Decoding Simultaneous Rational Evaluation Codes

Matteo Abbondati, Eleonora Guerrini, Romain Lebreton

2^{nd} year Phd student - Montpellier

JNCF24 - CIRM 06/03/2024

Motivation: Fault Tolerant Linear System Solving

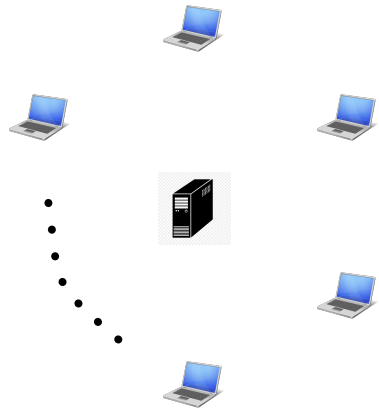
Linear System Solving over an Euclidean ring $\mathbb{A}$ ($\mathbb{A} = \mathbb{Z}$ or $\mathbb{A} = \mathbb{F}_q[x]$):

- Given $A \in \mathbb{A}^{r \times \ell}$ and $\vec{b} \in \mathbb{A}^r$
- Find $\vec{y}$ s.t. $A\vec{y} = \vec{b}$
  $\left(y = \left(\frac{f_1}{g}, \ldots, \frac{f_\ell}{g}\right) \in Frac(\mathbb{A})^\ell\right)$

Motivation: Fault Tolerant Linear System Solving

Linear System Solving over an Euclidean ring $\mathbb{A}$ ($\mathbb{A} = \mathbb{Z}$ or $\mathbb{A} = \mathbb{F}_q[x]$):

- Given $A \in \mathbb{A}^{r \times \ell}$ and $\vec{b} \in \mathbb{A}^r$
- Find $\vec{y}$ s.t. $A\vec{y} = \vec{b}$
  $\left(y = \left(\frac{f_1}{g}, \ldots, \frac{f_\ell}{g}\right) \in Frac(\mathbb{A})^\ell\right)$

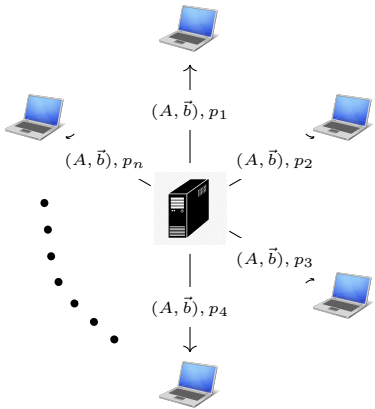


Motivation: Fault Tolerant Linear System Solving

Linear System Solving over an Euclidean ring $\mathbb{A}$ ($\mathbb{A} = \mathbb{Z}$ or $\mathbb{A} = \mathbb{F}_q[x]$):

- Given $A \in \mathbb{A}^{r \times \ell}$ and $\vec{b} \in \mathbb{A}^r$
- Find $\vec{y}$ s.t. $A\vec{y} = \vec{b}$
  $\left(y = \left(\frac{f_1}{g}, \ldots, \frac{f_\ell}{g}\right) \in Frac(\mathbb{A})^\ell\right)$

$\mathbb{A}=\mathbb{Z}$:p_j's distinct primes
$\mathbb{A}=\mathbb{F}_q[x]$:$p_j=x-\alpha_j$



Motivation: Fault Tolerant Linear System Solving

Linear System Solving over an Euclidean ring $\mathbb{A}$ ($\mathbb{A} = \mathbb{Z}$ or $\mathbb{A} = \mathbb{F}_q[x]$):

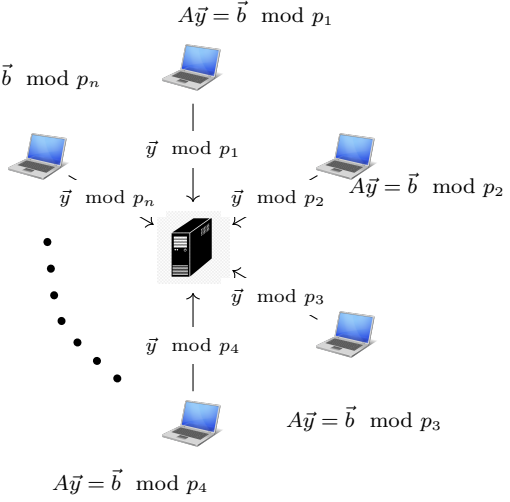
- Given $A \in \mathbb{A}^{r \times \ell}$ and $\vec{b} \in \mathbb{A}^r$
- Find $\vec{y}$ s.t. $A\vec{y} = \vec{b}$
  $\left(y = \left(\frac{f_1}{g}, \ldots, \frac{f_\ell}{g}\right) \in Frac(\mathbb{A})^\ell\right)$

$\mathbb{A}=\mathbb{Z}$:$p_j's$ distinct primes
$\mathbb{A}=\mathbb{F}_q[x]$:$p_j=x-\alpha_j$

