Fiat-Shamir Security of FRI and Related SNARKs

Alexander R. Block[1,2(✉)], Albert Garreta[3], Jonathan Katz[2], Justin Thaler[1,5], Pratyush Ranjan Tiwari[4], and Michał Zając[3]

[1] Georgetown University, Washington, D.C., USA
justin.thaler@georgetown.edu
[2] University of Maryland, College Park, USA
alexander.r.block@gmail.com,jkatz2@gmail.com
[3] Nethermind, London, UK
{albert,michal}@nethermind.io
[4] Johns Hopkins University, Baltimore, USA
pratyush@cs.jhu.edu
[5] A16z Crypto Research, Menlo Park, USA

Abstract. We establish new results on the Fiat-Shamir (FS) security of several protocols that are widely used in practice, and we provide general tools for establishing similar results for others. More precisely, we: (1) prove the FS security of the FRI and batched FRI protocols; (2) analyze a general class of protocols, which we call δ-*correlated*, that use low-degree proximity testing as a subroutine (this includes many "Plonk-like" protocols (e.g., Plonky2 and Redshift), ethSTARK, RISC Zero, etc.); and (3) prove FS security of the aforementioned "Plonk-like" protocols, and sketch how to prove the same for the others.

We obtain our first result by analyzing the round-by-round (RBR) soundness and RBR knowledge soundness of FRI. For the second result, we prove that if a δ-correlated protocol is RBR (knowledge) sound under the assumption that adversaries always send low-degree polynomials, then it is RBR (knowledge) sound in general. Equipped with this tool, we prove our third result by formally showing that "Plonk-like" protocols are RBR (knowledge) sound under the assumption that adversaries always send low-degree polynomials. We then outline analogous arguments for the remainder of the aforementioned protocols.

To the best of our knowledge, ours is the first formal analysis of the Fiat-Shamir security of FRI and widely deployed protocols that invoke it.

1 Introduction

Succinct Non-interactive ARguments of Knowledge (SNARKs) and their zero-knowledge variants (zkSNARKs) are a thriving field of study both in theory and practice. Allowing for fast verification of complex statements made by untrusted parties, zkSNARKs have now been deployed in a myriad of applications. A popular paradigm for constructing (zk)SNARKs is via the following two-step process:

© International Association for Cryptologic Research 2023
J. Guo and R. Steinfeld (Eds.): ASIACRYPT 2023, LNCS 14439, pp. 3–40, 2023.
https://doi.org/10.1007/978-981-99-8724-5_1

(1) construct a public-coin[1] interactive protocol; and (2) remove all interaction using the Fiat-Shamir (FS) transformation [31], adding zero-knowledge as necessary.

Non-interactivity is essential in many applications of zkSNARKs. In general, interactive protocols are not publicly verifiable and hence cannot be used in settings where anyone in the world should be able to verify the proof. There are various proposals (e.g., [5]) to render interactive protocols publicly verifiable using so-called randomness beacons [61] (i.e., publicly verifiable sources of random bits, such as contents blockchain block headers) and the transaction-ordering functionality offered by public blockchains (which enable the public to verify that the prover sent a message before it knew what the verifier's response to that message would be). However, to the best of our knowledge, such proposals have not been deployed at scale. They are also fraught with performance and security considerations; for example, blockchain headers are at least somewhat biasable [17,57], and splitting an interactive proof across many blockchain blocks can substantially increase latency and fees.

Regardless, the Fiat-Shamir transformation is pervasive and has been used extensively in a variety of schemes beyond zkSNARKs, e.g., signature schemes and non-interactive zero-knowledge [31,54,58], inspiring a rich line of research into understanding both its applicability and pitfalls. The FS transformation is typically modeled and analyzed in the random oracle model (ROM) for security proofs. When using FS in practice, one then assumes that a suitable concrete hash function (e.g., SHA256) is an adequate replacement for said random oracle.

However, there are surprisingly many open problems regarding specific applications of the FS transformation. In particular, the FS transformation is *not* secure in general [3,13,36], even in the random oracle model, when applied to many-round protocols. Specifically, its use can lead to a loss in the number of "bits of security" that is linear in the number of rounds r of the protocol to which it is applied. Here, the number of bits of security roughly refers to the logarithm of the amount of work an attacker has to do to succeed with probability close to 1.

Accordingly, the FS transformation is often applied to many-round protocols without formal security proofs for the resulting SNARKs' security. That is, the security analysis of these protocols is often provided only for their interactive versions. Without further analysis, the security (measured in bits) lost via the FS transformation may be a factor equal to the number of rounds of the protocol. Even a 30% loss in security would be devastating in practical deployments (e.g., reducing the number of bits of security from 100 down to 70), and (more than) such a loss can occur even when Fiat-Shamir is applied to protocols with just two rounds. There are also some works that claim FS-security of their protocols, but in fact show this only under the assumption that certain many-round subprotocols used in the overall protocol are FS-secure [25,26,44].

[1] A protocol is *public-coin* if all messages sent by the verifier are sampled uniformly at random from a challenge space and are independent of all prior prover and verifier messages.

In this work, we fill this gap in these security analyses and provide general tools for doing so for certain varieties of protocols. Specifically, we show that for the protocols we are interested in, the security of the FS-transformed protocol resembles the security of the interactive one (pre-FS) (or more precisely, *what is currently known* about the interactive security). This adds to a recent fruitful line of work has introduced many tools to understand FS security of many-round protocols. These include the notions of state-restoration soundness [9], round-by-round soundness [22], and (generalized) special soundness [2,28,72]. Nonetheless, in the literature on SNARKs, relatively few protocols are known to be FS-secure, despite the above tools existing. These include the GKR protocol [22,37] (or more generally, anything based on the sum-check protocol [49]), the GMW protocol and other natural classes of "commit-and-open" protocols [41], and any protocol satisfying the notion of (generalized) special soundness [2], which includes IPA/Bulletproofs [18,20]. Bulletproofs [18,20] and Sonic [50] have separately been shown to be FS-secure in the algebraic group model [35].

In this introduction, we informally refer to protocols that experience little-to-no loss in the number of bits of security when the FS transformation is applied in the random oracle model as *FS-secure*.

1.1 Our Results

We formally analyze and prove FS-security of the FRI protocol [4] and of some protocols that have wide use in practice which use low-degree proximity testing as a subroutine. For the latter, we build a general tool that allows us to prove FS-security of a certain type of protocol, which we call a δ-correlated IOP, by analyzing its round-by-round soundness assuming an adversary sends low-degree polynomials. We formally apply this tool to "Plonk-like" protocols such as Plonky2 [60], and we outline how the tool can be used on other constructions such as ethSTARK [65]. In particular, we either formally prove or we outline a proof that the security of all these protocols, after applying the Fiat-Shamir transformation, (nearly) matches what is known about its security when run interactively.

As mentioned, we focus on these protocols due to their current popularity. For example, FRI is currently used in various Layer-2 Ethereum projects [59,66] to help secure hundreds of millions of dollars of assets [46]. Some projects are deploying FRI with (at most) 80-bits (dYdX) or 96-bits (those using the SHARP prover) of *interactive security* before the FS transformation is applied [6,65,66]. More precisely, the *best known attacks* on these interactive protocols have success probability 2^{-80} or 2^{-96}. These attacks are conjectured to be optimal [65], though only partial results in this direction are known [6]. Similarly, Plonk-like protocols are utilized in a variety of blockchain networks and Layer 2 Ethereum projects (e.g., [30,51,55,56,67]),

When it comes to the FRI protocol, we *do not* address the gaps between the conjectured and known soundness of the interactive protocol. We merely analyze the security of the FS-compiled protocol as a function of the security of the interactive protocol.

1.2 Technical Details

In a nutshell, we formally establish the *round-by-round (knowledge) soundness* [22] of both FRI and several protocols that rely on a form of low-degree proximity testing. For analyzing round-by-round (RBR) soundness, there is a protocol *state* that can either be "doomed" or not. The state of the protocol starts off as doomed whenever a prover falsely claims that an input is valid. If at the end of interaction the state is doomed, the verifier rejects. The protocol is said to be RBR sound if, whenever the state is doomed, the protocol is still doomed in the next round of interaction, except with negligible probability, no matter what a prover does. RBR knowledge soundness is a similar notion, except that in this case, the protocol always starts off in a doomed state, and after each round, except with negligible probability, it remains doomed unless the prover knows a valid witness; see Sect. 2.1 for more discussion.

By establishing the round-by-round (knowledge) soundness of these protocols, we can then leverage the so-called BCS transformation [9], which (informally) compiles any interactive protocol[2] into a (zk)SNARK via (a variant of) the Fiat-Shamir transformation in the random oracle model. Applying the BCS transformation on a round-by-round (knowledge) sound protocol preserves (knowledge) soundness (yielding a SNARK) [25,26].[3] In fact, round-by-round soundness of the interactive protocol was even shown to imply that the BCS-transformed SNARK is secure against quantum adversaries [25]. Thus, we establish the Fiat-Shamir security of both FRI and the rest of protocols via proving their round-by-round (knowledge) soundness.

Round-by-Round Soundness of FRI. The FRI protocol [4], which stands for **F**ast **R**eed-Solomon **I**nteractive Oracle Proof of Proximity is a logarithmic round *interactive oracle proof*. Briefly, an interactive oracle proof (IOP) [9] is an interactive protocol where the verifier is given oracle (i.e., query) access to the (long) prover messages, and an IOP of Proximity (IOPP) is an IOP for proving proximity of a function to some pre-specified linear error-correcting code [4]. The FRI protocol proves that a function is close to the space of Reed-Solomon codewords [62] of a certain degree over some pre-specified domain over a finite field. This protocol is both of theoretical and practical interest. On the theory side, FRI gives a polylogarithmic-size proof for proving the proximity of messages to some pre-specified Reed-Solomon code, which is an important primitive in many proof systems [4]. On the practical side, FRI is used as a sub-protocol in the design and construction of many SNARKs and has the benefit of being plausibly post-quantum secure due to its avoidance of elliptic curve cryptography (and in fact, it follows from our results that FRI, when run non-interactively via Fiat-Shamir, is unconditionally secure in the quantum random oracle model).

[2] More formally, the BCS transformation is applied to *interactive oracle proofs* [9].

[3] Actually, [9,25] prove this for *state-restoration soundness*; however, subsequent works observed that round-by-round soundness is an upper bound on state-restoration soundness [22,25,26,44].

Despite intense interest from both theorists and practitioners, we are unaware of any formal security proof for FRI under Fiat-Shamir.

Theorem 1 (Informally Stated; see Theorem 6). *For finite field $\mathbb{F}$, evaluation domain $L \subset \mathbb{F}$ of size 2^n, constants $\rho \in (0,1)$, $\delta \in (0, 1-\sqrt{\rho})$, and positive integer ℓ, the FRI protocol has round-by-round (knowledge) soundness error*

$$\varepsilon_{\mathsf{rbr}}^{\mathsf{FRI}}(\mathbb{F}, \rho, \delta, n, \ell) = \max\{O(2^{2n}/(\rho^{3/2}|\mathbb{F}|)), (1-\delta)^{\ell}\}.$$

Establishing the round-by-round (knowledge) soundness of FRI is a crucial first step to establishing the Fiat-Shamir security of FRI. In particular, given the round-by-round soundness of FRI, we can now apply the BCS transformation [9] to obtain a secure non-interactive argument in the random oracle model using FRI.

Corollary 1 (Informally Stated; see Corollary 4). *For finite field $\mathbb{F}$, evaluation domain $L \subset \mathbb{F}$ of size 2^n, constants $\rho \in (0,1)$, $\delta \in (0, 1-\sqrt{\rho})$, and positive integer ℓ, given a random oracle with κ-bits of output and query bound $Q \geq 1$, compiling FRI with the BCS transformation yields a non-interactive argument in the random oracle model with adaptive soundness error and knowledge error*

$$\varepsilon_{\mathsf{fs}}^{\mathsf{FRI}}(\mathbb{F}, \rho, \delta, n, \ell, Q, \kappa) = Q\varepsilon_{\mathsf{rbr}}^{\mathsf{FRI}}(\mathbb{F}, \rho, \delta, n, \ell) + O(Q^2/2^{\kappa}).$$

Moreover, the transformed non-interactive argument has adaptive soundness error and knowledge error $\Theta(Q \cdot \varepsilon_{\mathsf{fs}}^{\mathsf{FRI}}(\mathbb{F}, \rho, \delta, n, \ell, Q))$ against $O(Q)$-query quantum adversaries.

Extension to Batched FRI. In practice, it is common to run a *Batched FRI* protocol, which allows a prover to simultaneously prove the δ-correlated agreement[4] of t functions $f_1, \ldots, f_t$ by running the FRI protocol on the batched function $G = \sum_i \alpha_i f_i$ for randomly sampled α_i provided by the verifier. We extend our analysis of FRI to this version of Batched FRI and establish its round-by-round (knowledge) soundness.

Theorem 2 (Informally Stated, see Theorem 7). *For finite field $\mathbb{F}$, evaluation domain $L \subset \mathbb{F}$ of size 2^n, constants $\rho \in (0,1)$, $\delta \in (0, 1-\sqrt{\rho})$, and positive integers ℓ, t, the Batched FRI protocol has round-by-round (knowledge) soundness error*

$$\varepsilon_{\mathsf{rbr}}^{\mathsf{bFRI}}(\mathbb{F}, \rho, \delta, n, \ell, t) = \max\{O((2^{2n})/(\rho^{3/2}|\mathbb{F}|)), (1-\delta)^{\ell}\}.$$

As before, establishing round-by-round soundness allows us to securely apply the BCS transformation, obtaining a non-interactive argument in the random oracle model.

[4] Informally, functions have δ-correlated agreement if they are all δ-close to some prespecified Reed-Solomon code and all have the same agreement set; see [14] for full details.

Corollary 2 (Informally Stated; see Corollary 5). *For finite field $\mathbb{F}$, evaluation domain $L \subset \mathbb{F}$ of size 2^n, constants $\rho \in (0,1)$, $\delta \in (0, 1-\sqrt{\rho})$, and positive integers ℓ, t, given a random oracle with κ-bits of output and query bound $Q \geq 1$, compiling Batched FRI with the BCS transformation yields a non-interactive argument in the random oracle model with adaptive soundness error and knowledge error*

$$\varepsilon_{\mathsf{fs}}^{\mathsf{bFRI}}(\mathbb{F}, \rho, \delta, n, \ell, t, Q, \kappa) = Q \cdot \varepsilon_{\mathsf{rbr}}^{\mathsf{bFRI}}(\mathbb{F}, \rho, \delta, n, \ell, t) + O(Q^2/2^\kappa).$$

Moreover, the transformed non-interactive argument has adaptive soundness error and knowledge error $\Theta(Q \cdot \varepsilon_{\mathsf{fs}}^{\mathsf{bFRI}}(\mathbb{F}, \rho, \delta, n, \ell, t, Q, \kappa))$ against $O(Q)$-query quantum adversaries.

To the best of our knowledge, our results are the first to establish the security of non-interactive analogs of FRI and Batched FRI in the random oracle model.

A Variant of Batched FRI. To save on communication costs, a variant of Batched FRI is sometimes used, where the batched function G has the form $G = \sum_i \alpha^{i-1} f_i$ for challenge α randomly sampled and sent by the verifier. In both the context of regular soundness and round-by-round soundness, this version of Batched FRI incurs some soundness loss proportional to t. In particular, in Theorem 2, the round-by-round soundness error for this Batched FRI protocol is $\varepsilon_{\mathsf{rbr}}^{\mathsf{bFRI}}(\mathbb{F}, \rho, \delta, n, \ell, t) = \max\{O((2^{2n} \cdot t)/(\rho^{3/2}|\mathbb{F}|)), (1-\delta)^\ell\}$; see [14] for complete details.

Round-by-Round Soundness Error versus Standard Soundness Error of FRI. Ben-Sasson et al. [6] give the best known provable soundness bounds for (Batched) FRI; in fact, we leverage many tools from their results to show our round-by-round soundness bounds. Roughly speaking, [6] show that the soundness error of (Batched) FRI is $\varepsilon_1 + \varepsilon_2 + \varepsilon_3$, where $\varepsilon_1 = O(2^{2n}/(\rho^{3/2}|\mathbb{F}|))$, $\varepsilon_2 = O((2^n \cdot n\sqrt{\rho})/|\mathbb{F}|)$, and $\varepsilon_3 = (1-\delta)^\ell$. Then our RBR soundness bound for (Batched) FRI is given by $\max\{\varepsilon_1, \varepsilon_3\}$.

Round-by-Round Knowledge Error. Both FRI and Batched FRI additionally have *round-by-round knowledge error* [25,26,44] identical to the round-by-round soundness errors given in Theorems 1 and 2. The BCS transformation preserves this type of knowledge soundness, yielding a SNARK. See Sect. 2.1 for more discussion.

