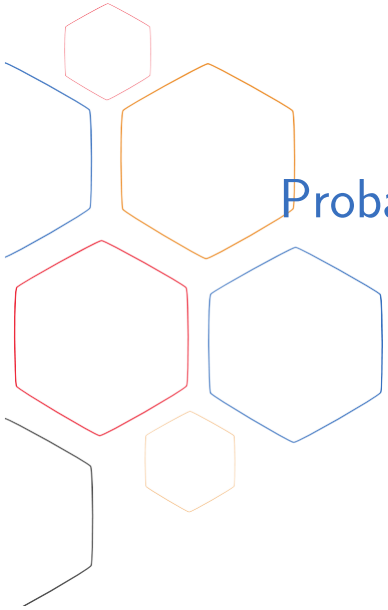


LIRMM

Probabilistic Analysis of LLL-based Decoder for ICR codes

Matteo Abbondati, Antoine Afflatet,
Eleonora Guerrini, Romain Lebreton

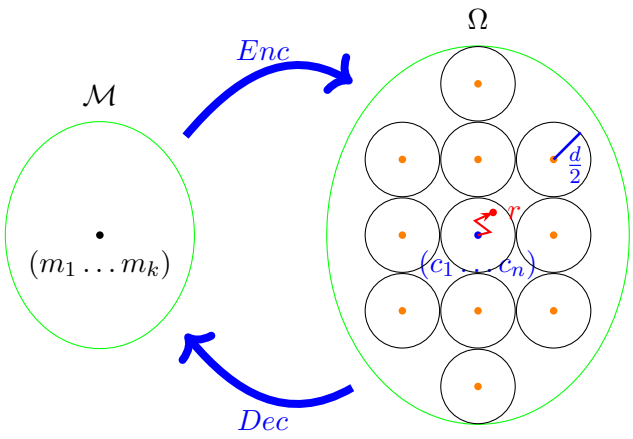
JNCF - Marseille 06/03/2023

cnrs

Table of Contents

Outline

What is a code?

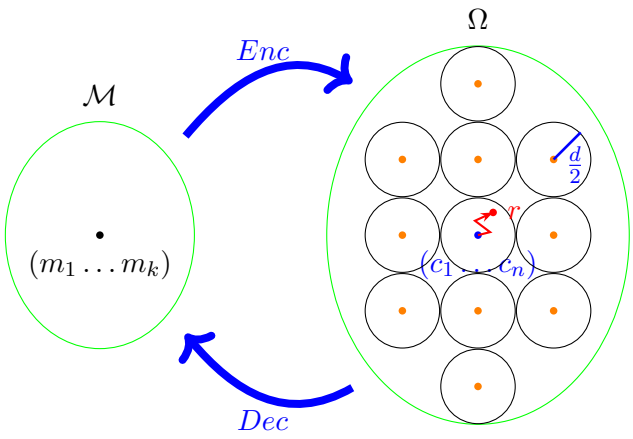


- $\mathcal{C} = Enc(\mathcal{M}) \subset \Omega$
- $n - k > 0$ Redundancy
- $d = \min_{c_1 \neq c_2} d(c_1, c_2)$ Minimum distance of the code
- $d \leq n - k + 1 (=$ MDS$)$

What a code can be used for?

- Correct errors in communications
- Correct errors in data storage
- Define fault tolerant algorithms

What is a code?

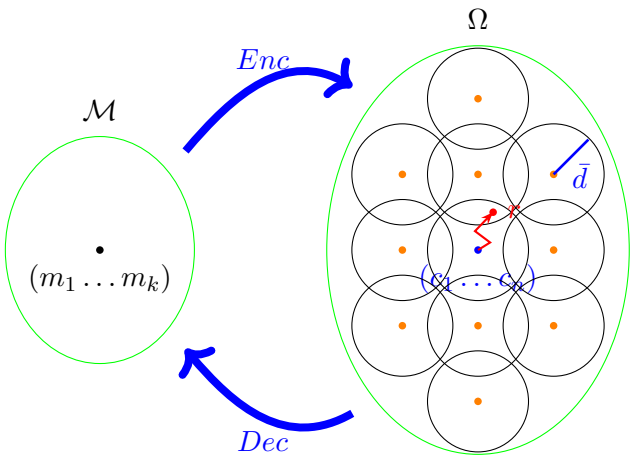


- $\mathcal{C} = Enc(\mathcal{M}) \subset \Omega$
- $n - k > 0$ Redundancy
- $d = \min_{c_1 \neq c_2} d(c_1, c_2)$ Minimum distance of the code
- $d \leq n - k + 1 (=$ MDS$)$

What a code can be used for?