Motivation: Fault Tolerant Linear System Solving

Linear System Solving over an Euclidean ring $\mathbb{A}$ ($\mathbb{A} = \mathbb{Z}$ or $\mathbb{A} = \mathbb{F}_q[x]$):

- Given $A \in \mathbb{A}^{r \times \ell}$ and $\vec{b} \in \mathbb{A}^r$
- Find $\vec{y}$ s.t. $A\vec{y} = \vec{b}$
  $\left(y = \left(\frac{f_1}{g}, \ldots, \frac{f_\ell}{g}\right) \in Frac(\mathbb{A})^\ell\right)$



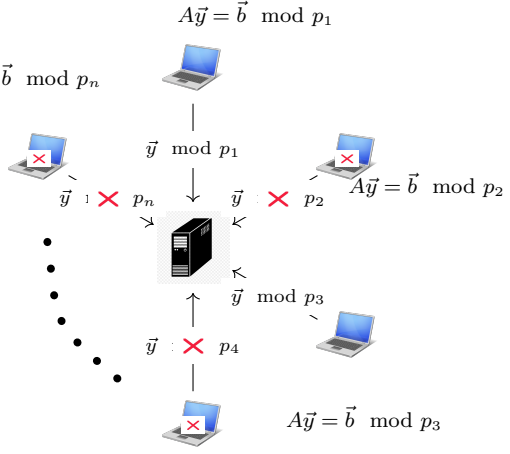
$\mathbb{A}=\mathbb{Z}$: $p_j's$ distinct primes
$\mathbb{A}=\mathbb{F}_q[x]$: $p_j = x - \alpha_j$

Assumption: $\forall j = 1, \ldots, n$ $\quad g(\alpha_j) \neq 0$ $\quad p_j \not| g$

Motivation: Fault Tolerant Linear System Solving

Linear System Solving over an Euclidean ring $\mathbb{A}$ ($\mathbb{A} = \mathbb{Z}$ or $\mathbb{A} = \mathbb{F}_q[x]$):

- Given $A \in \mathbb{A}^{r \times \ell}$ and $\vec{b} \in \mathbb{A}^r$
- Find $\vec{y}$ s.t. $A\vec{y} = \vec{b}$
  $\left(y = \left(\frac{f_1}{g}, \ldots, \frac{f_\ell}{g}\right) \in Frac(\mathbb{A})^\ell\right)$



$\mathbb{A}=\mathbb{Z}$: $p_j's$ distinct primes
$\mathbb{A}=\mathbb{F}_q[x]$: $p_j = x - \alpha_j$

Assumption: $\forall j = 1, \ldots, n$ $\quad g(\alpha_j) \neq 0$ $\quad p_j \not| g$

Simultaneous Rational Reconstruction with Errors

$\mathbb{A} = \mathbb{F}_q[x], \mathsf{Frac}(\mathbb{A}) = \mathbb{F}_q(x)$

SRFRwE

$\circ \alpha_1, \ldots, \alpha_n \in \mathbb{F}_q, M := \prod_{j=1}^{n}(x - \alpha_j)$

$\circ d_f, d_g, t \geq 0, \ell > 0$

$$\circ \boldsymbol{R} = \begin{pmatrix} \vdots & \cdots & \vdots \\ \vec{r}_1 & \cdots & \vec{r}_n \\ \vdots & \cdots & \vdots \end{pmatrix} \in \mathbb{F}_q^{\ell \times n}$$

$\star$ Find $\left(\frac{f_1}{g}, \ldots, \frac{f_\ell}{g}\right) \in \mathbb{F}_q(x)^\ell$ s.t.

- $\deg(f_i) < d_f, \deg(g) < d_g, \gcd(g, M) = 1$
- $d_H(f_i(\alpha_j)/g(\alpha_j), \boldsymbol{R}) \leq t$

$\mathbb{A} = \mathbb{Z}, \mathsf{Frac}(\mathbb{A}) = \mathbb{Q}$

SRNRwE

$\circ p_1, \ldots, p_n \in \mathbb{Z}, N := \prod_{j=1}^{n} p_j$

$\circ F, G, d \geq 0, \ell > 0$

$$\circ \boldsymbol{R} = \begin{pmatrix} \vdots & \cdots & \vdots \\ \vec{r}_1 & \cdots & \vec{r}_n \\ \vdots & \cdots & \vdots \end{pmatrix} \in \mathbb{Z}_{p_1}^{\ell} \times \ldots \times \mathbb{Z}_{p_n}^{\ell}$$

$\star$ Find $\left(\frac{f_1}{g}, \ldots, \frac{f_\ell}{g}\right) \in \mathbb{Q}^\ell$ s.t.

- $|f_i| < F, 0 < g < G, \gcd(g, N) = 1$
- $d_w([f_i/g]_{p_j}), \boldsymbol{R}) \leq d$

Codes come to help...