A General Tool for Proving RBR (Knowledge) Soundness. We go on to analyze proof systems that rely on the FRI protocol as a subroutine. To this end, we introduce a family of IOPs which we call *δ-correlated IOPs*, where $\delta \geq 0$ is a parameter. This family encompasses all of the aforementioned protocols. In a nutshell, we say an IOP is δ-correlated if the prover is supposed to send oracles to maps that are δ-close to low-degree polynomials in a *correlated* manner. Correlation here means that the domain where these maps agree with low-degree

polynomials is the same among all the maps. In a δ-correlated IOP, during the verification phase, the verifier: (1) checks some algebraic equalities involving some evaluations of these maps; and (2) verifies that all the received oracles correspond indeed to δ-correlated maps (we assume the verifier has another oracle to perform this check). When $\delta = 0$, a δ-correlated IOPs can be seen as a subclass of RS-encoded IOPs [8,26]. See [14] for a more in-depth comparison.

This points to a "recipe" for building a particular family of SNARKs: first, construct a δ-correlated IOP; then, instantiate the check for δ-correlation using an interactive protocol, e.g., Batched FRI [6]. This produces an IOP as a result. Finally, use the aforementioned BCS transformation on this IOP to produce a succinct non-interactive argument. If this argument is knowledge sound, one has obtained a SNARK. Figure 1 summarizes this construction. It is immediate to see that the previously mentioned protocols (Plonky2, RISC Zero, ethSTARKs, etc.) are actual instantiations of this construction.

Fig. 1. A recipe for building a succinct non-interactive argument.

We then provide general results for proving that the resulting succinct non-interactive argument is knowledge sound. Precisely, we prove the following:

1. **RBR soundness of Batched FRI.** As a general result, we prove that the (Batched) FRI protocol is RBR sound and RBR knowledge sound. We remark that Batched FRI can be used for checking δ-correlated agreement of a collection of maps [6].
2. **From RBR knowledge when the adversary sends low degree polynomials, to general RBR knowledge.** Consider a δ-correlated IOP Π, and suppose attackers always send oracles to low degree polynomials. We prove that if Π is RBR (knowledge) sound under this assumption, then it is also RBR (knowledge) sound in general, and that the soundness error only increases by a (relatively) small factor.
3. **From a RBR knowledge sound δ-correlated IOP to a RBR knowledge sound IOP.** Again let Π be a δ-correlated IOP. By using an interactive protocol Π_{CA} to check for δ-correlation, Π can be turned into a regular IOP Π_{compiled}. We prove that this compilation preserves RBR (knowledge) soundness, assuming Π_{CA} is RBR sound (not necessarily RBR knowledge sound).
4. **From a RBR knowledge sound IOP to a SNARK.** We then apply the BCS compilation results from [9] to obtain a SNARK.

In conclusion, we show that given any succinct non-interactive argument constructed as in Fig. 1 (using Batched FRI to check for δ-correlation), one can show its knowledge soundness simply by proving RBR knowledge soundness of the underlying δ-correlated IOP *under the assumption that the adversary is*

constrained to sending oracles to low-degree polynomials. The latter can greatly simplify the analysis since it allows one to work with the simplicity of IOPs (as opposed to arguments) and the convenient properties of polynomials.

Thus, our methods not only allow us to prove FS-security, they also remove the complexity of dealing with maps that are close to low-degree polynomials when using FRI within a protocol. This allows us to analyze the interactive version of these protocols in a similar way as when one studies *Polynomial* IOPs [21], where, by definition, soundness is only considered for adversaries that send low-degree polynomials.

According to our formalism, a δ-correlated IOP where we constrain adversaries to always send low-degree polynomials is in fact a 0-correlated IOP. Then, Item (2) above can be seen as a result that relates the RBR knowledge soundness of a δ-correlated IOP for $\delta = 0$ and for $\delta > 0$. Overall, our security results can be organized and depicted as in Fig. 2; also see Theorem 3.

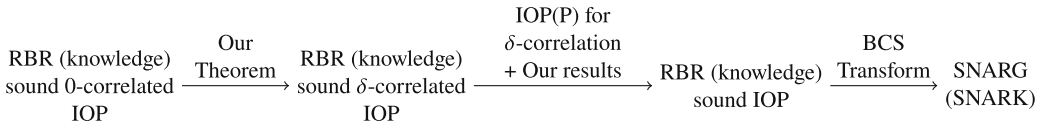

Fig. 2. Another recipe for building a SNARG/SNARK.

Theorem 3 (Informally Stated, see Theorem 8). *Let $\Pi_\delta^{\mathcal{O}}$ be a δ-correlated IOP, where $\mathcal{O}$ is an oracle for δ-correlated agreement. Let $0 < \rho, \eta \leq 1$ and $\delta = 1 - \sqrt{\rho} - \eta$. Assume Π_0 has RBR knowledge soundness with error ε. Then Π_δ has RBR knowledge soundness with error $\varepsilon/(2\sqrt{\rho}\eta)$.*

Moreover, if Π' is an IOP for testing δ-correlated agreement in a Reed-Solomon code with RBR soundness error ε', then the protocol Π_{compiled} obtained by replacing $\mathcal{O}$ in Π_δ with Π' has RBR knowledge soundness with error $\varepsilon_{\mathsf{compiled}} = \max\{\varepsilon/(2\sqrt{\rho}\eta), \varepsilon'\}$. Finally, given a random oracle with κ-bits of output and query bound $Q \geq 1$, compiling Π_{compiled} with the BCS transformation yields a succinct non-interactive argument in the random oracle model with knowledge error $Q \cdot \max\{\varepsilon/(2\sqrt{\rho}\eta), \varepsilon'\} + O(Q^2/2^\kappa)$.

Remark 1. As we mentioned, the notion of δ-correlated IOP is closely related to that of *RS-encoded IOP* from [8,26]. The works of [8,26] also provide a method for compiling a RBR (knowledge) sound RS-encoded IOP into RBR (knowledge sound IOPs); e.g., see [26, Theorem 8.2]. However, our result allows to use a proximity parameter up to the Johnson bound, i.e., we can select $\delta = 1 - \sqrt{\rho} - \eta$ for any arbitrarily small $\eta > 0$, while the compilation results from [8,26] constrain δ to be within the unique decoding radius $\delta < \frac{1-\rho}{2}$. On the other hand, in some sense, RS-encoded IOPs encompass a wider class of protocols than δ-correlated ones. See [14] for further discussion.

Remark 2. Many security analyses of SNARKs obtained by combining Plonk-like protocols with so-called KZG polynomial commitments [43] can assume that an adversary always sends oracles to polynomials of appropriate degree. Intuitively, this is due to the fact that the KZG polynomial commitment scheme ensures that a committed function is indeed a polynomial of appropriate degree.

However, in our scenario, due to the usage of the FRI protocol instead of KZG, adversaries are only constrained to sending (oracles to) maps that are *close to* polynomials of appropriate degree. This makes the soundness analysis of our protocols more subtle. Indeed, as we mentioned, besides showing that FRI itself is RBR sound, most of our work is concerned with reducing the analysis to the case when the adversary actually sends oracles to polynomials of the appropriate degree.

Round-by-Round Soundness of Specific δ-Correlated Proof Systems. We can apply all the tools developed so far to specific protocols whose construction follows the outline from Figs. 1 and 2. In short, these are protocols obtained by compiling a δ-correlated IOP into a succinct non-interactive argument via a protocol for δ-correlated agreement and the BCS transformation. Thanks to Theorems 2 and 3, we can prove the knowledge soundness of these protocols just by proving that the corresponding 0-correlated IOP has RBR knowledge soundness. Recall that in a 0-correlated IOP, the adversary is assumed to always send oracles to low-degree polynomials.

Some of the protocols that fit into this framework are many "Plonk-like" proof systems that use FRI instead of the KZG polynomial commitment scheme; e.g., Plonky2 [60], Redshift [44], and RISC Zero [68]. Here we use the term "Plonk-like" to loosely refer to protocols that use an interactive permutation argument [15,19,47,48,73] as a subroutine (we use the term "Plonk-like" because the Plonk SNARK [34] helped popularize the use of this permutation-checking procedure). Other protocols that fit in our framework but are not "Plonk-like" are ethSTARK or DEEP-ALI [10].

We focus our attention mostly on Plonky2 since we believe that, among all these protocols in 0-correlated IOP form, Plonky2 is the most involved to analyze. Indeed, Plonky2 was designed to be used over a small field (the 64-bit so-called Goldilocks field). Because of this, some checks are repeated in parallel in order to increase its security. The task of correctly designing these parallel repetitions is subtle, and indeed in the full version of our work [14], we describe an (arguably more natural) variation of Plonky2 with dramatically *less* security than Plonky2 itself. To the best of our knowledge, this variation is *not used* in practice—we are highlighting it here to illustrate a potential pitfall to be avoided.

Accordingly, we rigorously define a general "Plonk-like" δ-correlated IOP, which captures many "Plonk-like" protocols that rely on the FRI protocol. We denote this δ-correlated IOP by $\mathsf{OPlonky}(\delta)$. We then formally prove that when $\delta = 0$ (i.e., when adversaries are constrained to sending low-degree polynomials), $\mathsf{OPlonky}(0)$ has RBR soundness and knowledge soundness. Together with our general results and our results on batched FRI, this proves in particular that the SNARK version of Plonky2 is indeed knowledge sound (as well as all

other "Plonk-like" protocols of the form $\mathsf{OPlonky}(\delta)$). Adapting Theorem 3 to our abstraction $\mathsf{OPlonky}$, we obtain the following result.

Theorem 4 (Informally Stated, see Lemmas 1 and 3). *Let $\mathbb{F}$ be a finite field and $\mathbb{K}$ be a finite extension of $\mathbb{F}$ and let $D \subseteq \mathbb{F}$ be an evaluation domain for maps. Let $\mathcal{P} = \{P_1, \ldots, P_k\}$ be a list of $2r+\ell$-variate circuit constraint polynomials over $\mathbb{F}$ for $k, r, \ell \geq 1$. For parameters $n, t, u \geq 1$, $s = \lceil r/u \rceil$, and $m \geq 3, \rho = (n+1)/|D| \in (0,1), \eta \in (0, \sqrt{\rho}/2m)$ and $\delta = 1 - \sqrt{\rho} - \eta$, the protocol $\mathsf{OPlonky}^{O}$, when the verifier is given an oracle O for δ-correlated agreement in the Reed-Solomon code $\mathsf{RS}[\mathbb{F}, D, n+1]$, has round-by-round soundness error*

$$\varepsilon_{\mathsf{rbr}}^{\mathsf{OPlonky},O}(\mathbb{F}, \mathbb{K}, D, n, k, r, s, t, u, d, \rho, \eta) = \frac{1}{2\eta\sqrt{\rho}} \cdot \max\left\{ O\left(\left(\frac{n(r+u)}{|\mathbb{F}|}\right)^t\right), O\left(\left(\frac{k+st}{|\mathbb{F}|}\right)^t\right), \frac{n \cdot \max\{u+1, d\}}{|\mathbb{K} \setminus D|} \right\},$$

where $d = \max_i\{\deg(P_i)\}$ and D is an evaluation domain for RS codes. Moreover, when $\delta = 0$ then we have

$$\varepsilon_{\mathsf{rbr}}^{\mathsf{OPlonky},O}(\mathbb{F}, \mathbb{K}, D, n, k, r, s, t, u, d, \rho, \eta) = \max\left\{ O\left(\left(\frac{n(r+u)}{|\mathbb{F}|}\right)^t\right), O\left(\left(\frac{k+st}{|\mathbb{F}|}\right)^t\right), \frac{n \cdot \max\{u+1, d\}}{|\mathbb{K} \setminus D|} \right\}.$$

Remark 3. The parameter t in Theorem 4 controls the number of times certain checks in $\mathsf{OPlonky}$ are performed "in parallel". In most Plonk-like protocols, one uses $t = 1$ and a large field $\mathbb{F}$ to ensure an adequate security level. However, some projects (e.g., Plonky2) currently feature a 64-bit field $\mathbb{F}$, and use $t = 2$ to increase security.

We show in this paper that, if done properly, the resulting FS-transformed protocol does achieve the targeted security level. However, in the full version of our work [14], we explain that this result is surprisingly subtle: certain natural ways of implementing the t-fold repetition actually result in RBR security (and, correspondingly, the post-FS security [2]) that is much lower than the one attained in Theorem 4. While (to our knowledge) all existing projects do implement the t-fold repetition properly so as to ensure FS-security, we highlight this subtlety so that protocol designers continue to avoid this potential pitfall.

We can instantiate the oracle O in Theorem 4 with Batched FRI and obtain the following result.

Theorem 5 (Informally Stated, see Theorem 9). *Let $\mathbb{F}$ be a finite field, $\mathbb{K}$ be a finite extension of $\mathbb{F}$, and $D \subset \mathbb{F}^*$. Let $\mathcal{P} = \{P_1, \ldots, P_k\}$ be a list of $2r + \ell$-variate circuit constraint polynomials over $\mathbb{F}$ for $k, r, \ell, n \geq 1$. For integer $u \geq 1$, $s = \lceil r/u \rceil$, and parameters $\rho, \eta > 0$, $\delta = 1 - \sqrt{\rho} - \eta$, and $N, q \geq 1$, the protocol $\mathsf{OPlonky}$ composed with Batched FRI (replacing O) has round-by-round soundness error:*

$$\varepsilon_{\mathsf{rbr}}^{\mathsf{OPlonky}}(\mathbb{F}, \mathbb{K}, D, n, k, r, s, t, u, d, \rho, \eta, N, q) = \max\{\varepsilon_{\mathsf{rbr}}^{\mathsf{OPlonky},O}(\mathbb{F}, \mathbb{K}, D, n, k, r, s, t, u, d, \rho, \eta), \varepsilon_{\mathsf{rbr}}^{\mathsf{bFRI}}(\mathbb{F}, D, \rho, \delta, N, q)\},$$

where $d = \max_i\{\deg(P_i)\}$.

Given the above protocol is a round-by-round sound IOP, as in Theorem 3, we can now apply the BCS transformation to obtain a secure non-interactive argument in the random oracle model.

Corollary 3 (Informally Stated; see [14]). *Let* $\mathbb{F}$ *be a finite field,* $\mathbb{K}$ *be a finite extension of* $\mathbb{F}$*, and* $D \subset \mathbb{F}^*$*. Let* $\mathcal{P} = \{P_1, \ldots, P_k\}$ *be a list of* $2r+\ell$*-variate circuit constraint polynomials over* $\mathbb{F}$ *for* $k, r, \ell, n \geq 1$*. For integers* $u, t \geq 1$*,* $s = \lceil r/u \rceil$*, and parameters* $\rho, \eta > 0$*,* $\delta = 1 - \sqrt{\rho} - \eta$*, and* $N, q \geq 1$*, given a random oracle with* κ*-bits of output and a query bound* $Q \geq 1$*, using the BCS transformation to compile* OPlonky *composed with Batched FRI yields a non-interactive argument in the random oracle model with adaptive soundness error and knowledge error*

$$\varepsilon_{\mathsf{fs}}^{\mathsf{OPlonky}}(\mathbb{F}, \mathbb{K}, D, n, k, r, s, t, u, d, \rho, \eta, N, q, \kappa, Q) = Q\varepsilon_{\mathsf{rbr}}^{\mathsf{OPlonky}}(\mathbb{F}, \mathbb{K}, D, n, k, r, s, t, u, d, \rho, \eta, N, q) + O(Q^2/2^\kappa),$$

where $d = \max_i\{\deg(P_i)\}$*. Moreover, the the transformed non-interactive argument has adaptive soundness error and knowledge error*

$$\Theta(Q \cdot \varepsilon_{\mathsf{fs}}^{\mathsf{OPlonky}}(\mathbb{F}, \mathbb{K}, D, n, k, r, s, t, u, d, \rho, \eta, N, q, \kappa, Q))$$

versus $O(Q)$*-query quantum adversaries.*

Remark 4. We stress that the above theorems do not imply anything for the original work of Plonk [34], or any other Plonk variants that utilize the so-called KZG polynomial commitment scheme [43] as their low-degree test. The tools we leverage to show Fiat-Shamir security of our protocols rely on the low-degree test also being an IOP or an IOP of Proximity, which the KZG scheme is not. While it is likely one can extend our analysis to handle using the KZG scheme, we do not explore that direction in this work.

RISC Zero and ethSTARK. When it comes to RISC Zero and ethSTARK, we sketch why their 0-correlated formulations have RBR knowledge soundness, as opposed to fully formally proving these facts. We do that due to brevity (since formally describing these protocols is a lengthy task), and because proving that these 0-correlated IOPs are RBR knowledge sound is a relatively straightforward task, as our analysis of OPlonky indicates. Moreover, RISC Zero's whitepaper is in draft form at the moment of writing [68]. We hope practitioners can follow the techniques and ideas exposed in this paper to prove in a relatively simple way that their SNARKs are indeed FS-secure.

