Don't Use It Twice!
Solving Relaxed Linear Equivalence Problems

Advances in Cryptology – ASIACRYPT 2024

Jesús-Javier Chi-Domínguez [1], Alessandro Budroni [1], Giuseppe D'Alconzo[2], Antonio J. Di Scala[2], and Mukul Kulkarni[1]

[1] Cryptography Research Center, Technology Innovation Institute, Abu Dhabi, UAE
[2] Department of Mathematical Sciences, Polytechnic University of Turin, Italy

{jesus.dominguez,alessandro.budroni,mukul.kulkarni}@tii.ae,
{giuseppe.dalconzo,antonio.discala}@polito.it

December 10, 2024

Technology Innovation Institute

Politecnico di Torino

Cryptographic Group Action

Let $(\mathcal{G}, \circ)$ be a group with identity element $id \in \mathcal{G}$, and $\mathcal{X}$ a set. A map

$$\star : \mathcal{G} \times \mathcal{X} \to \mathcal{X}$$

is a group action if it satisfies the following properties:

1. Identity: $id \star x = x$ for all $x \in \mathcal{X}$.
2. Compatibility: $(g \circ h) \star x = g \star (h \star x)$ for all $g, h \in \mathcal{G}$ and $x \in \mathcal{X}$.

Vectorization problem [8]

Given $(x, y) \in \mathcal{X}^2$, determine $g \in \mathcal{G}$ such that $y = g \star x$.

Cryptographic Group Action

Let $(\mathcal{G}, \circ)$ be a group with identity element $id \in \mathcal{G}$, and $\mathcal{X}$ a set. A map

$$\star : \mathcal{G} \times \mathcal{X} \to \mathcal{X}$$

is a group action if it satisfies the following properties:

1. Identity: $id \star x = x$ for all $x \in \mathcal{X}$.
2. Compatibility: $(g \circ h) \star x = g \star (h \star x)$ for all $g, h \in \mathcal{G}$ and $x \in \mathcal{X}$.

Vectorization problem [8]

Given $(x, y) \in \mathcal{X}^2$, determine $g \in \mathcal{G}$ such that $y = g \star x$.

Cryptographic Group Action

Let $(\mathcal{G}, \circ)$ be a group with identity element $id \in \mathcal{G}$, and $\mathcal{X}$ a set. A map

$$\star : \mathcal{G} \times \mathcal{X} \to \mathcal{X}$$

is a group action if it satisfies the following properties:

1. Identity: $id \star x = x$ for all $x \in \mathcal{X}$.
2. Compatibility: $(g \circ h) \star x = g \star (h \star x)$ for all $g, h \in \mathcal{G}$ and $x \in \mathcal{X}$.

Vectorization problem [8]

Given $(x, y) \in \mathcal{X}^2$, determine $g \in \mathcal{G}$ such that $y = g \star x$.

To achieve some *advanced* properties of digital signatures, some relaxations of the above problem are usually used.

Given a polynomial number of pairs

$$(x_i, g \star x_i), \quad i = 1, \ldots, t.$$

- Find g, or
- Distinguish from random pairs in $\mathcal{X}^2$.

Cryptographic Group Action

Let $(\mathcal{G}, \circ)$ be a group with identity element $id \in \mathcal{G}$, and $\mathcal{X}$ a set. A map

$$\star : \mathcal{G} \times \mathcal{X} \to \mathcal{X}$$

is a group action if it satisfies the following properties:

1. Identity: $id \star x = x$ for all $x \in \mathcal{X}$.
2. Compatibility: $(g \circ h) \star x = g \star (h \star x)$ for all $g, h \in \mathcal{G}$ and $x \in \mathcal{X}$.

Vectorization problem [8]

Given $(x, y) \in \mathcal{X}^2$, determine $g \in \mathcal{G}$ such that $y = g \star x$.

To achieve some *advanced* properties of digital signatures, some relaxations of the above problem are usually used. This topic has been cryptanalyzed for LCE and MCE [9] and LIP [5].