- Correct errors in communications
- Correct errors in data storage
- Define fault tolerant algorithms

What is a code?



- $\mathcal{C} = Enc(\mathcal{M}) \subset \Omega$
- $n - k > 0$ Redundancy
- $d = \min_{c_1 \neq c_2} d(c_1, c_2)$ Minimum distance of the code
- $d \leq n - k + 1 (=$ MDS$)$

What a code can be used for?

- Correct errors in communications
- Correct errors in data storage
- Define fault tolerant algorithms

What is Interleaving?

$\mathcal{C}$ code of length n
- MDS
- Efficently decodable

$\ell > 0$

$$IC_\ell = \left\{ \begin{pmatrix} c_1^{(1)} & c_2^{(1)} & \cdots & c_n^{(1)} \\ \vdots & \vdots & \vdots & \vdots \\ c_1^{(\ell)} & c_2^{(\ell)} & \cdots & c_n^{(\ell)} \end{pmatrix} : c^{(i)} \in \mathcal{C} \right\}$$

Handle burst of errors

- Interleaved codeword $\left(c_1^{(1)}, c_1^{(2)}, \ldots, c_n^{(\ell)}\right)$ (column-wise).
- If $\mathcal{C}$ has error correction capacity t, then an interleaving of $\mathcal{C}$ of depth ℓ can handle burst of errors of length up to ℓt

Correct confidently errors beyond unique decoding radius

- Stack ℓ codewords together into a matrix
- Collaborative decoding algorithm
- Increase decoding radius $\bar{d} > \frac{d}{2}$
- Failure probability $\mathbb{P}_f(\bar{d}, \ell)$ $\nearrow \bar{d}, \searrow \ell$

What is Interleaving?

$\mathcal{C}$ code of length n

- MDS
- Efficently decodable

$\ell > 0$

$$IC_\ell = \left\{ \begin{pmatrix} c_1^{(1)} & c_2^{(1)} & \cdots & c_n^{(1)} \\ \vdots & \vdots & \vdots & \vdots \\ c_1^{(\ell)} & c_2^{(\ell)} & \cdots & c_n^{(\ell)} \end{pmatrix} : c^{(i)} \in \mathcal{C} \right\}$$

Handle burst of errors

- Interleaved codeword $\left(c_1^{(1)}, c_1^{(2)}, \ldots, c_n^{(\ell)}\right)$ (column-wise).
- If $\mathcal{C}$ has error correction capacity t, then an interleaving of $\mathcal{C}$ of depth ℓ can handle burst of errors of length up to ℓt

Correct confidently errors beyond unique decoding radius

- Stack ℓ codewords together into a matrix
- Collaborative decoding algorithm
- Increase decoding radius $\bar{d} > \frac{d}{2}$
- Failure probability $\mathbb{P}_f(\bar{d}, \ell)$ $\nearrow \bar{d}, \searrow \ell$

