FRI
Fast
Reed-Solomon (RS)
Interactive Oracle Proofs of Proximity (IOPP)

From ICALP 2018 presentation

Eli Ben-Sasson Iddo Bentov Yinon Horesh Michael Riabzev

February 2019

Overview

tl;dr: FRI is a fast, FFT-like, IOP solution for verifying $\deg(f) < d$

- motivation
- main result, applications
- FRI protocol dive-in

Reed Solomon (RS) codes [RS60]

- prominent role in algebraic coding and computational complexity
- For $S \subset \mathbb{F}$ a finite field and $\rho \in (0, 1]$ a *rate* parameter

$$\mathsf{RS}[\mathbb{F}, S, \rho] = \{f : S \to \mathbb{F} \mid \deg(f) < \rho|S|\}$$

Reed Solomon (RS) codes [RS60]

- prominent role in algebraic coding and computational complexity
- For $S \subset \mathbb{F}$ a finite field and $\rho \in (0, 1]$ a *rate* parameter

$$\mathrm{RS}[\mathbb{F}, S, \rho] = \{f : S \to \mathbb{F} \mid \deg(f) < \rho|S|\}$$

- RS codes have many desirable properties, like
  - maximum distance separable (MDS): rel. Hamming distance $1 - \rho$
  - efficient, quasi-linear time encoding via FFT
  - efficient unique decoding [BW83] and list decoding [GS99]
  - used in quasi-linear PCPs [BS05] and constant rate IOPs [BCGRS16]

Reed Solomon (RS) codes [RS60]

- prominent role in algebraic coding and computational complexity
- For $S \subset \mathbb{F}$ a finite field and $\rho \in (0, 1]$ a *rate* parameter

$$\mathrm{RS}[\mathbb{F}, S, \rho] = \{f : S \to \mathbb{F} \mid \deg(f) < \rho|S|\}$$

- RS codes have many desirable properties, like
    - maximum distance separable (MDS): rel. Hamming distance $1 - \rho$
    - efficient, quasi-linear time encoding via FFT
    - efficient unique decoding [BW83] and list decoding [GS99]
    - used in quasi-linear PCPs [BS05] and constant rate IOPs [BCGRS16]
- notation:
    - $d = \rho|S| - 1$ is degree;
    - $n = |S|$ is blocklength;
    - Δ is relative Hamming distance

RS proximity testing (RPT) problem

- Question: Construct a verifier V that has
  - oracle access to $f^{(0)} : S^{(0)} \to \mathbb{F}$
  - completeness: If $f^{(0)} \in \mathrm{RS}[\mathbb{F}, S, \rho]$, then $\Pr[V \text{ accepts } f^{(0)}] = 1$
  - soundness: otherwise, $\Pr[V \text{ rejects } f^{(0)}] \geq \Delta(f^{(0)}, \mathrm{RS}[\mathbb{F}, S^{(0)}, \rho])$

while minimizing query complexity q.

RS proximity testing (RPT) problem

- Question: Construct a verifier V that has
  - oracle access to $f^{(0)} : S^{(0)} \to \mathbb{F}$
  - completeness: If $f^{(0)} \in \mathrm{RS}[\mathbb{F}, S, \rho]$, then $\Pr[V \text{ accepts } f^{(0)}] = 1$
  - soundness: otherwise, $\Pr[V \text{ rejects } f^{(0)}] \geq \Delta(f^{(0)}, \mathrm{RS}[\mathbb{F}, S^{(0)}, \rho])$

  while minimizing query complexity q.
- Answers:
  - $\mathsf{q} = d + 1$ required and sufficient [folklore]

RS proximity testing (RPT) problem

- Question: Construct a verifier V that has
  - oracle access to $f^{(0)} : S^{(0)} \to \mathbb{F}$ and PCPP $\pi : S^{(1)} \to \mathbb{F}$
  - completeness: If $f^{(0)} \in \mathsf{RS}[\mathbb{F}, S, \rho]$, then $\Pr[V \text{ accepts } f^{(0)}] = 1$
  - soundness: otherwise, $\Pr[V \text{ rejects } f^{(0)}] \geq \Delta(f^{(0)}, \mathsf{RS}[\mathbb{F}, S^{(0)}, \rho])$

  while minimizing query complexity q.
- Answers:
  - $\mathsf{q} = d + 1$ required and sufficient [folklore]
  - $\mathsf{q} = O(1/\delta)$, if verifier has oracle access to PCPP [AS+ALMSS98]

RS proximity testing (RPT) problem

- Question: Construct a verifier V that has
    - oracle access to $f^{(0)} : S^{(0)} \to \mathbb{F}$ and PCPP $\pi : S^{(1)} \to \mathbb{F}$
    - completeness: If $f^{(0)} \in \text{RS}[\mathbb{F}, S, \rho]$, then $\Pr[V \text{ accepts } f^{(0)}] = 1$
    - soundness: otherwise, $\Pr[V \text{ rejects } f^{(0)}] \geq \Delta(f^{(0)}, \text{RS}[\mathbb{F}, S^{(0)}, \rho])$

  while minimizing query complexity q.
- Answers:
    - $\mathsf{q} = d + 1$ required and sufficient [folklore]
    - $\mathsf{q} = O(1/\delta)$, if verifier has oracle access to PCPP [AS+ALMSS98]
    - PCPP can have quasi-linear length $n \log^{O(1)} n$ [BS08, D07]
    - IOPP can have linear length $O(n)$ [BCF^{+}16, BBGR16]

RS proximity testing (RPT) problem

- Question: Construct a verifier V that has
  - oracle access to $f^{(0)} : S^{(0)} \to \mathbb{F}$ and PCPP $\pi : S^{(1)} \to \mathbb{F}$
  - completeness: If $f^{(0)} \in \mathrm{RS}[\mathbb{F}, S, \rho]$, then $\Pr[V \text{ accepts } f^{(0)}] = 1$
  - soundness: otherwise, $\Pr[V \text{ rejects } f^{(0)}] \geq \Delta(f^{(0)}, \mathrm{RS}[\mathbb{F}, S^{(0)}, \rho])$

  while minimizing query complexity q.
- Answers:
  - $\mathrm{q} = d + 1$ required and sufficient [folklore]
  - $\mathrm{q} = O(1/\delta)$, if verifier has oracle access to PCPP [AS+ALMSS98]
  - PCPP can have quasi-linear length $n \log^{O(1)} n$ [BS08, D07]
  - IOPP can have linear length $O(n)$ [BCF$^+$16, BBGR16]
- Interactive Oracle Proof of Proximity (IOPP) model [BCS16,RRR16,BCF+16]
  - prover sends $f^{(0)} : S^{(0)} \to \mathbb{F}$; verifier sends random $x^{(0)}$
  - prover sends $f^{(1)} : S^{(1)} \to \mathbb{F}$; verifier sends random $x^{(1)}$
  - repeat for r rounds
  - verifier queries $f^{(0)}, \ldots, f^{(\mathrm{r})}$; based on answers and $(x^{(0)}, \ldots, x^{(\mathrm{r}-1)})$ verifier decides to accept/reject claim "$f^{(0)} \in \mathrm{RS}\left[\mathbb{F}, S^{(0)}, \rho\right]$"

