Review of IOPP to Algebraic Geometry Codes (S. Bordage, M. Lhotel, J. Nardi, H. Randriam)

Rayan Sandid, Parsa Tasbihgou

December 2023

IOPP for AG codes

- Motivation behind using AG codes
- Mathematical background to construct a general protocol
- Special settings

AG codes: Motivation

Algebraic Geometry codes (AG) generalize Reed-Solomon (RS) codes:
RS codes $\subset$ *AG codes*.

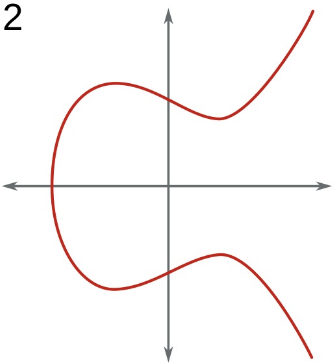
Drawbacks of RS-IOPPs:

- Alphabet size must be larger than block length of the code: $|\mathbb{F}| \geq$ *block length*;
- Specific algebraic structure
- Operations over $\mathbb{F}$ have high cost.

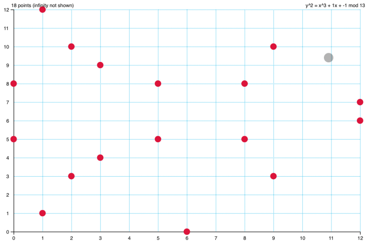
Advantages of AG code IOPPs:

- Constant rate and relative distance over constant-size fields.
- Closed under coordinate-wise multiplication.

(Informal) An *algebraic curve* defined on a field $\mathbb{F}$ is a set of points in space that are the zeros of a set of polynomials



(a) $\mathbb{R}$



(b) $\mathbb{F}_{13}$

Figure: $y^2 = x^3 + x - 1$

AG codes: Divisors and Riemann-Roch spaces

- A divisor D on $\mathcal{X}$ is a formal sum of points $D = \sum_{P \in \mathcal{X}} n_P P$. A divisor is effective if $n_P \geq 0$ for every point P.
- The set of divisors on a curve $\mathcal{X}$ forms an additive group, denoted by $Div(\mathcal{X})$. This group is endowed with a partial order relation $\geq$ such that $D \leq D'$ if $D' - D$ is effective.
- The Riemman-Roch space of a divisor $D \in Div(\mathcal{X})$ is the $\mathbb{F}-$vector space defined by

$$L_{\mathcal{X}}(D) = \{f \in \mathbb{F}(\mathcal{X}) |\ div_{\mathcal{X}}(f) + D \geq 0\} \cup \{0\}$$

Group action on curves

We say a group G acts on a curve $\mathcal{X}$ if there is an application (action)

$$\bullet \cdot \bullet : G \times \mathcal{X} \to \mathcal{X}$$

such that $\forall g, g' \in G^2, \forall x \in \mathcal{X}$, $e \cdot x = x$ and

$$g \cdot (g' \cdot x) = gg' \cdot x.$$

If finite Abelian group Γ of order p acts on $\mathcal{X}$ we can define $\mathcal{X}/\Gamma$ and we can have the projection map $\pi : \mathcal{X} \to \mathcal{X}/\Gamma$.

$$\mathcal{X}/\Gamma = \{Orb_\Gamma(P) | P \in \mathcal{X}\}$$

$$\mathbb{P}^1/\{1, -1\} = \{[x : 1] | x \geq 0\} \cup \{P_\infty = [1 : 0]\}$$

AG codes

- Take $D \in Div(\mathcal{X})$ and $\mathcal{P} = \{P_1, \dots, P_n\} \subset \mathcal{X}(\mathbb{F})$ of size $n := |\mathcal{P}|$. An AG code $C = C(\mathcal{X}, \mathcal{P}, D)$ over an algebraic curve $\mathcal{X}$ is the vector space of the image under the evaluations ev : $L(D) \to \mathbb{F}^n$ on the functions in the Riemann-Roch space $L(D)$;
- Particularly, the AG codes on the curve $\mathbb{P}^1$, the set of all lines through the origin in $\mathbb{R}^2$ correspond to the RS code.

Formally :

$$C(\mathbb{P}^1, \mathcal{P}, d \cdot P_\infty) = \mathrm{RS}[\mathbb{F}, \mathcal{P}, d]$$

where P_∞ is the point at infinity of $\mathbb{P}^1$.