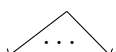
$\mathbb{A} = \mathbb{F}_q[x]$: Reed-Solomon codes	$\mathbb{A} = \mathbb{Z}$: Chinese Remainder codes
$f \in \mathbb{F}_q[x]_{<k}$	$C \in [0, K)$
Encoding: $\mod (x-\alpha_1)$, $\mod (x-\alpha_n)$	Encoding: $\mod p_1$, $\mod p_n$
$\vec{c} = (f(\alpha_1), \ldots, f(\alpha_n))$	$\vec{c} = ([C]_{p_1}, \ldots, [C]_{p_n})$
Channel: $\vec{e} = (e_1, \ldots, e_n)$	$\vec{e} = (e_1, \ldots, e_n)$
$\vec{r} = (r_1, \ldots, r_n) \leftrightarrow R(x) \in \mathbb{F}_q[x]_{<n}$	$\vec{r} = (r_1, \ldots, r_n) \leftrightarrow R \in \mathbb{Z}_N$
Decoding	
$f(x) \in \mathbb{F}_q[x]_{<k}$	$C \in [0, K)$

Rational Evaluation Codes [P][1]

Field of fractions of $\mathbb{F}_q[x] : \mathbb{F}_q(x)$

RF Codes

$$\mathbb{F}_q(x)$$
$$\cup|$$
$$\left\{\frac{f}{g} : \deg(f) < d_f, \deg(g) < d_g, \gcd(g, M) = 1\right\}$$



$$\vec{c} = \left(\frac{f}{g}(\alpha_1), \ldots, \frac{f}{g}(\alpha_n)\right)$$

Field of fractions of $\mathbb{Z} : \mathbb{Q}$

RN Codes

$$\mathbb{Q}$$
$$\cup|$$
$$\left\{\frac{f}{g} : |f| < F, 0 < g < G, \gcd(g, N) = 1\right\}$$

$$\vec{c} = ([f/g]_{p_1}, \ldots, [f/g]_{p_n})$$

[1] Pernet, Clément. High performance and reliable algebraic computing. Diss. Université Joseph Fourier, Grenoble 1, 2014.

Simultaneous Rational Evaluation Codes

SRF Codes

$$\mathbb{F}_q(x)^\ell$$
$$\cup|$$
$$\left\{ \left(\tfrac{f_i}{g}\right)_{1\le i\le \ell} : \begin{array}{c} \deg(f_i) < d_f, \deg(g) < d_g \\ \gcd(g, M) = 1 \end{array} \right\}$$

$$\boldsymbol{C} = \begin{pmatrix} \frac{f_1}{g}(\alpha_1) & \dots & \frac{f_1}{g}(\alpha_n) \\ \vdots & \vdots & \vdots \\ \frac{f_\ell}{g}(\alpha_1) & \dots & \frac{f_\ell}{g}(\alpha_n) \end{pmatrix}$$

SRN Codes

$$\mathbb{Q}^\ell$$
$$\cup|$$
$$\left\{ \left(\tfrac{f_i}{g}\right)_{1\le i\le \ell} : \begin{array}{c} |f_i| < F, 0 < g < G, \\ \gcd(g, N) = 1 \end{array} \right\}$$

$$\boldsymbol{C} = \begin{pmatrix} [f_1/g]_{p_1} & \dots & [f_1/g]_{p_n} \\ \vdots & \vdots & \vdots \\ [f_\ell/g]_{p_1} & \dots & [f_\ell/g]_{p_n} \end{pmatrix}$$

Polyalphabetic code? Different metric!

SRF code metric in $\mathbb{F}_q^{\ell \times n}$

$$\boldsymbol{C} = \begin{pmatrix} \vdots & \cdots & \vdots \\ \frac{\vec{f}}{g}(\alpha_1) & \cdots & \frac{\vec{f}}{g}(\alpha_n) \\ \vdots & \cdots & \vdots \end{pmatrix} \rightsquigarrow \begin{pmatrix} \vdots & \cdots & \vdots \\ \vec{r}_1 & \cdots & \vec{r}_n \\ \vdots & \cdots & \vdots \end{pmatrix} = \boldsymbol{R}$$

$$\xi := \left\{ j : \vec{r}_j \neq \frac{\vec{f}}{g}(\alpha_j) \right\} \text{ (Error Support)}$$

$$\Lambda := \prod_{j \in \xi} (x - \alpha_j) \text{ (Error Locator)}$$

$$d_H(\boldsymbol{R}, \boldsymbol{C}) := \#\xi = \deg(\Lambda) \text{ (Hamming Distance)}$$