1.3 Additional Related Work

The Fiat-Shamir (FS) transform [31] has been studied and used extensively to remove interaction from interactive protocols. While it is known that the FS

transformation is secure when applied to sound protocols with a constant number of rounds in the random oracle model (ROM) [1,31,58], it is well-known that there exist protocols that are secure under FS in the ROM but insecure for *any* concrete instantiation of the random oracle [3,13,36]. Furthermore, several natural classes of secure interactive protocols are rendered insecure when applying FS (e.g., sequential repetition of a protocol and parallel repetition of certain protocols) [2,22,72], and real-world implementations of FS are often done incorrectly, leading to vulnerabilities [12,29]. Despite this, FS is widely deployed and is a critical component in the majority of SNARG and SNARK constructions.

Recent work has extensively studied which protocols can be securely instantiated under Fiat-Shamir (either in the ROM or using suitable hash-function families). As mentioned before, the general tools of state-restoration soundness [9], round-by-round soundness [22], and special soundness [2,28,72] have been introduced as soundness notions that "behave nicely" under Fiat-Shamir. Prior to these tools, a variety of works [23,40,42] circumvented the impossibility results of [13] by utilizing stronger hardness assumptions to construct Fiat-Shamir compatible hash function families. Another line of work [7,16,27,37,63,64,69,71] follows the frameworks of Kilian [45] and Micali [53] to compile interactive oracle proofs [9] into efficient arguments and SNARKs via collision-resistant hash functions [9,45] or in the random oracle model [9,53].

1.4 Organization

In Sect. 2, we give an overview of our main technical results. Section 3 presents our main results in full detail. Section 4 discusses some future directions. Due to space constraints, most technical details are deferred to the full version of our paper [14].

2 Technical Overview

Our main technical contributions are three-fold. First, we formally prove the round-by-round (knowledge) soundness of the FRI protocol. Second, we build a general tool for proving round-by-round (knowledge) soundness of δ-correlated IOPs. Third, we give a δ-correlated IOP called OPlonky, prove its round-by-round (knowledge) soundness, and showcase how it captures many "Plonk-like" protocols used in practice. Additionally, we sketch how to extend the OPlonky analysis to the ethSTARK protocol. In Sect. 2.1, we briefly discuss round-by-round soundness and its relation to Fiat-Shamir; in Sect. 2.2, we give an overview of the round-by-round soundness of FRI and Batched FRI; in Sect. 2.3, we introduce the concept of δ-correlated IOP and prove our general results about them; in Sect. 2.4, we give an overview of the round-by-round (knowledge) soundness of OPlonky; in Sect. 2.5, we discuss how a similar analysis can be done for the ethSTARK protocol.

2.1 Round-by-Round Soundness and Fiat-Shamir

Our tool of choice for establishing Fiat-Shamir security is *round-by-round soundness* [22]. Informally, a public-coin interactive protocol for a language L is *round-by-round sound* (RBR sound) if at any point during the execution of the protocol, the protocol is in a well-defined state (depending on the protocol execution so far) and some of these states are "doomed", where being "doomed" means that no matter what message the prover sends, with overwhelming probability over the verifier messages, the protocol remains "doomed". A bit more formally, RBR soundness error ε states that: (1) if $x \notin L$ the initial state of the protocol is "doomed"; (2) if the protocol is in a "doomed" state during any non-final round of the protocol, then for any message sent by the prover, the protocol remains doomed with probability at least $1 - \varepsilon$ over the verifier messages; and (3) if the protocol terminates in a "doomed" state, then the verifier rejects. Chiesa et al. [25] extend RBR soundness to *RBR knowledge soundness*, which roughly says that if (1) the protocol is in a "doomed" state during any round of interaction, and (2) *every* prover message can force the protocol to leave this "doomed" state with probability at least ε_{k} (over the verifier randomness), then an extractor can efficiently extract a witness (with probability 1) simply by examining the current protocol state and the prover's next message.

Canetti et al. [22] introduced RBR soundness as a tool for showing Fiat-Shamir security of interactive proofs [38] when used in conjunction with a suitable family of correlation intractable hash functions [24]. In particular, random oracles are correlation intractable when the set of "doomed" states of a protocol is sufficiently sparse; i.e., for small enough RBR soundness error. RBR soundness readily extends to the language of interactive oracle proofs (IOPs) [9], and hence the Fiat-Shamir compiler result of [22] readily extends to IOPs, and can be readily adapted to the random oracle model as well. However, applying this compiler to IOPs directly introduces some undesirable effects: the constructed non-interactive argument would have proof lengths proportional to the length of the oracle sent by the prover since the compiler of [22] does not compress prover messages in any way. This leads to long proofs and verification times, negating any succinct verification the IOP may have had. Moreover, the transformation of [22] says nothing about the knowledge soundness of the resulting non-interactive argument, even in the random oracle model.

While it is likely that, in the random oracle model, one could argue that the transformation of [22] retains knowledge soundness if the underlying IOP is RBR knowledge sound, we do not prove this fact; moreover, the loss of verifier succinctness is still an issue even if knowledge soundness is retained. Thus to circumvent the above issues, we utilize the BCS transformation [9] for IOPs. Informally, the BCS transformation first compresses oracles sent by the prover using a Merkle tree [52] and then replaces any queries made by the verifier to prover oracles with additional rounds of interaction where the verifier asks the prover its queries, and the prover responds with said queries and Merkle authentication paths to verify consistency. It was shown that if an IOP is round-by-round sound then applying BCS to this IOP gives a SNARK in the random oracle model [25,26]. Thus

showing the RBR soundness of FRI and OPlonkyallows us to readily show Fiat-Shamir security of these protocols under the BCS transformation in the random oracle model, yielding our results. Thus in what follows, we give a high-level overview of the round-by-round soundness proofs for both FRI and OPlonky.

2.2 Round-by-Round Soundness of FRI

We give a high-level sketch of the round-by-round soundness of FRI in this section; for full details, see [14]. As previously stated, FRI is an interactive oracle proof of proximity for testing whether or not a polynomial specified by a prover is "close to" a particular space of Reed-Solomon codewords. More formally, for finite field $\mathbb{F}$, multiplicative subgroup $L_0 \subset \mathbb{F}^*$ of size $N = 2^n$, and degree bound $d_0 = 2^k$ for $k \in \mathbb{N}$, $\mathsf{RS} := \mathsf{RS}[\mathbb{F}, L_0, d_0] \subset \mathbb{F}^N$ is the set of all polynomials $f : L_0 \to \mathbb{F}$ of degree at most $d_0 - 1$, and the FRI protocol allows for a prover to succinctly prove to a verifier that a function $G_0 : L_0 \to \mathbb{F}$ is within some proximity bound δ of the RS code. That is, if a verifier accepts the interaction, then the verifier is convinced that there exists $f \in \mathsf{RS}$ such that $\Delta(G_0, f) < \delta N$, where Δ is the Hamming distance between G_0 and f (when viewing them as vectors in $\mathbb{F}^N$). We say that such a G_0 is δ-*close* to RS; otherwise, we say that G_0 is δ-*far* from RS (i.e., $\Delta(G_0, f) \geq \delta N$ for all $f \in \mathsf{RS}$).

To achieve succinct verification, the FRI protocol first interactively compresses G_0 during a *folding phase*,[5] which proceeds as follows. First, the prover sends oracle G_0 to the verifier. Next, the verifier samples $x_0 \in \mathbb{F}$ uniformly at random and sends it to the prover. Now the prover defines new oracle $G_1 : L_1 \to \mathbb{F}$ over the new domain $L_1 = (L_0)^2 := \{z^2 : z \in L_0\}$ of size $N/2$, where for any $s \in L_1$, if $s', s'' \in L_0$ are the square roots of s, then we have

$$G_1(s) = (x_0 - s')(s'' - s')^{-1} G_0(s'') + (x_0 - s'')(s' - s'')^{-1} G_0(s'). \tag{1}$$

Given G_1, the prover and verifier now recursively engage in the above folding procedure with the function G_1, where the claim is that G_1 is δ-close to a new Reed-Solomon code $\mathsf{RS}[\mathbb{F}, L_1, d_1]$ for $d_1 = d_0/2$; this recursion continues $\log(d_0) = k$ times which results in prover oracles $G_0, G_1, \ldots, G_{k-1}$ and verifier challenges $x_0, x_1, \ldots, x_{k-1}$.

After the folding phase, the prover and verifier now engage in the *query phase*. During this phase, the prover sends a constant value $G_k \in \mathbb{F}$ to the verifier, and the verifier samples a uniformly random challenge $s_0 \in L_0$ tto check the consistency of all pairs of functions G_{i-1}, G_i for $i \in \{1, \ldots, k\}$ as follows. The verifier first checks consistency of G_0 and G_1 using Eq. (1); in particular, if we set $s_1 = (s_0)^2$ and let t_0 be the other square root of s_1 (i.e., $(t_0)^2 = s_1$ and $t_0 \neq s_0$), the verifier checks that $G_1(s_1)$ is consistent with $G_0(s_0)$ and $G_0(t_0)$ via Eq. (1). This check is then performed for every pair of functions G_{i-1} and G_i via Eq. (1) using challenge x_{i-1} and $G_i(s_i)$, $G_{i-1}(s_{i-1})$, and $G_{i-1}(t_{i-1})$, where $s_i = (s_{i-1})^2$ and $t_{i-1} \neq s_{i-1}$ is the other square root of s_i. The verifier accepts if and only if all of

[5] [4] refers to this as the *commit phase*. We view the term "folding phase" as more appropriate given the nature of the compression.

these checks pass. More generally, the verifier performs the above query phase (in parallel) $\ell \geq 1$ times and outputs accept if and only if all checks pass.

To show RBR soundness of FRI, we first turn to the prior soundness analyses of FRI. Suppose that G_0 is δ-far from $\mathsf{RS}[\mathbb{F}, L_0, d_0]$, then it turns out a malicious prover has two strategies for fooling the verifier: (1) "luck out" in the sense that for $x_0 \xleftarrow{\$} \mathbb{F}$ sent by the verifier, the new function G_1 is δ-close to $\mathsf{RS}[\mathbb{F}, L_1, d_1]$; or (2) send some $G_1' \neq G_1$ that is δ-close to $\mathsf{RS}[\mathbb{F}, L_1, d_1]$. Intuitively, strategy (2) never increases the probability the prover can fool the verifier since even though G_1' is closer to the Reed-Solomon codespace, this improvement is offset by the fact that G_1 and G_1' will differ at many different points. Thus the optimal prover strategy is to simply behave honestly by sending the correct function during every round using Eq. (1), and hoping to "luck out" from the verifier challenge during that round.

FRI Round-by-Round Soundness Overview. We adapt the above intuition for the RBR soundness of FRI. Let P^* be our (possibly malicious) prover. Let ε_1 be the probability that P^* "lucks out" as described above. First, since G_0 is assumed to be δ-far, and moreover G_0 is honestly sent to the verifier, the protocol, begins in a doomed state. Then if the verifier sends x_0 such that P^* "lucks out" and the function G_1 is δ-close, then we say the protocol is no longer in a doomed state. This happens with probability at most ε_1.

Building on this, suppose the partial transcript so far consists of (G_0, x_0) and suppose that this state is doomed; that is, both G_0 and G_1 are δ-far functions. Now the prover P^* may send some function G_1' that may or may not be equal to G_1 (as given in Eq. (1)), and then the verifier responds with challenge x_1. However, as described before, sending $G_1' \neq G_1$ doesn't increase the probability that the prover fools the verifier, and we want the RBR soundness analysis to reflect this as well. Thus we say that the current state of the protocol, given by (G_0, x_0, G_1', x_1) is not doomed if and only if $G_1' = G_1$ and P^* "lucks out" with the function G_2 (again defined via Eq. (1) using x_1 and G_1). In other words, the protocol remains in a doomed state if: (1) $G_1' \neq G_1$; or G_2 is δ-far (i.e., the prover didn't "luck out"). Thus the protocol leaves its doomed state with probability at most ε_1. This analysis generalizes to all rounds of the folding phase: given any partial transcript $(G_0, x_0, G_1', x_1, \ldots, G_{i-1}', x_{i-1})$ that is in a doomed state, if P^* sends function G_i' and the verifier sends challenge x_i, then the protocol is no longer doomed if and only if (1) the prover "lucked out" and G_{i+1} is δ-close; and (2) *all* $G_j' = G_j$ for $j \in \{1, \ldots, i-1\}$. And again, the protocol is no longer doomed with probability at most ε_1.

To complete the analysis, we now consider the final round of the protocol, which consists of the query phase. Suppose that the partial transcript for this round is given by $(G_0, x_0, G_1', x_1, \ldots, G_{k-1}', x_{k-1})$ and suppose the protocol is in a doomed state. At this point, P^*'s hands are tied: it must send a constant $G_k \in \mathbb{F}$ to the verifier, and the verifier then uniformly samples $s_0^{(1)}, \ldots, s_0^{(\ell)} \in L_0$ and performs its checks. Thus, the only way the protocol can leave the doomed state is if *all* of these checks pass; in particular, if a single check fails then the protocol

remains doomed (and, in fact, the verifier rejects). Let ε_2 denote the probability that a single verifier check passes; that is, a single chain of checks depending on $s_0^{(1)}$ passes (i.e., computing the squares and square roots at every level, and checking consistency across all levels with this check). Then the probability P^* can leave the doomed state is exactly ε_2; extending this to ℓ checks (which are performed uniformly and independently at random) gives us that the protocol leaves the doomed state with probability at most ε_2^ℓ. Considering the folding and query phases, the discussion above shows that the FRI protocol has RBR soundness error $\varepsilon_{\mathsf{rbr}}^{\mathsf{FRI}} = \max\{\varepsilon_1, \varepsilon_2^\ell\}$.

Batched FRI Round-by-Round Soundness Overview. Extending the above analysis to Batched FRI is straightforward. Briefly, Batched FRI invokes FRI on a random linear combination of t functions $f_1, \ldots, f_t : L_0 \to \mathbb{F}$. In more detail, first the prover sends oracles $f_1, \ldots, f_t$ to the verifier, then the verifier responds with random challenges $\alpha_1, \ldots, \alpha_t$. The prover and verifier then engage in the FRI protocol using function $G_0 = \sum_i \alpha_i f_i$.[6] Finally, Batched FRI modifies the query phase of FRI to also check consistency between f_i and G_0 exactly via the equation $G_0 = \sum_i \alpha_i f_i$. Key to Batched FRI is that if all f_i are δ-close to $\mathsf{RS}[\mathbb{F}, L_0, d_0]$, then G_0 is also δ-close, and if even one f_j is δ-far, then with high probability G_0 is also δ-far.

The RBR soundness analysis of Batched FRI proceeds as follows. Let P^* again denote our (possibly malicious) prover. The protocol begins in a doomed state; namely, there exists at least one f_j that is δ-far from $\mathsf{RS}[\mathbb{F}, L_0, d_0]$. Then P^* honestly sends $f_1, \ldots, f_t$ to the verifier,[7] and the verifier responds with $\alpha_1, \ldots, \alpha_t \in \mathbb{F}$ sampled uniformly and independently at random. Let ε_t be the probability that G_0 is δ-close given that there exists at least one f_j that is δ-far, where the probability is taken over $\alpha_1, \ldots, \alpha_t$. Then we say the protocol is no longer in a doomed state if and only if G_0 is δ-close; thus during this round, P^* can leave the doomed state with probability at most ε_t. Now suppose that $(f_0, \ldots, f_t, \alpha_1, \ldots, \alpha_t)$ is the current protocol state and that this state is doomed. The prover and verifier now engage in FRI using some function G_0' constructed by P^* as input. The observation here is that we can now invoke the RBR soundness analysis of FRI directly, with the following slight change for the first round of FRI. Suppose P^* sends G_0' to the verifier and the verifier responds with x_0. Then the protocol is no longer in a doomed state if and only if $G_0' = G_0$ and G_1 is δ-close, where G_1 is defined via Eq. (1) with respect to the correct function G_0. In particular, the intuition behind the prover's strategy remains the same: if P^* sends some other $G_0' \neq G_0$, then the verifier is more likely to detect this change when checking consistency of G_0' and $f_1, \ldots, f_t$, so P^* can only leaved the doomed state of the protocol if it behaves honestly and "lucks out" with verifier challenge x_0. Finally, we remark

[6] In practice to save on communication, only a single α is sent and the linear combination is computed with $\alpha_i = \alpha^{i-1}$, at the cost of an increased soundness error; see [14] for details.

