FRI Lecture 10/10/24

Credit: LOTS if content is stolen from Professor Chiesa course

Lesson Plan

1. Background and Motivation
2. Protocol Construction
3. Soundness Overview
4. Extensions/Further Work

} Focus

Goal: Teach Intuition

This is a very rich topic.

1a Motivation (as pertaining to information-theoretic proof systems)

So far

- We have discussed various PCP constructions for NP
  - for exponential-sized PCPs we used linear extensions (Hadamard Code) $NP \subseteq PCP[poly, O(1)]$
  - LPCP $\longrightarrow$ PCP
    - P_{PCP} would write down every query answer
    - V_{PCP} would read its query answers and run V_{LPCP}
    - V_{PCP} would run a linearity test on π_{PCP}

Recall in LPCP verifier has query access to a linear function $q \mapsto \langle \pi, q \rangle$

- for poly-sized PCPs we used multivariate Low-Degree Extensions (Reed-Muller Codes)
  - We chose NP-Complete Language QuadEq
  - Prover low-degree extends assignment vector $\vec{w}$ into $\hat{w}$
  - Verifier runs sumcheck to test that $\hat{w}$ satisfies the QuadEq instance
  - Verifier runs Low-Degree Test to ensure $\hat{w}$ is well-formed

$NP \subseteq PCP[\text{poly}(n), \text{polylog}(n)]$

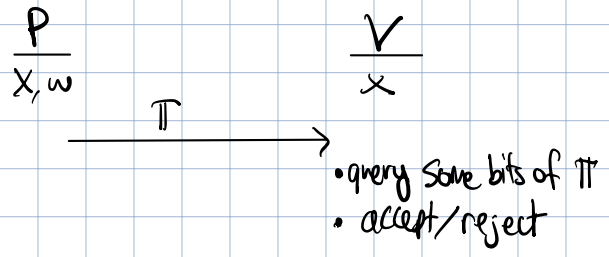
Now

- What about linear-sized PCPs for NP? (linear in $(\vec{x, w})$)
  - we can hope for linear-sized PCPs for some NP-complete language at best
  - eg R1CS (a relaxation of QuadEq still NP-Complete)

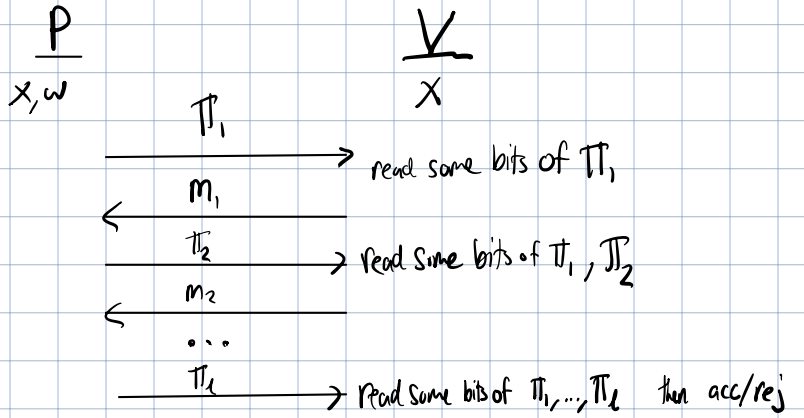
Detour

Defn IOP (Interactive Oracle Proof) is a generalization of PCP to multiple rounds [BCS 16]

PCP



IOP (k-rounds)



Define size of IOP proof as $\sum_{i=1}^{\ell} |\pi_i|$

Thm: $\exists$ linear-sized IOP $= (P, V)$ protocol for RICS

Key ingredients

- univariate low-degree extensions of witness vector $\vec{w} = (w_0, w_1, \ldots, w_m)$
  - i.e. vector of evaluations of univariate polynomial $w(x)$ of low degree st $w(i) = w_i \quad \forall i \in \{0, \ldots, m\}$
  - also known as a Reed-Solomon Codeword (definition to come)
- IOP for univariate sumcheck
- IOP for univariate low-degree test

(Jumping Ahead)

In other words, we want an IOP to test proximity to a Reed-Solomon Code

FRI = Fast Reed-Solomon IOPs of Proximity

1b Background (warning, some linear algebra necessary)

Defn: A linear code (over field $\mathbb{F}$) is a subspace of $\mathbb{F}^n$ (where $\mathbb{F}^n$ = n-dimensional $\mathbb{F}$-vector space)

Say $C \subseteq \mathbb{F}^n$ is our code of dimension K with basis vectors $\vec{c}_0, \ldots, \vec{c}_{K-1} \in \mathbb{F}^n$