Outline

Chinese Remainder Codes

Definition

Let $p_1 < \ldots < p_n$ be n distinct primes, $1 \leq k < n$, $N = \prod_{i=1}^{n} p_i$, $K = \prod_{i=1}^{k} p_i$.
$\mathcal{C} = CR(N,k) = \{([C]_{p_1}, \ldots, [C]_{p_n}) : 0 \leq C < K\} \subset \mathbb{Z}_{p_1} \times \ldots \times \mathbb{Z}_{p_n}$

$$c = (\overset{r_1}{\underset{\shortparallel}{c_1}}, \ldots, \overset{r_{i_1}}{\underset{\shortparallel}{c_{i_1}}}, \ldots, \overset{r_{i_t}}{\underset{\shortparallel}{c_{i_t}}}, \ldots, \overset{r_n}{\underset{\shortparallel}{c_n}}) \rightsquigarrow r \in \mathbb{Z}_{p_1} \times \ldots \times \mathbb{Z}_{p_n}$$

$$\updownarrow \qquad CRT \qquad \updownarrow$$

$$C \in \mathbb{Z}_N \ 0 \leq C < K \qquad\qquad R \in \mathbb{Z}_N$$

Important notions:

- $\xi_{r,c} = \{i : r_i \neq c_i\}$ Error support
- $\Lambda_{r,c} = \prod_{i \in \xi_{r,c}} p_i$ Error locator $\rightsquigarrow$ $d(r,c) = \log_2(\Lambda_{r,c})$

Chinese Remainder Codes

Definition

Let $p_1 < \ldots < p_n$ be n distinct primes, $1 \leq k < n$, $N = \prod_{i=1}^{n} p_i$, $K = \prod_{i=1}^{k} p_i$.
$\mathcal{C} = CR(N,k) = \{([C]_{p_1}, \ldots, [C]_{p_n}) : 0 \leq C < K\} \subset \mathbb{Z}_{p_1} \times \ldots \times \mathbb{Z}_{p_n}$

$$
\begin{array}{ccccc}
c = (\overset{r_1}{\overset{\shortparallel}{c_1}}, \ldots, \cancel{\overset{r_{i_1}}{\overset{\shortparallel}{c_{i_1}}}}, \ldots, \cancel{\overset{r_{i_t}}{\overset{\shortparallel}{c_{i_t}}}}, \ldots, \overset{r_n}{\overset{\shortparallel}{c_n}}) & \rightsquigarrow & r \in \mathbb{Z}_{p_1} \times \ldots \times \mathbb{Z}_{p_n} \\
\updownarrow & & \updownarrow \\
C \in \mathbb{Z}_N \ 0 \leq C < K & CRT & R \in \mathbb{Z}_N
\end{array}
$$

Important notions:

- $\xi_{r,c} = \{i : r_i \neq c_i\}$ Error support
- $\Lambda_{r,c} = \prod_{i \in \mathcal{E}_{r,c}} p_i$ Error locator $\rightsquigarrow$ $d(r,c) = \log_2(\Lambda_{r,c})$

Lemma

The minimal distance $d(\mathcal{C})$ of the $CR(N,k)$ code is such that

$$d(\mathcal{C}) > \log_2 \left(\frac{N}{K} \right)$$

Key Equation & Decoding idea

$$\Lambda R = \Lambda C \mod N \quad \overset{\text{Linearize}}{\Rightarrow} \quad \begin{cases} \varphi = \Lambda \\ \psi = \Lambda C \end{cases} \quad \begin{cases} \varphi R = \psi \mod N \\ 0 \leq \varphi \leq \sqrt{N/K} \\ 0 \leq \psi < \sqrt{NK} \end{cases} \quad \leftarrow \text{Rational reconstruction - EEA}$$

Decoding CR codes below half the minimum distance [3]

Algorithm 1: Unique decoding of CR codes

Input: a code $CR(P; n, K)$ and a received word $R \in \mathbb{Z}_N$

Output: a codeword C such that $d(C, R) \leq \log\left(\sqrt{\frac{N}{K}}\right)$ or "decoding failure"

Compute (φ, ψ) by rational reconstruction
if φ *divides* ψ *and* $0 \leq \psi/\varphi < K$ **then**
 | **return** $C := \psi/\varphi$
end
else
 | **return** *"decoding failure"*
end