[7] This is necessary, if a malicious prover is allowed to send dishonest $f_1^*, \ldots, f_t^*$ such that all are δ-close, then the protocol reduces to the honest prover analysis.

that the final round (i.e., the query phase) of Batched FRI with the additional checks between $f_1, \ldots, f_t$ and G'_0 has the same RBR soundness error ε_2 as with FRI. Thus the RBR soundness error of Batched FRI is $\varepsilon_{\mathsf{rbr}}^{\mathsf{bFRI}} = \max\{\varepsilon_t, \varepsilon_1, \varepsilon_2^\ell\}$, where ℓ is the number of times the query phase is repeated.

Instantiating ε_1, ε_2, and ε_3. For the query phase, the best one can hope for is $\varepsilon_2 = (1 - \delta)$ [4,6,11,70]; for the folding phase, there is a long line of work done towards improving the bounds on ε_1 [4,6,11]. In our work, we utilize the best known provable bounds on ε_1 given by Ben-Sasson et al. [6], and note that any improvements for ε_1 directly improve the round-by-round soundness error of FRI. In particular, we have $\varepsilon_1 = O(2^{2n}/(\rho \cdot |\mathbb{F}|))$, where $\rho = d_0/|L_0|$ and $|L_0| = 2^n$. This yields our stated round-by-round soundness error in Theorem 1. Finally, [6] also show that $\varepsilon_t = \varepsilon_1$ for Batched FRI, which gives us Batched FRI round-by-round soundness error $\varepsilon_{\mathsf{rbr}}^{\mathsf{bFRI}} = \max\{\varepsilon_1, \varepsilon_2^\ell\}$, yielding our stated round-by-round soundness error in Theorem 2. See [14] for a complete discussion and proof of the round-by-round soundness of FRI and Batched FRI.

FRI Round-by-Round Knowledge Overview. Recall that a protocol has RBR knowledge error ε_{k} if for any "doomed" state of the protocol, if every message the prover can send will put the protocol in a non-"doomed" state with probability at least ε_{k} over the verifier randomness, then an extractor can efficiently recover a witness (with probability 1) when given the current protocol state and the prover's next message. In the context of FRI, RBR knowledge soundness means we can extract a δ-close function G, and for Batched FRI we can extract t functions $f_1, \ldots, f_t$ that are all δ-close. For both FRI and Batched FRI, it turns out we obtain RBR knowledge soundness more or less for free. Recall that both protocols have RBR soundness error $\max\{\varepsilon_1, \varepsilon_2^\ell\}$ from our discussion above. Then we claim that these protocols both have RBR knowledge error exactly $\varepsilon_{\mathsf{k}} = \max\{\varepsilon_1, \varepsilon_2^\ell\}$.

We give an efficient extractor for the RBR knowledge soundness of FRI. First consider any intermediate round i of the folding phase of FRI (the analysis for Batched FRI is identical). Then the current protocol state is doomed and is given by the transcript $(G_0, x_0, G'_1, x_1, \ldots, G'_{i-1}, x_{i-1})$. Suppose that for any function G'_i sent by the prover, for $x_i \xleftarrow{\$} \mathbb{F}$ sampled by the verifier, the protocol state $(G_0, x_0, G'_1, x_1, \ldots, G'_i, x_i)$ is not doomed with probability at least ε_{k}. In particular, this happens with probability at least $\varepsilon_1 = O(2^{2n}/(\rho|\mathbb{F}|))$. Then our extractor, given $(G_0, x_0, G'_1, x_1, \ldots, G'_i)$ simply reads and outputs the oracle G_0. For the query phase, the analysis is identical: let the current protocol state be doomed for transcript $(G_0, x_0, G'_1, x_1, \ldots, G'_{k-1}, x_{k-1})$. Suppose for every $G_k \in \mathbb{F}$ sent by the prover and verifier challenges $s_{0,1}, \ldots, s_{0,\ell} \xleftarrow{\$} L_0$, the protocol state $(G_0, x_0, G'_1, x_1, \ldots, G_k, (s_{0,j})_{j \le \ell})$ is not doomed with probability at least ε_{k}. In particular, this happens with probability at least $\varepsilon_2^\ell = (1-\delta)^\ell$. Then our extractor again simply reads and outputs oracle G_0.

Now why should we expect G_0 to be a δ-close function? It turns out that by the choices of ε_1 and ε_2, if *all* prover messages can leave the doomed state

with the above probabilities, it *unconditionally* implies that G_0 must be δ-close in both cases, a result shown by [6]. First, for any round of the folding, the function G'_i can leave the doomed set if and only if $G'_i = G_i$ (i.e., it is computed as an honest prover would compute it) and G_{i+1} is δ-close. If G_{i+1} is δ-close with probability greater than ε_1 over the verifier randomness, then it unconditionally implies that G_i must have been δ-close as well [6]. This then recursively applies to G_{i-1}, and so on, finally yielding that G_0 must have been δ-close as well. [6] show that a similar result must hold for the query phase: if all verifier checks pass with probability at least ε_2^ℓ during the query phase for any $G_k \in \mathbb{F}$ sent by the prover, then G_0 must be δ-close as well. Thus the RBR knowledge error of FRI is identical to the RBR soundness error. Finally, the above analysis proceeds identically for Batched FRI as well; i.e., if during any round of folding or batching phase the prover can leave with probability at least ε_1, then it unconditionally implies that $f_1, \ldots, f_t$ must be δ-close functions. The Batched FRI query phase is analogous.

2.3 Correlated IOPs and Round-by-Round Knowledge Soundness

To conduct our security analysis beyond FRI, we formulate an abstract type of IOP which we call a *δ-correlated IOP*. This is a notion related and inspired by that of *Reed-Solomon Encoded IOPs* [8,26] (see [14] for further comparison). In a nutshell, when $\delta = 0$, a 0-correlated IOP is an IOP where:

- The verifier has access to an oracle $\mathcal{O}$ that, given any number of maps $f_1, \ldots, f_k : D \to \mathbb{F}$, determines whether each of the f_i is the evaluation map of a polynomial of degree at most d, for any $d < |D|$. Here D is a subset of $\mathbb{F}$, called *evaluation domain*.
  In other words, $\mathcal{O}$ determines whether the maps (or words) f_i belong to the Reed-Solomon code $\mathsf{RS}[\mathbb{F}, D, d+1]$.
- During the interactive phase, the prover sends oracle access to some maps $g_1, \ldots, g_m : D \to \mathbb{F}$ (across several rounds of interaction).
- In the last round of interaction, the verifier sends a field element $\mathfrak{z} \in \mathbb{K} \setminus D$ to the prover, and the prover replies with values

$$\left\{g_i(k_{i,j}\mathfrak{z}) \mid i \in [m], j \in [n_i]\right\} \tag{2}$$

  where $k_{i,j}$ are some pre-defined field elements and $n_i \geq 1$ are predefined positive integers. Here $\mathbb{K}$ is either $\mathbb{F}$ or a field extension of $\mathbb{F}$. Importantly, each map g_i appears at least once in the list Eq. (2).
- To decide whether to reject or accept the prover's proof, the verifier:
  - **Check 1.** Asserts that the values $\left\{g_i(k_{i,j}\mathfrak{z}) \mid i \in [m], j \in [n_i]\right\}$ satisfy certain polynomial equations.
  - **Check 2.** Uses its oracle $\mathcal{O}$ to check that the following maps belong to $\mathsf{RS}[\mathbb{F}, D, d]$:

$$\mathsf{quotients} := \left\{(g_i(X) - g_i(k_{i,j}\mathfrak{z}))/(X - \mathfrak{z}) \mid i \in [m], j \in [n_i]\right\}. \tag{3}$$

When $\delta > 0$, a δ-correlated IOP has the exact same form as above, except that now $\mathcal{O}$ is an oracle for checking *δ-correlated agreement* in $\mathsf{RS}[\mathbb{F}, D, d+1]$ for any $d < |D|$. A sequence of maps $g_1, \ldots, g_m : D \to \mathbb{F}$ has δ-correlated agreement if there exists a subset $S \subseteq D$ and polynomials $q_1, \ldots, q_m$ of degree $\leq d$ such that g_i coincides with q_i on S, for all $i \in [m]$, and $|S| \geq (1-\delta)|D|$.

These type of IOP's are interesting to us because several modern IOP's can be understood as being built on top of a 0-correlated or δ-correlated IOP for $\delta > 0$, e.g., all Plonk-like protocols that use FRI instead of KZG [34,44,60], ethSTARK (or DEEP-ALI) [10,65], RISC Zero [68], etc.

We prove the following results about δ-correlated IOPs:

- **Result 1.** If a 0-correlated IOP Π_0 has round-by-round (RBR) soundness or knowledge ε, then replacing $\delta = 0$ by a larger $\delta > 0$ results in a δ-correlated IOP with RBR soundness or knowledge $\ell\varepsilon$, where ℓ is certain constant related to list decodability of Reed-Solomon (RS) codes. Namely, ℓ is the maximum number of distinct RS codewords that can be δ-close to any given word. Here, by "replacing $\delta = 0$ by a larger $\delta > 0$" we refer to the δ-correlated IOP that results from taking Π_0 and replacing the verifier's oracle for checking membership to $\mathsf{RS}[\mathbb{F}, D, d+1]$ (so, checking 0-correlated agreement) by an oracle that checks for δ-correlated agreement in $\mathsf{RS}[\mathbb{F}, D, d+1]$.
- **Result 2.** Given a δ-correlated IOP Π with RBR soundness or knowledge ε, and given a IOP or IOP of Proximity Π_{CA} for checking δ-correlated agreement, we can construct a new IOP (in the standard sense, i.e., an "uncorrelated IOP"), denoted by Π_{compiled}, by replacing the oracle $\mathcal{O}$ with the protocol Π_{CA}. We show that if Π_{CA} has RBR soundness $\varepsilon_{\mathsf{CA}}$, then Π_{compiled} has RBR (knowledge) soundness $\max\{\varepsilon, \varepsilon_{\mathsf{CA}}\}$. Notice that, for RBR knowledge soundness, we don't need Π_{CA} to have RBR knowledge soundness. It suffices for Π to have RBR knowledge soundness, and for Π_{CA} to be RBR sound.

First, we explain how these results can be applied to existing protocols, then give a high-level overview of their proofs.

Using the Above Results. Given these results, one strategy for proving that an IOP Π has RBR (knowledge) soundness is to first try to formulate the IOP as a δ-correlated IOP that has been compiled with the method described above, then prove that the corresponding 0-correlated IOP has RBR (knowledge) soundness. Once this is done, our results provide RBR (knowledge) soundness error bounds for the initial IOP Π. Figure 2 gives an overview of this methodology.

The latter task can be a significant simplification in comparison to analyzing the initial IOP Π directly. This is because when $\delta = 0$, the verifier in Π has an oracle for checking that the maps from the verifier's Check 2 are low-degree polynomials. This effectively forces the prover to send (oracles to) low-degree polynomials throughout the interaction and to provide correct openings in its last message. As a consequence, and roughly speaking, our methods allows to study the IOP as if it was a *Polynomial* IOP (PIOP), with the Batched FRI protocol acting as a Polynomial Commitment Scheme (PCS) used to compile the PIOP into an interactive argument. However, note that FRI cannot be used

as a PCS (unless δ lies in the unique decoding radius) since it only guarantees δ-closeness to low-degree polynomials.

Later, we show how these methods can be used on "Plonk-like" protocols, and briefly discuss how to use them on other protocols such as ethSTARK and RISC Zero.

Proof Sketch of Result 1. Let $\delta > 0$, let Π_δ be a δ-correlated IOP, and let Π_0 be the same IOP except that the verifier has access to an oracle for 0-correlated agreement instead of δ-correlated agreement (equivalently, it has an oracle for checking membership to $\mathsf{RS}[\mathbb{F}, D, d'+1]$ for any $d' < |D|$). Suppose Π_0 is RBR (knowledge) sound with error ε. We first focus on RBR soundness and discuss RBR knowledge soundness later. Let τ be a partial transcript produced during some rounds of interaction between the prover and the verifier from Π_δ. For ease of presentation, assume the prover sends maps to the verifier, as opposed to sending oracle access to these maps. Let $g_1, \ldots, g_k$ be all prover's maps in τ and write $\tau = \tau(g_1, \ldots, g_k)$ to denote that τ contains such maps. Let $\tau' = \tau'(g'_1, \ldots, g'_k)$ be another partial transcript. We informally say τ' is a *low-degree partial transcript* if all of the maps $g'_1, \ldots, g'_k$ are codewords from $\mathsf{RS}[\mathbb{F}, D, d+1]$. We also say τ' has *δ-correlated agreement* with τ if there is $S \subseteq D$ such that g_i coincides with g'_i on S for all $i \in [k]$ and $|S| \geq (1-\delta)|D|$. Then we say that τ is "doomed" in Π_δ if and only if one of the following holds:

- All low-degree partial transcripts τ' that are δ-correlated with τ are doomed in Π_0.
- τ is a complete transcript and Check 2 of the verifier fails, i.e., the maps $\mathsf{quotients}$ from Eq. (3) do not have δ-correlated agreement in $\mathsf{RS}[\mathbb{F}, D, d+1]$.

This defines the doomed states for Π_δ, i.e., the doomed states are those where the partial transcript so far is doomed.

Now it remains to be shown that Π_δ has RBR (knowledge) soundness error $\varepsilon/(2\sqrt{\rho}\eta)$ with respect to these doomed states. In what follows, we say that a partial transcript is *doomed in* Π_δ or *doomed in* Π_0 depending on whether it is doomed with respect to the doomed states of Π_δ or Π_0. By a i-round partial transcript we mean a partial transcript where both prover and verifier have sent i messages each.

Let τ be a i-partial transcript after that is doomed in Π_δ. By definition, all low-degree partial transcripts that are δ-correlated with τ are doomed in Π_0. Let m be a prover's message for round $i+1$. We want to show that the probability that (τ, m, c) is not doomed in Π_δ is at most $\varepsilon/(2\sqrt{\rho}\eta)$, where the probability is taken over the verifier's $(i+1)$-th message c. Assume (τ, m, c) is not doomed in Π_δ for some c. Then, by definition of the doomed states of Π_δ, there is a low-degre partial transcript ν that is δ-correlated with (τ, m, c) and that is not doomed in Π_0. This transcript must have the form $\nu = (\tau', m', c)$, where τ' is a i-round low-degree partial transcript that is δ-correlated with τ. In particular, τ' is doomed in Π_0.

Since Π_0 is RBR sound with error ε, the fraction of challenges c such that τ' is doomed in Π_0 but (τ', m', c) is not is at most ε. Thus the fraction of challenges

c such that τ is doomed in Π_δ but (τ, m, c) is not doomed in Π_δ is at most $\ell\varepsilon$, where ℓ is the number of i-round low-degree partial transcripts τ' that are δ-correlated with τ. Using a lemma from [65], we can bound ℓ by $1/(2\sqrt{\rho}\eta)$.

It remains to argue that doomed complete transcripts are rejected by the verifier. Let $\tau = \tau(g_1, \ldots, g_m)$ be a doomed complete partial transcript and let $\mathsf{quotients}$ be as in Eq. (3). If the maps $\mathsf{quotients}$ do not have δ-correlated agreement in $\mathsf{RS}[\mathbb{F}, D, d]$, then the verifier rejects and we are done. Hence assume they do have δ-correlated agreement. Thus, for each $i \in [m]$ and $j \in [n_i]$ we have that $(g_i(X) - g_i(k_{i,j}\vartheta))/(X - k_{i,j}\vartheta)$ agrees with a polynomial $q_{i,j}(X)$ on a set S (this set is the same for all i, j). In other words, $g_i(X)$ agrees with the polynomial $u_{i,j}(X) := q_{i,j}(X)(X - k_{i,j}\vartheta) + g_i(k_{i,j}\vartheta)$ on S. Moreover, both g_i and $u_{i,j}$ take the same value on $X = k_{i,j}\vartheta$, i.e., $g_i(k_{i,j}\vartheta) = u_{i,j}(k_{i,j}\vartheta)$. Additionally, we have $|S| > (1-\delta)|D|$, and by how δ is chosen, $(1-\delta)|D| \geq d+1$. This makes $u_{i,j}(X)$ the same for all $j \in [n_i]$; thus, we denote any $u_{i,j}(X)$ simply as $u_i(X)$.

We have seen so far that $g_i(X)$ agrees with the polynomial $u_i(X)$ on S, for all $i \in [m]$, and that $g_i(k_{i,j}\vartheta) = u_i(k_{i,j}\vartheta)$, for all i, j. Thus $\tau' = \tau(u_1, \ldots, u_m)$ is a low-degree partial transcript that is δ-correlated with τ. Since τ is a doomed transcript and $\mathsf{quotients}$ have δ-correlated agreement in $\mathsf{RS}[\mathbb{F}, D, d]$, we must have that τ' is doomed in Π_0. Note that τ' is a complete transcript, and so Π_0's verifier rejects it. Clearly, τ' passes the 0-correlated agreement check of Π_0's verifier. Hence the first check of the verifier fails, i.e., the values $\{u_i(k_{i,j}\vartheta) \mid i \in [m], j \in [n_i]\}$ do not satisfy some required polynomial identities. However, these values coincide with $\{g_i(k_{i,j}\vartheta) \mid i \in [m], j \in [n_i]\}$, and so the verifier of Π_δ rejects τ for the same reason: the values do not satisfy the appropriate polynomial equations. This proves that Π_δ has the claimed RBR soundness error.