RS proximity testing (RPT) problem

- Question: Construct a verifier V that has
  - oracle access to $f^{(0)} : S^{(0)} \to \mathbb{F}$ and PCPP $\pi : S^{(1)} \to \mathbb{F}$
  - completeness: If $f^{(0)} \in \mathrm{RS}[\mathbb{F}, S, \rho]$, then $\Pr[V \text{ accepts } f^{(0)}] = 1$
  - soundness: otherwise, $\Pr[V \text{ rejects } f^{(0)}] \geq \Delta(f^{(0)}, \mathrm{RS}[\mathbb{F}, S^{(0)}, \rho])$

  while minimizing query complexity q.
- Answers:
  - $\mathsf{q} = d + 1$ required and sufficient [folklore]
  - $\mathsf{q} = O(1/\delta)$, if verifier has oracle access to PCPP [AS+ALMSS98]
  - PCPP can have quasi-linear length $n \log^{O(1)} n$ [BS08, D07]
  - IOPP can have linear length $O(n)$ [BCF+16, BBGR16]
- This work: IOPP model, minimize q *and*
  1. total proof length $\ell = |\pi_1| + \ldots + |\pi_r|$
  2. prover arithmetic complexity $\mathsf{t_p}$
  3. verifier arithmetic complexity $\mathsf{t_v}$

RS proximity testing (RPT) problem

- Question: Construct a verifier V that has
  - oracle access to $f^{(0)} : S^{(0)} \to \mathbb{F}$ and PCPP $\pi : S^{(1)} \to \mathbb{F}$
  - completeness: If $f^{(0)} \in \mathrm{RS}[\mathbb{F}, S, \rho]$, then $\Pr[V \text{ accepts } f^{(0)}] = 1$
  - soundness: otherwise, $\Pr[V \text{ rejects } f^{(0)}] \geq \Delta(f^{(0)}, \mathrm{RS}[\mathbb{F}, S^{(0)}, \rho])$

  while minimizing query complexity q.
- Answers:
  - $\mathsf{q} = d + 1$ required and sufficient [folklore]
  - $\mathsf{q} = O(1/\delta)$, if verifier has oracle access to PCPP [AS+ALMSS98]
  - PCPP can have quasi-linear length $n \log^{O(1)} n$ [BS08, D07]
  - IOPP can have linear length $O(n)$ [BCF$^+$16, BBGR16]
- This work: IOPP model, minimize q *and*
  1. total proof length $\ell = |\pi_1| + \ldots + |\pi_r|$
  2. prover arithmetic complexity $\mathsf{t_p}$
  3. verifier arithmetic complexity $\mathsf{t_v}$
  4. for "small", concrete, non-asymptotic values of n, $(< 2^{50})$, using non-asymptotic bounds ($\cancel{O}$, $\cancel{\Omega}$, $\cancel{\Theta}$)

RS proximity testing (RPT) problem

- Question: Construct a verifier V that has
  - oracle access to $f^{(0)} : S^{(0)} \to \mathbb{F}$ and PCPP $\pi : S^{(1)} \to \mathbb{F}$
  - completeness: If $f^{(0)} \in \mathsf{RS}[\mathbb{F}, S, \rho]$, then $\Pr[V \text{ accepts } f^{(0)}] = 1$
  - soundness: otherwise, $\Pr[V \text{ rejects } f^{(0)}] \geq \Delta(f^{(0)}, \mathsf{RS}[\mathbb{F}, S^{(0)}, \rho])$

  while minimizing query complexity q.
- Answers:
  - $\mathsf{q} = d + 1$ required and sufficient [folklore]
  - $\mathsf{q} = O(1/\delta)$, if verifier has oracle access to PCPP [AS+ALMSS98]
  - PCPP can have quasi-linear length $n \log^{O(1)} n$ [BS08, D07]
  - IOPP can have linear length $O(n)$ [BCF+16, BBGR16]
- This work: IOPP model, minimize q *and*
  1. total proof length $\ell = |\pi_1| + \ldots + |\pi_r|$
  2. prover arithmetic complexity $\mathsf{t_p}$
  3. verifier arithmetic complexity $\mathsf{t_v}$
  4. for "small", concrete, non-asymptotic values of n, ($< 2^{50}$), using non-asymptotic bounds ($\cancel{O}, \cancel{\Omega}, \cancel{\Theta}$)
- Why? 1–3 interesting theoretically, 4 important practically, for ZK systems like Ligero [AHIV17], STARK [BBHR18], Aurora [BCRSVW19], ...

Prior RS proximity testing (RPT) results

	prover comp.	proof length	verifier comp.	query comp.	round comp.
folklore	0	0	$\tilde{O}(\rho n)$	ρn	0
PCP **[ALM+92]**	$n^{O(1)}$	$n^{O(1)}$	$n^{O(1)}$	$O\left(\frac{1}{\delta}\right)$	1
PCP **[BFL+90]**	$n^{1+\epsilon}$	$n^{1+\epsilon}$	$\frac{1}{\delta}\log^{1/\epsilon} n$	$\frac{1}{\delta}\log^{1/\epsilon} n$	1
PCPP **[BS+05]**	$n\log^{1.5} n$	$n\log^{1.5} n$	$\frac{1}{\delta}\log^{5.8} n$	$\frac{1}{\delta}\log^{5.8} n$	1
PCPP **[D07, M09]**	$n\log^{c} n$	$n\log^{c} n$	$\frac{1}{\delta}\log^{c} n$	$O\left(\frac{1}{\delta}\right)$	1
IOPP **[BCF+16]**	$n\log^{c} n$	$> 4\cdot n$	$\frac{1}{\delta}\log^{c} n$	$O\left(\frac{1}{\delta}\right)$	$\log\log n$
This work	$< 6\cdot n$	$< \frac{n}{3}$	$\leq 21\cdot\log n$	$2\log n$	$\frac{\log n}{2}$

Overview

- motivation ✓
- main result, applications
- FRI protocol dive-in

Main Result — Fast RS IOPP (FRI)

Theorem (Informal)

For “nice” RS codes RS $\left[\mathbb{F}, S^{(0)}, \rho\right]$, the FRI protocol satisfies

- $\mathsf{t_p}(n) \leq 6 \cdot n$ and $\ell(n) \leq n/3$
- $\mathsf{t_v}(n) \leq 21 \cdot \log n$ and $\mathsf{q}(n) \leq 2 \log n$
- $\mathsf{r}(n) \leq \frac{1}{2} \log n$ (round complexity)
- soundness (rejection prob.) $\delta - \frac{2n}{|\mathbb{F}|}$ for all $f^{(0)}$ that are $\delta < \delta_0$-far from code, $\delta_0 \approx \frac{1-\rho}{4}$

Main Result — Fast RS IOPP (FRI)

Theorem (Informal)

For "nice" RS codes RS $\left[\mathbb{F}, S^{(0)}, \rho\right]$, the FRI protocol satisfies

- $\mathsf{t_p}(n) \leq 6 \cdot n$ and $\ell(n) \leq n/3$
- $\mathsf{t_v}(n) \leq 21 \cdot \log n$ and $\mathsf{q}(n) \leq 2 \log n$
- $\mathsf{r}(n) \leq \frac{1}{2} \log n$ (round complexity)
- soundness (rejection prob.) $\delta - \frac{2n}{|\mathbb{F}|}$ for all $f^{(0)}$ that are $\delta < \delta_0$-far from code, $\delta_0 \approx \cancel{\frac{1-\rho}{4}}$