SRN code metric in $\mathbb{Z}_{p_1}^{\ell} \times \ldots \times \mathbb{Z}_{p_n}^{\ell}$

$$\boldsymbol{C} = \begin{pmatrix} \vdots & \cdots & \vdots \\ \left[\frac{\vec{f}}{g}\right]_{p_1} & \cdots & \left[\frac{\vec{f}}{g}\right]_{p_n} \\ \vdots & \cdots & \vdots \end{pmatrix} \rightsquigarrow \begin{pmatrix} \vdots & \cdots & \vdots \\ \vec{r}_1 & \cdots & \vec{r}_n \\ \vdots & \cdots & \vdots \end{pmatrix} = \boldsymbol{R}$$

$$\xi := \left\{ j : \vec{r}_j \neq \left[\frac{\vec{f}}{g}\right]_{p_j} \right\} \text{ (Error Support)}$$

$$\Lambda := \prod_{j \in \xi} p_j \text{ (Error Locator)}$$

$$d_w(\boldsymbol{R}, \boldsymbol{C}) := \log_2(\Lambda) \text{ (Weighted Distance)}$$

$$d_{min}(SRF_\ell(M, d_f, d_g)) = n - d_f - d_g + 2$$

$$d_H(\boldsymbol{R}, \boldsymbol{C}) \leq \lfloor \tfrac{d_{min}-1}{2} \rfloor := d_u \Rightarrow \text{Unique Solution}$$

$$d_{min}(SRN_\ell(N, F, G)) \gtrsim \log_2\left(\tfrac{N}{2FG}\right)$$

$$d_w(\boldsymbol{R}, \boldsymbol{C}) \leq \log_2\left(\sqrt{\tfrac{N}{2FG}}\right) \Rightarrow \text{Unique Solution}$$

Idea of the Decoding beyond Unicity

SRF/SRN code

$$\begin{cases} \varphi = \Lambda g \\ \psi_i = \Lambda f_i \end{cases} \qquad (M(x) := \prod_{j=1}^{n}(x-\alpha_j)) \quad (N := \prod_{j=1}^{n} p_j)$$

Linearize

$$\Lambda g R_i = \Lambda f_i \mod M(x) \text{ or } N \quad \Rightarrow \quad \varphi R_i = \psi_i \mod M(x) \text{ or } N$$

$$S_{\boldsymbol{R},t} := \{(\varphi, \psi_1, \ldots, \psi_\ell) \in \mathbb{F}_q[x]^{\ell+1} : \varphi R_i = \psi_i \mod M(x), \deg(\varphi) < d_g + t, \deg(\psi_i) < d_f + t\}$$

$$S_{\boldsymbol{R},d} := \{(\varphi, \psi_1, \ldots, \psi_\ell) \in \mathbb{Z}^{\ell+1} : \varphi R_i = \psi_i \mod N, 0 < \varphi < 2^d G, |\psi_i| < 2^d F\}$$

Idea of the Decoding beyond Unicity

SRF/SRN code

$$\begin{cases} \varphi = \Lambda g \\ \psi_i = \Lambda f_i \end{cases} \qquad (M(x) := \prod_{j=1}^{n}(x-\alpha_j)) \quad (N := \prod_{j=1}^{n} p_j)$$

Linearize

$$\Lambda g R_i = \Lambda f_i \mod M(x) \text{ or } N \quad \Rightarrow \quad \varphi R_i = \psi_i \mod M(x) \text{ or } N$$

$$S_{\boldsymbol{R},t} := \{(\varphi, \psi_1, \ldots, \psi_\ell) \in \mathbb{F}_q[x]^{\ell+1} : \varphi R_i = \psi_i \mod M(x), \deg(\varphi) < d_g + t, \deg(\psi_i) < d_f + t\}$$

$$S_{\boldsymbol{R},d} := \{(\varphi, \psi_1, \ldots, \psi_\ell) \in \mathbb{Z}^{\ell+1} : \varphi R_i = \psi_i \mod N, 0 < \varphi < 2^d G, |\psi_i| < 2^d F\}$$

Decoding Algorithm SRN Codes:
Input: Received matrix $\boldsymbol{R}$, distance bound d
Output: Code word $\boldsymbol{C}$ s.t. $d(\boldsymbol{R}, \boldsymbol{C}) \leq d$ or "Decoding Failure"
1. Compute $v_s = (\varphi, \psi_1, \ldots, \psi_\ell) \in S_{\boldsymbol{R},d}$ (β-Approx-SVP)
2. if $v_s = v_{\boldsymbol{C}} := (\Lambda g, \Lambda f_1, \ldots, \Lambda f_\ell)$:
   return $\left(\frac{\psi_1}{\varphi}, \ldots, \frac{\psi_\ell}{\varphi}\right)$
3. else
   return "Decoding failure"

Goal: Decode for $t > \lfloor \frac{n - d_f - d_g + 1}{2} \rfloor$

- $\mathbb{P}_f \leq \mathbb{P}(S_{\boldsymbol{R},t} \not\subseteq v_{\boldsymbol{C}} \mathbb{F}_q[x])$