The proof that Π_δ has RBR knowledge soundness uses similar ideas. Precisely, suppose τ is a i-round partial transcript that is doomed in Π_δ. Let m be a prover's $(i+1)$-th round message and assume the probability (over the verifier's $(i+1)$-th challenge c) that (τ, m, c) is not doomed is larger than $\varepsilon/(2\sqrt{\rho}\eta)$. Since, as we argued, there are at most $1/(2\sqrt{\rho}\eta)$ i-round low-degree partial transcripts τ' that are δ-correlated with τ, there must exist at least one such transcript τ' that is doomed in Π_0 such that (τ', m', c) is not doomed in Π_0 with probability larger than ε. Then we can use the RBR knowledge soundness of Π_0 to extract a valid witness from τ'. We can build an extractor that, given τ, computes all low-degree partial transcripts τ' that are δ-correlated with τ. This can be done in polynomial time using a method from [65]. Then for each such τ', the new extractor uses the extractor of Π_0 on τ', until a valid witness is found.

Proof Sketch of Result 2. The second general result stated above can be proved as follows: define a partial transcript τ for Π_{compiled} to be doomed if one of the following hold:

1. τ is a partial transcript for Π and τ is a doomed state in Π.
2. τ is a partial transcript of the form $\tau = (\tau_1, \tau_2)$, where τ_1 is a complete transcript of Π, and τ_2 is a (possibly empty) partial transcript corresponding to some rounds of Π_{CA}, and either

(a) τ_2 is a doomed state in Π_{CA}, or
(b) the verifier V_Π from Π would reject τ_1 due to Check 1 not passing.

We then prove that Π_{compiled} is RBR (knowledge) sound with respect to these doomed states and with error $\max\{\varepsilon, \varepsilon_{\mathsf{CA}}\}$. As before, we discuss first RBR soundness then RBR knowledge.

The key observation is that if τ is a doomed partial transcript of Type 1 above, then it remains doomed in the next round except with probability ε due to the RBR soundness of Π. A similar argument can be used for a partial transcript of Type 2 of the form $\tau = (\tau_1, \tau_2)$, with $\tau_2 \neq \emptyset$. The most noteworthy case is when τ is of Type 2 and of the form $\tau = (\tau_1, \emptyset)$, i.e., the case when τ is exactly a complete transcript for Π. In this case, since τ is doomed, the verifier V_Π in Π would reject τ. Hence τ fails either Check 1 or Check 2 of V_Π. In the first case, the probability of leaving the doomed state in Π_{compiled} is 0 since any partial transcript $\tau' = (\tau'_1, \tau'_2)$ of Type 2 such that τ'_1 fails Check 1 of V_Π is doomed by definition. In the latter case, Π_{CA} is executed with input a set of words that do not have δ-correlated agreement. As such, Π_{CA} starts off in a doomed state and so the probability that the state is not doomed in the next round of interaction is at most $\varepsilon_{\mathsf{CA}}$. This shows that Π_{compiled} is RBR sound with error $\max\{\varepsilon, \varepsilon_{\mathsf{CA}}\}$.

For RBR knowledge soundness, we make the following observations. First, we define doomed states for Π_{compiled} as before, using the doomed states given by the RBR knowledge (as opposed to RBR soundness) for Π, and the doomed states given by the RBR soundness for Π_{CA}. Now, let τ be a doomed partial transcript for Π_{compiled}. Assume the probability θ that τ stops being doomed at the next round is larger than $\max\{\varepsilon_k, \varepsilon_{\mathsf{CA}}\}$, where ε_k is the RBR knowledge error of Π. Then, if τ is of Type 1, we can use the extractor given by the RBR knowledge of Π to obtain a valid witness from τ. On the other hand, if $\theta > \max\{\varepsilon_k, \varepsilon_{\mathsf{CA}}\}$ then $\tau = (\tau_1, \tau_2)$ cannot be of Type 2 because:

- If τ_2 is a doomed state in Π_{CA}, then by definition of RBR soundness, the probability that τ_2 is not doomed in the next round of Π_{CA} is at most $\varepsilon_{\mathsf{CA}}$.
- If τ_1 would be rejected by Π's verifier due to Check 1 failing, then the partial transcript will be doomed at the next round because of the same reason, and so in this case τ has probability 0 of not being doomed in the next round.

In other words, doomed partial transcripts of Type 2 are always doomed at the next round, except with probability at most $\max\{\varepsilon_k, \varepsilon_{\mathsf{CA}}\}$. Thus, we do not need to describe an extractor for this type of partial transcripts.

Remark 5. This approach yields better RBR soundness bounds than some prior known methods. For example, in [44] the authors introduce RedShift, a Plonk-like IOP. The authors obtain a RBR knowledge error (modulo FRI) for RedShift which has a factor of the form, roughly, ℓ^m, where ℓ is the aforementioned "maximum list decoding set size", and m is the number of oracles sent by the prover during the interactive phase. For RedShift, m is set to 6, but similar (though not fully identical) protocols such as Plonky2 [60] use $m \geq 130$. On the contrary, as we mention later in this paper, with our method the factor ℓ^m would be reduced

to ℓ. We remark again that [44] also does not obtain FS security of their protocol, as that work does not analyze the FS security of FRI.

In Sect. 2.5 we also point out that, when applied to the ethSTARK protocol, our approach leads to better knowledge soundness error than the one in [65] (this improvement was already demonstrated in [39]).

2.4 Round-by-Round Knowledge of Plonk-Like Protocols

We generalize and abstract Plonk-like protocols as a correlated IOP, which we call $\mathsf{OPlonky}$, where again by "Plonk-like" we specifically mean the interactive oracle proof abstractions underlying the protocols related to and built upon the (IOP underlying the) Plonk SNARK. The abstraction is inspired mostly by Plonky2 [60], which we believe to be one of the most general Plonk-like IOP's currently published.

The protocol $\mathsf{OPlonky}$ is an IOP for a Plonk-like relation $\mathbf{R}_{\mathbf{OPlonky}}$ (related to [32]), which generalizes arithmetic circuit satisfiability and seamlessly supports custom gates. Simplifying greatly, an instance of $\mathbf{R}_{\mathbf{OPlonky}}$ is characterized by some multivariate polynomial equations $P_1 = 0, \ldots, P_k = 0$, two integers n, r representing the dimensions of a matrix (usually called *execution trace*), and a permutation $\sigma : [n] \times [r] \to [n] \times [r]$. An input and witness pair $(\mathbb{x}, \mathbb{w})$ satisfies such an instance if $\mathbb{w}$ is a $n \times r$ matrix of field elements, $\mathbb{x}$ is a vector of field elements, and

- The values in each row $\mathbb{w}_i$ of $\mathbb{w}$ satisfy $P_1(\mathbb{w}_i) = \ldots = P_k(\mathbb{w}_i) = 0$.
- Certain pre-specified cells in $\mathbb{w}$ have the values $\mathbb{x}$.
- The entries in $\mathbb{w}$ satisfy the *copy constraints* induced by σ. More precisely, $\mathbb{w}_{(i,j)} = \mathbb{w}_{\sigma(i,j)}$ for all $i, j \in [n] \times [r]$.

The IOP $\mathsf{OPlonky}$ proceeds in the following 4-round process. For the sake of presentation, we provide a greatly simplified exposition.

1. **Round 1.** The prover sends r polynomials $a_1(X), \ldots, a_r(X)$ of degree $< n$ to the verifier as oracles. Each of these polynomials is the result of interpolating the columns of $\mathbb{w}$ over a multiplicative subgroup H of $\mathbb{F}$ or order n. The verifier then replies with some random challenges.
2. **Round 2.** The prover uses the verifier randomness from the prior round, to construct and send oracle access to so-called *permutation polynomials* $\pi_1(X), \ldots, \pi_s(X)$ of degree less than n. These polynomials will later be used to (again roughly) check that the copy constraints are satisfied. The verifier responds with a random challenge α.
3. **Round 3.** At this point, the goal of the prover is to convince the verifier that the polynomials $Q_j := P_j(a_1(X), \ldots, a_r(X))$ and certain polynomials of the form $\delta_i(X) := R_i(\pi_1(X), \ldots, \pi_s(X))$ vanish on H, where the R_i is some multivariate polynomial. To this end, the prover *batches* these constraints together by computing

$$d(X) = Q_1(X) + \alpha Q_2(X) + \ldots + \alpha^{k-1} Q_k(X) + \alpha^k \delta_1(X) + \ldots + \alpha^{k+s-1} \delta_s(X) \quad (4)$$

and proving that $d(X)$ vanishes in H. To do so, the prover sends the verifier oracle access to the polynomial $q(X) := d(X)/\mathsf{Z}_H(X)$, where $\mathsf{Z}_H(X)$ is the vanishing polynomial of H. The verifier replies with a random field element $\mathfrak{z}$.

4. **Round 4.** The prover replies with the values $(a_i(\mathfrak{z}))_i, (\pi_j(\mathfrak{z}))_j$ and $q(\mathfrak{z})$.
5. **Verification phase.** The verifier performs two assertions. It accepts the proof if and only if both of them return true.
   - Assert whether $q(\mathfrak{z})\mathsf{Z}_H(\mathfrak{z}) = d(\mathfrak{z})$, where the value $d(\mathfrak{z})$ is obtained by replacing X by $\mathfrak{z}$ in Eq. (4), and querying the oracles to $a_i(X), \pi_j(X)$ for all $i \in [r]$ and $j \in [s]$.
   - Use an oracle to assert whether the following set of words has δ-correlated agreement in some Reed-Solomon code:

$$\left\{ \frac{q(X) - q(\mathfrak{z})}{X - \mathfrak{z}}, \left(\frac{a_i(X) - q(\mathfrak{z})}{X - \mathfrak{z}} \right)_i, \left(\frac{\pi_j(X) - q(\mathfrak{z})}{X - \mathfrak{z}} \right)_j \right\}.$$

It is apparent from the description above that $\mathsf{OPlonky}$ is indeed a δ-correlated IOP.

When compiled with the Batched FRI protocol, $\mathsf{OPlonky}_{\mathsf{compiled}}$ becomes almost identical to Plonky2's IOP [60] with some similarities to Redshift [44]. Alternatively, $\mathsf{OPlonky}$ could also be compiled somehow with the KZG commitment scheme (which, in a sense, can act as a protocol for 0-correlated agreement). This would yield generalized versions of the original Plonk protocol and its variations (e.g., TurboPlonk). We leave this as future work.

Round-by-Round Soundness of $\mathsf{OPlonky}$. With the above observations in mind, we then go on to show that $\mathsf{OPlonky}$ with $\delta = 0$ has RBR soundness and knowledge. We now provide an intuitive idea of the proof, focusing on RBR soundness. To do so, we use the simplified description of $\mathsf{OPlonky}$ provided above. As such, our analysis and resulting error bounds are also simplified.

We let $\mathsf{OPlonky}^{\mathcal{O}}$ denote the $\mathsf{OPlonky}$ protocol where the verifier has oracle access to 0-correlated agreement oracle $\mathcal{O}$. To prove that $\mathsf{OPlonky}^{\mathcal{O}}$ has RBR (knowledge) soundness, we need to define a set of "doomed states" the protocol can be in. As a general rule, we will always set the state to "doomed" if the prover has sent the verifier an oracle to a map that is not a polynomial of appropriate degree. As argued in Sect. 2.3, in this scenario it is impossible for a malicious prover to "recover" and eventually convince the verifier, since the verifier will detect the dishonesty when using $\mathcal{O}$ in its Check 2. Moreover, by similar reasons, we can also assume that the prover provides correct openings as its last message.

Next we describe the rest of scenarios in which we set the state to "doomed" and analyze the probabilities of "recovering", i.e., of not being in a doomed set in the next round. We proceed in a round-by-round fashion.

- Given an input $\mathbb{x}$ for the relation $\mathbf{R}_{\mathbf{ROPlonky}}$, if $\mathbb{x}$ is not in the language $\mathcal{L}_{\mathbf{R}_{\mathbf{ROPlonky}}}$ induced by $\mathbf{R}_{\mathbf{ROPlonky}}$, we set the state to "doomed".
- Now assume that at the end of round 1, it is not possible for the prover to compute polynomials $\pi_1(X), \ldots, \pi_s(X)$ of degree $< n$ such that all the polynomials $\delta_i(X)$ vanish on H. Then we set the state to "doomed". We see that if

the state was doomed before round 1, then the chances of receiving verifier randomness such that the state is not doomed at the end of round 1 are, roughly, $\mathsf{rn}/|\mathbb{F}|$. This probability comes from the soundness of permutation checking procedure used in Plonk and many other protocols.
- Now suppose that at the end of round 2, the polynomial $d(X)$ does not vanish on $\mathsf{Z}_H(X)$. Then we set the state to "doomed". In this case, the probability of starting round 2 in a doomed state and finishing it in a non-doomed state is at most, roughly, $(k+s)/|\mathbb{F}|$. This is deduced by taking an arbitrary $x \in H$ and looking at the equality $d(x) = 0$ as a polynomial equation on α. This equation either has degree $\approx k+s$ (on α), or it is identically zero. However, we see that if round 2 started in a doomed state, then $R(x) = 0$ is not identically zero for at least one $x \in H$. Hence, there are at most $\approx k+s$ distinct values of α such that $d(x) = 0$ for all $x \in H$.
- Finally, suppose that at the end of round 3, one has $q(\mathfrak{z})\mathsf{Z}_H(\mathfrak{z}) \neq d(\mathfrak{z})$. Then we set the state to "doomed". In this case, the probability of ending round 3 in a non-doomed state if the state was previously doomed is at most, roughly, $\max_j\{\deg P_j\} \cdot n/|\mathbb{F}|$.[8] This is because either $q(X)Z_H(X) - d(X)$ is the zero polynomial (as it should be), or it is a polynomial of degree $\max_j\{\deg P_j\}n$ and $\mathfrak{z}$ is a root of it. We then see that if the protocol is in a doomed state when round 3 starts, then $q(X)Z_H(X) - d(X)$ is a nonzero polynomial. Notice as well that if the protocol ends in a doomed state as per our definition, then the verifier rejects.

The above argument, at a high-level, establishes the round-by-round security of the 0-correlated hIOP $\mathsf{OPlonky}^{\mathcal{O}}$; complete details are given in the full version of our work [14].

Round-by-Round Knowledge of RISC Zero. RISC Zero [68] is similar to the Plonky2 protocol. More precisely, and modulo technicalities, it can be thought of as being built on top of $\mathsf{OPlonky}$ with the addition that RISC Zero implements a lookup argument [33] in the same round as the permutation check is performed. We believe that similar methods as the ones presented in the previous section can be used to establish the RBR knowledge soundness of RISC Zero, and thus, the knowledge soundness of the Fiat-Shamir transformed version of RISC Zero. Since RISC Zero's whitepaper is in draft form at the moment of writing, we leave formally proving this claim as an open task.

2.5 Round-by-Round Knowledge of EthSTARK

We begin by discussing the ethSTARK protocol Π_{ethSTARK} [65], which is a close variation of the DEEP-ALI protocol [10]. We briefly provide a rough overview of the protocol; see [65] for complete details.

[8] This is not entirely accurate; for precise bounds, see Theorem 8.

Description of the Protocol. In Π_{ethSTARK}, the honest prover first sends oracle access to a list of degree $\leq d$ polynomials $f_1, \ldots, f_m$ that interpolate the columns of a so-called Algebraic Intermediate Representation (AIR) instance over a multiplicative subgroup H of a field $\mathbb{F}$ (simply put, these polynomials encode the witness). Supposedly, these polynomials are such that certain maps of the form

$$[Q_i(X, f_1(g_{i,1}X), \ldots, f_m(g_{i,m}X))]/Z_{H_i}(X), \quad i \in I, \tag{5}$$

are low-degree polynomials. Here, each $Q_i(X, Y_1, \ldots, Y_m)$ is a pre-specified $(m+1)$-variate polynomial; the $g_{i,j}$'s are field elements; $Z_{H_i}(X)$ is the vanishing polynomial of a subgroup H_i of H; and I is a list of indices.