Evaluation - Interpolation codes

RS	CR
$\mathcal{M} = \{f \in \mathbb{F}_q[x] : \deg(f) < k\}$	$\mathcal{M} = \{C \in \mathbb{Z} : 0 \leq C < K\}$
$Enc(f) = (f(\alpha_1), \ldots, f(\alpha_n))$	$Enc(C) = ([C]_{p_1}, \ldots, [C]_{p_n})$
Lagrange interpolation	CRT
$\Lambda f = \Lambda Y \mod \prod(x - \alpha_i)$	$\Lambda C = \Lambda R \mod N$
Linear, MDS: $d = n - k + 1$	Not Linear, $d > \log_2(N/K)$
IRS [4][5]	ICR [2]
$\tau_{IRS} = \lfloor \frac{\ell(n-k)}{\ell+1} \rfloor$, $\mathbb{P}_f \leq \frac{\exp(q^{2-\ell})}{q-1}$	Li, Wenhui, Vladimir Sidorenko, Johan SR Nielsen.

Evaluation - Interpolation codes

RS	CR
$\mathcal{M} = \{f \in \mathbb{F}_q[x] : \deg(f) < k\}$	$\mathcal{M} = \{C \in \mathbb{Z} : 0 \leq C < K\}$
$Enc(f) = (f(\alpha_1), \ldots, f(\alpha_n))$	$Enc(C) = ([C]_{p_1}, \ldots, [C]_{p_n})$
Lagrange interpolation	CRT
$\Lambda f = \Lambda Y \mod \prod(x - \alpha_i)$	$\Lambda C = \Lambda R \mod N$
Linear, MDS: $d = n - k + 1$	Not Linear, $d > \log_2(N/K)$
IRS [4][5]	ICR [2]
$\tau_{IRS} = \lfloor \frac{\ell(n-k)}{\ell+1} \rfloor$, $\mathbb{P}_f \leq \frac{\exp(q^{2-\ell})}{q-1}$	Li, Wenhui, Vladimir Sidorenko, Johan SR Nielsen.

Evaluation - Interpolation codes

RS	CR
$\mathcal{M} = \{f \in \mathbb{F}_q[x] : \deg(f) < k\}$	$\mathcal{M} = \{C \in \mathbb{Z} : 0 \leq C < K\}$
$Enc(f) = (f(\alpha_1), \ldots, f(\alpha_n))$	$Enc(C) = ([C]_{p_1}, \ldots, [C]_{p_n})$
Lagrange interpolation	CRT
$\Lambda f = \Lambda Y \mod \prod(x - \alpha_i)$	$\Lambda C = \Lambda R \mod N$
Linear, MDS: $d = n - k + 1$	Not Linear, $d > \log_2(N/K)$

IRS [4][5]	ICR [2]
$\tau_{IRS} = \lfloor \frac{\ell(n-k)}{\ell+1} \rfloor$, $\mathbb{P}_f \leq \frac{\exp(q^{2-\ell})}{q-1}$	Li, Wenhui, Vladimir Sidorenko, Johan SR Nielsen.

Heuristic arguments

No hueristics in our work

Outline

What do we work with?

Definition(ICR)

$$ICR(N,k,\ell) = \left\{ \mathbf{C} = \begin{pmatrix} c_1^{(1)} & c_2^{(1)} & \dots & c_n^{(1)} \\ c_1^{(2)} & c_2^{(2)} & \dots & c_n^{(2)} \\ \vdots & \vdots & \ddots & \vdots \\ c_1^{(\ell)} & c_2^{(\ell)} & \dots & c_n^{(\ell)} \end{pmatrix} : c^{(j)} = (c_1^{(j)}, ..., c_n^{(j)}) \in \mathcal{C} \ \forall j \right\}$$

What do we work with?