Goal: Decode for $d > \log_2\left(\sqrt{\frac{N}{2FG}}\right)$

- $\mathbb{P}_f \leq \mathbb{P}(S_{\boldsymbol{R},d} \not\subseteq v_{\boldsymbol{C}} \mathbb{Z})$

Interleaving

Burst Errors Channels

$$\ldots |c_{1,1}, c_{1,2}, c_{1,3}, c_{1,4}, c_{1,5}|c_{2,1}, c_{2,2}, c_{2,3}, c_{2,4}, c_{2,5}|c_{3,1}, c_{3,2}, c_{3,3}, c_{3,4}, c_{3,5}| \ldots$$

Interleaving

Burst Errors Channels (length bursts $\ell \approx 3$)

$$\ldots |c_{1,1}, c_{1,2}, c_{1,3}, c_{1,4}, c_{1,5}|c_{2,1}, \cancel{c_{2,2}}, \cancel{c_{2,3}}, \cancel{c_{2,4}}, c_{2,5}|c_{3,1}, c_{3,2}, c_{3,3}, c_{3,4}, c_{3,5}| \ldots$$

Interleaving

Burst Errors Channels (length bursts $\ell \approx 3$)

$$\ldots |\underline{c_{1,1}}, c_{1,2}, c_{1,3}, c_{1,4}, c_{1,5}| c_{2,1}, \cancel{c_{2,2}}, \cancel{c_{2,3}}, \cancel{c_{2,4}}, c_{2,5}| c_{3,1}, c_{3,2}, c_{3,3}, c_{3,4}, c_{3,5}| \ldots$$

$$\ldots |c_{1,1}$$

Interleaving

Burst Errors Channels (length bursts $\ell \approx 3$)

$$\ldots |c_{1,1}, c_{1,2}, c_{1,3}, c_{1,4}, c_{1,5}| \underline{c_{2,1}}, \cancel{c_{2,2}}, \cancel{c_{2,3}}, \cancel{c_{2,4}}, c_{2,5} |c_{3,1}, c_{3,2}, c_{3,3}, c_{3,4}, c_{3,5}| \ldots$$

$$\ldots |c_{1,1}, c_{2,1}$$

Interleaving

Burst Errors Channels (length bursts $\ell \approx 3$)

$$\ldots |c_{1,1}, c_{1,2}, c_{1,3}, c_{1,4}, c_{1,5}|c_{2,1}, \cancel{c_{2,2}}, \cancel{c_{2,3}}, \cancel{c_{2,4}}, c_{2,5}|\underline{c_{3,1}}, c_{3,2}, c_{3,3}, c_{3,4}, c_{3,5}| \ldots$$

$$\ldots |c_{1,1}, c_{2,1}, c_{3,1}$$

Interleaving

Burst Errors Channels (length bursts $\ell \approx 3$)

$$\ldots |c_{1,1}, c_{1,2}, c_{1,3}, c_{1,4}, c_{1,5}|c_{2,1}, \cancel{c_{2,2}}, \cancel{c_{2,3}}, \cancel{c_{2,4}}, c_{2,5}|c_{3,1}, c_{3,2}, c_{3,3}, c_{3,4}, c_{3,5}| \ldots$$

$$\ldots |c_{1,1}, c_{2,1}, c_{3,1}, c_{1,2}$$

Interleaving

Burst Errors Channels (length bursts $\ell \approx 3$)

$$\ldots |c_{1,1}, c_{1,2}, c_{1,3}, c_{1,4}, c_{1,5}|c_{2,1}, \underline{c_{2,2}}, c_{2,3}, c_{2,4}, c_{2,5}|c_{3,1}, c_{3,2}, c_{3,3}, c_{3,4}, c_{3,5}| \ldots$$

$$\ldots |c_{1,1}, c_{2,1}, c_{3,1}, c_{1,2}, c_{2,2}$$

Interleaving

Burst Errors Channels (length bursts $\ell \approx 3$)

$$\ldots |c_{1,1}, c_{1,2}, c_{1,3}, c_{1,4}, c_{1,5}|c_{2,1}, \cancel{c_{2,2}}, \cancel{c_{2,3}}, \cancel{c_{2,4}}, c_{2,5}|c_{3,1}, c_{3,2}, c_{3,3}, c_{3,4}, c_{3,5}| \ldots$$

$$\ldots |c_{1,1}, c_{2,1}, c_{3,1}, c_{1,2}, c_{2,2}, c_{3,2}, \cancel{c_{1,3}}, \cancel{c_{2,3}}, \cancel{c_{3,3}}, c_{1,4}, c_{2,4}, c_{3,4}, c_{1,5}, c_{2,5}, c_{3,5}| \ldots$$