The verifier replies with $2|I|$ random elements $r_1, r'_1, \ldots, r_{|I|}, r'_{|I|}$ from a field extension $\mathbb{K}$ of $\mathbb{F}$. As its second message, the honest prover sends oracle access to low-degree polynomials $q_1(X), \ldots, q_k(X)$ such that

$$\sum_{i \in I} (r_i + r'_i X^{c_i})[Q_i(X, f_1(g_{i,1}X), \ldots, f_m(g_{i,m}X))]/Z_{H_i}(X) = \sum_{j=1}^{k} X^{j-1} q_j(X^k), \tag{6}$$

where the c_i's are pre-agreed positive integers such that that each summand on the left-hand side of Eq. (6) has the same degree, and k is a conveniently pre-agreed positive integer. The reason why the prover sends k polynomials $q_1(X), \ldots, q_k(X)$ instead of just one polynomial $q(X)$ that equals the left-hand side of Eq. (6) is because the degree of $q(X)$ would be "too large", and hence it is "split" into low-degree polynomials.

The verifier replies with a field element $\mathfrak{z}$ uniformly sampled in a large subset S of $\mathbb{K}$. The honest prover replies with evaluations

$$\left\{ f_1(g_{i,j}\mathfrak{z}), \ldots, f_m(g_{i,j}\mathfrak{z}), q_1(\mathfrak{z}), \ldots, q_k(\mathfrak{z}) \;\middle|\; i \in I, j \in [m] \right\}. \tag{7}$$

Then, the verifier checks that Eq. (6) holds after replacing X by $\mathfrak{z}$ (using the purported evaluations in Eq. (7)), and it engages with the prover in the Batched FRI protocol to verify that the following maps have δ-correlated agreement in some Reed-Solomon code:

$$\left\{ \frac{f_j(X) - f_j(g_{i,j}\mathfrak{z})}{X - g_{i,j}\mathfrak{z}} \;\middle|\; i \in I, j \in [m] \right\} \bigcup \left\{ \frac{q_t(X) - q_t(\mathfrak{z})}{X - \mathfrak{z}} \;\middle|\; t \in [k] \right\}. \tag{8}$$

RBR Knowledge Soundness of the ethSTARK Protocol. It is clear that Π_{ethSTARK} is the compilation of a δ-correlated IOP using the Batched FRI protocol for δ-correlated agreement. Thus, one can prove that Π_{ethSTARK} has RBR (knowledge) soundness by showing that the underlying δ-correlated IOP has RBR (knowledge) soundness when $\delta = 0$. Once this is done, we obtain as a consequence that compiling Π_{ethSTARK} with Merkle tree commitments and the Fiat-Shamir transformation (i.e., the BCS transform) yields a knowledge sound succinct non-interactive argument, i.e., a SNARK. Here, the "underlying δ-correlated IOP" is precisely the protocol Π_{ethSTARK} without applying Batched FRI. Instead, we assume the verifier has an oracle that allows for checking δ-correlated agreement

of the maps that are batched together in Batched FRI. These are the quotient polynomials in Eq. (8).

As we mentioned, due to our results (Theorem 3), it suffices to analyze RBR knowledge soundness when $\delta = 0$. This corresponds to the case when the verifier has an oracle for checking that the maps of Eq. (8) are low-degree polynomials. Note that if the maps in Eq. (8) have 0-correlated agreement, then so do all the (oracles to) maps sent by the adversary during the protocol execution. This is because if a map of the form $(h(X) - y)/(X - z)$ for constants y, z agree with a polynomial $q(X)$ on a set S, then $h(X)$ agrees with the polynomial $q(X)(X-z)+y$ on the same set S. Moreover, for any map $h(X)$ sent by the prover there is a map of the form $(h(X)-y)/(X-z)$ in the list of Eq. (8). Hence we only need consider adversaries that send (oracles to) low-degree polynomials. Moreover, the check for 0-correlated agreement also forces the prover to provide correct openings for Eq. (7).

We say that a 1-round partial transcript is doomed if the left-hand side of Eq. (6) is not a polynomial of appropriate degree. We say that a 2-round partial transcript is doomed if Eq. (6) does not hold for the received challenge $\mathfrak{z}$. Clearly, if a 1-round partial transcript is doomed, then a 2-round partial transcript is doomed except with probability $d'/|S|$, where d' is the degree of the polynomial equation obtained from Eq. (6) after multiplying it by $Z_H(X)$ on each side. Moreover, any doomed 2-round partial transcript is eventually rejected by the verifier, no matter how it is completed, since Eq. (6) does not hold for $X = \mathfrak{z}$. Finally, if $f_1(X), \ldots, f_m(X)$ do not "encode a valid witness", then by how the AIRs are constructed, not all maps in Eq. (5) are polynomials of appropriate degree. Then we claim there are at most $|\mathbb{K}|^{2|I|-1}$ tuples $(r_1, r'_1, \ldots, r_{|I|}, r'_{|I|})$ such that the right-hand side of Eq. (6) is a polynomial of appropriate degree. If the claim is true, then an incorrect initial message $f_1(X), \ldots, f_m(X)$ leads to a doomed state after Round 1 except with probability $1/|\mathbb{K}|$. To prove the claim, consider the expression

$$\sum_{i \in I} (r_i + r'_i X^{c_i}) Q_i(X, f_1(g_{i,1}X), \ldots, f_m(g_{i,m}X))(Z_H(X)/Z_{H_i}(X)) \tag{9}$$

where we view $Z_H(X)/Z_{H_i}(X)$ as a polynomial since $Z_{H_i}(X)$ divides $Z_H(X)$. Then the right-hand side of Eq. (6) is a polynomial of appropriate degree if and only if Eq. (9) vanishes on H. The latter means that for each $x \in H$, the elements $(r_1, r'_1, \ldots, r_{|I|}, r'_{|I|})$ form a solution to the equation

$$\sum_{i \in I} (r_i + r'_i x^{c_i}) Q_i(x, f_1(g_{i,1}x), \ldots, f_m(g_{i,m}x))(Z_H(x)/Z_{H_i}(x)) = 0$$

on the variables $\{r_i, r'_i \mid i \in I\}$. Unless the right-hand side of the equation is identically zero, there are at most $|\mathbb{K}|^{2|I|-1}$ such solutions. On the other hand, if for all $x \in H$ the right-hand side of the equation was identically zero, then each of the maps Eq. (5) would be polynomials of appropriate degree (recall that the adversary is constrained to sending low-degree polynomials), contradicting the

assumption that $f_1(X), \ldots, f_m(X)$ encode an incorrect witness. This proves the claim.

It follows that, in its 0-correlated form, Π_{ethSTARK} has RBR (knowledge) soundness error $\varepsilon_0 := \max\{1/|\mathbb{K}|, d'/|S|\}$. Then, due to the results from Sects. 2.2 and 2.3, Π_{ethSTARK} (as an IOP) has RBR (knowledge) soundness $\varepsilon := \max\{\ell/|\mathbb{K}|, \ell d'/|S|, \varepsilon_{\mathsf{rbr}}^{\mathsf{bFRI}}\}$, where $\ell = 1/(2\sqrt{\rho}\eta)$ (here ρ and η are parameters related to the RS codes used within the protocol), and $\varepsilon_{\mathsf{rbr}}^{\mathsf{bFRI}}$ is the RBR soundness error of batched FRI.

Remark 6. This analysis can slightly improve the knowledge soundness error for Π_{ethSTARK} when compared with [65]. This improvement is already demonstrated in [39]. Using the notation of [65, Theorem 4], the improved knowledge soundness error is

$$(\ell/|\mathbb{K}|) + \ell \cdot (\mathsf{d}_{\mathsf{max}} + 2^{\mathsf{h}} + \mathsf{a})/(|\mathbb{K}| - \mathsf{a} \cdot |D| - |\mathsf{H}_0|) + \varepsilon_{\mathsf{FRI}}.$$

The improvement here is in having the factor ℓ in the second summand, instead of ℓ^2.

2.6 From Round-by-Round Soundness to Fiat-Shamir Security

As stated in Sect. 2.1, we utilize the BCS transformation for IOPs due to Ben-Sasson et al. [9] to compile our round-by-round sound IOPs into secure non-interactive protocols in the random oracle model. At a high level, the transformation works by first compressing oracles sent by the prover with a Merkle tree [52]; i.e., instead of sending oracle f to the verifier, the prover sends M_f, where M_f is the root of the Merkle tree with leaves corresponding to evaluations of f (in some canonical way). Then whenever the verifier would query oracle f at position i, instead the prover provides the verifier with pair $(f(i), \pi_i)$, where π_i is the Merkle authentication path for proving that $f(i)$ is consistent with M_f. Finally, once the IOP is transformed in this way, it is then compressed using Fiat-Shamir to obtain a non-interactive protocol.

Ben-Sasson et al. showed that applying the BCS transformation to an IOP yields a secure non-interactive protocol in the random oracle model if the IOP satisfied a notion of soundness called *state-restoration soundness*, which roughly says that the IOP remains secure even if the prover is allows to rewind the verifier to any prior state at most b times for some upper bound $b \geq 1$; see [9] for full details. However, it is known that round-by-round soundness is an upper bound on state-restoration soundness: in particular, if a protocol has state-restoration soundness error $\varepsilon_{\mathsf{sr}}(b)$ and round-by-round soundness error $\varepsilon_{\mathsf{rbr}}$, then $\varepsilon_{\mathsf{sr}}(b) \leq b\varepsilon_{\mathsf{rbr}}$ [25,26,44]. Moreover, Chiesa et al. [25,26] showed that if an IOP is both round-by-round sound and round-by-round knowledge sound, then the BCS transformed IOP is both (adaptively) sound and (adaptively) knowledge sound versus both classical and quantum adversaries in the random oracle model.

Applying BCS to FRI and Batched FRI directly gives us a SNARK in the random oracle model, establishing the Fiat-Shamir security for FRI and Batched FRI (i.e., Corollary 2). Similarly, for $\mathsf{OPlonky}^{\mathcal{O}}$, we replace the δ-correlated oracle

$\mathcal{O}$ with the Batched FRI protocol, leveraging our δ-correlated IOP techniques to obtain a round-by-round sound IOP. Then again applying BCS to $\mathsf{OPlonky}$ composed with Batched FRI gives us a SNARK in the random oracle model, establishing the Fiat-Shamir security of $\mathsf{OPlonky}$ composed with Batched FRI (i.e., Corollary 3). Finally, our results allow us to obtain FS security for a variety of Plonk-like protocols; see [14] for details.

3 Our Results

In this section, we formally state all of our results. Due to space constraints, the discussion and proofs for these results can be found in the full version of our work [14].

3.1 Round-by-Round Soundness of FRI and Batched FRI

We formally present the FRI IOPP algorithm in Algorithm 1. The following theorem captures the round-by-round soundness of FRI.

Theorem 6. *Let $\mathbb{F}$ be a finite field, $L_0 \subset \mathbb{F}^*$ be a smooth multiplicative subgroup of size 2^n, $d_0 = 2^k$, $\rho = d_0/|L_0| = 2^{-(n-k)}$, and $\ell \in \mathbb{Z}^+$. For any integer $m \geq 3$, $\eta \in (0, \sqrt{\rho}/(2m))$, $\delta \in (0, 1 - \sqrt{\rho} - \eta)$, and function $G_0 \colon L_0 \to \mathbb{F}$ that is δ-far from $\mathsf{RS}[\mathbb{F}, L_0, d_0]$, Algorithm 1 has round-by-round soundness error*

$$\varepsilon_{\mathsf{rbr}}^{\mathsf{FRI}} := \varepsilon_{\mathsf{rbr}}^{\mathsf{FRI}}(\mathbb{F}, L_0, \rho, \delta, m, \ell) = \max\{[(m + 1/2)^7 |L_0|^2]/[3\rho^{3/2} \cdot |\mathbb{F}|], (1 - \delta)^\ell\}.$$

We extend the above theorem to the Batched FRI protocol, a variant of Algorithm 1 where the prover first sends t oracles $f_1, \ldots, f_t$ to the verifier, and the verifier replies with $\alpha_1, \ldots, \alpha_t \xleftarrow{\$} \mathbb{F}$. The prover and verifier then engage in the FRI protocol for polynomial $G_0 = \sum_i \alpha_i f_i$. The following theorem captures the RBR soundness of Batched FRI.

Theorem 7. *Let $\mathbb{F}$ be a finite field, $L_0 \subset \mathbb{F}^*$ be a smooth multiplicative subgroup of size 2^n, $d_0 = 2^k$, $\rho = d_0/|L_0| = 2^{-(n-k)}$, and $\ell \in \mathbb{Z}^+$. For any integer $m \geq 3$, $\eta \in (0, \sqrt{\rho}/(2m))$, $\delta \in (0, 1 - \sqrt{\rho} - \eta)$, and functions $f_1^{(0)}, \ldots, f_t^{(0)} \colon L_0 \to \mathbb{F}$ for $t \geq 2$ such that at least one $f_i^{(0)}$ that is δ-far from $\mathsf{RS}^{(0)}$, the Batched FRI protocol has round-by-round soundness error*

$$\varepsilon_{\mathsf{rbr}}^{\mathsf{bFRI}} := \varepsilon_{\mathsf{rbr}}^{\mathsf{bFRI}}(\mathbb{F}, L_0, \rho, \delta, m, \ell, t) = \max\{[(m + 1/2)^7 |L_0|^2]/[3\rho^{3/2} \cdot |\mathbb{F}|], (1 - \delta)^\ell\}.$$

Fiat-Shamir Security of FRI and Batched FRI. Given the BCS transformation [9] (also see [14]), we can apply the BCS transformation to transform both FRI and Batched FRI into SNARKs in the random oracle model. The following corollaries capture this result.

Corollary 4 (FS Security of FRI). *Let $\mathbb{F}$ be a finite field, $L_0 \subset \mathbb{F}^*$ be a smooth multiplicative subgroup of size 2^n, $d_0 = 2^k$, $\rho = d_0/|L_0| = 2^{-(n-k)}$, and $\ell \in \mathbb{Z}^+$. For any integer $m \geq 3$, $\eta \in (0, \sqrt{\rho}/(2m))$, $\delta \in (0, 1 - \sqrt{\rho} - \eta)$, random*

oracle $\mathcal{H}\colon \{0,1\}^* \to \{0,1\}^\kappa$*, query bound* $Q \in \mathbb{N}$*, and function* $G_0\colon L_0 \to \mathbb{F}$ *that is* δ*-far from* $\mathsf{RS}[\mathbb{F}, L_0, d_0]$*, compiling Algorithm 1 with the BCS transformation [9] gives a non-interactive random oracle proof with adaptive soundness error and knowledge error*

$$\varepsilon_{\mathsf{fs}}^{\mathsf{FRI}} := \varepsilon_{\mathsf{fs}}^{\mathsf{FRI}}(\mathbb{F}, L_0, \rho, \delta, m, \ell, Q, \kappa) = Q \cdot \varepsilon_{\mathsf{rbr}}^{\mathsf{FRI}}(\mathbb{F}, L_0, \rho, \delta, m, \ell) + (3(Q^2+1)/2^\kappa).$$

Moreover, if $\gamma := \gamma(\mathbb{F}, L_0, \rho, \delta, \ell)$ *denotes the length of a FRI proof for parameters* $\mathbb{F}, L_0, \rho, \delta, \ell$*, then the above non-interactive random oracle proof has adaptive soundness error and knowledge error*

$$\varepsilon_{\mathsf{fs-q}}^{\mathsf{FRI}} := \varepsilon_{\mathsf{fs-q}}^{\mathsf{FRI}}(\mathbb{F}, L_0, \rho, \delta, m, \ell, Q, \kappa) = \Theta(Q \cdot \varepsilon_{\mathsf{fs}}^{\mathsf{FRI}}(\mathbb{F}, L_0, \rho, \delta, m, \ell, Q, \kappa))$$

against quantum adversaries that can make at most $Q - O(\ell \cdot \log(\gamma))$ *queries.*

Corollary 5 (FS Security of Batched FRI). *Let* $\mathbb{F}$ *be a finite field,* $L_0 \subset \mathbb{F}^*$ *be a smooth multiplicative subgroup of size* 2^n*,* $d_0 = 2^k$*,* $\rho = d_0/|L_0| = 2^{-(n-k)}$*, and* $\ell \in \mathbb{Z}^+$*. For any integer* $m \geq 3$*,* $\eta \in (0, \sqrt{\rho}/(2m))$*,* $\delta \in (0, 1-\sqrt{\rho}-\eta)$*, random oracle* $\mathcal{H}\colon \{0,1\}* \to \{0,1\}\kappa$*, query bound* $Q \in \mathbb{N}$*, and functions* $f_1^{(0)}, \ldots, f_t^{(0)}\colon L_0 \to \mathbb{F}$ *for* $t \geq 2$ *such that at least one* $f_i^{(0)}$ *is* δ*-far from* $\mathsf{RS}[\mathbb{F}, L_0, d_0]$*, compiling Batched FRI with the BCS transformation [9] gives a non-interactive random oracle proof with adaptive soundness error and knowledge error*

$$\varepsilon_{\mathsf{fs}}^{\mathsf{bFRI}} := \varepsilon_{\mathsf{fs}}^{\mathsf{bFRI}}(\mathbb{F}, L_0, \rho, \delta, m, \ell, t, Q, \kappa) = Q\varepsilon_{\mathsf{rbr}}^{\mathsf{bFRI}}(\mathbb{F}, L_0, \rho, \delta, m, \ell, t) + (3(Q^2+1)/2^\kappa).$$

Moreover, if $\gamma := \gamma(\mathbb{F}, L_0, \rho, \delta, \ell, t)$ *denotes the length of a Batched FRI proof for parameters* $\mathbb{F}, L_0, \rho, \delta, \ell, t$*, then the above non-interactive random oracle proof has adaptive soundness error and knowledge error*

$$\varepsilon_{\mathsf{fs-q}}^{\mathsf{bFRI}} := \varepsilon_{\mathsf{fs-q}}^{\mathsf{bFRI}}(\mathbb{F}, L_0, \rho, \delta, \ell, t, Q, \kappa) = \Theta(Q \cdot \varepsilon_{\mathsf{fs}}^{\mathsf{bFRI}}(\mathbb{F}, L_0, \rho, \delta, \ell, t, Q, \kappa))$$

against quantum adversaries that can make at most $Q - O(\ell \cdot \log(\gamma))$ *queries.*

Remark 7. A variety of works (e.g., [6,65]) make conjectures about the security of the FRI and Batched FRI protocols. We similarly adapt our above results when assuming these conjectured security bounds; see [14] for full details.

