-\rho)$, partition $\mathcal{S}^{(0)}$ into two sets: a set $\mathcal{S}'$ of fraction $\delta^{(0)}$ and $\mathcal{S}'' = \mathcal{S} \setminus \mathcal{S}'$ of fraction $1-\delta^{(0)}$. Pick an arbitrary polynomial P of degree $2^\eta - 1$ that vanishes on a set of size $2^\eta - 1$ that is disjoint from $\bigcup_{S \in \mathcal{S}'} S$. Let $f^{(0)}$ be defined as follows: for $S \in \mathcal{S}'$ let $f^{(0)}|_S$ be the evaluation of P on S and for $S \in \mathcal{S}''$ let $f^{(0)}|_S = \mathbf{0}$. Furthermore, for $i > 0$ let $f^{(i)} = \mathbf{0}$.

We claim $f^{(0)}, \ldots, f^{(\mathsf{r})}$ satisfy the two bullets of Part 3. By construction, $f^{(0)}$ agrees with $\mathbf{0}$ on $\mathcal{S}''$ therefore $f^{(0)}$ is precisely $\delta^{(0)}$-far from $\mathbf{0}$. Similarly by construction, only the 0th layer has a positive round error $\mathsf{err}^{(0)} = \delta^{(0)}$ and this proves the first bullet of Part 3. Finally, if $x^{(0)}$ is a root of P, which happens with probability $(2^\eta - 1)/|\mathbb{F}|$ over $x^{(0)} \in \mathbb{F}$, then $f^{(0)}, f^{(1)}, \ldots, f^{(\mathsf{r})}$ are accepted with probability 1 during the QUERY phase. This proves the second bullet of Part 3 and completes the proof.

4.4 Prover complexity — Part 4

Consider the computation performed by the prover during the ith step of the protocol. At this point the prover has already committed to $f^{(i)} : L^{(i)} \to \mathbb{F}$ and has received $x^{(i)} \in \mathbb{F}$ and needs to compute $f^{(i+1)} : L^{(i+1)} \to \mathbb{F}$ as explained in steps 2–4 of the COMMIT protocol.

During step 2, for each distinct coset $S_y \in \mathcal{S}^{(i)}$, the prover needs to interpolate the polynomial $P_y(X)$ defined there and evaluate it on $x^{(i)}$ to obtain $f^{(i+1)}(y) = P_y\left(x^{(i)}\right)$. If $x^{(i)} \in S_y$ then $f^{(i+1)}(y) = f^{(i)}\left(x^{(i)}\right)$ and the computation terminates in single step. Otherwise, using Lagrange interpolation,

$$P_y\left(x^{(i)}\right) = \sum_{\alpha \in S_y} f^{(i)}(\alpha) \cdot \frac{\prod_{\beta \in S_y}\left(\beta - x^{(i)}\right)}{\prod_{\beta \in S_y \setminus \{\alpha\}}(\beta - \alpha)} = \mathsf{Zero}_{S_y}\left(x^{(i)}\right) \cdot \sum_{\alpha \in S_y} \frac{f^{(i)}(\alpha)}{c_\alpha \cdot \left(x^{(i)} - \alpha\right)} \tag{33}$$

where $c_\alpha = \prod_{\gamma \in S_y \setminus \{\alpha\}}(\gamma - \alpha)$ can be precomputed in advance on a PRAM with sufficiently many processors because c_α does not depend on $f^{(i)}, x^{(i)}$. The polynomial Zero_{S_y} is linearized and has $\eta + 1$ terms, hence can be valuated on $x^{(i)}$ via repeated squaring in $3\eta + 3$ cycles in the PRAM-CREW model (each cycle is a single arithmetic operation in $\mathbb{F}$), using η processors and a total of $3\eta + 3$ arithmetic operations: (i) $\eta + 1$ squarings to obtain the relevant powers of $x^{(i)}$, (ii) $\eta + 1$ multiplications and (iii) $\eta + 1$ additions to evaluate the polynomial once the powers of $x^{(i)}$ are known. The summation on the right hand side of (33) has 2^η terms so it can be computed separately in parallel using 2^η processors and $\eta + 2$ cycles for a total of $4 \cdot 2^{\eta+2}$ arithmetic operations. The total PRAM-CREW number of cycles is $2\eta + O(1)$ using 2^η processors and at most $4(2^\eta + \eta + 1)$ arithmetic operations.

