

Notes on FRI

arnaucube

February 2023

Abstract

Notes taken from Vincenzo Iovino explanations about FRI [1], [2].

These notes are for self-consumption, are not complete, don't include all the steps neither all the proofs.

An implementation of FRI can be found at <https://github.com/arnaucube/fri-commitment>.

Contents

| | |
|---------------------------------------|----------|
| 1 Preliminaries | 1 |
| 1.1 Low degree testing | 1 |
| 1.1.1 General degree d test | 1 |
| 2 FRI protocol | 2 |
| 2.1 Intuition | 2 |
| 2.2 FRI | 2 |
| 3 FRI as polynomial commitment | 5 |

1 Preliminaries

1.1 Low degree testing

V wants to ensure that $\deg(f(x)) \leq d$.

We are in the IOP setting, V asks on a point, P sends back the opening at that point.

TODO

1.1.1 General degree d test

Query at points $\{x_i\}_0^{d+1}$, z (with $\text{rand } z \in \mathbb{F}$). Interpolate $p(x)$ at $\{f(x_i)\}_0^{d+1}$ to reconstruct the unique polynomial p of degree d such that $p(x_i) = f(x_i) \forall i = 1, \dots, d+1$.

V checks $p(z) = f(z)$, if the check passes, then V is convinced with high probability.

This needs $d + 2$ queries, is linear, $\mathcal{O}(n)$. With FRI we will have the test in $\mathcal{O}(\log d)$.

2 FRI protocol

Allows to test if a function f is a poly of degree $\leq d$ in $\mathcal{O}(\log d)$.

Note: "P sends $f(x)$ to V", "sends", in the ideal IOP model means that all the table of $f(x)$ is sent, in practice is sent a commitment to $f(x)$.

2.1 Intuition

V wants to check that two functions g, h are both polynomials of degree $\leq d$.

Consider the following protocol:

1. V sends $\alpha \in \mathbb{F}$ to P. P sends $f(x) = g(x) + \alpha h(x)$ to V.
2. P sends $f(x) = g(x) + \alpha h(x)$ to V.
3. V queries $f(r), g(r), h(r)$ for rand $r \in \mathbb{F}$.
4. V checks $f(r) = g(r) + \alpha h(r)$. (Schwartz-Zippel lema). If holds, V can be certain that $f(x) = g(x) + \alpha h(x)$.
5. P proves that $\deg(f) \leq d$.
6. If V is convinced that $\deg(f) \leq d$, V believes that both g, h have $\deg \leq d$.

With high probability, α will not cancel the coeffs with $\deg \geq d + 1$.

Let $g(x) = a \cdot x^{d+1}$, $h(x) = b \cdot x^{d+1}$, and set $f(x) = g(x) + \alpha h(x)$. Imagine that P can chose α such that $ax^{d+1} + \alpha \cdot bx^{d+1} = 0$, then, in $f(x)$ the coefficients of degree $d + 1$ would cancel.

Here, P proves g, h both have $\deg \leq d$, but instead of doing $2 \cdot (d + 2)$ queries ($d + 2$ for g , and $d + 2$ for h), it is done in $d + 2$ queries (for f). So we halved the number of queries.

2.2 FRI

Both P and V have oracle access to function f .

V wants to test if f is polynomial with $\deg(f) \leq d$.

Let $f_0(x) = f(x)$.

Each polynomial $f(x)$ of degree that is a power of 2, can be written as

$$f(x) = f^L(x^2) + x f^R(x^2)$$

for some polynomials f^L , f^R of degree $\frac{\deg(f)}{2}$, each one containing the even and odd degree coefficients as follows:

$$f^L(x) = \sum_0^{\frac{d+1}{2}-1} c_{2i}x^i, \quad f^R(x) = \sum_0^{\frac{d+1}{2}-1} c_{2i+1}x^i$$

eg. for $f(x) = x^4 + x^3 + x^2 + x + 1$,

$$\left. \begin{aligned} f^L(x) &= x^2 + x + 1 \\ f^R(x) &= x + 1 \end{aligned} \right\} \begin{aligned} f(x) &= f^L(x^2) + x \cdot f^R(x^2) \\ &= (x^2)^2 + (x^2) + 1 + x \cdot ((x^2) + 1) \\ &= x^4 + x^2 + 1 + x^3 + x \end{aligned}$$

Proof generation P starts from $f(x)$, and for $i = 0$ sets $f_0(x) = f(x)$.

1. $\forall i \in \{0, \log(d)\}$, with $d = \deg f(x)$,
P computes $f_i^L(x)$, $f_i^R(x)$ for which

$$f_i(x) = f_i^L(x^2) + x f_i^R(x^2) \quad (\text{eq. } A_i)$$

holds.