Remarks

1. "nice" codes means $S^{(0)}$ is either of following two:
   1.1 2-smooth multiplicative group, i.e., $|S^{(0)}| = 2^k, k \in \mathbb{N}$, or
   1.2 binary additive groups, i.e., $\mathrm{S}^{(0)}$ an $\mathbb{F}_2$-linear space
2. first PCPP/IOPP for RS codes achieving simultaneous
   - linear prover complexity, $\mathsf{t_p} = O(n)$, and
   - sub-linear verifier complexity, $\mathsf{t_v} = o(n)$

Main Result — Fast RS IOPP (FRI)

Theorem (Informal)

For "nice" RS codes RS $\left[\mathbb{F}, S^{(0)}, \rho\right]$, the FRI protocol satisfies

- $\mathsf{t_p}(n) \leq 6 \cdot n$ and $\ell(n) \leq n/3$
- $\mathsf{t_v}(n) \leq 21 \cdot \log n$ and $\mathsf{q}(n) \leq 2 \log n$
- $\mathsf{r}(n) \leq \frac{1}{2} \log n$ (round complexity)
- soundness (rejection prob.) $\delta - \frac{2n}{|\mathbb{F}|}$ for all $f^{(0)}$ that are $\delta < \delta_0$-far from code, $\delta_0 \approx \cancel{\frac{1-\rho}{4}}$ $1 - \rho^{\frac{1}{4}}$ [BGKS19]

Remarks

1. "nice" codes means $S^{(0)}$ is either of following two:
    1.1 2-smooth multiplicative group, i.e., $|S^{(0)}| = 2^k$, $k \in \mathbb{N}$, or
    1.2 binary additive groups, i.e., $\mathrm{S}^{(0)}$ an $\mathbb{F}_2$-linear space
2. first PCPP/IOPP for RS codes achieving simultaneous
    - linear prover complexity, $\mathsf{t_p} = O(n)$, and
    - sub-linear verifier complexity, $\mathsf{t_v} = o(n)$

Main Result — Fast RS IOPP (FRI)

Theorem (Informal)

For “nice” RS codes RS $\left[\mathbb{F}, S^{(0)}, \rho\right]$, the FRI protocol satisfies

- $\mathsf{t_p}(n) \le 6 \cdot n$ and $\ell(n) \le n/3$
- $\mathsf{t_v}(n) \le 21 \cdot \log n$ and $\mathsf{q}(n) \le 2 \log n$
- $\mathsf{r}(n) \le \frac{1}{2} \log n$ (round complexity)
- soundness (rejection prob.) $\delta - \frac{2n}{|\mathbb{F}|}$ for all $f^{(0)}$ that are $\delta < \delta_0$-far from code, $\delta_0 \approx \cancel{\frac{1-\rho}{4}}\ 1 - \rho^{\cancel{\frac{1}{4}}\frac{1}{3}}$ [BGKS19]

Remarks

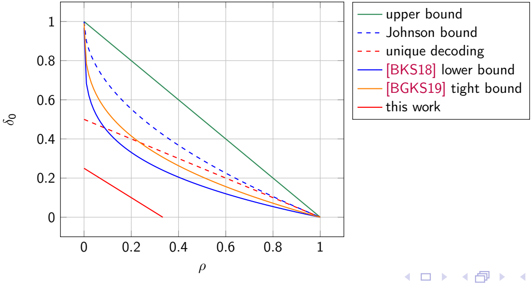
1. “nice” codes means $S^{(0)}$ is either of following two:
   1.1 2-smooth multiplicative group, i.e., $|S^{(0)}| = 2^k, k \in \mathbb{N}$, or
   1.2 binary additive groups, i.e., $\mathrm{S}^{(0)}$ an $\mathbb{F}_2$-linear space
2. first PCPP/IOPP for RS codes achieving simultaneous
   - linear prover complexity, $\mathsf{t_p} = O(n)$, and
   - sub-linear verifier complexity, $\mathsf{t_v} = o(n)$

Main Result — Fast RS IOPP (FRI)

Theorem (Informal)

For "nice" RS codes $\mathrm{RS}\left[\mathbb{F}, S^{(0)}, \rho\right]$, the FRI protocol satisfies

- $\mathsf{t_p}(n) \leq 6 \cdot n$ and $\ell(n) \leq n/3$
- $\mathsf{t_v}(n) \leq 21 \cdot \log n$ and $\mathsf{q}(n) \leq 2 \log n$
- $\mathsf{r}(n) \leq \frac{1}{2} \log n$ (round complexity)
- soundness (rejection prob.) $\delta - \frac{2n}{|\mathbb{F}|}$ for all $f^{(0)}$ that are $\delta < \delta_0$-far from code, $\delta_0 \approx \cancel{\frac{1-\rho}{4}}\ 1 - \rho^{\cancel{\frac{1}{4}}\frac{1}{3}}$ [BGKS19]



FRI applications: (i) computational integrity and (ii) privacy

Definition (Computational Integrity (CI))

is the language of quadruples $(M, \mathcal{T}, x_{\text{in}}, x_{\text{out}})$ such that nondeterministic machine M, on input x_{in} reaches output x_{out} after $\mathcal{T}$ cycles, $\mathcal{T}$ in binary.

FRI applications: (i) computational integrity and (ii) privacy

Definition (Computational Integrity (CI))

is the language of quadruples $(M, \mathcal{T}, x_{\text{in}}, x_{\text{out}})$ such that nondeterministic machine M, on input x_{in} reaches output x_{out} after $\mathcal{T}$ cycles, $\mathcal{T}$ in binary.

Lemma

CI is NEXP-complete

FRI applications: (i) computational integrity and (ii) privacy

Definition (Computational Integrity (CI))

is the language of quadruples $(M, \mathcal{T}, x_{\text{in}}, x_{\text{out}})$ such that nondeterministic machine M, on input x_{in} reaches output x_{out} after $\mathcal{T}$ cycles, $\mathcal{T}$ in binary.

Lemma

CI is NEXP-complete

Definition (proof system)

An proof system S for L is a pair $\mathrm{S} = (\mathrm{V}, \mathrm{P})$ satisfying

- **efficiency** V is randomized polynomial time; P unbounded item **completeness** $x \in L \Rightarrow \Pr\left[\mathrm{V}(x) \leftrightarrow \mathrm{P}(x) \rightsquigarrow \mathrm{acc}\right] = 1$
- **soundness** $x \notin L \Rightarrow \Pr\left[\mathrm{V}(x) \leftrightarrow \mathrm{P}(x) \rightsquigarrow \mathrm{acc}\right] \leq 1/2$

FRI applications: (i) computational integrity and (ii) privacy

Definition (Computational Integrity (CI))

is the language of quadruples $(M, \mathcal{T}, x_{\text{in}}, x_{\text{out}})$ such that nondeterministic machine M, on input x_{in} reaches output x_{out} after $\mathcal{T}$ cycles, $\mathcal{T}$ in binary.