$\exists$ Enc: $\mathbb{F}^K \hookrightarrow \mathbb{F}^n$

$$(m_0, m_1, \ldots, m_{K-1}) \longmapsto (m_0 \vec{c}_0 + \cdots + m_{K-1} \vec{c}_{K-1})$$

st
- Enc is injective
- $C = \text{im}(\text{Enc})$

Jargon

K: msg length

n: block length

Notes: $0 \in C$

$\forall \vec{x}, \vec{y} \in C \quad \forall a, b \in \mathbb{F}, \quad a\vec{x} + b\vec{y} \in C$

(notably $x - y \in C$)

Defn: Hamming Distance function $\Delta : \mathbb{F}^n \times \mathbb{F}^n \longrightarrow \{0, \dots, n\}$

$$(\vec{x}, \vec{y}) \longmapsto \left| \{ i \mid i \in [n] \wedge x_i \neq y_i \} \right|$$

(This is a metric, and obeys triangle inequality)

Defn: Distance of linear code is $\min_{x,y \in C} \{ \Delta(x,y) \}$ or equivalently $\min_{x \in C} \{ \Delta(x,0) \}$

(minimum weight codeword)

Defn: Reed-Solomon Codes

- Fix $n \in \mathbb{N}$, Fix domain $L = \{ x_0, \dots, x_{n-1} \} \subseteq \mathbb{F}_q$
- We define RS code of msg length $K \leq n$ and block length n as

$$C = RS[n, K]_q := \left\{ \left(P(x_0), P(x_1), \dots, P(x_{n-1}) \right) \;\middle|\; P(x) \in \mathbb{F}_q[x]^{<K} \right\}$$

Suppose $P(x) = a_0 + a_1 x + \cdots + a_{K-1} x^{K-1}$, $Q(x) = b_0 + b_1 x + \cdots + b_{K-1} x^{K-1}$

then $Enc(a_0, \dots, a_{K-1}) \longrightarrow \left(P(x_0), \dots, P(x_{n-1}) \right) =: C_P$

$Enc(b_0, \dots, b_{K-1}) \longrightarrow \left(Q(x_0), \dots, Q(x_{n-1}) \right) =: C_Q$

If $P \neq Q$ then $\Delta(C_P, C_Q) \geq n - (k-1) = n-k+1$

↑ can agree on at most $k-1$ evaluations

Thus min distance of C is $n-k+1$

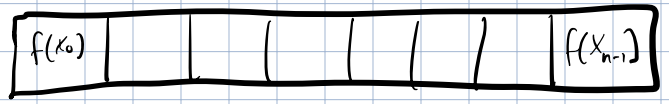
Fact: Reed-Solomon Codes are linear-codes

Defn Rate of a code $\rho := \frac{k}{n}$

Defn: Distance of word w to code C

$$\Delta(w, C) := \min_{c \in C} \{ \Delta(w, c) \}$$

Can think of any word in $\mathbb{F}^n$ as a function $f: L \to \mathbb{F}$

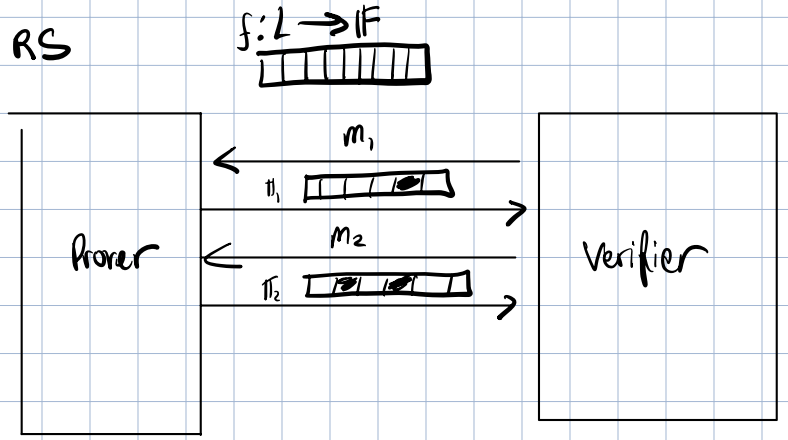


$\equiv$ $f: L \to \mathbb{F}$ where $L = \{x_0, \ldots, x_{n-1}\}$

2a Initial Protocol Attempts:

- Given some $f: L \to \mathbb{F}$ equivalently some vector $\vec{f} = (f(x))_{x \in L}$ where $|L| = n$
- Want proximity test for $RS[n, K]$ (i.e. $\exists P \in \mathbb{F}[x]^{<K}$ s.t. $P(x) = f(x)\ \forall x \in L$)
  - ✱ completeness: if $f \in RS[n,K]$, tester always accepts
  - ✱ "Soundness": if f is δ-far from $RS[n,d]$, tester accepts w.p. $\leq \varepsilon(\delta)$