Given a polynomial number of pairs

$$(x_i, g \star x_i), \quad i = 1, \ldots, t.$$

- Find g, or
- Distinguish from random pairs in $\mathcal{X}^2$.

Contributions

Our results concern LCE and MCE.

$\rightarrow$ We improve the bound on the number of necessary pairs

$$(x_i, g \star x_i)$$

to retrieve g.

$\rightarrow$ For the case of LCE with $k = \frac{n}{2}$, we show that two pairs are enough to retrieve g in polynomial time.

Linear and Matrix Code Equivalence Problems

Let $\mathbf{G}$ and $\mathbf{G}'$ be the generator matrices of two (n, k)-linear codes $\mathcal{C}, \mathcal{C}'$. We say that $\mathcal{C}$ and $\mathcal{C}'$ are *equivalent* if there exist $\mathbf{S} \in \mathrm{GL}_k(\mathbb{F}_q)$ and $\mathbf{Q} \in \mathrm{Mono}_n(\mathbb{F}_q)$ such that

$$\mathbf{G}' = \mathbf{SGQ}.$$

Let $\mathbf{G}$ and $\mathbf{G}'$ be the generator matrices of two (n, k)-linear codes $\mathcal{C}, \mathcal{C}'$. We say that $\mathcal{C}$ and $\mathcal{C}'$ are *equivalent* if there exist $\mathbf{S} \in \mathrm{GL}_k(\mathbb{F}_q)$ and $\mathbf{Q} \in \mathrm{Mono}_n(\mathbb{F}_q)$ such that

$$\mathbf{G}' = \mathbf{SGQ}.$$

Definition (Linear Code Equivalence (LCE) Problem)

Given $\mathbf{G}, \mathbf{G}' \in \mathbb{F}_q^{k\times n}$. Find (if they exists) matrices $\mathbf{S} \in \mathrm{GL}_k(\mathbb{F}_q)$ and $\mathbf{Q} \in \mathrm{Mono}_n(\mathbb{F}_q)$ such that $\mathbf{G}' = \mathbf{SGQ}$.

Let $\mathbf{G}$ and $\mathbf{G}'$ be the generator matrices of two (n, k)-linear codes $\mathcal{C}, \mathcal{C}'$. We say that $\mathcal{C}$ and $\mathcal{C}'$ are *equivalent* if there exist $\mathbf{S} \in \mathrm{GL}_k(\mathbb{F}_q)$ and $\mathbf{Q} \in \mathrm{Mono}_n(\mathbb{F}_q)$ such that

$$\mathbf{G}' = \mathbf{SGQ}.$$

Definition (Linear Code Equivalence (LCE) Problem)

Given $\mathbf{G}, \mathbf{G}' \in \mathbb{F}_q^{k \times n}$. Find (if they exists) matrices $\mathbf{S} \in \mathrm{GL}_k(\mathbb{F}_q)$ and $\mathbf{Q} \in \mathrm{Mono}_n(\mathbb{F}_q)$ such that $\mathbf{G}' = \mathbf{SGQ}$.

- If $\mathbf{Q} \in \mathrm{Perm}_n(\mathbb{F}_q)$, then it is **Permutation Code Equivalence (PCE) Problem**.
- If $\mathcal{C}$ and $\mathcal{C}'$ determine two subspaces of the $m \times r$ matrix space, then it becomes the **Matrix Code Equivalence (MCE) Problem**.
- Cryptographic constructions assume the matrix code generators are in systematic form (SF), corresponding with its reduced row-echelon form.

Let $\mathbf{G}$ and $\mathbf{G}'$ be the generator matrices of two (n,k)-linear codes $\mathcal{C}, \mathcal{C}'$. We say that $\mathcal{C}$ and $\mathcal{C}'$ are *equivalent* if there exist $\mathbf{S} \in \mathrm{GL}_k(\mathbb{F}_q)$ and $\mathbf{Q} \in \mathrm{Mono}_n(\mathbb{F}_q)$ such that

$$\mathbf{G}' = \mathbf{SGQ}.$$

Definition (Linear Code Equivalence (LCE) Problem)

Given $\mathbf{G}, \mathbf{G}' \in \mathbb{F}_q^{k \times n}$. Find (if they exists) matrices $\mathbf{S} \in \mathrm{GL}_k(\mathbb{F}_q)$ and $\mathbf{Q} \in \mathrm{Mono}_n(\mathbb{F}_q)$ such that $\mathbf{G}' = \mathbf{SGQ}$.