The calculation above refers to a single $y \in L^{(i+1)}$. Summing over all such y shows that the ith step requires a total of $|L^{(i)}| + \eta$ processors, and is computed in $3\eta + 1$ cycles using a total of $3\left(|L^{(i)}| + \eta + 1\right)$ arithmetic operations. For $i = \mathsf{r}$ the function $f^{(\mathsf{r})}$ — which is evaluated on $L^{(\mathsf{r})}, |L^{(\mathsf{r})}| \leq 2^\eta$ — needs to be interpolated; this can be done using 2^η processors in 3η cycles because $\deg\left(P^{(\mathsf{r})}\right) < |L^{(\mathsf{r})}| \leq 2^\eta$ (details omitted).

Summing over all r steps completes the proof of this part.

Remark 4.8 (Arithmetic complexity for smooth RS codes)**.** *For smooth codes, prover complexity is somewhat smaller than mentioned above, because* $\mathsf{Zero}_{S_y} = X^{2^\eta} - \zeta$ *for a constant* ζ *depending on* S_y*; i.e.,* Zero_{S_y} *has only* 2 *terms as opposed to* $\eta + 1$ *terms in the additive case. Similar savings ore obtained for the verifier arithmetic complexity (discussed next) in the smooth case. Notice, howoever, that Theorem 1.2 sets* $\eta = 2$ *and hence the difference between arithmetic complexity in the additive and smooth cases is minor.*

4.5 Verifier complexity — Part 5

The unit of measurement for communication and query complexity is field elements of $\mathbb{F}$. During the COMMIT phase the verifier sends a total of $\mathsf{r} \leq (k^{(0)} - \mathcal{R})/\eta$ field elements. During the QUERY phase, the verifier precomputes $q^{(i)}\left(s^{(i)}\right)$ where $q^{(i)}$ is a linearized polynomial with $\eta + 1$ terms whose coefficients are precomputed because they are independent of all verifier messages $x^{(i)}$ and prover oracles $f^{(i)}$. Using the explanation above (for evaluating Zero_{S_y}), each $q^{(i)}\left(s^{(i)}\right)$ evaluation costs $3(\eta + 1)$ cycles and arithmetic operations.

Having specified the query set $s^{(i)}$, the verifier now receives a total of $\ell \cdot \mathsf{r} \cdot 2^\eta$ field elements as answers, and solves $\ell \cdot \mathsf{r}$ interpolation and evaluation problems of the kind described in (33). Using the explanation provided in Section 4.4 we conclude that the verifiers work can be performed on a PRAM with exclusive read and write (EREW) using $6(\eta + 1)$ cycles for, requiring $\ell \cdot \mathsf{r} \cdot 2^\eta$ processors and a total of $6\ell \cdot \mathsf{r} \cdot (2^\eta + \eta + 1) \leq 6\ell \cdot \frac{k^{(0)} - \mathcal{R}}{\eta} \cdot (2^\eta + \eta + 1)$ arithmetic operations.

References

[ALM+92] Sanjeev Arora, Carsten Lund, Rajeev Motwani, Madhu Sudan, and Mario Szegedy. Proof verification and hardness of approximation problems. In *Proceedings of the 33rd Annual Symposium on Foundations of Computer Science*, pages 14–23, 1992.

[ALM+98] Sanjeev Arora, Carsten Lund, Rajeev Motwani, Madhu Sudan, and Mario Szegedy. Proof verification and the hardness of approximation problems. *Journal of the ACM*, 45(3):501–555, 1998. Preliminary version in FOCS '92.

[AS98] Sanjeev Arora and Shmuel Safra. Probabilistic checking of proofs: a new characterization of NP. *Journal of the ACM*, 45(1):70–122, 1998. Preliminary version in FOCS '92.