Definition(ICR)

$$ICR(N,k,\ell) = \left\{ \mathbf{C} = \begin{pmatrix} c_1^{(1)} & c_2^{(1)} & \dots & c_n^{(1)} \\ c_1^{(2)} & c_2^{(2)} & \dots & c_n^{(2)} \\ \vdots & \vdots & \ddots & \vdots \\ c_1^{(\ell)} & c_2^{(\ell)} & \dots & c_n^{(\ell)} \end{pmatrix} : c^{(j)} = (c_1^{(j)}, ..., c_n^{(j)}) \in \mathcal{C} \ \forall j \right\}$$

Multidimensional key equation

$$\begin{cases} \Lambda R^{(1)} \equiv \Lambda C^{(1)} \mod N \\ \Lambda R^{(2)} \equiv \Lambda C^{(2)} \mod N \\ \quad \vdots \\ \Lambda R^{(\ell)} \equiv \Lambda C^{(\ell)} \mod N \end{cases}, \ \Lambda = lcm(\Lambda_1, \dots, \Lambda_\ell)$$

$\mathbb{Z}$-module minimization

A few words on LLL

- A. Lenstra, H. Lenstra, L. Lovász reduction algorithm on lattices.

Definition (Lattice)

Given $n > 0$ and $\{v_1, \ldots, v_n\} \subset \mathbb{R}^n$ a basis, then

$$\mathcal{L} = \sum_{1 \leq i \leq n} \mathbb{Z} v_i$$

- $LLL(\mathcal{L}) = v \in \mathcal{L} \setminus \{0\}$ such that $\|v\|_2 \leq \gamma \lambda_1(\mathcal{L})$
- $\gamma = \sqrt{2}^n$: Approximation factor of LLL
- $\lambda_1(\mathcal{L})$: 2-norm shortest non-zero vector of $\mathcal{L}$

Why LLL?

$$\exists \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_\ell \end{pmatrix} \in \mathbb{Z}^\ell s.t. \begin{cases} \Lambda R^{(1)} - a_1 N &= \Lambda C^{(1)} \\ \Lambda R^{(2)} - a_2 N &= \Lambda C^{(2)} \\ & \vdots \\ \Lambda R^{(\ell)} - a_\ell N &= \Lambda C^{(\ell)} \end{cases} \Leftrightarrow v_C = (\Lambda, \Lambda C^{(1)}, \ldots, \Lambda C^{(\ell)}) \in \mathcal{L}$$

$$\mathcal{L} = \begin{pmatrix} 1 & R^{(1)} & R^{(2)} & \ldots & R^{(\ell)} \\ 0 & N & 0 & \ldots & 0 \\ 0 & 0 & N & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \ldots & N \end{pmatrix}$$

Why LLL?

$$\exists \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_\ell \end{pmatrix} \in \mathbb{Z}^\ell s.t. \begin{cases} \Lambda R^{(1)} - a_1 N &= \Lambda C^{(1)} \\ \Lambda R^{(2)} - a_2 N &= \Lambda C^{(2)} \\ & \vdots \\ \Lambda R^{(\ell)} - a_\ell N &= \Lambda C^{(\ell)} \end{cases} \Leftrightarrow v_C = (\overset{\varphi}{\downarrow}\Lambda, \overset{\psi_1}{\downarrow}\Lambda C^{(1)}, \ldots, \overset{\psi_\ell}{\downarrow}\Lambda C^{(\ell)}) \in \mathcal{L}$$

$$\mathcal{L} = \begin{pmatrix} 1 & R^{(1)} & R^{(2)} & \ldots & R^{(\ell)} \\ 0 & N & 0 & \ldots & 0 \\ 0 & 0 & N & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \ldots & N \end{pmatrix}$$

Short vector in the lattice

$$v_C \in S_{\boldsymbol{R}} = \left\{ (\varphi, \psi_1, \ldots, \psi_\ell) \in \mathbb{Z}^{\ell+1} \;\middle|\; \begin{array}{c} \varphi R_i \equiv \psi_i \mod N \\ |\psi_i| < 2^\tau K \text{ and } |\varphi| \leq 2^\tau \end{array} \right\}$$

τ : Parameter related to the decoding radius ($\Lambda \leq 2^\tau$)

Outline

How do we use LLL here?