- If $\mathbf{Q} \in \mathrm{Perm}_n(\mathbb{F}_q)$, then it is **Permutation Code Equivalence (PCE) Problem**.
- If $\mathcal{C}$ and $\mathcal{C}'$ determine two subspaces of the $m \times r$ matrix space, then it becomes the **Matrix Code Equivalence (MCE) Problem**.
- Cryptographic constructions assume the matrix code generators are in systematic form (SF), corresponding with its reduced row-echelon form.

Let $\mathbf{G}$ and $\mathbf{G}'$ be the generator matrices of two (n, k)-linear codes $\mathcal{C}, \mathcal{C}'$. We say that $\mathcal{C}$ and $\mathcal{C}'$ are *equivalent* if there exist $\mathbf{S} \in \mathrm{GL}_k(\mathbb{F}_q)$ and $\mathbf{Q} \in \mathrm{Mono}_n(\mathbb{F}_q)$ such that

$$\mathbf{G}' = \mathbf{SGQ}.$$

Definition (LCE Systematic Form Version)

Given the generators $\mathbf{G}, \mathbf{G}' \in \mathbb{F}_q^{k \times n}$ in systematic form. Find $\mathbf{Q} \in \mathrm{Mono}_n(\mathbb{F}_q)$ such that $\mathbf{G}' = \mathrm{SF}(\mathbf{GQ})$.

- If $\mathbf{Q} \in \mathrm{Perm}_n(\mathbb{F}_q)$, then it is **Permutation Code Equivalence (PCE) Problem**.
- If $\mathcal{C}$ and $\mathcal{C}'$ determine two subspaces of the $m \times r$ matrix space, then it becomes the **Matrix Code Equivalence (MCE) Problem**.
- Cryptographic constructions assume the matrix code generators are in systematic form (SF), corresponding with its reduced row-echelon form.

In the context of linkable ring signatures, [2] introduced the following problem.

Definition (Inverse LCE (ILCE) Systematic Form Version)

Given the generators $\mathbf{G}, \mathbf{G}', \mathbf{G}'' \in \mathbb{F}_q^{k \times n}$ in systematic form, find $\mathbf{Q} \in \mathrm{Mono}_n(\mathbb{F}_q)$ such that $\mathbf{G}' = \mathrm{SF}(\mathbf{GQ})$ and $\mathbf{G}'' = \mathrm{SF}(\mathbf{GQ}^{-1})$.

In the context of linkable ring signatures, [2] introduced the following problem.

Definition (Inverse LCE (ILCE) Systematic Form Version)

Given the generators $\mathbf{G}, \mathbf{G}', \mathbf{G}'' \in \mathbb{F}_q^{k \times n}$ in systematic form, find $\mathbf{Q} \in \mathrm{Mono}_n(\mathbb{F}_q)$ such that $\mathbf{G}' = \mathrm{SF}(\mathbf{GQ})$ and $\mathbf{G}'' = \mathrm{SF}(\mathbf{GQ}^{-1})$.

Remark

From a given ILCE instance

$$\{ (\mathbf{G}, \mathbf{G}' = \mathrm{SF}(\mathbf{GQ})) , \quad (\mathbf{G}, \mathbf{G}'' = \mathrm{SF}(\mathbf{GQ}^{-1}))\}$$

one obtains

$$\{ (\mathbf{G}, \mathbf{G}' = \mathrm{SF}(\mathbf{GQ})) , \quad (\mathbf{G}'', \mathbf{G} = \mathrm{SF}(\mathbf{G}''\mathbf{Q})) \},$$

which is *almost* like having two random problem instances with the same secret.