[Bab85] László Babai. Trading group theory for randomness. In *Proceedings of the 17th Annual ACM Symposium on Theory of Computing*, STOC '85, pages 421–429, 1985.

[BBC+16] Eli Ben-Sasson, Iddo Bentov, Alessandro Chiesa, Ariel Gabizon, Daniel Genkin, Matan Hamilis, Evgenya Pergament, Michael Riabzev, Mark Silberstein, Eran Tromer, and Madars Virza. Computational integrity with a public random string from quasi-linear PCPs. *IACR Cryptology ePrint Archive*, 2016:646, 2016.

[BBGR16a] Eli Ben-Sasson, Iddo Bentov, Ariel Gabizon, and Michael Riabzev. A security analysis of probabilistically checkable proofs. *Electronic Colloquium on Computational Complexity (ECCC)*, 23:149, 2016.

[BBGR16b] Eli Ben-Sasson, Iddo Bentov, Ariel Gabizon, and Michael Riabzev. A security analysis of probabilistically checkable proofs. *Electronic Colloquium on Computational Complexity (ECCC)*, 23:149, 2016.

[BBHR17] Eli Ben-Sasson, Iddo Bentov, Ynon Horesh, and Michael Riabzev. Scalable, transparent, and post-quantum secure computational integrity, 2017. Unpublished manuscript.

[BCCT12] Nir Bitansky, Ran Canetti, Alessandro Chiesa, and Eran Tromer. From extractable collision resistance to succinct non-interactive arguments of knowledge, and back again. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, ITCS '12, pages 326–349, 2012.

[BCF+16] Eli Ben-Sasson, Alessandro Chiesa, Michael A. Forbes, Ariel Gabizon, Michael Riabzev, and Nicholas Spooner. On probabilistic checking in perfect zero knowledge. *Electronic Colloquium on Computational Complexity (ECCC)*, 23:156, 2016.

[BCG+13] Eli Ben-Sasson, Alessandro Chiesa, Daniel Genkin, Eran Tromer, and Madars Virza. SNARKs for C: Verifying program executions succinctly and in zero knowledge. In *Proceedings of the 33rd Annual International Cryptology Conference*, CRYPTO '13, pages 90–108, 2013.

[BCG+14] Eli Ben-Sasson, Alessandro Chiesa, Christina Garman, Matthew Green, Ian Miers, Eran Tromer, and Madars Virza. Zerocash: Decentralized anonymous payments from Bitcoin. In *Proceedings of the 2014 IEEE Symposium on Security and Privacy*, SP '14, 2014.

[BCG+16] Eli Ben-Sasson, Alessandro Chiesa, Ariel Gabizon, Michael Riabzev, and Nicholas Spooner. Short interactive oracle proofs with constant query complexity, via composition and sumcheck. *Electronic Colloquium on Computational Complexity (ECCC)*, 23:46, 2016.

[BCGT13] Eli Ben-Sasson, Alessandro Chiesa, Daniel Genkin, and Eran Tromer. On the concrete efficiency of probabilistically-checkable proofs. In *Proceedings of the 45th ACM Symposium on the Theory of Computing*, STOC '13, pages 585–594, 2013.

[BCGV16] Eli Ben-Sasson, Alessandro Chiesa, Ariel Gabizon, and Madars Virza. Quasilinear-size zero knowledge from linear-algebraic PCPs. In *Proceedings of the 13th Theory of Cryptography Conference*, TCC '16, pages 33–64, 2016.

[BCH+96] Mihir Bellare, Don Coppersmith, Johan Håstad, Marcos A. Kiwi, and Madhu Sudan. Linearity testing in characteristic two. *IEEE Trans. Information Theory*, 42(6):1781–1795, 1996.

[BCS16] Eli Ben-Sasson, Alessandro Chiesa, and Nicholas Spooner. *Interactive Oracle Proofs*, pages 31–60. Springer Berlin Heidelberg, Berlin, Heidelberg, 2016.

[BFL90] László Babai, Lance Fortnow, and Carsten Lund. Nondeterministic exponential time has two-prover interactive protocols. In *Proceedings of the 31st Annual Symposium on Foundations of Computer Science*, SFCS '90, pages 16–25, 1990.