IOPP for RS



want better than querying $K+1$ points b/c $K = \Theta(n)$ in many applications

Want LDPP for $RS[N,K]_q$ (assumptions $\mathbb{F} = \mathbb{Z}/q\mathbb{Z}$ for some prime q; $|L| = N = 2^n$ for some $n \in \mathbb{N}$, $K = 2^k$ for some $k \in \mathbb{N}$)

Instance is $f: L \to \mathbb{F}$

What is L? we can choose $L \subseteq \mathbb{F}$ so long as $|L| = N = 2^n$

let $L = \langle w \rangle$ where w is some order N element in $\mathbb{F}_q^*$

(exists iff $N \mid \phi(q)$ where $\phi(q) = q-1$)

ex) $q = 2^{64} - 2^{32} + 1$ can use $N = 2^{32}$

$$L = \{1, w, w^2, w^3, \ldots, w^{N-1}\}$$

Define $L^2 = \{x^2 \mid x \in L\}$

$$L^2 = \{1, w^2, w^4, \ldots, w^{\frac{N}{2}-1}\} = \langle w^2 \rangle$$

↳ order $\frac{N}{2}$

Observe: Given $P(X) \in \mathbb{F}_q[X]^{<K}$ s.t. $f(x) = P(x)\ \forall x \in L$

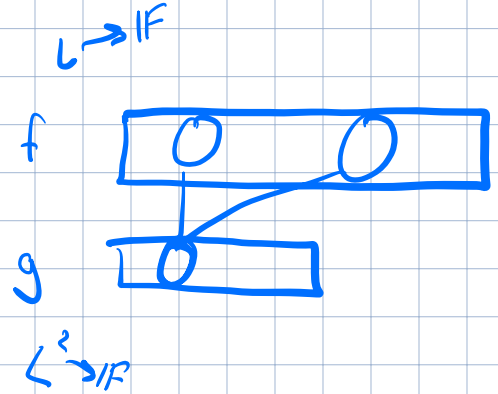
Can find polynomials $P_{even}(X), P_{odd}(X) \in \mathbb{F}_q[X]^{<K/2}$

such that

$$P(X) = P_{even}(X^2) + X \cdot P_{odd}(X^2)$$

let $g, h: L^2 \to \mathbb{F}$ st

$g(x) := P_{even}(x)$ $h(x) := P_{odd}(x)$ $\forall x \in L^2$



ex// $P(x) \equiv 1 + 2x + 3x^2 + 4x^3 + 5x^4 + 6x^5$

$P_{even}(x) \equiv 1 + 3x + 5x^2$, $P_{odd}(x) \equiv 2 + 4x + 6x^2$

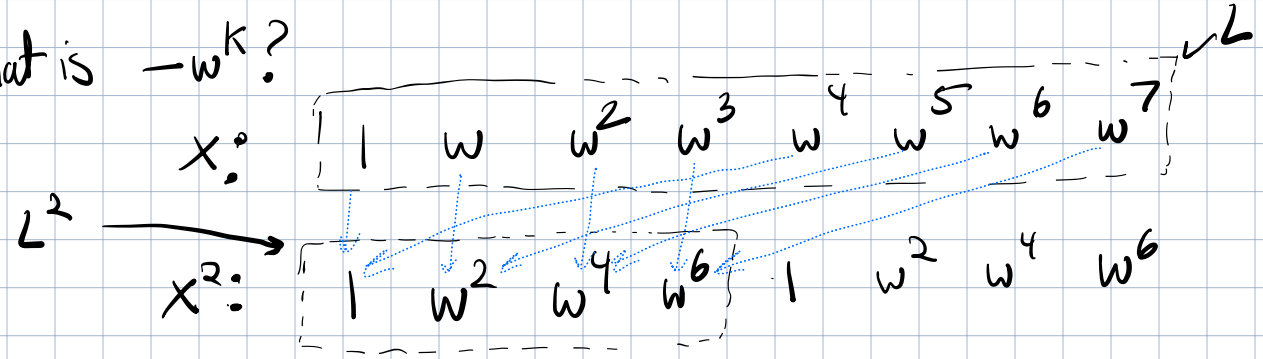
$P_{even}(x^2) \equiv 1 + 3x^2 + 5x^4$, $P_{odd}(x^2) \equiv 2 + 4x^2 + 6x^4$

$$P_{even}(x^2) \equiv \frac{P(x) + P(-x)}{2} \qquad P_{odd}(x^2) \equiv \frac{P(x) - P(-x)}{2x}$$

$\forall w^{2k} \in L^2$

$$g(w^{2k}) = \frac{f(w^k) + f(-w^k)}{2} \qquad h(w^{2k}) = \frac{f(w^k) - f(-w^k)}{2 \cdot w^k}$$

What is $-w^k$?