Interleaving

Burst Errors Channels (length bursts $\ell \approx 3$)

$$\ldots |c_{1,1}, c_{1,2}, c_{1,3}, c_{1,4}, c_{1,5}|c_{2,1}, \cancel{c_{2,2}}, \cancel{c_{2,3}}, \cancel{c_{2,4}}, c_{2,5}|c_{3,1}, c_{3,2}, c_{3,3}, c_{3,4}, c_{3,5}| \ldots$$

$$\ldots |c_{1,1}, c_{2,1}, c_{3,1}, c_{1,2}, c_{2,2}, c_{3,2}, \cancel{c_{1,3}}, \cancel{c_{2,3}}, \cancel{c_{3,3}}, c_{1,4}, c_{2,4}, c_{3,4}, c_{1,5}, c_{2,5}, c_{3,5}| \ldots$$

$$I_\ell(\mathcal{C}) = \left\{ \begin{pmatrix} \cdots & \vec{c}_1 & \cdots \\ \cdots & \vec{c}_2 & \cdots \\ & \vdots & \\ \cdots & \vec{c}_\ell & \cdots \end{pmatrix} : \vec{c}_i \in \mathcal{C} \right\}$$

Interleaving

Burst Errors Channels (length bursts $\ell \approx 3$)

$$\ldots | c_{1,1}, c_{1,2}, c_{1,3}, c_{1,4}, c_{1,5} | c_{2,1}, \cancel{c_{2,2}}, \cancel{c_{2,3}}, \cancel{c_{2,4}}, c_{2,5} | c_{3,1}, c_{3,2}, c_{3,3}, c_{3,4}, c_{3,5} | \ldots$$

$$\ldots | c_{1,1}, c_{2,1}, c_{3,1}, c_{1,2}, c_{2,2}, c_{3,2}, \cancel{c_{1,3}}, \cancel{c_{2,3}}, \cancel{c_{3,3}}, c_{1,4}, c_{2,4}, c_{3,4}, c_{1,5}, c_{2,5}, c_{3,5} | \ldots$$

$$I_\ell(\mathcal{C}) = \left\{ \begin{pmatrix} \cdots & \vec{c}_1 & \cdots \\ \cdots & \vec{c}_2 & \cdots \\ & \vdots & \\ \cdots & \vec{c}_\ell & \cdots \end{pmatrix} : \vec{c}_i \in \mathcal{C} \right\}$$

Interleaving

Burst Errors Channels (length bursts $\ell \approx 3$)

$$\ldots | c_{1,1}, c_{1,2}, c_{1,3}, c_{1,4}, c_{1,5} | c_{2,1}, \cancel{c_{2,2}}, \cancel{c_{2,3}}, \cancel{c_{2,4}}, c_{2,5} | c_{3,1}, c_{3,2}, c_{3,3}, c_{3,4}, c_{3,5} | \ldots$$

$$\ldots | c_{1,1}, c_{2,1}, c_{3,1}, c_{1,2}, c_{2,2}, c_{3,2}, \cancel{c_{1,3}}, \cancel{c_{2,3}}, \cancel{c_{3,3}}, c_{1,4}, c_{2,4}, c_{3,4}, c_{1,5}, c_{2,5}, c_{3,5} | \ldots$$

$$I_\ell(\mathcal{C}) = \left\{ \begin{pmatrix} \cdots & \vec{c}_1 & \cdots \\ \cdots & \vec{c}_2 & \cdots \\ & \vdots & \\ \cdots & \vec{c}_\ell & \cdots \end{pmatrix} : \vec{c}_i \in \mathcal{C} \right\}$$

Interleaving

Burst Errors Channels (length bursts $\ell \approx 3$)

$$\ldots |c_{1,1}, c_{1,2}, c_{1,3}, c_{1,4}, c_{1,5}|c_{2,1}, \cancel{c_{2,2}}, \cancel{c_{2,3}}, \cancel{c_{2,4}}, c_{2,5}|c_{3,1}, c_{3,2}, c_{3,3}, c_{3,4}, c_{3,5}| \ldots$$

$$\ldots |c_{1,1}, c_{2,1}, c_{3,1}, c_{1,2}, c_{2,2}, c_{3,2}, \cancel{c_{1,3}}, \cancel{c_{2,3}}, \cancel{c_{3,3}}, c_{1,4}, c_{2,4}, c_{3,4}, c_{1,5}, c_{2,5}, c_{3,5}| \ldots$$

$$I_\ell(\mathcal{C}) = \left\{ \begin{pmatrix} \vdots & \vec{c}_1 & \cdots \\ \vdots & \vec{c}_2 & \cdots \\ & \vdots & \\ \vdots & \vec{c}_\ell & \cdots \end{pmatrix} : \vec{c}_i \in \mathcal{C} \right\}$$