[BFLS91] László Babai, Lance Fortnow, Leonid A. Levin, and Mario Szegedy. Checking computations in polylogarithmic time. In *Proceedings of the 23rd Annual ACM Symposium on Theory of Computing*, STOC '91, pages 21–32, 1991.

[BGH+05] Eli Ben-Sasson, Oded Goldreich, Prahladh Harsha, Madhu Sudan, and Salil Vadhan. Short PCPs verifiable in polylogarithmic time. In *Proceedings of the 20th Annual IEEE Conference on Computational Complexity*, CCC '05, pages 120–134, 2005.

[BGH+06] Eli Ben-Sasson, Oded Goldreich, Prahladh Harsha, Madhu Sudan, and Salil P. Vadhan. Robust PCPs of proximity, shorter PCPs, and applications to coding. *SIAM Journal on Computing*, 36(4):889–974, 2006.

[BKK+16] Eli Ben-Sasson, Yohay Kaplan, Swastik Kopparty, Or Meir, and Henning Stichtenoth. Constant rate pcps for circuit-sat with sublinear query complexity. *J. ACM*, 63(4):32:1–32:57, 2016.

[BLR93] Manuel Blum, Michael Luby, and Ronitt Rubinfeld. Self-testing/correcting with applications to numerical problems. *J. Comput. Syst. Sci.*, 47(3):549–595, 1993.

[BS08] Eli Ben-Sasson and Madhu Sudan. Short PCPs with polylog query complexity. *SIAM Journal on Computing*, 38(2):551–607, 2008. Preliminary version appeared in STOC '05.

[CMT12] Graham Cormode, Michael Mitzenmacher, and Justin Thaler. Practical verified computation with streaming interactive proofs. In *Proceedings of the 4th Symposium on Innovations in Theoretical Computer Science*, ITCS '12, pages 90–112, 2012.

[CT65] James W Cooley and John W Tukey. An algorithm for the machine calculation of complex fourier series. *Mathematics of computation*, 19(90):297–301, 1965.

[DG08] Irit Dinur and Elazar Goldenberg. Locally testing direct product in the low error range. In *49th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2008, October 25-28, 2008, Philadelphia, PA, USA*, pages 613–622, 2008.

[Din07] Irit Dinur. The PCP theorem by gap amplification. *Journal of the ACM*, 54(3):12, 2007.

[DR04] Irit Dinur and Omer Reingold. Assignment testers: Towards a combinatorial proof of the PCP theorem. In *Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science*, FOCS '04, pages 155–164, 2004.

[FS86] Amos Fiat and Adi Shamir. How to prove yourself: practical solutions to identification and signature problems. In *Proceedings of the 6th Annual International Cryptology Conference*, CRYPTO '86, pages 186–194, 1986.

[FS95] K. Friedl and M. Sudan. Some improvements to total degree tests. In *Proceedings Third Israel Symposium on the Theory of Computing and Systems*, pages 190–198, Jan 1995.

[GGPR13] Rosario Gennaro, Craig Gentry, Bryan Parno, and Mariana Raykova. Quadratic span programs and succinct NIZKs without PCPs. In *Proceedings of the 32nd Annual International Conference on Theory and Application of Cryptographic Techniques*, EUROCRYPT '13, pages 626–645, 2013.

[GKdO+17] Sivakanth Gopi, Swastik Kopparty, Rafael Mendes de Oliveira, Noga Ron-Zewi, and Shubhangi Saraf. Locally testable and locally correctable codes approaching the gilbert-varshamov bound. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2017, Barcelona, Spain, Hotel Porta Fira, January 16-19*, pages 2073–2091, 2017.

[GKR08] Shafi Goldwasser, Yael Tauman Kalai, and Guy N. Rothblum. Delegating computation: Interactive proofs for Muggles. In *Proceedings of the 40th Annual ACM Symposium on Theory of Computing*, STOC '08, pages 113–122, 2008.

[GS00] Oded Goldreich and Shmuel Safra. A combinatorial consistency lemma with application to proving the pcp theorem. *SIAM Journal on Computing*, 29(4):1132–1154, 2000.