Lemma

CI is NEXP-complete

Definition (argument system)

An argument system S for L is a pair $\mathrm{S} = (\mathrm{V}, \mathrm{P})$ satisfying

- **efficiency** V is randomized polynomial time; P is similarly bounded
- **completeness** $x \in L \Rightarrow \Pr[\mathrm{V}(x) \leftrightarrow \mathrm{P}(x) \rightsquigarrow \mathrm{acc}] = 1$
- **soundness** $x \notin L \Rightarrow \Pr[\mathrm{V}(x) \leftrightarrow \mathrm{P}(x) \rightsquigarrow \mathrm{acc}] \leq 1/2$

FRI applications: (i) computational integrity and (ii) privacy

Definition (Computational Integrity (CI))

is the language of quadruples $(M, \mathcal{T}, x_{\text{in}}, x_{\text{out}})$ such that nondeterministic machine M, on input x_{in} reaches output x_{out} after $\mathcal{T}$ cycles, $\mathcal{T}$ in binary.

Lemma

CI is NEXP-complete

Theorem ([BM88, GMR88, BFL88, BFL91 , BGKW88, FLS90, BFLS91, AS92, ALMSS92, K92, M94])

CI *has an argument system* $\mathsf{S} = (\mathsf{V}, \mathsf{P})$ *that is*

- **succinct:** *Verifier run-time* $\text{poly}(n, \log \mathcal{T})$*; this bounds proof length*
- **transparent (AM):** *verifier sends only public random coins*
- **private (ZK):** *proof preserves privacy of nondeterministic witness*

FRI applications: (i) computational integrity and (ii) privacy

Theorem ([BM88, GMR88, BFL88, BFL91 , BGKW88, FLS90, BFLS91, AS92, ALMSS92, K92, M94])

CI *has an argument system* $\mathsf{S} = (\mathsf{V}, \mathsf{P})$ *that is*

- **succinct:** *Verifier run-time* $\mathrm{poly}(n, \log \mathcal{T})$*; this bounds proof length*
- **transparent (AM):** *verifier sends only public random coins*
- **private (ZK):** *proof preserves privacy of nondeterministic witness*

FRI applications: (i) computational integrity and (ii) privacy

Theorem ([BM88, GMR88, BFL88, BFL91 , BGKW88, FLS90, BFLS91, AS92, ALMSS92, K92, M94])

CI *has an argument system* $\mathsf{S} = (\mathsf{V}, \mathsf{P})$ *that is*

- **succinct:** *Verifier run-time* $\mathrm{poly}(n, \log \mathcal{T})$*; this bounds proof length*
- **transparent (AM):** *verifier sends only public random coins*
- **private (ZK):** *proof preserves privacy of nondeterministic witness*

1. *privacy-preserving* proof of computational integrity
    - Proof and verification time may be longer than $\mathcal{T}$
    - Useful for asserting properties of private, crypto-committed data

FRI applications: (i) computational integrity and (ii) privacy

Theorem ([BM88, GMR88, BFL88, BFL91 , BGKW88, FLS90, BFLS91, AS92, ALMSS92, K92, M94])

CI *has an argument system* $\mathsf{S} = (\mathsf{V}, \mathsf{P})$ *that is*

- **succinct:** *Verifier run-time* $\text{poly}(n, \log \mathcal{T})$*; this bounds proof length*
- **transparent (AM):** *verifier sends only public random coins*
- **private (ZK):** *proof preserves privacy of nondeterministic witness*

1. *privacy-preserving* proof of computational integrity
   - Proof and verification time may be longer than $\mathcal{T}$
   - Useful for asserting properties of private, crypto-committed data
2. *compression* of computation/data, with computational integrity
   - meaningful when $\mathsf{t_v} \ll \mathcal{T}$ or $\ell \ll$ witness-size
   - useful for compressing blockchain history

- Scalable Transparent ARguments of Knowledge [BBHR18]
  - C++ implementation: github.com/elibensasson/libSTARK
  - achieves Thm above, quasi-linear $\mathsf{t_p}$, "post-quantum secure"
  - FRI is a major contributor to STARK efficiency

Overview

- motivation ✓
- main result, applications ✓
- FRI protocol dive-in

Overview of FRI protocol

Theorem (Informal)

For "nice" RS codes RS $\left[\mathbb{F}, S^{(0)}, \rho\right]$, the FRI protocol satisfies

- $\mathsf{t_p}(n) \leq 6 \cdot n$ and $\ell(n) \leq n/3$
- $\mathsf{t_v}(n) \leq 21 \cdot \log n$ and $\mathsf{q}(n) \leq 2 \log n$
- $\mathsf{r}(n) \leq \frac{1}{2} \log n$ (round complexity)
- soundness (rejection prob.) $\delta - \frac{2n}{|\mathbb{F}|}$ for all $f^{(0)}$ that are $\delta < \delta_0$-far from code, $\delta_0 \approx \cancel{\frac{1-\rho}{4}}\ 1 - \rho^{\cancel{\frac{1}{4}}\,\frac{1}{3}}$ [BGKS19]

Recall the inverse Fast Fourier Transform (iFFT)

- evaluate $P(X), \deg(P) < n$ on $\langle\omega\rangle$, ω is root of unity of order $n = 2^k$

Overview of FRI protocol

Theorem (Informal)

For “nice” RS codes RS $\left[\mathbb{F}, S^{(0)}, \rho\right]$, the FRI protocol satisfies

- $\mathsf{t_p}(n) \leq 6 \cdot n$ and $\ell(n) \leq n/3$
- $\mathsf{t_v}(n) \leq 21 \cdot \log n$ and $\mathsf{q}(n) \leq 2 \log n$
- $\mathsf{r}(n) \leq \frac{1}{2} \log n$ (round complexity)
- soundness (rejection prob.) $\delta - \frac{2n}{|\mathbb{F}|}$ for all $f^{(0)}$ that are $\delta < \delta_0$-far from code, $\delta_0 \approx \cancel{\frac{1-\rho}{4}}\ 1 - \rho^{\cancel{\frac{1}{4}}\frac{1}{3}}$ [BGKS19]

Recall the inverse Fast Fourier Transform (iFFT)

- evaluate $P(X), \deg(P) < n$ on $\langle \omega \rangle$, ω is root of unity of order $n = 2^k$
- write $P(X) = P_0(X^2) + X \cdot P_1(X^2)$

Overview of FRI protocol

Theorem (Informal)

For "nice" RS codes RS $\left[\mathbb{F}, S^{(0)}, \rho\right]$, the FRI protocol satisfies

- $\mathsf{t_p}(n) \leq 6 \cdot n$ and $\ell(n) \leq n/3$
- $\mathsf{t_v}(n) \leq 21 \cdot \log n$ and $\mathsf{q}(n) \leq 2 \log n$
- $\mathsf{r}(n) \leq \frac{1}{2} \log n$ (round complexity)
- soundness (rejection prob.) $\delta - \frac{2n}{|\mathbb{F}|}$ for all $f^{(0)}$ that are $\delta < \delta_0$-far from code, $\delta_0 \approx \cancel{\frac{1-\rho}{4}}\ 1 - \rho^{\cancel{\frac{1}{4}} \frac{1}{3}}$ [BGKS19]

Recall the inverse Fast Fourier Transform (iFFT)

- evaluate $P(X), \deg(P) < n$ on $\langle \omega \rangle$, ω is root of unity of order $n = 2^k$
- write $P(X) = P_0(X^2) + X \cdot P_1(X^2)$
- equivalently, $P(X) \equiv P_0(Y) + X \cdot P_1(Y) \mod Y - X^2$