(because x^2 is 2-to-1 map in $\mathbb{Z}_{qq}^*$)

so $-w^k = w^{k+N/2}$ (here $N = 8$)

If Prover hires $\exists P(X) \in \mathbb{F}[X]^{<K}$ s.t. $\forall \gamma \in L$, $f(\gamma) = P(\gamma)$

Attempt 1

Prover

$X = (N, K, f)$

witness $W = P(X)$

Compute g, h

Verifier

$X = (N, K, f)$

- $r \xleftarrow{\$} L$
- check $f(r) \stackrel{?}{=} g(r^2) + r h(r^2)$

$g: L^2 \to \mathbb{F}$ $\quad$ $h: L^2 \to \mathbb{F}$

Recurse on $(N/2, K/2, g)$ and $(N/2, K/2, h)$

Problem: Too many queries ($K-1$ total subproblems)

Problem: Distance Decay

ex //

f:

1	w	w^2	w^3	w^4	w^5	w^6	w^7	w^8	w^9	w^{10}	w^{11}	w^{12}	w^{13}	w^{14}	w^{15}
1	0	0	0	0	1	0	0	1	0	0	0	0	-1	0	0

$\delta = \frac{1}{4}$

g:

1	w^2	w^4	w^6	w^8	w^{10}	w^{12}	w^{14}
$\frac{1+1}{2}$	0	0	0	0	$\frac{1-1}{2}$	0	0

$\delta = \frac{1}{8}$

h:

1	w^2	w^4	w^6	w^8	w^{10}	w^{12}	w^{14}
$\frac{1-1}{2w^0} = 0$	0	0	0	0	$\frac{1-(-1)}{2 \cdot w^5}$	0	0

$\delta = \frac{1}{8}$

After r rounds distance can decay $\delta, \frac{\delta}{2}, \ldots, \frac{\delta}{2^r}$

Generally want $r = \Theta(\log K) = \Theta(\log N)$ rounds, VERY BAD!

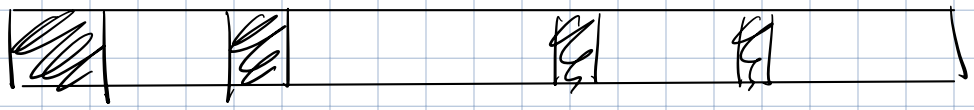
Attempt 2:

Prover		Verifier
$X = (N, K, f)$ $w = P(X)$		$X = (N, K, f)$
• Compute $g, h : L^2 \to \mathbb{F}$		
	$\xrightarrow{g, h}$	• $u \in_R L$, check $f(u) = g(u^2) + u h(u^2)$
		• Sample $\alpha \in_R \mathbb{F}$
	$\xleftarrow{\alpha}$	
• Set $f_\alpha := g + \alpha h$		
	$\xrightarrow{f_\alpha : L^2 \to \mathbb{F}}$	• check $f_\alpha(u^2) = g(u^2) + \alpha h(u^2)$

Recurse on $X = (\frac{N}{2}, \frac{K}{2}, f_\alpha)$

Fixes both problems! Now $\log K$ subproblems only, each of $\frac{1}{2}$ size ✓

Also, distance-preserving*



$\delta = 20\%$

$\frac{\delta}{2} = 10\%$

$\frac{\delta}{2} = 10\%$

$\delta = 20\%$ (for most α)

Attempt 3 (Simplifying Attempt 2)

Defn Given $f: L \to \mathbb{F}$, $\alpha \in \mathbb{F}$, define

$$\text{Fold}(f, \alpha): L^2 \to \mathbb{F} \qquad g(w^{2k}) + \alpha\, h(w^{2k})$$

$$w^{2k} \longmapsto \underbrace{\frac{f(w^k) + f(w^{k+N/2})}{2}}_{\text{previously } g(w^{2k})} + \alpha \underbrace{\frac{f(w^k) - f(w^{k+N/2})}{2 \cdot w^k}}_{\text{previously } h(w^{2k})}$$

Prover

$X = (N, K, f)$ $w = P(X)$

Verifier

$X = (N, K, f)$

- Sample $\alpha \in_R \mathbb{F}$

α ←

- Set $f_1 := \text{Fold}(f, \alpha)$

f_1 →

- Sample $u \in_R L$
- check $f_1(u^2) = g(u^2) + \alpha\, h(u^2)$ // has error $\epsilon_{\text{rd\_consistency\_check}}$

Can compute $g(u^2)$ from $f(u), f(-u)$

Can compute $h(u^2)$ similarly

Recall $K = 2^k$, so $r = k$ rounds

Soundness Err $\geq r \cdot \epsilon_{\text{rd\_consistency check}}$

Recurse on $X = (\frac{N}{2}, \frac{K}{2}, f_1)$

Ethan asked why this 1 check suffices instead of the 2 before...
I unfortunately could not come up with a clean answer on the spot but Attempt 3 proof size is factor of 3 smaller than Attempt 2.