3.2 Correlated IOPs

A key technical tool we introduce is the notion of a δ*-correlated (holographic) interactive oracle proof*, or δ-correlated hIOP in short. A δ-correlated hIOP is an hIOP for indexed $(\mathbb{F}, H, d)$-polynomial oracle relations, where we fix some $0 \leq \delta < 1$ and assume the verifier has an oracle $\mathsf{OCoAgg}(\delta)$ for the correlated agreement relation $\mathbf{CoAgg}(\delta)$ (see [14] for complete details). Furthermore, we assume that the final offline verification process consists of: (1) checking that the oracles sent by the prover satisfy a certain polynomial equation on a random point $\mathfrak{z}$ (not necessarily from H); and (2) using $\mathsf{OCoAgg}(\delta)$ to check that the

maps corresponding to certain oracles have correlated agreement in $\mathsf{RS}[\mathbb{F}, H, d]$ (see [14] for details). We denote such a protocol as $\Pi^{\mathsf{OCoAgg}(\delta)}$.

Given a δ-correlated hIOP, our first main result is showing that given a round-by-round sound 0-correlated hIOP, when replacing the oracle $\mathsf{OCoAgg}(0)$ with another suitable IOP, results in a new hIOP that is also round-by-round sound.

Theorem 8. *Let $\Pi^{\mathsf{OCoAgg}(0)} = (\mathsf{Ind}, \mathsf{P}, \mathsf{V}^{\mathsf{OCoAgg}(0)})$ be a μ-round 0-correlated hIOP for an indexed $(\mathbb{F}, D, d)$-polynomial oracle relation $\mathbf{R}$. Let $0 < \delta < 1 - \sqrt{\rho}$, where $\rho = d/|D|$, and let Π_{CA} be a IOPP for δ-correlated agreement in $\mathsf{RS}[\mathbb{F}, D, d]$. Let $\eta > 0$ be such that $\delta = 1 - \sqrt{\rho} - \eta$. Assume Π_{CA} is RBR sound with error $\varepsilon_{\mathsf{CA}}$. Then:*

- *Suppose that $\Pi^{\mathsf{OCoAgg}(0)}$ is RBR sound with error $\varepsilon_{\mathsf{rbr-s}}$. Then there exists a hIOP Π for $\mathbf{R}$ with RBR soundness error $\varepsilon'(\mathbb{i}) := \max\left\{\varepsilon_{\mathsf{rbr-s}}(\mathbb{i})(2\eta\sqrt{\rho}), \varepsilon_{\mathsf{CA}}(\mathbb{i}_{\mathsf{CA}})\right\}$, where $\mathbb{i}_{\mathsf{CA}} = (\mathbb{F}, D, d, \delta, N)$, and N is the number of words whose δ-correlated agreement is checked in the last verification check of $\Pi^{\mathsf{OCoAgg}(\delta)}$.*
- *Suppose $\mu(\mathbb{i}, \mathbb{x}) \geq 1$ for all $\mathbb{i}, \mathbb{x}$ and $\Pi^{\mathsf{OCoAgg}(0)}$ has RBR knowledge error $\varepsilon_{\mathsf{rbr-k}}$, then Π has RBR knowledge error $\max\left\{\varepsilon_{\mathsf{rbr-k}}(\mathbb{i})/(2\eta\sqrt{\rho}), \varepsilon_{\mathsf{CA}}(\mathbb{i}_{\mathsf{CA}})\right\}$, where $\mathbb{i}_{\mathsf{CA}}$ has the same meaning as in above.*

The proof of the above theorem relies on two technical lemmas. The first lemma states that if you have a round-by-round sound 0-correlated hIOP when given access to $\mathsf{OCoAgg}(0)$, then when given access to $\mathsf{OCoAgg}(\delta)$ for $\delta > 0$, the same hIOP is now δ-correlated and is round-by-round sound (with some loss in the soundness error).

Lemma 1. *Let $\Pi^{\mathsf{OCoAgg}(0)} = (\mathsf{Ind}, \mathsf{P}, \mathsf{V}^{\mathsf{OCoAgg}(0)})$ be a μ-round 0-correlated hIOP for an indexed $(\mathbb{F}, D, d)$-polynomial oracle relation $\mathbf{R}$. Let $\delta = 1 - \sqrt{\rho} - \eta$. Then:*

- *Suppose that $\Pi^{\mathsf{OCoAgg}(0)}$ is RBR sound with error $\varepsilon_{\mathsf{rbr-s}}$. Then $\Pi^{\mathsf{OCoAgg}(\delta)}$ has RBR soundness error $\varepsilon_{\mathsf{rbr-s}}(\mathbb{i})/(2\eta\sqrt{\rho})$.*
- *Suppose that $\Pi^{\mathsf{OCoAgg}(0)}$ has RBR knowledge with error $\varepsilon_{\mathsf{rbr-k}}$. Then $\Pi^{\mathsf{OCoAgg}(\delta)}$ has RBR knowledge error $\varepsilon_{\mathsf{rbr-k}}(\mathbb{i})/(2\eta\sqrt{\rho})$,*

The second lemma then states that when one replaces the oracle $\mathsf{OCoAgg}(\delta)$ in the above hIOP with another round-by-round sound IOP for δ-correlated agreement, then the resulting composed protocol remains round-by-round sound.

Lemma 2. *Assume the notation and hypotheses of Theorem 8. Then there exists a hIOP Π_{compiled} for $\mathbf{R}$ with the following properties:*

- *Suppose $\Pi^{\mathsf{OCoAgg}(\delta)}$ has RBR soundness error $\varepsilon_{\mathsf{rbr-s},\delta}$. Then Π_{compiled} has RBR soundness error $\max\left\{\varepsilon_{\mathsf{rbr-s},\delta}(\mathbb{i}), \varepsilon_{\mathsf{CA}}(\mathbb{i}_{\mathsf{CA}})\right\}$.*
- *Suppose $\Pi^{\mathsf{OCoAgg}(\delta)}$ has RBR knowledge soundness error $\varepsilon_{\mathsf{rbr-k},\delta}$. Then Π_{compiled} has RBR knowledge soundness error $\max\left\{\varepsilon_{\mathsf{rbr-k},\delta}(\mathbb{i}), \varepsilon_{\mathsf{CA}}(\mathbb{i}_{\mathsf{CA}})\right\}$.*

3.3 A Plonk-Like Protocol Abstraction $\mathsf{OPlonky}$

Building upon the δ-correlated hIOP framework, we introduce a δ-correlated hIOP we call $\mathsf{OPlonky}$, which abstracts the polynomial IOPs underlying many of the variants of Plonk. This generalization is inspired in part by Plonky2 [60]. Our main technical result is establishing the round-by-round soundness of $\mathsf{OPlonky}(0) := \mathsf{OPlonky}^{\mathsf{OCoAgg}(0)}$, where we assume the verifier has oracle access to the 0-correlated agreement oracle $\mathsf{OCoAgg}(0)$.

Lemma 3. *The 0-correlated agreement encoded hIOP* $\mathsf{OPlonky}(0)$ *has RBR soundness and RBR knowledge error* $\varepsilon(\mathtt{i}) := \max_{i\in[3]}\{\varepsilon_i(\mathtt{i})\}$*, where*

$$\varepsilon_1(\mathtt{i}) := ([3\mathsf{n}(\mathsf{r}'+\mathsf{u})]/|\mathbb{F}|)^{\mathsf{t}}, \quad \varepsilon_2(\mathtt{i}) := ([|\mathcal{P}|+(\mathsf{s}+2)\mathsf{t}-1]/|\mathbb{F}|)^{\mathsf{t}},$$
$$\varepsilon_3(\mathtt{i}) := \max\{\deg(P_j)_{j\in[|\mathcal{P}|]}, \mathsf{u}+1\}\cdot(\mathsf{n}/|\mathbb{K}\setminus D|)$$

for all index $\mathtt{i} = (\mathcal{P}, \mathcal{Q}, H, \sigma, \mathsf{PI}, \mathsf{r}, \mathsf{r}', \ell, \mathsf{t})$*, any potential input* $\mathtt{x}$*, and* $\mathsf{n} = |H|$*.*

Given the above lemma and our δ-correlated hIOP results, we obtain our main theorem for $\mathsf{OPlonky}$: compiling $\mathsf{OPlonky}^{\mathsf{OCoAgg}(\delta)}$ with the Batched FRI protocol.

Theorem 9. *Let* $\mathbb{F}$ *be a finite field,* $D \subseteq \mathbb{F}^*$ *a smooth multiplicative subgroup of* $\mathbb{F}$ *of order* 2^n*, and* H *a subgroup of* D *of order* n*. Let* $m \geq 3$*,* $\delta = 1-\sqrt{\rho}-\eta$ *for some* $\eta \in (0, \sqrt{\rho}/2m)$*, and let* $\mathsf{Plonky2hIOP}$ *be the hIOP obtained from* $\mathsf{OPlonky}(\delta)$ *after compiling it with the Batched FRI protocol (see [14]). Then* $\mathsf{Plonky2hIOP}$ *is RBR sound and has RBR knowledge. For each* $\mathtt{i} = (\mathcal{P}, \mathcal{Q}, H, \sigma, \mathsf{PI}, \mathsf{r}, \mathsf{r}', \ell, \mathsf{t})$ *and all* $q \geq 1$*, the error in both cases is given by*

$$\varepsilon_{\mathsf{rbr}}^{\mathsf{OPlonky}}(\mathtt{i}, q) = \max\{\left(\varepsilon_i(\mathtt{i})/(2\eta\sqrt{\rho})\right)_{i\in[3]}, \varepsilon_{\mathsf{rbr}}^{\mathsf{bFRI}}(\mathbb{F}, D, \rho, \delta, N, q)\},$$

where N *is the total number of codewords that are batched together in the batched FRI protocol,* $\varepsilon_{\mathsf{rbr}}^{\mathsf{bFRI}}$ *is the RBR soundness error of* $\varepsilon_{\mathsf{rbr}}^{\mathsf{bFRI}}$ *(which equals its RBR knowledge error, see [14]) and*

$$\varepsilon_1(\mathtt{i}) := ([3\mathsf{n}(\mathsf{r}'+\mathsf{u})]/|\mathbb{F}|)^{\mathsf{t}}, \qquad \varepsilon_2(\mathtt{i}) := ([|\mathcal{P}|+(\mathsf{s}+2)\mathsf{t}-1]/|\mathbb{F}|)^{\mathsf{t}},$$

$$\varepsilon_3(\mathtt{i}) := \max\{\deg(P_j)_{j\in[|\mathcal{P}|]}, \mathsf{u}+1\}(\mathsf{n}/|\mathbb{K}\setminus D|).$$

Algorithm 1: FRI-IOPP

Input: Finite field $\mathbb{F}$, smooth multiplicative subgroup $L_0 \subset \mathbb{F}^*$ of size 2^n, degree bound $d_0 = 2^k$, and $\ell \in \mathbb{N}$.
P has function $G_0 \colon L_0 \to \mathbb{F}$ and V has oracle $(G_0(z))_{z \in L_0}$.
Output: The verifier V outputs accept or reject.

foreach $i \in [k]$ **do** // *Fold Phase*
  V sends $x_{i-1} \xleftarrow{\$} \mathbb{F}$ to P.
  P and V set $d_i := d_{i-1}/2$ and $L_i := \{z^2 \colon z \in L_{i-1}\}$.
  P computes unique bi-variate polynomial $Q_{i-1}(X, Y)$ such that
    1. $\deg_X(Q_{i-1}) = 1$;
    2. $\deg_Y(Q_{i-1}) < d_i$; and
    3. $G_{i-1}(r) = Q_{i-1}(r, r^2)$ for all $r \in L_{i-1}$.
  P defines $G_i(Y) := Q_{i-1}(x_{i-1}, Y)$.
  if $i = k$ **then** P sends $G_k = C \in \mathbb{F}$ to V.
  else P sends oracle $(G_i(z))_{z \in L_i}$ to V.
forall $j \in [\ell]$ **do** // *Query Phase; processed in parallel*
  V samples $s_{0,j} \xleftarrow{\$} L_0$.
  foreach $i \in [k]$ **do**
    V computes $s_{i,j} = (s_{i-1,j})^2$ and $s'_{i-1,j} \neq s_{i-1,j}$ such that $(s'_{i-1,j})^2 = s_{i,j}$.
    V queries and obtains $q_{i-1,j} = G_{i-1}(s_{i-1,j})$ and $q'_{i-1,j} = G_{i-1}(s'_{i-1,j})$.
    V computes linear polynomial $\widetilde{Q}_{i-1,j}(X)$ via Lagrange interpolation on the set $\{(s_{i-1,j}, q_{i-1,j}), (s'_{i-1,j}, q'_{i-1,j})\}$.
    V checks that $G_i(s_{i,j}) = \widetilde{Q}_{i-1,j}(x_{i-1})$ by querying G_i.
    if $G_i(s_{i,j}) \neq \widetilde{Q}_{i-1,j}(x_{i-1})$ **then** V outputs reject.
V outputs accept.

4 Conclusions and Open Problems

In this work, we formalized the FS-security of FRI and related SNARKs, particularly Plonk-like protocols captured by δ-correlated IOPs. Our results on Plonk-like protocols cover multiple variants, some of which are already in production. There are other protocols that are amenable to our general framework for correlated IOP's, e.g., ethSTARK [65] and RISC Zero [68]. We leave it as future work to perform a RBR soundness/knowledge and FS analysis of these protocols.

Our generalization OPlonky of IOPs using Plonk-like arithmetization along with a protocol for low-degree testing (specifically, FRI) does not address KZG-based Plonk-like schemes. Compiling a 0-correlated IOP with RBR soundness and knowledge using other commitment schemes and the FS-security of such schemes remain open problems.

Acknowledgements. Alexander R. Block was supported by DARPA under Contract Nos. HR00112020022 and HR00112020025. Albert Garreta and Michał Zając were supported by Ethereum Foundation grant FY23-0885. Work of Jonathan Katz

was supported by NSF award CNS-2154705 and by DARPA under Contract No. HR00112020025. Work of Justin Thaler was supported by NSF CAREER award CCF-1845125 and by DARPA under Contract No. HR00112020022. Pratyush Ranjan Tiwari was supported by NSF award CNS-1814919 and a Google Security and Privacy research award to Matthew Green. The views, opinions, findings, conclusions, and/or recommendations expressed in this material are those of the authors and should not be interpreted as reflecting the position or policy of DARPA or the United States Government, and no official endorsement should be inferred.