Overview of FRI protocol

Theorem (Informal)

For “nice” RS codes $\mathrm{RS}\left[\mathbb{F}, S^{(0)}, \rho\right]$, the FRI protocol satisfies

- $\mathrm{t_p}(n) \leq 6 \cdot n$ and $\ell(n) \leq n/3$
- $\mathrm{t_v}(n) \leq 21 \cdot \log n$ and $\mathrm{q}(n) \leq 2 \log n$
- $\mathrm{r}(n) \leq \frac{1}{2} \log n$ (round complexity)
- soundness (rejection prob.) $\delta - \frac{2n}{|\mathbb{F}|}$ for all $f^{(0)}$ that are $\delta < \delta_0$-far from code, $\delta_0 \approx \cancel{\frac{1-\rho}{4}}$ $1 - \rho^{\cancel{\frac{1}{4}}\frac{1}{3}}$ [BGKS19]

Recall the inverse Fast Fourier Transform (iFFT)

- evaluate $P(X), \deg(P) < n$ on $\langle \omega \rangle$, ω is root of unity of order $n = 2^k$
- write $P(X) = P_0(X^2) + X \cdot P_1(X^2)$
- equivalently, $P(X) \equiv P_0(Y) + X \cdot P_1(Y) \mod Y - X^2$
- notice $\langle \omega^2 \rangle$ has size $n/2$

Overview of FRI protocol

Theorem (Informal)

For "nice" RS codes RS $\left[\mathbb{F}, S^{(0)}, \rho\right]$, the FRI protocol satisfies

- $\mathsf{t_p}(n) \leq 6 \cdot n$ and $\ell(n) \leq n/3$
- $\mathsf{t_v}(n) \leq 21 \cdot \log n$ and $\mathsf{q}(n) \leq 2 \log n$
- $\mathsf{r}(n) \leq \frac{1}{2} \log n$ (round complexity)
- soundness (rejection prob.) $\delta - \frac{2n}{|\mathbb{F}|}$ for all $f^{(0)}$ that are $\delta < \delta_0$-far from code, $\delta_0 \approx \cancel{\frac{1-\rho}{4}}\ 1 - \rho^{\cancel{\frac{1}{4}}\,\frac{1}{3}}$ [BGKS19]

Recall the inverse Fast Fourier Transform (iFFT)

- evaluate $P(X), \deg(P) < n$ on $\langle\omega\rangle$, ω is root of unity of order $n = 2^k$
- write $P(X) = P_0(X^2) + X \cdot P_1(X^2)$
- equivalently, $P(X) \equiv P_0(Y) + X \cdot P_1(Y) \mod Y - X^2$
- notice $\langle\omega^2\rangle$ has size $n/2$
- so evaluate each of $P_0(Y), P_1(Y)$ on $\langle\omega^2\rangle$, . . . , $O(n \log n)$ runtime

FRI Protocol

- Let $S^{(0)} \subset \mathbb{F}^*$ be 2-smooth mult. group: $|S^{(0)}| = 2^{k^{(0)}}$, $k^{(0)} \in \mathbb{N}$
- Let $f^{(0)} : S^{(0)} \to \mathbb{F}$, FRI for $\mathsf{RS}^{(0)} = \mathsf{RS}\left[\mathbb{F}, S^{(0)}, \rho = \frac{1}{8}\right]$

FRI Protocol

- Let $S^{(0)} \subset \mathbb{F}^*$ be 2-smooth mult. group: $|S^{(0)}| = 2^{k^{(0)}}$, $k^{(0)} \in \mathbb{N}$
- Let $f^{(0)} : S^{(0)} \to \mathbb{F}$, FRI for $\mathsf{RS}^{(0)} = \mathsf{RS}\left[\mathbb{F}, S^{(0)}, \rho = \frac{1}{8}\right]$
- Two-phase protocol
  - COMMIT: while $i < k^{(0)} - \log \frac{1}{\rho}$
    - verifier sends randomness $x^{(i)}$
    - prover sends oracle $f^{(i+1)} : S^{(i+1)} \to \mathbb{F}, |S^{(i+1)}| = |S^{(i)}|/2$

FRI Protocol

- Let $S^{(0)} \subset \mathbb{F}^*$ be 2-smooth mult. group: $|S^{(0)}| = 2^{k^{(0)}}$, $k^{(0)} \in \mathbb{N}$
- Let $f^{(0)} : S^{(0)} \to \mathbb{F}$, FRI for $\mathrm{RS}^{(0)} = \mathrm{RS}\left[\mathbb{F}, S^{(0)}, \rho = \frac{1}{8}\right]$
- Two-phase protocol
  - COMMIT: while $i < k^{(0)} - \log \frac{1}{\rho}$
    - verifier sends randomness $x^{(i)}$
    - prover sends oracle $f^{(i+1)} : S^{(i+1)} \to \mathbb{F}, |S^{(i+1)}| = |S^{(i)}|/2$
    - completeness: If $f^{(i)} \in \mathrm{RS}[\mathbb{F}, S^{(i)}, \rho]$ then $f^{(i+1)} \in \mathrm{RS}[\mathbb{F}, S^{(i+1)}, \rho]$

FRI Protocol

- Let $S^{(0)} \subset \mathbb{F}^*$ be 2-smooth mult. group: $|S^{(0)}| = 2^{k^{(0)}}$, $k^{(0)} \in \mathbb{N}$
- Let $f^{(0)} : S^{(0)} \to \mathbb{F}$, FRI for $\mathrm{RS}^{(0)} = \mathrm{RS}\left[\mathbb{F}, S^{(0)}, \rho = \frac{1}{8}\right]$
- Two-phase protocol
  - COMMIT: while $i < k^{(0)} - \log \frac{1}{\rho}$
    - verifier sends randomness $x^{(i)}$
    - prover sends oracle $f^{(i+1)} : S^{(i+1)} \to \mathbb{F}, |S^{(i+1)}| = |S^{(i)}|/2$
    - completeness: If $f^{(i)} \in \mathrm{RS}[\mathbb{F}, S^{(i)}, \rho]$ then $f^{(i+1)} \in \mathrm{RS}[\mathbb{F}, S^{(i+1)}, \rho]$
    - each entry of $f^{(i+1)}$ computed from 2 distinct entries of $f^{(i)}$ via $O(1)$ arithmetic operations (so $\mathsf{t_p} = O(n)$)

FRI Protocol

- Let $S^{(0)} \subset \mathbb{F}^*$ be 2-smooth mult. group: $|S^{(0)}| = 2^{k^{(0)}}$, $k^{(0)} \in \mathbb{N}$
- Let $f^{(0)} : S^{(0)} \to \mathbb{F}$, FRI for $\mathsf{RS}^{(0)} = \mathsf{RS}\left[\mathbb{F}, S^{(0)}, \rho = \frac{1}{8}\right]$
- Two-phase protocol
  - COMMIT: while $i < k^{(0)} - \log \frac{1}{\rho}$
    - verifier sends randomness $x^{(i)}$
    - prover sends oracle $f^{(i+1)} : S^{(i+1)} \to \mathbb{F}, |S^{(i+1)}| = |S^{(i)}|/2$
    - completeness: If $f^{(i)} \in \mathsf{RS}[\mathbb{F}, S^{(i)}, \rho]$ then $f^{(i+1)} \in \mathsf{RS}[\mathbb{F}, S^{(i+1)}, \rho]$
    - each entry of $f^{(i+1)}$ computed from 2 distinct entries of $f^{(i)}$ via $O(1)$ arithmetic operations (so $\mathsf{t_p} = O(n)$)
    - #rounds $\leq k^{(0)} = \log n$
    - last round ($i = k^{(0)} - \log 1/\rho$): prover sends constant function