FINAL ATTEMPT:

Prover(x, w) | **Verifier(x)**

α_0 ←

$f_1 := \text{Fold}(f, \alpha_0)$

f_1 →

α_1 ←

$f_2 := \text{Fold}(f_1, \alpha_1)$

f_2 →

...

α_{r-1} ←

$f_r := \text{Fold}(f_{r-1}, \alpha_{r-1})$

f_r →

- Sample $\alpha_0, \alpha_1, \ldots, \alpha_{r-1} \in_R \mathbb{F}$
- Consistency check randomness $u \in L$ [can boost t times]
- Consistency check on $f_0, \ldots, f_r$

$$f_1(u^2) \stackrel{?}{=} \text{Fold}(f_0, \alpha_0)(u^2)$$

requires $f_0(u), f_0(-u)$

$$f_2(u^4) \stackrel{?}{=} \text{Fold}(f_1, \alpha_1)(u^4)$$

requires $f_1(u^2), f_1(-u^2)$

...

$$f_r(u^{2^r}) \stackrel{?}{=} \text{Fold}(f_{r-1}, \alpha_{r-1})(u^{2^r})$$

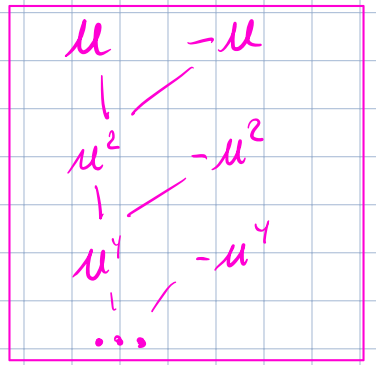
- Last function is low degree:

$$f_r \stackrel{?}{\in} RS\left[N/2^r, K/2^r\right]$$

Can think of FRI as having 2 phases: FOLD, QUERY

Why these consistency checks better? Coming...

Query Access Pattern



3 Soundness Intuition

3.1 Lower-Bounding Soundness Error (Shaving attack)

Claim: $\exists \tilde{P}$ that makes V accept some δ-far f with probability $\geq \max\left\{\frac{1}{|\mathbb{F}|}, (1-\delta)^t\right\}$

(Consider $\delta < \frac{1-\rho}{2}$)

Proof: Partition L into S_0, S_1 s.t. $|L_0| = (1-\delta)|L|$ and $|L_1| = \delta|L|$

where $u \in S_b \Rightarrow u^2 \in S_b$

Consider $f_0 = (\underbrace{0, \dots, 0}_{S_0}, \underbrace{Q, \dots}_{S_1})$ where Q is any line w/ no zeros on S_1

let $f_1 = f_2 = \dots = f_r = 0$

- $\epsilon_{\text{consistency\_check}} = \Pr[\text{all consistency checks fail}] = \Pr[\text{consistency check fails}]^t \geq (1-\delta)^t$
- $\epsilon_{\text{distortion}} = \Pr[f_1 = \text{Fold}(f_0, \alpha_0)] = \frac{1}{|\mathbb{F}|}$ (if α_0 is root of Q)

Distortion exercise

Let $\mathbb{F} = \mathbb{Z}/17\mathbb{Z}$ Let $L = \langle 2 \rangle$ multiplicative subgroup of $(\mathbb{Z}/17\mathbb{Z})^*$

Fact: $|L| = 8$

Let $S_1 := \langle 4 \rangle$ and $S_0 := L \setminus S_1$ Fact) $|L_1| = 4$ $4^3 = 64 \bmod 17 = 13$

Consider any $Q(X) = a + bX$ s.t. $Q(1), Q(4), Q(16), Q(13) \neq 0$

($1 = 4^0$, $4 = 4^1$, $16 = 4^2$, $13 = 4^3$)

Define $f(x) := 0 \ \forall x \in S_0 = \{2, 8, 15, 9\}$

$f(x) := Q(x) \ \forall x \in S_1 = \{1, 4, 16, 13\}$

Then Claim: $\text{Fold}(f, \alpha) = 0 \iff \alpha = \frac{-a}{b}$ i.e. $Q(\alpha) = 0$

3.2 Upper Bounding Soundness (proving soundness err bound)

"Easy" Case Prover $\tilde{P}$ is "consistent but noisy" and all functions f_i are "close"