- Scale the lattice $\mathcal{L}$ by K on the first entry (homogenaize the size contraints):

$$\sigma_K((v_0, v_1, \ldots, v_\ell)) = (v_0 K, v_1, \ldots, v_\ell), \quad \sigma_K(\mathcal{L}) = \begin{pmatrix} K & R^{(1)} & R^{(2)} & \ldots & R^{(\ell)} \\ 0 & N & 0 & \ldots & 0 \\ 0 & 0 & N & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \ldots & N \end{pmatrix}$$

- Run LLL on $\sigma_K(\mathcal{L})$: $\bar{v}_s = LLL(\sigma_K(\mathcal{L}))$
- Unscale the result: $v_s = \sigma_K^{-1}(\bar{v}_s)$

Constraint 1

There exists a codeword C such that $\gamma\sqrt{\ell+1}\Lambda_{C,R} \leq 2^\tau$.

$\Rightarrow$ $v_s \in S_R$

How do we use LLL here?

- Scale the lattice $\mathcal{L}$ by K on the first entry (homogenaize the size contraints):

$$\sigma_K((v_0, v_1, \ldots, v_\ell)) = (v_0 K, v_1, \ldots, v_\ell), \quad \sigma_K(\mathcal{L}) = \begin{pmatrix} K & R^{(1)} & R^{(2)} & \ldots & R^{(\ell)} \\ 0 & N & 0 & \ldots & 0 \\ 0 & 0 & N & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \ldots & N \end{pmatrix}$$

- Run LLL on $\sigma_K(\mathcal{L})$: $\bar{v}_s = LLL(\sigma_K(\mathcal{L}))$
- Unscale the result: $v_s = \sigma_K^{-1}(\bar{v}_s)$

Constraint 1

There exists a codeword C such that $\gamma\sqrt{\ell+1}\Lambda_{C,R} \leq 2^\tau$.

$\Rightarrow$ $v_s \in S_{\boldsymbol{R}}$

Decoding ICR codes

Algorithm 2: Interleaved CR codes decoder

Input: $\mathcal{C} = ICR(N, k, \ell)$, received word R, parameter τ
Output: A codeword C s.t. $d(C, R) \leq \tau$ or "decoding failure"

Let $\bar{\mathcal{L}} = \sigma_K(\mathcal{L})$ be the scaled lattice
Compute a short vector $\bar{v}_s := LLL(\bar{\mathcal{L}})$
Unscale the vector: $v_s = (\varphi, \psi_1, \ldots, \psi_\ell) := \sigma_K^{-1}(\bar{v}_s)$
if $(|\varphi| \leq 2^\tau)$ *and* $(\varphi$ *divides all the* $(\psi_j)_{j=1,\ldots,\ell})$ *and* $(0 \leq \psi_j/\varphi < K$ *for all* $1 \leq j \leq \ell)$
then
  return $(C_1, \ldots, C_\ell) := (\psi_1/\varphi, \ldots, \psi_\ell/\varphi)$
end
else
  return *"decoding failure"*
end

Lemma

$Alg(\mathcal{C}, R, \tau) \to C \Rightarrow C \in \mathcal{C}, \quad d(C, R) \leq \tau$

($\Leftarrow$) is probabilistic

Decoding ICR codes

Algorithm 3: Interleaved CR codes decoder

Input: $\mathcal{C} = ICR(N, k, \ell)$, received word R, parameter τ
Output: A codeword C s.t. $d(C, R) \leq \tau$ or "decoding failure"

Let $\bar{\mathcal{L}} = \sigma_K(\mathcal{L})$ be the scaled lattice
Compute a short vector $\bar{v}_s := LLL(\bar{\mathcal{L}})$
Unscale the vector: $v_s = (\varphi, \psi_1, \ldots, \psi_\ell) := \sigma_K^{-1}(\bar{v}_s)$
if $(|\varphi| \leq 2^\tau)$ *and* $(\varphi$ *divides all the* $(\psi_j)_{j=1,\ldots,\ell})$ *and* $(0 \leq \psi_j/\varphi < K$ *for all* $1 \leq j \leq \ell)$
then
  return $(C_1, \ldots, C_\ell) := (\psi_1/\varphi, \ldots, \psi_\ell/\varphi)$
end
else
  return *"decoding failure"*
end

Lemma

$Alg(\mathcal{C}, R, \tau) \to C \Rightarrow C \in \mathcal{C}, \quad d(C, R) \leq \tau$