FRI Protocol

- Let $S^{(0)} \subset \mathbb{F}^*$ be 2-smooth mult. group: $|S^{(0)}| = 2^{k^{(0)}}$, $k^{(0)} \in \mathbb{N}$
- Let $f^{(0)} : S^{(0)} \to \mathbb{F}$, FRI for $\mathsf{RS}^{(0)} = \mathsf{RS}\left[\mathbb{F}, S^{(0)}, \rho = \frac{1}{8}\right]$
- Two-phase protocol
  - COMMIT: while $i < k^{(0)} - \log \frac{1}{\rho}$
    - verifier sends randomness $x^{(i)}$
    - prover sends oracle $f^{(i+1)} : S^{(i+1)} \to \mathbb{F}, |S^{(i+1)}| = |S^{(i)}|/2$
    - completeness: If $f^{(i)} \in \mathsf{RS}[\mathbb{F}, S^{(i)}, \rho]$ then $f^{(i+1)} \in \mathsf{RS}[\mathbb{F}, S^{(i+1)}, \rho]$
    - each entry of $f^{(i+1)}$ computed from 2 distinct entries of $f^{(i)}$ via $O(1)$ arithmetic operations (so $\mathsf{t_p} = O(n)$)
    - $\#\text{rounds} \leq k^{(0)} = \log n$
    - last round $(i = k^{(0)} - \log 1/\rho)$: prover sends constant function
    - (notice $|f^{(i+1)}| = |f^{(i)}|/2$ so total proof length $O(n)$)
  - QUERY: verifier queries oracles (prover not involved)

Example: $S^{(0)} = \mathbb{F}_{17}^{*}, n = 2^4, \rho = 2^{-2}$

COMMIT phase has $\log|S^{(0)}| - \log \rho = 2$ rounds; during ith round

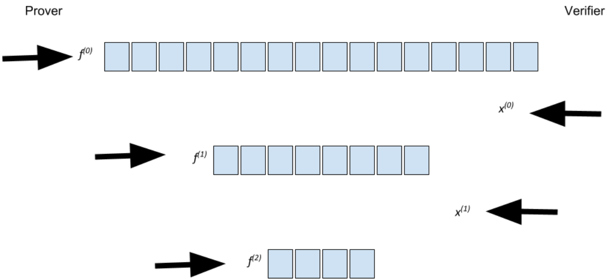
- verifier sends random $x^{(i)} \in \mathbb{F}$
- prover sends next oracle $f^{(i+1)} : S^{(i+1)} \to \mathbb{F}$
    - $S^{(i+1)}$ is 2-smooth multiplicative group, $|S^{(i+1)}| = |S^{(i)}|/2$
    - each entry of $f^{(i+1)}$ computed from 2 distinct entries of $f^{(i)}$
- termination: When $i = k^{(0)} - \log 1/\rho$ prover sends constant function

Example: $S^{(0)} = \mathbb{F}_{17}^*, n = 2^4, \rho = 2^{-2}$

COMMIT phase has $\log|S^{(0)}| - \log\rho = 2$ rounds; during ith round

- verifier sends random $x^{(i)} \in \mathbb{F}$
- prover sends next oracle $f^{(i+1)} : S^{(i+1)} \to \mathbb{F}$
  - $S^{(i+1)}$ is 2-smooth multiplicative group, $|S^{(i+1)}| = |S^{(i)}|/2$
  - each entry of $f^{(i+1)}$ computed from 2 distinct entries of $f^{(i)}$
- termination: When $i = k^{(0)} - \log 1/\rho$ prover sends constant function

QUERY phase: pick random $s^{(0)} \in S^{(0)}$ and check path-to-root

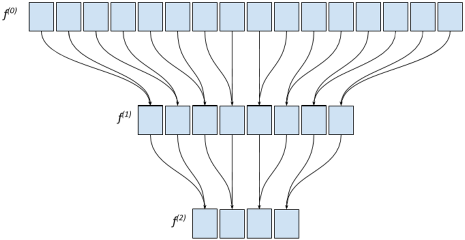


Example: $S^{(0)} = \mathbb{F}_{17}^{*}, n = 2^4, \rho = 2^{-2}$

COMMIT phase has $\log|S^{(0)}| - \log\rho = 2$ rounds; during ith round

- verifier sends random $x^{(i)} \in \mathbb{F}$
- prover sends next oracle $f^{(i+1)} : S^{(i+1)} \to \mathbb{F}$
  - $S^{(i+1)}$ is 2-smooth multiplicative group, $|S^{(i+1)}| = |S^{(i)}|/2$
  - each entry of $f^{(i+1)}$ computed from 2 distinct entries of $f^{(i)}$
- termination: When $i = k^{(0)} - \log 1/\rho$ prover sends constant function

QUERY phase: pick random $s^{(0)} \in S^{(0)}$ and check path-to-root

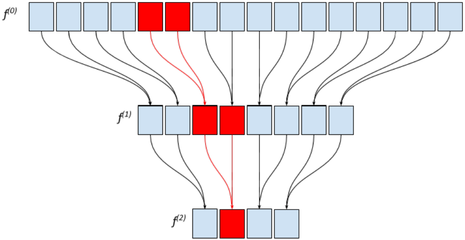


Example: $S^{(0)} = \mathbb{F}_{17}^*, n = 2^4, \rho = 2^{-2}$

COMMIT phase has $\log|S^{(0)}| - \log \rho = 2$ rounds; during ith round

- verifier sends random $x^{(i)} \in \mathbb{F}$
- prover sends next oracle $f^{(i+1)} : S^{(i+1)} \to \mathbb{F}$
    - $S^{(i+1)}$ is 2-smooth multiplicative group, $|S^{(i+1)}| = |S^{(i)}|/2$
    - each entry of $f^{(i+1)}$ computed from 2 distinct entries of $f^{(i)}$
- termination: When $i = k^{(0)} - \log 1/\rho$ prover sends constant function

QUERY phase: pick random $s^{(0)} \in S^{(0)}$ and check path-to-root



FRI COMMIT — single round

- suppose $f^{(0)} : \mathbb{F}_{17}^{*} \to \mathbb{F}_{17}$ satisfies $\deg(f^{(0)}) < 4$

FRI COMMIT — single round

- suppose $f^{(0)} : \mathbb{F}_{17}^{*} \rightarrow \mathbb{F}_{17}$ satisfies $\deg(f^{(0)}) < 4$
- let $P(X)$ interpolate $f^{(0)}$, $\deg(P) < 4$

FRI COMMIT — single round

- suppose $f^{(0)} : \mathbb{F}_{17}^* \to \mathbb{F}_{17}$ satisfies $\deg(f^{(0)}) < 4$
- let $P(X)$ interpolate $f^{(0)}$, $\deg(P) < 4$
- write $P(X) = P_0(X^2) + X \cdot P_1(X^2)$, FFT-style