(not today) Hard Case Prover $\tilde{P}$ is inconsistent OR some f_i is "far"

Definition $\forall f, g : L \to \mathbb{F}$ Define $\hat{\Delta}(f,g) := \dfrac{\left| \{ \gamma^2 \mid \gamma^2 \in L^2 \text{ and } (f(\gamma) \neq g(\gamma) \text{ OR } f(-\gamma) \neq g(-\gamma) \} \right|}{|L^2|}$

($\hat{\Delta}$: blockwise distance)

Let's abuse notation and define $\Delta(f,g) := \dfrac{\left| \{ \gamma \mid \gamma \in L \wedge f(\gamma) \neq g(\gamma) \} \right|}{|L|}$

Claim $\hat{\Delta}(f,g) \geq \Delta(f,g)$ Why? because blocks all same size (2)

Naturally define $\Delta(f,C)$ and $\hat{\Delta}(f,C)$ as $\min_{c \in C} \{ \Delta(f,c) \}$ and $\min_{c \in C} \{ \hat{\Delta}(f,c) \}$ respectively

Define $Drop(f, \delta) := \{ \alpha \in \mathbb{F} \mid \Delta(Fold(f, \alpha), RS[N/2, K/2]) < \delta \}$

- Suppose $\hat{\Delta}(f, RS[N,K]) = \delta$, $Drop(f, \delta)$ = set of bad folding challenges α

Define

$i = 0, \dots, r$ • $\delta_i := \hat{\Delta}(f_i, RS[N/2^i, K/2^i])$ recall $\rho = \frac{K}{N}$

$i = 0, \dots, r$ • $\tilde{f}_i := \underset{c \in C}{\operatorname{argmin}} \{ \hat{\Delta}(f_i, c) \}$ break ties arbitrarily, unique if $\delta_i < \frac{1-\rho}{2}$

$i = 0, \dots, r$ • $Err_i := \{ a \in L_i \mid f_i(a) \neq \tilde{f}_i(a) \text{ OR } f_i(-a) \neq \tilde{f}_i(-a) \}$

$i = 1, \dots, r$ • $Fail_i := \{ a \in L_i \mid \delta_i(a) \neq Fold(f_{i-1}, \alpha_{i-1})(a) \}$

Back to Easy Case (Unique Decoding) (the unique corrections are consistent)

We assume $\forall i$, $\delta_i < \frac{1-\rho}{2}$ AND $Fold(\tilde{f}_i, \alpha_i) \equiv \tilde{f}_{i+1}$ $\forall i \in \{0, \dots, r-1\}$

Also, assume $\forall i$, $\alpha_i \notin Drop(f_i, \delta_i)$ (no distortion)

Lemma: $\Pr_u[\text{reject}] \geq \frac{|Err_0|}{|L|} = \delta_0 \geq \frac{\Delta(f_0, RS[N,K])}{2}$

Proof Idea: Sampling $u_0 \in L$ determines $u_1, \ldots u_r$ $(u_{j+1} = u_j^2)$

WLOG, Asm $f_0 \equiv 0$, Sps $u_0 \in Err_0$

Since $Err_r = \emptyset$ $(\delta_r = 0)$

Then $\exists j < r$ st $u_j \in Err_j \wedge u_{j+1} \notin Err_{j+1}$

Thus $f_{j+1}(u_{j+1}) = \hat{f}_{j+1}(u_{j+1}) = 0$

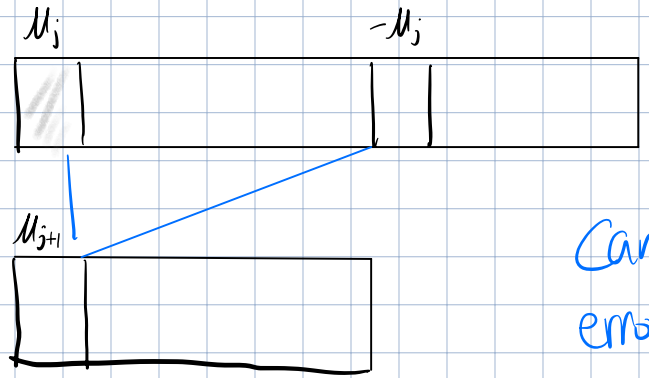
BUT $Fold(f_j, \alpha_j)(u_{j+1}) \neq Fold(\tilde{f}_j, \alpha_j)(u_{j+1})$ ✱

[Because $\alpha_j \in Bad(f_j, \delta_j) \wedge u_j \in Err_j$]

$\therefore u_{j+1} \in Fail_{j+1}$

(minor detail, must show $\widetilde{Fold(f_j, \alpha_j)} = \widehat{f_{j+1}}$)