[Hås01] Johan Håstad. Some optimal inapproximability results. *Journal of the ACM*, 48(4):798–859, 2001.

[HBHW17] Daira Hopwood, Sean Bowe, Taylor Hornby, and Nathan Wilcox, March 2017.

[IKW12] Russell Impagliazzo, Valentine Kabanets, and Avi Wigderson. New direct-product testers and 2-query pcps. *SIAM Journal on Computing*, 41(6):1722–1768, 2012.

[IMSX15] Yuval Ishai, Mohammad Mahmoody, Amit Sahai, and David Xiao. On zero-knowledge PCPs: Limitations, simplifications, and applications, 2015. Available at `http://www.cs.virginia.edu/~mohammad/files/papers/ZKPCPs-Full.pdf`.

[IW14] Yuval Ishai and Mor Weiss. Probabilistically checkable proofs of proximity with zero-knowledge. In *Theory of Cryptography - 11th Theory of Cryptography Conference, TCC 2014, San Diego, CA, USA, February 24-26, 2014. Proceedings*, pages 121–145, 2014.

[Kil92] Joe Kilian. A note on efficient zero-knowledge proofs and arguments. In *Proceedings of the 24th Annual ACM Symposium on Theory of Computing*, STOC '92, pages 723–732, 1992.

[KMRS16] Swastik Kopparty, Or Meir, Noga Ron-Zewi, and Shubhangi Saraf. High-rate locally-correctable and locally-testable codes with sub-polynomial query complexity. In *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016, Cambridge, MA, USA, June 18-21, 2016*, pages 202–215, 2016.

[KPT97] Joe Kilian, Erez Petrank, and Gábor Tardos. Probabilistically checkable proofs with zero knowledge. In *Proceedings of the 29th Annual ACM Symposium on Theory of Computing*, STOC '97, pages 496–505, 1997.

[KR08] Yael Kalai and Ran Raz. Interactive PCP. In *Proceedings of the 35th International Colloquium on Automata, Languages and Programming*, ICALP '08, pages 536–547, 2008.

[LFKN92] Carsten Lund, Lance Fortnow, Howard J. Karloff, and Noam Nisan. Algebraic methods for interactive proof systems. *Journal of the ACM*, 39(4):859–868, 1992.

[LN97] Rudolf Lidl and Harald Niederreiter. *Finite Fields*. Cambridge University Press, second edition edition, 1997.

[Mic00] Silvio Micali. Computationally sound proofs. *SIAM Journal on Computing*, 30(4):1253–1298, 2000. Preliminary version appeared in FOCS '94.

[Mie09] Thilo Mie. Short PCPPs verifiable in polylogarithmic time with o(1) queries. *Annals of Mathematics and Artificial Intelligence*, 56:313–338, 2009.

[MR10] Dana Moshkovitz and Ran Raz. Two-query PCP with subconstant error. *J. ACM*, 57(5):29:1–29:29, 2010.

[Ore33] O. Ore. On a special class of polynomials. *Trans. Amer. Math. Soc.*, 35(3):559–584, 1933.

[Ore34] O. Ore. Contributions to the theory of finite fields. *Trans. Amer. Math. Soc.*, 36(2):243–274, 1934.

[Pec16] M. Peck. A blockchain currency that beat s bitcoin on privacy [news]. *IEEE Spectrum*, 53(12):11–13, December 2016.

[PS94] Alexander Polishchuk and Daniel A. Spielman. Nearly-linear size holographic proofs. In *Proceedings of the 26th Annual ACM Symposium on Theory of Computing*, STOC '94, pages 194–203, 1994.

[PZ03] John Proos and Christof Zalka. Shor's discrete logarithm quantum algorithm for elliptic curves. *Quantum Info. Comput.*, 3(4):317–344, July 2003.

[Raz95] Ran Raz. A parallel repetition theorem. In *Proceedings of the 27th Annual ACM Symposium on Theory of Computing*, STOC '95, pages 447–456, 1995.