References

1. Abdalla, M., An, J.H., Bellare, M., Namprempre, C.: From identification to signatures via the Fiat-Shamir transform: minimizing assumptions for security and forward-security. In: Knudsen, L.R. (ed.) EUROCRYPT 2002. LNCS, vol. 2332, pp. 418–433. Springer, Heidelberg (2002). https://doi.org/10.1007/3-540-46035-7_28
2. Attema, T., Fehr, S., Klooß, M.: Fiat-Shamir transformation of multi-round interactive proofs. In: Kiltz, E., Vaikuntanathan, V. (eds.) TCC 2022, Part I. LNCS, vol. 13747, pp. 113–142. Springer, Heidelberg (2022). https://doi.org/10.1007/978-3-031-22318-1_5
3. Barak, B.: How to go beyond the black-box simulation barrier. In: 42nd FOCS, pp. 106–115. IEEE Computer Society Press (2001). https://doi.org/10.1109/SFCS.2001.959885
4. Ben-Sasson, E., Bentov, I., Horesh, Y., Riabzev, M.: Fast reed-solomon interactive oracle proofs of proximity. In: Chatzigiannakis, I., Kaklamanis, C., Marx, D., Sannella, D. (eds.) ICALP 2018. LIPIcs, vol. 107, pp. 14:1–14:17. Schloss Dagstuhl (2018). https://doi.org/10.4230/LIPIcs.ICALP.2018.14
5. Ben-Sasson, E., Bentov, I., Horesh, Y., Riabzev, M.: Scalable, transparent, and post-quantum secure computational integrity. Cryptology ePrint Archive, Report 2018/046 (2018). https://eprint.iacr.org/2018/046
6. Ben-Sasson, E., Carmon, D., Ishai, Y., Kopparty, S., Saraf, S.: Proximity gaps for reed-solomon codes. Cryptology ePrint Archive, Paper 2020/654 (2020). https://eprint.iacr.org/2020/654, full version of the same work published at FOCS 2020. https://doi.org/10.1109/FOCS46700.2020.00088
7. Ben-Sasson, E., Chiesa, A., Genkin, D., Tromer, E.: Fast reductions from RAMs to delegatable succinct constraint satisfaction problems: extended abstract. In: Kleinberg, R.D. (ed.) ITCS 2013, pp. 401–414. ACM (2013). https://doi.org/10.1145/2422436.2422481
8. Ben-Sasson, E., Chiesa, A., Riabzev, M., Spooner, N., Virza, M., Ward, N.P.: Aurora: transparent succinct arguments for R1CS. In: Ishai, Y., Rijmen, V. (eds.) EUROCRYPT 2019, Part I. LNCS, vol. 11476, pp. 103–128. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-17653-2_4
9. Ben-Sasson, E., Chiesa, A., Spooner, N.: Interactive oracle proofs. In: Hirt, M., Smith, A. (eds.) TCC 2016, Part II. LNCS, vol. 9986, pp. 31–60. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-53644-5_2
10. Ben-Sasson, E., Goldberg, L., Kopparty, S., Saraf, S.: DEEP-FRI: sampling outside the box improves soundness. In: Vidick, T. (ed.) ITCS 2020, vol. 151, pp. 5:1–5:32. LIPIcs (2020). https://doi.org/10.4230/LIPIcs.ITCS.2020.5

11. Ben-Sasson, E., Kopparty, S., Saraf, S.: Worst-case to average case reductions for the distance to a code. In: Servedio, R.A. (ed.) 33rd Computational Complexity Conference, CCC 2018, 22–24 June 2018, San Diego, CA, USA. LIPIcs, vol. 102, pp. 24:1–24:23. Schloss Dagstuhl - Leibniz-Zentrum für Informatik (2018). https://doi.org/10.4230/LIPIcs.CCC.2018.24
12. Bernhard, D., Pereira, O., Warinschi, B.: How not to prove yourself: pitfalls of the Fiat-Shamir heuristic and applications to Helios. In: Wang, X., Sako, K. (eds.) ASIACRYPT 2012. LNCS, vol. 7658, pp. 626–643. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-34961-4_38
13. Bitansky, N., et al.: Why "Fiat-Shamir for proofs" lacks a proof. In: Sahai, A. (ed.) TCC 2013. LNCS, vol. 7785, pp. 182–201. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-36594-2_11
14. Block, A.R., Garreta, A., Katz, J., Thaler, J., Tiwari, P.R., Zając, M.: Fiat-Shamir security of FRI and related snarks. Cryptology ePrint Archive, Paper 2023/1071 (2023). https://eprint.iacr.org/2023/1071
15. Blum, M., Evans, W., Gemmell, P., Kannan, S., Naor, M.: Checking the correctness of memories. Algorithmica **12**, 225–244 (1994)
16. Blumberg, A.J., Thaler, J., Vu, V., Walfish, M.: Verifiable computation using multiple provers. Cryptology ePrint Archive, Report 2014/846 (2014). https://eprint.iacr.org/2014/846
17. Bonneau, J., Clark, J., Goldfeder, S.: On bitcoin as a public randomness source. IACR Cryptology ePrint Archive, p. 1015 (2015)
18. Bootle, J., Cerulli, A., Chaidos, P., Groth, J., Petit, C.: Efficient zero-knowledge arguments for arithmetic circuits in the discrete log setting. In: Fischlin, M., Coron, J.-S. (eds.) EUROCRYPT 2016, Part II. LNCS, vol. 9666, pp. 327–357. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-49896-5_12
19. Bootle, J., Cerulli, A., Groth, J., Jakobsen, S., Maller, M.: Arya: nearly linear-time zero-knowledge proofs for correct program execution. In: Peyrin, T., Galbraith, S. (eds.) ASIACRYPT 2018. LNCS, vol. 11272, pp. 595–626. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-03326-2_20
20. Bünz, B., Bootle, J., Boneh, D., Poelstra, A., Wuille, P., Maxwell, G.: Bulletproofs: short proofs for confidential transactions and more. In: 2018 IEEE Symposium on Security and Privacy, pp. 315–334. IEEE Computer Society Press (2018). https://doi.org/10.1109/SP.2018.00020
21. Bünz, B., Fisch, B., Szepieniec, A.: Transparent SNARKs from DARK compilers. In: Canteaut, A., Ishai, Y. (eds.) EUROCRYPT 2020, Part I. LNCS, vol. 12105, pp. 677–706. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-45721-1_24
22. Canetti, R., et al.: Fiat-Shamir: from practice to theory. In: Charikar, M., Cohen, E. (eds.) 51st ACM STOC, pp. 1082–1090. ACM Press (2019). https://doi.org/10.1145/3313276.3316380
23. Canetti, R., Chen, Y., Reyzin, L., Rothblum, R.D.: Fiat-Shamir and correlation intractability from strong KDM-secure Encryption. In: Nielsen, J.B., Rijmen, V. (eds.) EUROCRYPT 2018, Part I. LNCS, vol. 10820, pp. 91–122. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-78381-9_4
24. Canetti, R., Goldreich, O., Halevi, S.: The random oracle methodology, revisited. J. ACM **51**(4), 557–594 (2004). https://doi.org/10.1145/1008731.1008734
25. Chiesa, A., Manohar, P., Spooner, N.: Succinct arguments in the quantum random oracle model. In: Hofheinz, D., Rosen, A. (eds.) TCC 2019, Part II. LNCS, vol. 11892, pp. 1–29. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-36033-7_1

26. Chiesa, A., Ojha, D., Spooner, N.: FRACTAL: post-quantum and transparent recursive proofs from holography. In: Canteaut, A., Ishai, Y. (eds.) EUROCRYPT 2020, Part I. LNCS, vol. 12105, pp. 769–793. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-45721-1_27
27. Cormode, G., Mitzenmacher, M., Thaler, J.: Practical verified computation with streaming interactive proofs. In: Goldwasser, S. (ed.) ITCS 2012, pp. 90–112. ACM (2012). https://doi.org/10.1145/2090236.2090245
28. Cramer, R., Damgård, I., Schoenmakers, B.: Proofs of partial knowledge and simplified design of witness hiding protocols. In: Desmedt, Y.G. (ed.) CRYPTO 1994. LNCS, vol. 839, pp. 174–187. Springer, Heidelberg (1994). https://doi.org/10.1007/3-540-48658-5_19
29. Dao, Q., Miller, J., Wright, O., Grubbs, P.: Weak fiat-shamir attacks on modern proof systems. Cryptology ePrint Archive, Paper 2023/691 (2023). https://eprint.iacr.org/2023/691
30. Dusk Network: Plonkup. https://github.com/dusk-network/plonkup. Accessed 24 May 2023
31. Fiat, A., Shamir, A.: How to prove yourself: practical solutions to identification and signature problems. In: Odlyzko, A.M. (ed.) CRYPTO 1986. LNCS, vol. 263, pp. 186–194. Springer, Heidelberg (1987). https://doi.org/10.1007/3-540-47721-7_12
32. Gabizon, A., Williamson, Z.J.: The turbo-plonk program syntax for specifying snark programs. https://docs.zkproof.org/pages/standards/accepted-workshop3/proposal-turbo_plonk.pdf. Accessed 23 May 2023
33. Gabizon, A., Williamson, Z.J.: plookup: a simplified polynomial protocol for lookup tables. Cryptology ePrint Archive, Paper 2020/315 (2020). https://eprint.iacr.org/2020/315
34. Gabizon, A., Williamson, Z.J., Ciobotaru, O.: PLONK: permutations over lagrange-bases for oecumenical noninteractive arguments of knowledge. Cryptology ePrint Archive, Report 2019/953 (2019). https://eprint.iacr.org/2019/953
35. Ghoshal, A., Tessaro, S.: Tight state-restoration soundness in the algebraic group model. In: Malkin, T., Peikert, C. (eds.) CRYPTO 2021, Part III. LNCS, vol. 12827, pp. 64–93. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-84252-9_3
36. Goldwasser, S., Kalai, Y.T.: On the (in)security of the Fiat-Shamir paradigm. In: 44th FOCS, pp. 102–115. IEEE Computer Society Press (2003). https://doi.org/10.1109/SFCS.2003.1238185
37. Goldwasser, S., Kalai, Y.T., Rothblum, G.N.: Delegating computation: interactive proofs for muggles. In: Ladner, R.E., Dwork, C. (eds.) 40th ACM STOC, pp. 113–122. ACM Press (2008). https://doi.org/10.1145/1374376.1374396
38. Goldwasser, S., Micali, S., Rackoff, C.: The knowledge complexity of interactive proof systems. SIAM J. Comput. **18**(1), 186–208 (1989). https://doi.org/10.1137/0218012
39. Haböck, U.: A summary on the FRI low degree test. Cryptology ePrint Archive, Report 2022/1216 (2022). https://eprint.iacr.org/2022/1216
40. Holmgren, J., Lombardi, A.: Cryptographic hashing from strong one-way functions (or: one-way product functions and their applications). In: Thorup, M. (ed.) 59th FOCS, pp. 850–858. IEEE Computer Society Press (2018). https://doi.org/10.1109/FOCS.2018.00085
41. Holmgren, J., Lombardi, A., Rothblum, R.D.: Fiat-Shamir via list-recoverable codes (or: parallel repetition of GMW is not zero-knowledge). In: Khuller, S., Williams, V.V. (eds.) 53rd ACM STOC, pp. 750–760. ACM Press (2021). https://doi.org/10.1145/3406325.3451116

42. Kalai, Y.T., Rothblum, G.N., Rothblum, R.D.: From obfuscation to the security of Fiat-Shamir for proofs. In: Katz, J., Shacham, H. (eds.) CRYPTO 2017, Part II. LNCS, vol. 10402, pp. 224–251. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-63715-0_8
43. Kate, A., Zaverucha, G.M., Goldberg, I.: Constant-size commitments to polynomials and their applications. In: Abe, M. (ed.) ASIACRYPT 2010. LNCS, vol. 6477, pp. 177–194. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-17373-8_11
44. Kattis, A.A., Panarin, K., Vlasov, A.: RedShift: transparent SNARKs from list polynomial commitments. In: Yin, H., Stavrou, A., Cremers, C., Shi, E. (eds.) ACM CCS 2022, pp. 1725–1737. ACM Press (2022). https://doi.org/10.1145/3548606.3560657
45. Kilian, J.: A note on efficient zero-knowledge proofs and arguments (extended abstract). In: 24th ACM STOC, pp. 723–732. ACM Press (1992). https://doi.org/10.1145/129712.129782
46. L2BEAT: L2BEAT total value locked. https://l2beat.com/scaling/tvl. Accessed 22 May 2023
47. Lipton, R.J.: Fingerprinting sets. Princeton University, Department of Computer Science (1989)
48. Lipton, R.J.: Efficient checking of computations. In: Choffrut, C., Lengauer, T. (eds.) STACS 1990. LNCS, vol. 415, pp. 207–215. Springer, Heidelberg (1990). https://doi.org/10.1007/3-540-52282-4_44
49. Lund, C., Fortnow, L., Karloff, H.J., Nisan, N.: Algebraic methods for interactive proof systems. J. ACM **39**(4), 859–868 (1992). https://doi.org/10.1145/146585.146605
50. Maller, M., Bowe, S., Kohlweiss, M., Meiklejohn, S.: Sonic: Zero-knowledge SNARKs from linear-size universal and updatable structured reference strings. In: Cavallaro, L., Kinder, J., Wang, X., Katz, J. (eds.) ACM CCS 2019, pp. 2111–2128. ACM Press (2019). https://doi.org/10.1145/3319535.3339817
51. Matter Labs: zksync 2.0: Hello ethereum! https://blog.matter-labs.io/zksync-2-0-hello-ethereum-ca48588de179. Accessed 24 May 2023
52. Merkle, R.: Secrecy, authentication, and public key systems (1979)
53. Micali, S.: CS proofs (extended abstracts). In: 35th FOCS, pp. 436–453. IEEE Computer Society Press (1994). https://doi.org/10.1109/SFCS.1994.365746
54. Micali, S.: Computationally sound proofs. SIAM J. Comput. **30**(4), 1253–1298 (2000). https://doi.org/10.1137/S0097539795284959
55. Mina: Mina book: Background on plonk. https://o1-labs.github.io/proof-systems/plonk/overview.html. Accessed 24 May 2023
56. =nil; Foundation: Circuit definition library for =nil; foundation's cryptography suite. https://github.com/NilFoundation/zkllvm-blueprint. Accessed 24 May 2023
57. Pierrot, C., Wesolowski, B.: Malleability of the blockchain's entropy. Cryptogr. Commun. **10**(1), 211–233 (2018)
58. Pointcheval, D., Stern, J.: Security proofs for signature schemes. In: Maurer, U. (ed.) EUROCRYPT 1996. LNCS, vol. 1070, pp. 387–398. Springer, Heidelberg (1996). https://doi.org/10.1007/3-540-68339-9_33
59. Polygon Labs: FRI verification procedures. https://wiki.polygon.technology/docs/miden/user_docs/stdlib/crypto/fri/. Accessed 23 May 2023
60. Polygon Zero Team: Plonky2: Fast recursive arguments with plonk and FRI. https://github.com/mir-protocol/plonky2/tree/main/plonky2
61. Rabin, M.O.: Transaction protection by beacons. J. Comput. Syst. Sci. **27**(2), 256–267 (1983). https://doi.org/10.1016/0022-0000(83)90042-9

62. Reed, I.S., Solomon, G.: Polynomial codes over certain finite fields. J. Soc. Ind. Appl. Math. **8**(2), 300–304 (1960). https://doi.org/10.1137/0108018
63. Ron-Zewi, N., Rothblum, R.D.: Local proofs approaching the witness length [extended abstract]. In: 61st FOCS, pp. 846–857. IEEE Computer Society Press (2020). https://doi.org/10.1109/FOCS46700.2020.00083
64. Setty, S.: Spartan: efficient and general-purpose zkSNARKs without trusted setup. In: Micciancio, D., Ristenpart, T. (eds.) CRYPTO 2020, Part III. LNCS, vol. 12172, pp. 704–737. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-56877-1_25
65. StarkWare: ethstark documentation. Cryptology ePrint Archive, Paper 2021/582 (2021). https://eprint.iacr.org/2021/582
66. StarkWare Industries: Starkex documentation: Customers and their deployment contract addresses. https://docs.starkware.co/starkex/deployments-addresses.html. Accessed 22 May 2023
67. Succinct Labs: gnark-plonky2-verifier. https://github.com/succinctlabs/gnark-plonky2-verifier. Accessed 24 May 2023
68. Team, R.Z.: RISC zero's proof system for a zkVM (2023). https://github.com/risc0/risc0. github repository
69. Thaler, J.: Time-optimal interactive proofs for circuit evaluation. In: Canetti, R., Garay, J.A. (eds.) CRYPTO 2013, Part II. LNCS, vol. 8043, pp. 71–89. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-40084-1_5
70. Thaler, J.: Proofs, arguments, and zero-knowledge (2022). https://people.cs.georgetown.edu/jthaler/ProofsArgsAndZK.html
71. Wahby, R.S., Tzialla, I., shelat, a., Thaler, J., Walfish, M.: Doubly-efficient zkSNARKs without trusted setup. In: 2018 IEEE Symposium on Security and Privacy, pp. 926–943. IEEE Computer Society Press (2018). https://doi.org/10.1109/SP.2018.00060
72. Wikström, D.: Special soundness in the random oracle model. Cryptology ePrint Archive, Report 2021/1265 (2021). https://eprint.iacr.org/2021/1265
73. Zhang, Y., Genkin, D., Katz, J., Papadopoulos, D., Papamanthou, C.: vRAM: faster verifiable ram with program-independent preprocessing. In: 2018 IEEE Symposium on Security and Privacy (SP), pp. 908–925. IEEE (2018)