✱ Intuition



Can prove if folding destroyed error on u_{j+1} then $\Delta(f_{j+1}, \hat{f}_{j+1}) < \delta_j$

Claim: $\mathrm{Fold}(f_j, \alpha_j)(u_{j+1}) \neq \mathrm{Fold}(\tilde{f}_j, \alpha_j)(u_{j+1}) = 0$

Proof of Claim

Lemma 1: $\widetilde{\mathrm{Fold}(f_j, \alpha_j)} \equiv \tilde{f}_{j+1}$

Proof of Lemma 1: $\forall a \notin \mathrm{Err}_j,\ \mathrm{Fold}(f_j, \alpha_j)(a^2) = \mathrm{Fold}(\tilde{f}_j, \alpha_j)(a^2)$

$$|L_{j+1}| \cdot \underbrace{\Delta}_{\text{relative distance}}\left(\mathrm{Fold}(f_j, \alpha_j),\ \underbrace{\mathrm{Fold}(\tilde{f}_j, \alpha_j)}_{=\tilde{f}_{j+1}}\right) \leq \frac{1}{2}|\mathrm{Err}_j| = \delta_j |L_{j+1}|$$

$$\Rightarrow \Delta\left(\mathrm{Fold}(f_j, \alpha_j),\ \tilde{f}_{j+1}\right) \leq \delta_j < \frac{1-\rho}{2} \quad \text{(unique decoding radius)} \quad \square$$

Lemma 2 $\forall a \in \mathrm{Err}_j,\ \mathrm{Fold}(f_j, \alpha_j)(a^2) = \tilde{f}_{j+1}(a^2) \Rightarrow \alpha_j \in \mathrm{Drop}(f_j, \delta_j)$

English: "Bad folding challenge α_j for a^2" $\Rightarrow$ α_j caused distortion

Pf of Lemma 2: Asm premise.

$$\Delta\left(\mathrm{Fold}(f_j, \alpha_j),\ RS\left[N/2^j,\ K/2^j\right]\right) = \overset{\text{Lemma 1}}{\Delta}\left(\mathrm{Fold}(f_j, \alpha_j),\ \tilde{f}_{j+1}\right) < \delta_j$$

Proof of Claim: $u_j \in \mathrm{Err}_j \wedge \alpha_j \notin \mathrm{Drop}(f_j, \alpha_j)$ Thus by lemma 2

$(u_{j+1} = u_j^2)$

$$\mathrm{Fold}(f_j, \alpha_j)(u_j^2) \neq \tilde{f}_{j+1}(u_j^2)$$

■ (done w/ easy case)

Harder Case: $\tilde{P}$ may jump to a "far" or "inconsistent" function.

At least one function is far

$\exists i \in \{0,\dots,r-1\}$ $\delta_i \geq \frac{1-\rho}{2}$ ($\delta_r = 0$ always)

OR

(unique) correction of close function is inconsistent

$\exists i \in \{0,\dots,r-1\}$ $\delta_i < \frac{1-\rho}{2}$ and $\text{Fold}(\tilde{f}_i, \alpha_i) \neq \tilde{f}_{i+1}$

Thm: $\Pr_u[\text{reject}] \geq \min\{\frac{1-\rho}{2}, \delta^*(\rho)\}$ where $\delta^*(\rho) := \frac{1-s\rho}{4}$

Let i be largest index for which one of two above conditions hold.

$\delta_{i+1} < \frac{1-\rho}{2}$ so $\tilde{f}_{i+1}$ and Err_{i+1} well-defined

Lemma 1: $\Pr_u[\text{reject}] \geq \frac{|\text{Fail}_{i+1} \cup \text{Err}_{i+1}|}{|L_{i+1}|}$

Proof Lemma 1: Fix $u_{i+1} \in \text{Fail}_{i+1} \cup \text{Err}_{i+1}$, will show we reject

- If $u_{i+1} \in \text{Fail}_{i+1} \Rightarrow$ rejects
- Else if $i+1 = r$, then $\text{Err}_{i+1} = \emptyset$, so case impossible
- Else $(i+1 < r)$, by easy case, $\exists j > i$ st $u_j \in \text{Fail}_j \Rightarrow$ rejects ∎

Lemma 2: $\frac{|\text{Fail}_{i+1} \cup \text{Err}_{i+1}|}{|L_{i+1}|} \geq \Delta\left(\tilde{f}_{i+1}, \text{Fold}(f_i, \alpha_i)\right)$

Proof Lemma 2:

$u_{i+1} \in L_{i+1} \setminus \text{Err}_{i+1} \Rightarrow f_{i+1}(u_{i+1}) = \tilde{f}_{i+1}(u_{i+1})$