[RRR16] Omer Reingold, Guy N. Rothblum, and Ron D. Rothblum. Constant-round interactive proofs for delegating computation. In *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016, Cambridge, MA, USA, June 18-21, 2016*, pages 49–62, 2016.

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

[RS92] Ronitt Rubinfeld and Madhu Sudan. Self-testing polynomial functions efficiently and over rational domains. In *Proceedings of the Third Annual ACM/SIGACT-SIAM Symposium on Discrete Algorithms, 27-29 January 1992, Orlando, Florida.*, pages 23–32, 1992.

[Sha92] Adi Shamir. IP = PSPACE. *Journal of the ACM*, 39(4):869–877, 1992.

[Sho94] P. W. Shor. Algorithms for quantum computation: discrete logarithms and factoring. In *Proceedings 35th Annual Symposium on Foundations of Computer Science*, pages 124–134, Nov 1994.

[Spi95] Daniel A. Spielman. *Computationally Efficient Error-Correcting Codes and Holographic Proofs*. PhD thesis, MIT, 1995.

[Spi96] Daniel A. Spielman. Linear-time encodable and decodable error-correcting codes. *IEEE Transactions on Information Theory*, 42(6):1723–1731, 1996. Preliminary version appeared in STOC '95.

[Sud92] Madhu Sudan. *Efficient Checking of Polynomials and Proofs and the Hardness of Approximation Problems*. PhD thesis, UC Berkeley, Berkeley, CA, USA, 1992. UMI Order No. GAX93-30747.

[Val08] Paul Valiant. Incrementally verifiable computation or proofs of knowledge imply time/space efficiency. In *Proceedings of the 5th Theory of Cryptography Conference*, TCC '08, pages 1–18, 2008.

[WB15] Michael Walfish and Andrew J. Blumberg. Verifying computations without reexecuting them. *Commun. ACM*, 58(2):74–84, 2015.

A Proof of Lemma 4.7

We restate Lemma 4.7 using the same notation as in [Spi96].

Lemma A.1. *Let $E(X, Y)$ be a polynomial of degree $(\alpha m, \beta n)$ and $P(X, Y)$ a polynomial of degree $((\alpha + \delta)m, (\beta + \epsilon)n)$. If there exist distinct $x_1, \dots, x_m$ such that $E(x_i, Y)|P(x_i, Y)$ and $y_1, \dots, y_n$ such that $E(X, y_i)|P(X, y_i)$ and*

$$1 > \max\left\{\beta + \epsilon, 2\alpha + \delta + \frac{\epsilon}{\beta}\right\} \tag{34}$$

then $E(X, Y)|P(X, Y)$.

The difference between the version above and the original one is that (34) is replaced in [Spi96] with

$$1 > \alpha + \beta + \delta + \epsilon \tag{35}$$

So, to prove the statement above we guide the reader through the proof in [Spi96] (pages 97–98) using the notation there and point out the modifications needed to use (34) instead of (35). Details follow.

First, we do not assume $\beta \geq \alpha$ as there. Next, the inequality

$$\alpha + \beta + \delta + \epsilon \geq \frac{\alpha m - a}{m - a} + \frac{\delta m - a}{m - a} + \frac{\beta n - b}{n - b} + \frac{\epsilon n - b}{n - b}$$

there is replaced with the two inequalities

$$\beta + \epsilon \geq \frac{\beta n - b}{n - b} + \frac{\epsilon n - b}{n - b}$$

and

$$2\alpha + \delta + \frac{\epsilon}{\beta} \geq 2\frac{\alpha m - a}{m - a} + \frac{\delta m - a}{m - a} + \frac{\epsilon n - b}{\beta n - b}$$

which hold because each term on the left hand side is less than 1, as follows from (34).

Finally, before the very last inequality on page 98 there, replace both (i) the use of (35) and (ii) the assumption $\beta \geq \alpha$, with the assumption (34) to show

$$\beta mn > mn(\beta\alpha + \beta\delta + \alpha\beta + \alpha\epsilon)$$

To see that this inequality follows from (34), simply divide both sides by βmn. Therefore Lemma A.1, the restatement of Lemma 4.7, holds as claimed.

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il