$(\Leftarrow)$ is probabilistic

Our Theorem

Theorem (Main result)

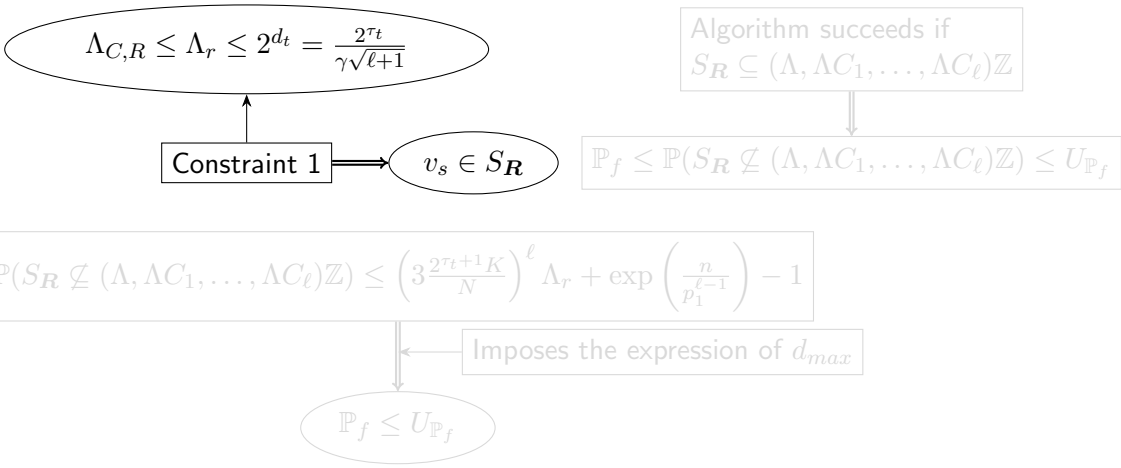
Given an ICR code $\mathcal{C}$ with parameters N, K, ℓ, and the approximation constant γ of LLL, set

$$d_{\max} := \frac{\ell}{\ell+1}\left[\log(N/K) - \log(6\gamma\sqrt{\ell+1})\right].$$

Choose a decoding distance bound $d_t < d_{\max}$, and set the parameter $\tau_t := d_t + \log(\gamma\sqrt{\ell+1})$ in Algorithm 2. Consider a random received word $R \sim D_{C,\mathcal{E}_r}$, for some codeword $C \in \mathcal{C}$ and error support $\mathcal{E}_r$ such that $\log \Lambda_r \leq d_t$.
Then, Algorithm 2 on random input R outputs the center codeword C of the distribution $D_{C,\mathcal{E}_r}$, with a probability of failure $\mathbb{P}_f$ upper-bounded by

$$\mathbb{P}_f \leq 2^{-(\ell+1)(d_{\max}-d_t)} + \exp\left(\frac{n}{p_1^{\ell-1}}\right) - 1 =: U_{\mathbb{P}_f}.$$

Ideas of the proof



Ideas of the proof

$\Lambda_{C,R} \leq \Lambda_r \leq 2^{d_t} = \frac{2^{\tau_t}}{\gamma\sqrt{\ell+1}}$

Constraint 1 $\Longrightarrow$ $v_s \in S_{\boldsymbol{R}}$

Algorithm succeeds if $S_{\boldsymbol{R}} \subseteq (\Lambda, \Lambda C_1, \ldots, \Lambda C_\ell)\mathbb{Z}$

$\mathbb{P}_f \leq \mathbb{P}(S_{\boldsymbol{R}} \not\subseteq (\Lambda, \Lambda C_1, \ldots, \Lambda C_\ell)\mathbb{Z}) \leq U_{\mathbb{P}_f}$

$\mathbb{P}(S_{\boldsymbol{R}} \not\subseteq (\Lambda, \Lambda C_1, \ldots, \Lambda C_\ell)\mathbb{Z}) \leq \left(3\frac{2^{\tau_t+1}K}{N}\right)^{\ell} \Lambda_r + \exp\left(\frac{n}{p_1^{\ell-1}}\right) - 1$