$u_{i+1} \in L_{i+1} \setminus \text{Fail}_{i+1} \Rightarrow f_{i+1}(u_{i+1}) = \text{Fold}(f_i, \alpha_i)(u_{i+1})$

$\therefore u_{i+1} \in L_{i+1} \setminus (\text{Err}_{i+1} \cup \text{Fail}_{i+1}) \Rightarrow \tilde{f}_{i+1}(u_{i+1}) = \text{Fold}(f_i, \alpha_i)(u_{i+1})$ ∎

Lemma 3: $\Delta(\tilde{f}_{i+1}, \text{Fold}(f_{i+1}, \alpha_i)) \geq \min\left\{\frac{1-\rho}{2}, \delta^*(\rho)\right\}$

[aside, distortion guarantee we assume is the following (proof omitted, very complex)]

Thm: Distortion $f: L \to \mathbb{F}$, $\delta := \hat{\Delta}(f, RS[N,K])$

① $\delta < \frac{1-\rho}{2}$ then $\Pr_{\alpha}[\alpha \in \text{Drop}(f, \delta)] \leq |L|/|\mathbb{F}|$

② $\delta \geq \frac{1-\rho}{2}$ then $\Pr_{\alpha}[\alpha \in \text{Drop}(f, \delta^*(\rho))] \leq |L|/|\mathbb{F}|$

↑ caps out at some max parameter, like lots of proximity testing

We assume in both easy + hard cases no distortion and union bound in final FRI soundness result

(at least one fn far) (correction of close function inconsistent)

If $\delta_i \geq \frac{1-\rho}{2}$, then $\text{Fold}(f_i, \alpha_i)$ is $\delta^*(\rho)$-far from $RS[N/2^{i+1}, K/2^{i+1}] \ni \tilde{f}_{i+1}$

Else $\delta_i < \frac{1-\rho}{2}$

so $\text{Fold}(\tilde{f}_i, \alpha_i) \not\equiv \tilde{f}_{i+1}$, both codewords, so differ in at least $1-\rho$ fraction of locations.

$$1-\rho \leq \Delta\left(\tilde{f}_{i+1}, \text{Fold}(\tilde{f}_i, \alpha_i)\right) \leq \Delta(\tilde{f}_{i+1}, \text{Fold}(f_i, \alpha_i)) + \Delta(\text{Fold}(f_i, \alpha_i), \text{Fold}(\tilde{f}_i, \alpha_i))$$

(Δ is metric, triangle inequality)

$$= \Delta(\tilde{f}_{i+1}, \text{Fold}(f_i, \alpha_i)) + \delta_i$$

$$< \Delta(\tilde{f}_{i+1}, \text{Fold}(f_i, \alpha_i)) + \frac{1-\rho}{2}$$

$$\therefore \frac{1-\rho}{2} \leq \Delta(\tilde{f}_{i+1}, \text{Fold}(f_i, \alpha_i))$$ ∎

Proof of Thm: Combine lemmas 1, 2, 3 ∎

(4) Further Work

- FRI-based univariate PCS (K2G trick $\frac{\phi(x) - \phi(i)}{x - i}$ is degree $< k-1$)
- Basefold multilinear PCS (the coeffs of $P_0(X)$ = evaluation of some multilinear polynomial @ $(\alpha_0, \dots, \alpha_{n-1})$)
- FRI-Binius multilinear PCS over towers of binary fields (my previous job ☺)
- STIR // increase redundancy in later rounds to help verifier consistency checks (code switching)
- Distortion (Proximity Gap)

↳ Extreme Simplification: If random l.c. is far from code, then whp all are far from code

For more, watch Ben Diamond Proximity Gap Talk on YouTube

Small note on Basefold Trick for Multilinear PCS

$$f(r_0, \dots, r_{n-1}) = \sum_{x_0, \dots, x_{n-1} \in \{0,1\}} f(x_0, \dots, x_{n-1}) \cdot \widetilde{eq}(r_0, \dots, r_{n-1}, x_0, \dots, x_{n-1})$$

↑ multilinear

This sumcheck will reduce to form

$$f(\alpha_0, \ldots, \alpha_{n-1}) \cdot \tilde{eq}(r_0, \ldots, r_{n-1}, \alpha_0, \ldots, \alpha_{n-1})$$

where α_i is ith round verifier challenge (random)

Basefold: When $\alpha_0, \ldots, \alpha_{n-1}$ are folding challenges...

then the final FRI constant is $f(\alpha_0, \ldots, \alpha_{n-1})$ evaluation

super cute trick, I encourage people to read paper directly for more

Thanks for reading :)