FRI COMMIT — single round

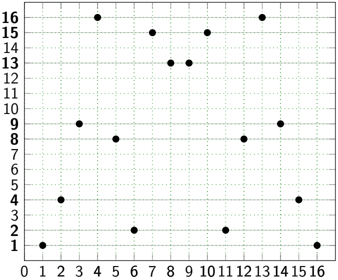
- suppose $f^{(0)} : \mathbb{F}_{17}^{*} \to \mathbb{F}_{17}$ satisfies $\deg(f^{(0)}) < 4$
- let $P(X)$ interpolate $f^{(0)}$, $\deg(P) < 4$
- write $P(X) = P_0(X^2) + X \cdot P_1(X^2)$, FFT-style
- then $P(X) \equiv P_0(Y) + X \cdot P_1(Y) \mod Y - X^2$

FRI COMMIT — single round

- suppose $f^{(0)} : \mathbb{F}_{17}^{*} \to \mathbb{F}_{17}$ satisfies $\deg(f^{(0)}) < 4$
- let $P(X)$ interpolate $f^{(0)}$, $\deg(P) < 4$
- write $P(X) = P_0(X^2) + X \cdot P_1(X^2)$, FFT-style
- then $P(X) \equiv P_0(Y) + X \cdot P_1(Y) \mod Y - X^2$
- let $Q(X, Y) \triangleq P_0(Y) + X \cdot P_1(Y)$,
    - $Q(X, Y) \equiv P(X) \mod Y - X^2$

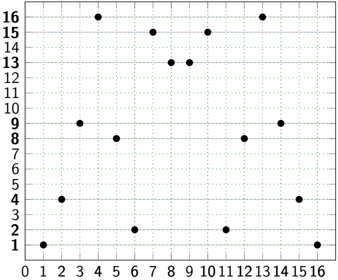
FRI COMMIT — single round

- suppose $f^{(0)} : \mathbb{F}_{17}^{*} \to \mathbb{F}_{17}$ satisfies $\deg(f^{(0)}) < 4$
- let $P(X)$ interpolate $f^{(0)}$, $\deg(P) < 4$
- write $P(X) = P_0(X^2) + X \cdot P_1(X^2)$, FFT-style
- then $P(X) \equiv P_0(Y) + X \cdot P_1(Y) \mod Y - X^2$
- let $Q(X, Y) \triangleq P_0(Y) + X \cdot P_1(Y)$,
  - $Q(X, Y) \equiv P(X) \mod Y - X^2$
  - consider points in $\mathbb{F} \times \mathbb{F}$ on curve $Y - X^2$,



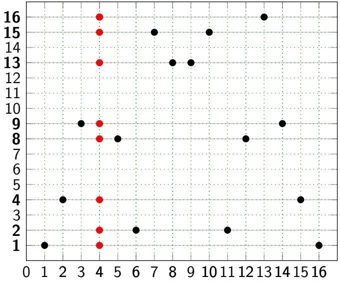
FRI COMMIT — single round

- suppose $f^{(0)} : \mathbb{F}_{17}^{*} \to \mathbb{F}_{17}$ satisfies $\deg(f^{(0)}) < 4$
- let $P(X)$ interpolate $f^{(0)}$, $\deg(P) < 4$
- write $P(X) = P_0(X^2) + X \cdot P_1(X^2)$, FFT-style
- then $P(X) \equiv P_0(Y) + X \cdot P_1(Y) \mod Y - X^2$
- let $Q(X, Y) \triangleq P_0(Y) + X \cdot P_1(Y)$,
  - $Q(X, Y) \equiv P(X) \mod Y - X^2$
  - $\deg_X(Q) < 2$; $\deg_Y(Q) \leq \deg(P)/2$
  - $S^{(1)} = \left\{x^2 \mid x \in S^{(0)}\right\}$ is mult. group, $|S^{(1)}| = |S^{(0)}|/2$



FRI COMMIT — single round

- suppose $f^{(0)} : \mathbb{F}_{17}^{*} \to \mathbb{F}_{17}$ satisfies $\deg(f^{(0)}) < 4$
- let $P(X)$ interpolate $f^{(0)}$, $\deg(P) < 4$
- write $P(X) = P_0(X^2) + X \cdot P_1(X^2)$, FFT-style
- then $P(X) \equiv P_0(Y) + X \cdot P_1(Y) \mod Y - X^2$
- let $Q(X, Y) \triangleq P_0(Y) + X \cdot P_1(Y)$,
  - $Q(X, Y) \equiv P(X) \mod Y - X^2$
  - $\deg_X(Q) < 2$; $\deg_Y(Q) \leq \deg(P)/2$
  - $S^{(1)} = \left\{x^2 \mid x \in S^{(0)}\right\}$ is mult. group, $|S^{(1)}| = |S^{(0)}|/2$

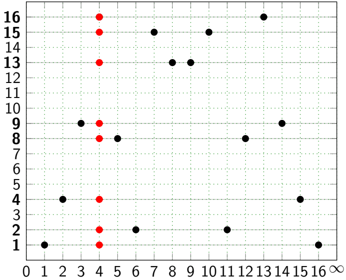


COMMIT round

- Verifier picks random $x^{(0)} \in \mathbb{F}$
- $f^{(1)} = Q(x^{(0)}, Y)|_{S^{(1)}}$
- each entry of $f^{(1)}$ interpolated from two entries of $f^{(0)}$
- $\deg(f^{(1)}) = \deg_Y(Q) < \rho|S^{(1)}|$

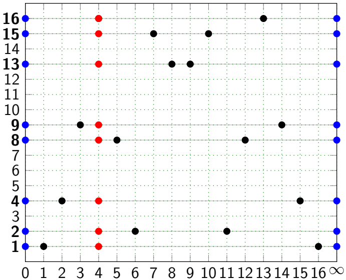
FRI vs. inverse FFT

- suppose $f^{(0)} : \mathbb{F}_{17}^{*} \to \mathbb{F}_{17}$ satisfies $\deg(f^{(0)}) < 4$
- find $P(X)$ that interpolates $f^{(0)}$
- write $P(X) = P_0(X^2) + X \cdot P_1(X^2)$, FFT-style
- then $P(X) \equiv P_0(Y) + X \cdot P_1(Y) \mod Y - X^2$
- let $Q(X, Y) \triangleq P_0(Y) + X \cdot P_1(Y)$,
  - $Q(X, Y) \equiv P(X) \mod Y - X^2$
  - $\deg_X(Q) < 2$; $\deg_Y(Q) \leq \deg(P)/2$
  - $S^{(1)} = \left\{ x^2 \mid x \in S^{(0)} \right\}$ is mult. group, $|S^{(1)}| = |S^{(0)}|/2$