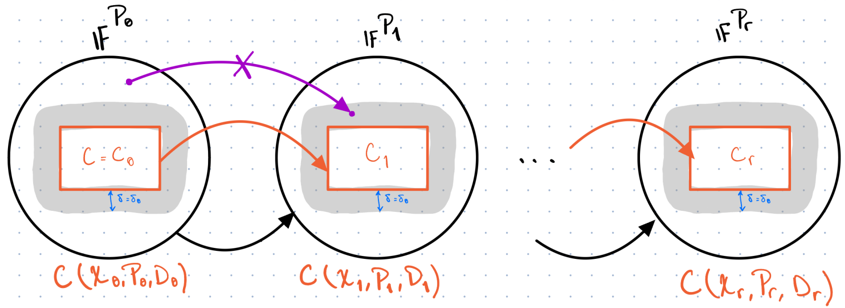
Generalizing the FRI protocol

Three steps of FRI:

- Splitting polynomials:
  decomposing the vector space of degree $\leq d$ polynomials over $\mathbb{F}$ into two copies of vector space of degree $\leq d/2$ polynomials.
- Randomized folding:
  reducing the evaluation domain (block length) and the vector space of functions.
- Distance preservation:
  if f is δ-far from code $C = \mathrm{RS}(\mathbb{F}, \mathcal{P}, d)$, then $\mathrm{Fold}(f, z)$ is "almost" δ-far from code $C = \mathrm{RS}(\mathbb{F}, \mathcal{P}, d/2)$

Sequence of codes

We want to iteratively reduce the problem size so we can check proximity to a smaller code. We need a sequence of codes $\{C_i(\mathcal{X}_i, \mathcal{P}_i, D_i)\}_{i\in[r]}$



Sequence of Curves

Let $G \leq \mathrm{Aut}(\mathcal{X})$ be a "large" finite solvable group that acts on $\mathcal{X}$.
$G = G_0 \rhd G_1 \rhd \ldots \rhd G_r = \{1\}$ is a normal sequence for G,
$\Gamma_i = G_i/G_{i+1}$ is Abelian and $|\Gamma_i| = [G_i; G_{i+1}] = p_i$.
Γ_0 acts on $\mathcal{X}_0 = \mathcal{X}$ and defines $\mathcal{X}_1 = \mathcal{X}_0/\Gamma_0$. Similarly Γ_i acts on $\mathcal{X}_i$ and defines $\mathcal{X}_{i+1} = \mathcal{X}_i/\Gamma_i$.
Now we can define:

- **$(\mathcal{X}, G)$-sequence of curves and projection maps**

$$\mathcal{X} = \mathcal{X}_0 \xrightarrow{\pi_0} \mathcal{X}_1 \xrightarrow{\pi_1} \mathcal{X}_2 \xrightarrow{\pi_2} \ldots \xrightarrow{\pi_{r-1}} \mathcal{X}_r = \mathcal{X}/G$$

$$\mathcal{X}_r = \frac{X_{r-1}}{\Gamma_{r-1}} = \frac{X_{r-2}}{\Gamma_{r-1}\Gamma_{r-2}} = \ldots = \frac{X_0}{\Gamma_{r-1}\ldots\Gamma_0} = \frac{X_0}{\frac{G_{r-1}}{G_r}\ldots\frac{G_0}{G_1}} = \frac{\mathcal{X}}{\frac{G_0}{G_r}} = \frac{\mathcal{X}}{G}$$

- **A sequence of evaluation points**

$$\mathcal{P} = \mathcal{P}_0 \xrightarrow{\pi_0} \mathcal{P}_1 \xrightarrow{\pi_1} \mathcal{P}_2 \xrightarrow{\pi_2} \ldots \xrightarrow{\pi_{r-1}} \mathcal{P}_r = \mathcal{P}/G$$

Kani's theorem

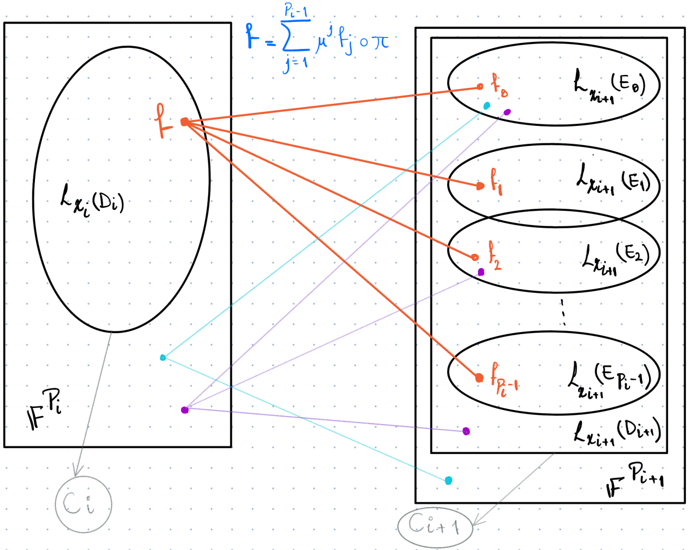
Given a function $f : \mathcal{P}_i \to \mathbb{F}$, define p_i functions $\{f_j\}_{0 \le j \le p_i}$ such that f is the evaluation of a function in $L_{\mathcal{X}_i}(D_i)$ iff f_j is the evaluation of some function in $L_{\mathcal{X}_{i+1}}(E_j) \subseteq L_{\mathcal{X}_{i+1}}(D_{i+1})$.

$$f = \sum_{j=0}^{p-1} \mu^j f_j \circ \pi_i$$

for all $f \in L_{\mathcal{X}}(D), \mu \in \mathbb{F}(\mathcal{X}), f_j \in L_{\mathcal{X}/\Gamma}(E_j)$

Divisors E_j and functions μ_i are explicitly expressed in terms of D_i and $\mathcal{X}_i$.