Imposes the expression of d_{max}

$\mathbb{P}_f \leq U_{\mathbb{P}_f}$

Ideas of the proof

$\Lambda_{C,R} \leq \Lambda_r \leq 2^{d_t} = \frac{2^{\tau_t}}{\gamma\sqrt{\ell+1}}$

Constraint 1 $\Longrightarrow$ $v_s \in S_{\boldsymbol{R}}$

Algorithm succeeds if $S_{\boldsymbol{R}} \subseteq (\Lambda, \Lambda C_1, \ldots, \Lambda C_\ell)\mathbb{Z}$

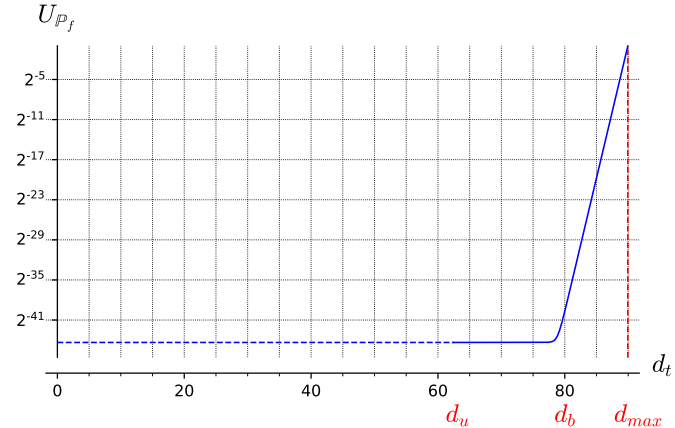
$\mathbb{P}_f \leq \mathbb{P}(S_{\boldsymbol{R}} \not\subseteq (\Lambda, \Lambda C_1, \ldots, \Lambda C_\ell)\mathbb{Z}) \leq U_{\mathbb{P}_f}$

$$\mathbb{P}(S_{\boldsymbol{R}} \not\subseteq (\Lambda, \Lambda C_1, \ldots, \Lambda C_\ell)\mathbb{Z}) \leq \left(3\frac{2^{\tau_t+1}K}{N}\right)^{\ell} \Lambda_r + \exp\left(\frac{n}{p_1^{\ell-1}}\right) - 1$$

Imposes the expression of d_{max}

$\mathbb{P}_f \leq U_{\mathbb{P}_f}$

An example



- $n = 50$ [~ 25 bits]
- $k = 45$
- $\ell = 2$

◇ $d_u = \log_2\left(\sqrt{\frac{N}{K}}\right)$

◇ d_{max} as in the main Thm.

◇ d_b such that $2^{-(\ell+1)(d_{\max}-d_b)} = \exp\left(n/p_1^{\ell-1}\right) - 1$

Future work

- Extension to rational number codes.
- Generalization to composite moduli.
- Fault tolerant algorithms for integer matrix problems (linear systems, matrix multiplication, matrix inversion...).

Thank you for your attention!

References

- M. Abbondati, A. Afflatet, E. Guerrini, R. Lebreton, "Probabilistic analysis of LLL-based decoder for ICR codes", ITW 2023.
- W. Li, V. Sidorenko, and J. S. R. Nielsen, "On decoding interleaved chinese remainder codes," in 2013 IEEE International Symposium on Information Theory, 2013, pp. 1052–1056
- O. Goldreich, D. Ron, and M. Sudan, "Chinese remaindering with errors" Information Theory, IEEE Transactions on, vol. 46, pp. 1330– 1338, 08 2000.
- Daniel Bleichenbacher, Aggelos Kiayias, Moti Yung, "Decoding Interleaved Reed-Solomon codes over Noisy Data." ICALP Springer, pp. 97- 108, 2003.
- Andrew Brown, Lorenz Minder, Amin Shokrollahi, "Probabilistic Decoding of Interleaved RS-Codes on the Q-ary symmetric channel." In Proc. of IEEE Intern. Symposium on Inf. Theory, p. 327, 2004.