2. V sends challenge α_i
3. P commits to the random linear combination f_{i+1} , for

$$f_{i+1}(x) = f_i^L(x) + \alpha_i f_i^R(x) \quad (\text{eq. } B_i)$$

4. P sets $f_i(x) := f_{i+1}(x)$ and starts again the iteration.

Notice that at each step, $\deg(f_i)$ halves.

This is done until the last step, where $f_i^L(x)$, $f_i^R(x)$ are constant (degree 0 polynomials). For which P does not commit but gives their values directly to V.

Data sent from P to V

Commitments: $\{Comm(f_i)\}_0^{\log(d)}$
eg. $\{Comm(f_0), Comm(f_1), Comm(f_2), \dots, Comm(f_{\log(d)})\}$

Openings: $\{f_i(z^{2^i}), f_i(-(z^{2^i}))\}_0^{\log(d)}$
for a challenge $z \in \mathbb{F}$ set by V

eg. $f_0(z), f_0(-z), f_1(z^2), f_1(-z^2), f_2(z^4), f_2(-z^4), f_3(z^8), f_3(-z^8), \dots$

Constant values of last iteration: $\{f_k^L, f_k^R\}$, for $k = \log(d)$

Verification V receives:

Commitments: $Comm(f_i), \forall i \in \{0, \log(d)\}$

Openings: $\{o_i, o'_i\} = \{f_i(z^{2^i}), f_i(-(z^{2^i}))\}, \forall i \in \{0, \log(d)\}$

Constant vals: $\{f_k^L, f_k^R\}$

For all $i \in \{0, \log(d)\}$, V knows the openings at z^{2^i} and $-(z^{2^i})$ for $Comm(f_i(x))$, which are $o_i = f_i(z^{2^i})$ and $o'_i = f_i(-(z^{2^i}))$ respectively.

V, from (eq. A_i), knows that

$$f_i(x) = f_i^L(x^2) + x f_i^R(x^2)$$

should hold, thus

$$f_i(z) = f_i^L(z^2) + z f_i^R(z^2)$$

where $f_i(z)$ is known, but $f_i^L(z^2)$, $f_i^R(z^2)$ are unknown. But, V also knows the value for $f_i(-z)$, which can be represented as

$$f_i(-z) = f_i^L(z^2) - z f_i^R(z^2)$$

(note that when replacing x by $-z$, it loses the negative in the power, not in the linear combination).

Thus, we have the system of independent linear equations

$$\begin{aligned} f_i(z) &= f_i^L(z^2) + z f_i^R(z^2) \\ f_i(-z) &= f_i^L(z^2) - z f_i^R(z^2) \end{aligned}$$

for which V will find the value of $f_i^L(z^{2^i})$, $f_i^R(z^{2^i})$. Equivalently it can be represented by

$$\begin{pmatrix} 1 & z \\ 1 & -z \end{pmatrix} \begin{pmatrix} f_i^L(z^2) \\ f_i^R(z^2) \end{pmatrix} = \begin{pmatrix} f_i(z) \\ f_i(-z) \end{pmatrix}$$

where V will find the values of $f_i^L(z^{2^i})$, $f_i^R(z^{2^i})$ being

$$\begin{aligned} f_i^L(z^{2^i}) &= \frac{f_i(z) + f_i(-z)}{2} \\ f_i^R(z^{2^i}) &= \frac{f_i(z) - f_i(-z)}{2z} \end{aligned}$$

Once, V has computed $f_i^L(z^{2^i})$, $f_i^R(z^{2^i})$, can use them to compute the linear combination of

$$f_{i+1}(z^2) = f_i^L(z^2) + \alpha_i f_i^R(z^2)$$

obtaining then $f_{i+1}(z^2)$. This comes from (eq. B_i).

Now, V checks that the obtained $f_{i+1}(z^2)$ is equal to the received opening $o_{i+1} = f_{i+1}(z^2)$ from the commitment done by P. V checks also the commitment of $Comm(f_{i+1}(x))$ for the opening $o_{i+1} = f_{i+1}(z^2)$. If the checks pass, V is convinced that $f_1(x)$ was committed honestly.

Now, sets $i := i + 1$ and starts a new iteration.

For the last iteration, V checks that the obtained $f_i^L(z^{2^i})$, $f_i^R(z^{2^i})$ are equal to the constant values $\{f_k^L, f_k^R\}$ received from P.

It needs $\log(d)$ iterations, and the number of queries (commitments + openings sent and verified) needed is $2 \cdot \log(d)$.

3 FRI as polynomial commitment

[WIP. Unfinished document]

References

- [1] Eli Ben-Sasson, Iddo Bentov, Yinon Horesh, and Michael Riabzev. Fast reed-solomon interactive oracle proofs of proximity, 2018. <https://eccc.weizmann.ac.il/report/2017/134/>.
- [2] Ulrich Haböck. A summary on the fri low degree test. Cryptology ePrint Archive, Paper 2022/1216, 2022. <https://eprint.iacr.org/2022/1216>.