Kani's theorem continued



Balancing functions

We want make sure that no f_j is in $L_{\mathcal{X}_{i+1}}(D_{i+1}) \setminus L_{\mathcal{X}_{i+1}}(E_j)$. We do this using balancing functions ν_j. They are defined such that:

$$f_j \in L_{\mathcal{X}_{i+1}}(D_{i+1}) \text{ and } \nu_j f_j \in L_{\mathcal{X}_{i+1}}(D_{i+1}) \\ \text{iff } f_j \in L_{\mathcal{X}_{i+1}}(E_j).$$

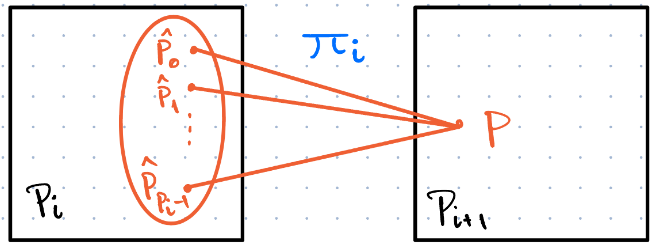
Existence of balancing functions depends on the Weierstrass semigroup of $support(D_{i+1})$.
If balancing functions exist for D_{i+1}, we say D_{i+1} is D_i-compatible.
The folding operator is:

$$\mathrm{Fold}[f, \vec{z}] = \sum_{j=0}^{p_i - 1} z_1^j f_j + \sum_{j=0}^{p_i - 1} z_2^{j+1} \nu_{i+1,j} f_j$$

Local computability

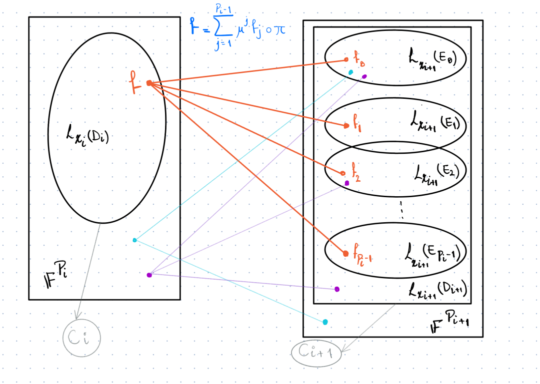
We want to compute Fold$[f, z]$ on some point $p \in \mathcal{P}_{i+1}$. This point has exactly p_i pre-images under π_i.
Define $A_p = \{(\mu_i(\hat{p}), f(\hat{p})\}_{\hat{p}\in\pi^{-1}(p)}$ and $I_{f,p}(x) = \sum_{j=0}^{p_i-1} x^j a_{j,p}$ the A_p-interpolating polynomial, so $I_{f,p}(\mu_i(\hat{p})) = f(\hat{p})$. Now, $f_j(p) = a_{j,p}$. So, given A_p, $f_j(P)$ can be found by interpolation.

$$\text{Fold}[f, \vec{z}] = \sum_{j=0}^{p_i-1} z_1^j f_j + \sum_{j=0}^{p_i-1} z_2^{j+1} \nu_{i+1,j} f_j$$



Completeness

Since $L_{\mathcal{X}_{i+1}}(E_j) \subseteq L_{\mathcal{X}_{i+1}}(D_{i+1})$, any linear combination of $L_{\mathcal{X}_{i+1}}(E_j)$ is a linear subspace of $L_{\mathcal{X}_{i+1}}(D_{i+1})$
so if $f \in C_i$ then for any $\vec{z} \in \mathbb{F}^2$, $\text{Fold}[f, \vec{z}] \in C_{i+1}$.



IOPP for AG codes

- **Setup**
  The prover and the verifyer agree on:
  - Starting curve $\mathcal{X}$
  - Group $G \subset Aut(\mathcal{X})$
  - Sequence $(\mathcal{X}, G)$
  - Functions μ_i
  - Divisors D_i.
- **Commit** The verifyer sends random field elements z_i to the prover and the prover sends the foldings $f_{i+1} = Fold[f_i, z_i]$ as oracles to the verifyer.
- **Query** First, the verifyer does a round consistancy check to see if all the oracles are really the folding of the previous one. The verifyer does this by comparing the two functions at a random point. Second, the veryfier reads the last code-word entirely and checks if it is in the last code C_r.