Interleaving

Burst Errors Channels (length bursts $\ell \approx 3$)

$$\ldots |c_{1,1}, c_{1,2}, c_{1,3}, c_{1,4}, c_{1,5}|c_{2,1}, \cancel{c_{2,2}}, \cancel{c_{2,3}}, \cancel{c_{2,4}}, c_{2,5}|c_{3,1}, c_{3,2}, c_{3,3}, c_{3,4}, c_{3,5}| \ldots$$

$$\ldots |c_{1,1}, c_{2,1}, c_{3,1}, c_{1,2}, c_{2,2}, c_{3,2}, \cancel{c_{1,3}}, \cancel{c_{2,3}}, \cancel{c_{3,3}}, c_{1,4}, c_{2,4}, c_{3,4}, c_{1,5}, c_{2,5}, c_{3,5}| \ldots$$

$$I_\ell(\mathcal{C}) = \left\{ \begin{pmatrix} \vdots & \vec{c}_1 & \cdots \\ \vdots & \vec{c}_2 & \cdots \\ & \vdots & \\ \vdots & \vec{c}_\ell & \cdots \end{pmatrix} : \vec{c}_i \in \mathcal{C} \right\}$$

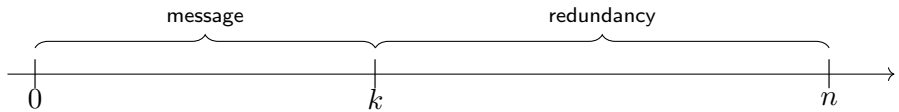
- Localized errors on common coordinates
- Increases the decoding radius beyond d_u
- Involves a failure probability analysis

The power of Interleaving

k bits message $\xmapsto{n-k \text{ redundant bits}}$ n bits codeword

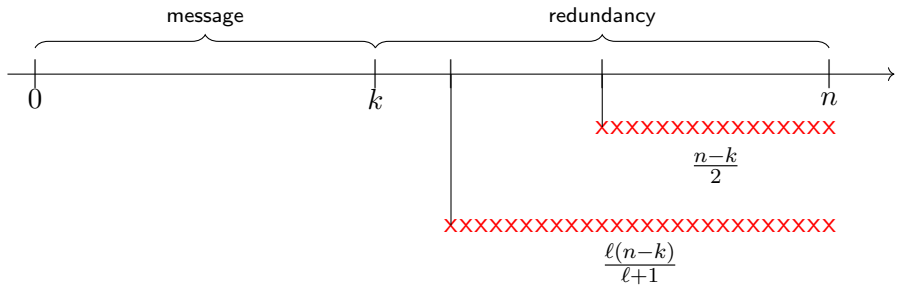
The power of Interleaving

k bits message $\xmapsto{n-k \text{ redundant bits}}$ n bits codeword



The power of Interleaving

k bits message $\xmapsto{n-k \text{ redundant bits}}$ n bits codeword



Distributions Received Codewords: SRF codes

Error Model I: $D^{I}_{\boldsymbol{C},\xi} := \{\boldsymbol{R} = \boldsymbol{C} + \boldsymbol{E}\}$

$$\boldsymbol{E} = \begin{pmatrix} \vdots & & \vdots \\ \vec{E}_1 & \cdots & \vec{E}_n \\ \vdots & & \vdots \end{pmatrix}$$

Fix error support $\xi \subseteq \{1, \ldots, n\}, \#\xi = t$

$$\Lambda = \prod_{j \in \xi}(x - \alpha_j)$$

$$\vec{E}_j = \begin{cases} \vec{0} & \text{if } j \notin \xi \\ \sim \mathcal{U}\left(\mathbb{F}_q^{\ell} \setminus \{\vec{0}\}\right) & \text{if } j \in \xi \end{cases}$$

Error Model II: $D^{II}_{\boldsymbol{C},\xi_r} := \{\boldsymbol{R} = \boldsymbol{C} + \boldsymbol{E}\}$

$$\boldsymbol{E} = \begin{pmatrix} \vdots & & \vdots \\ \vec{E}_1 & \cdots & \vec{E}_n \\ \vdots & & \vdots \end{pmatrix}$$

Fix maximal error support $\xi_r \subseteq \{1, \ldots, n\}, \#\xi_r = t$

$$\Lambda_r = \prod_{j \in \xi_r}(x - \alpha_j)$$

$$\vec{E}_j = \begin{cases} \vec{0} & \text{if } j \notin \xi_r \\ \sim \mathcal{U}\left(\mathbb{F}_q^{\ell}\right) & \text{if } j \in \xi_r \end{cases}$$

Distributions Received Codewords: SRN codes

Error Model I: $D^{I}_{\boldsymbol{C},\xi} := \{\boldsymbol{R} = \boldsymbol{C} + \boldsymbol{E}\}$

$$\boldsymbol{E} = \begin{pmatrix} \vdots & & \vdots \\ \vec{E}_1 & \cdots & \vec{E}_n \\ \vdots & & \vdots \end{pmatrix}$$