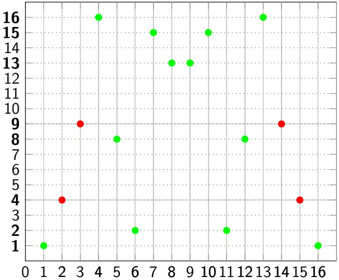
FRI vs. inverse FFT

- suppose $f^{(0)} : \mathbb{F}_{17}^{*} \to \mathbb{F}_{17}$ satisfies $\deg(f^{(0)}) < 4$
- find $P(X)$ that interpolates $f^{(0)}$
- write $P(X) = P_0(X^2) + X \cdot P_1(X^2)$, FFT-style
- then $P(X) \equiv P_0(Y) + X \cdot P_1(Y) \mod Y - X^2$
- let $Q(X, Y) \triangleq P_0(Y) + X \cdot P_1(Y)$,
  - $Q(X, Y) \equiv P(X) \mod Y - X^2$
  - $\deg_X(Q) < 2$; $\deg_Y(Q) \leq \deg(P)/2$
  - $S^{(1)} = \left\{ x^2 \mid x \in S^{(0)} \right\}$ is mult. group, $|S^{(1)}| = |S^{(0)}|/2$



- $P_0(Y) = Q(0, Y)$,
  $P_1(Y) = Q(\infty, Y)$
- let $g_0 = Q(0, Y)|_{S^{(1)}}$,
  $g_1 = Q(\infty, Y)|_{S^{(1)}}$
- compute g_0, g_1, $O(n)$ steps
- recurse on g_0, g_1

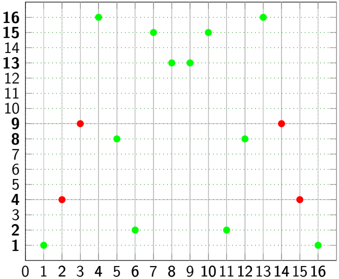
Soundness analysis — low error



For simplicity, suppose $f^{(0)}$ is $\delta < \frac{1-\rho}{4}$-far from $\mathbf{0}$

- $y \in S^{(1)}$ good if $f^{(0)}(x_0) = f^{(0)}(x_1) = 0$ for $x_0^2 = x_1^2 = y$
- otherwise, $y \in S^{(1)}$ bad
- fraction of bad y's in $S^{(1)}$ between δ and 2δ

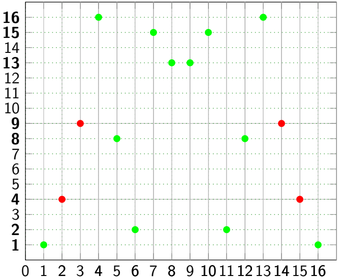
Soundness analysis — low error



For simplicity, suppose $f^{(0)}$ is $\delta < \frac{1-\rho}{4}$-far from $\mathbf{0}$

- $y \in S^{(1)}$ good if $f^{(0)}(x_0) = f^{(0)}(x_1) = 0$ for $x_0^2 = x_1^2 = y$
- otherwise, $y \in S^{(1)}$ bad
- fraction of bad y's in $S^{(1)}$ between δ and 2δ
- interpolant of bad row has at most 1 root

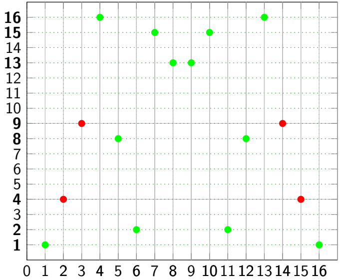
Soundness analysis — low error



For simplicity, suppose $f^{(0)}$ is $\delta < \frac{1-\rho}{4}$-far from $\mathbf{0}$

- $y \in S^{(1)}$ good if $f^{(0)}(x_0) = f^{(0)}(x_1) = 0$ for $x_0^2 = x_1^2 = y$
- otherwise, $y \in S^{(1)}$ bad
- fraction of bad y's in $S^{(1)}$ between δ and 2δ
- interpolant of bad row has at most 1 root
- w.p. $1 - \frac{|S^{(1)}|}{|\mathbb{F}|}$, $x^{(0)}$ misses roots of bad rows; call such $x^{(0)}$ good

Soundness analysis — low error



For simplicity, suppose $f^{(0)}$ is $\delta < \frac{1-\rho}{4}$-far from $\mathbf{0}$

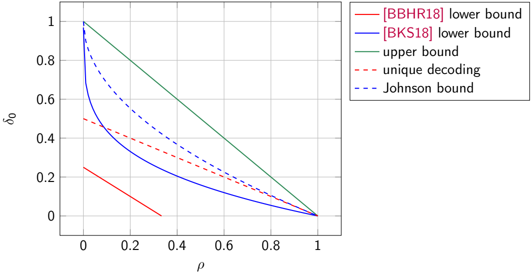
- $y \in S^{(1)}$ good if $f^{(0)}(x_0) = f^{(0)}(x_1) = 0$ for $x_0^2 = x_1^2 = y$
- otherwise, $y \in S^{(1)}$ bad
- fraction of bad y's in $S^{(1)}$ between δ and 2δ
- interpolant of bad row has at most 1 root
- w.p. $1 - \frac{|S^{(1)}|}{|\mathbb{F}|}$, $x^{(0)}$ misses roots of bad rows; call such $x^{(0)}$ good
- prover left with two bad options:
  - let $f^{(1)}$ “jump” to be closer to non-zero RS-codeword; large error;
  - continue with $f^{(1)}$ close to $\mathbf{0}$;

Summary

- first RPT solution with $t_p = O(n)$ and $t_v = O(\log n)$
- nearly optimal soundness for $\delta < \delta_0$

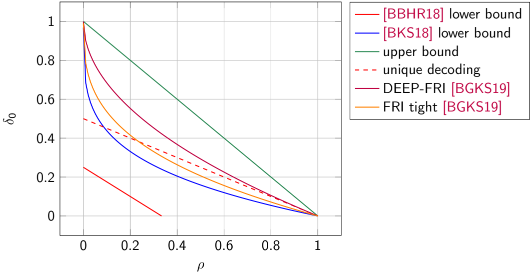
Summary

- first RPT solution with $t_p = O(n)$ and $t_v = O(\log n)$
- nearly optimal soundness for $\delta < \delta_0$
- what's δ_0? (higher lines are better)



Summary

- first RPT solution with $\mathsf{t_p} = O(n)$ and $\mathsf{t_v} = O(\log n)$
- nearly optimal soundness for $\delta < \delta_0$
- what's δ_0? (higher lines are better)



- New protocol: DEEP-FRI [B, Goldberg, Kopparty, Saraf 2019]
  - DEEP-FRI: Domain Extending for Eliminating Pretenders FRI
  - like FRI, has linear proving complexity, logarithmic verifer complexity
  - DEEP-FRI soundness reaches Johnson bound $\delta_0 \approx 1 - \sqrt{\rho}$
  - Under plausible list decoding conjecture, reaches $\delta_0 \approx 1 - \rho$

Summary

- first RPT solution with $\mathsf{t_p} = O(n)$ and $\mathsf{t_v} = O(\log n)$
- nearly optimal soundness for $\delta < \delta_0$
- what's δ_0? (higher lines are better)

- Questions
  - "sliding scale" soundness-error $\approx 1/\text{poly}(|\mathbb{F}|)$ for RS-IOPPs?

Summary

- first RPT solution with $\mathrm{t_p} = O(n)$ and $\mathrm{t_v} = O(\log n)$
- nearly optimal soundness for $\delta < \delta_0$
- what's δ_0? (higher lines are better)

- Questions
    - "sliding scale" soundness-error $\approx 1/\mathrm{poly}(|\mathbb{F}|)$ for RS-IOPPs?
    - want to learn more? workshop@starkware.co
    - want to realize in practice? jobs@starkware.co

STARKWARE