Fix error support $\xi \subseteq \{1, \ldots, n\}, \#\xi = t$

$$\vec{E}_j = \begin{cases} \vec{0} & \text{if } j \notin \xi \\ \sim \mathcal{U}\left(\mathbb{Z}_{p_j}^{\ell} \setminus \{\vec{0}\}\right) & \text{if } j \in \xi \end{cases}$$

$$\Lambda = \prod_{j \in \xi} p_j$$

Error Model II: $D^{II}_{\boldsymbol{C},\xi_r} := \{\boldsymbol{R} = \boldsymbol{C} + \boldsymbol{E}\}$

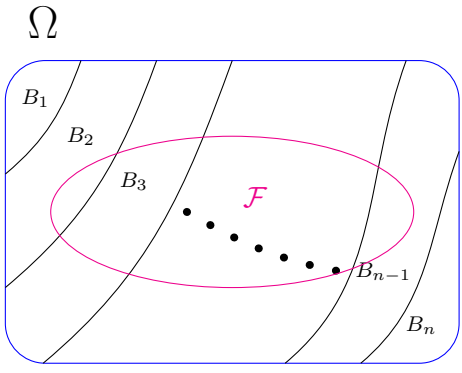
$$\boldsymbol{E} = \begin{pmatrix} \vdots & & \vdots \\ \vec{E}_1 & \cdots & \vec{E}_n \\ \vdots & & \vdots \end{pmatrix}$$

Fix maximal error support $\xi_r \subseteq \{1, \ldots, n\}, \#\xi_r = t$

$$\vec{E}_j = \begin{cases} \vec{0} & \text{if } j \notin \xi_r \\ \sim \mathcal{U}\left(\mathbb{Z}_{p_j}^{\ell}\right) & \text{if } j \in \xi_r \end{cases}$$

$$\Lambda_r = \prod_{j \in \xi_r} p_j$$

From Model I to Model II (Law of Total Probability)



$$\mathbb{P}^{II}_{\xi_r}(\mathcal{F}) = \sum_{\xi \subseteq \xi_r} \overbrace{\mathbb{P}^{II}_{\xi_r}(\mathcal{F}|\xi_{\boldsymbol{E}} = \xi)}^{\mathbb{P}^{I}_{\xi}(\mathcal{F})} \mathbb{P}^{II}_{\xi_r}(\xi_{\boldsymbol{E}} = \xi)$$

No $\mathcal{F}$

$$\mathbb{P}(\mathcal{F}) = \sum_{i=1}^{n} \mathbb{P}(\mathcal{F}|B_i)\mathbb{P}(B_i)$$

Our Results (SRNRwE unicity)

$$d_{max} := \frac{\ell}{\ell+1}\left[\log_2\left(\frac{N}{2FG}\right) - \log_2(3\beta)\right]$$

Model I: $\boldsymbol{R} \in D^{I}_{\boldsymbol{C},\xi}, \log_2\left(\prod_{p\in\xi} p\right) \leq d \leq d_{max} \Longrightarrow \mathbb{P}_f \leq 2^{-(\ell+1)(d_{max}-d)} \exp\left(\frac{n}{p_1^{\ell}}\right)$

Model II: $\boldsymbol{R} \in D^{II}_{\boldsymbol{C},\xi_r}, \log_2\left(\prod_{p\in\xi_r} p\right) \leq d \leq d_{max} \Longrightarrow \mathbb{P}_f \leq 2^{-(\ell+1)(d_{max}-d)}$

Improves & Generalizes [AAGL][2]

[2] Abbondati, Matteo, et al. "Probabilistic Analysis of LLL-based Decoder of Interleaved Chinese Remainder Codes." ITW 2023-IEEE Information Theory Workshop. 2023.

Our Results (SRFRwE unicity)

$$t_{max} := \frac{\ell}{\ell+1}\left[n - d_f - d_g + 1\right]$$

Model I: $\boldsymbol{R} \in D^{I}_{\boldsymbol{C},\xi}, t := \#\xi \leq t_{max} \implies \mathbb{P}_f \leq \left(\frac{q^{\ell}-\frac{1}{q}}{q^{\ell}-1}\right)^{t} \frac{q^{-(\ell+1)(t_{\max}-t)}}{q-1}$ $d_g = 1$ (no fractions) [SSB][3]

Model II: $\boldsymbol{R} \in D^{II}_{\boldsymbol{C},\xi_r}, t := \#\xi_r \leq t_{max} \implies \mathbb{P}_f \leq \frac{q^{-(\ell+1)(t_{\max}-t)}}{q-1}$ $\mathbb{P}_f \leq \frac{d_g+t}{q}$ [GLZ][4]

[3] Schmidt, Georg, Vladimir R. Sidorenko, and Martin Bossert. "Collaborative decoding of interleaved Reed–Solomon codes and concatenated code designs." IEEE Transactions on Information Theory 55.7 (2009): 2991-3012.

[4] Guerrini, E., Lebreton, R., & Zappatore, I. (2020). Enhancing simultaneous rational function recovery: adaptive error correction capability and new bounds for applications. arXiv preprint arXiv:2003.01793.

Thank you for your attention!