Define the following equivalence relation

$$A \simeq_{\mathsf{SF}} B \iff \mathsf{SF}(A) = \mathsf{SF}(B), \qquad A, B \in \mathbb{F}_q^{k\times n}.$$

Consider the base set as $\mathcal{X} = \mathbb{F}_q^{k\times n} / \simeq_{\mathsf{SF}}$ and the group as $\mathcal{G} = \mathsf{Mono}_n(\mathbb{F}_q)$. Then, the group action $\star$ is defined as

$$\star\colon \mathcal{G} \times \mathcal{X} \to \mathcal{X}, \quad (\mathbf{Q}, \mathbf{G}) \mapsto \mathbf{Q} \star \mathbf{G} := \mathsf{SF}(\mathbf{GQ}).$$

Similarly, PCE and MCE are modeled as group actions following the same framework.

Sample complexity

We define and study the following problem in the context of LCE and MCE.

Multiple Sample Setting

Let $\mathbf{Q} \in \mathrm{Mono}_n(\mathbb{F}_q)$ be fixed and secret. Given t random instances

$$(\mathbf{G_i}, \mathbf{G_i'} = \mathbf{Q} \star \mathbf{G_i}) \in \mathcal{X}^2, \quad i = 1, \ldots, t.$$

The t-LCE problem is to find $\mathbf{Q}$.

Similarly, one defines t-PCE and t-MCE.

Sample complexity

D'Alconzo and Di Scala [9] showed that with $t = kn$ instances of the form

$$(\mathbf{G_i}, \mathbf{G_i'} = \mathbf{S}\mathbf{G_i}\mathbf{Q})$$

one can retrieve $\mathbf{S} \in \mathrm{GL}_k(\mathbb{F}_q)$ and $\mathbf{Q} \in \mathrm{Mono}_n(\mathbb{F}_q)$ in polynomial time.

We improve this result in two ways:

- → We require a much smaller number of samples.
- → Our result also works with instances in systematic form, where the matrix $\mathbf{S}$ is different for each $\mathbf{G_i'} = \mathrm{SF}(\mathbf{G_i}\mathbf{Q}) = \mathbf{S_i}\mathbf{G_i}\mathbf{Q}$.

Sample complexity

Let $(\mathbf{G}, \mathbf{G}' = \mathrm{SF}(\mathbf{GQ}))$ be an LCE instance, and let $\mathbf{H}'$ be a parity check matrix of $\mathbf{G}'$. Then we have that

$$\mathbf{GQH}'^{\top} = \mathbf{0} \Leftrightarrow (\mathbf{G} \otimes \mathbf{H}')\mathrm{vec}(\mathbf{Q}) = \mathbf{0}$$

where $\mathrm{vec}(\mathbf{Q})$ is the column vector whose entries are the entries of $\mathbf{Q}$ row-by-row.

Let $(\mathbf{G}, \mathbf{G}' = \mathrm{SF}(\mathbf{GQ}))$ be an LCE instance, and let $\mathbf{H}'$ be a parity check matrix of $\mathbf{G}'$. Then we have that

$$\mathbf{GQH}'^{\top} = \mathbf{0} \Leftrightarrow (\mathbf{G} \otimes \mathbf{H}')\mathrm{vec}(\mathbf{Q}) = \mathbf{0}$$

where $\mathrm{vec}(\mathbf{Q})$ is the column vector whose entries are the entries of $\mathbf{Q}$ row-by-row.

In particular, if $\mathbf{G} = (\mathbf{I}_k|\mathbf{M})$ and $\mathbf{G}' = (\mathbf{I}_k|\mathbf{M}')$, then we have

$$\left[\begin{pmatrix} \mathbf{I}_k & | & \mathbf{M} \end{pmatrix} \otimes \begin{pmatrix} -\mathbf{M}'^{\top} & | & \mathbf{I}_{n-k} \end{pmatrix}\right] \mathrm{vec}(\mathbf{Q}) = \mathbf{0}.$$

Let $(\mathbf{G}, \mathbf{G}' = \mathrm{SF}(\mathbf{GQ}))$ be an LCE instance, and let $\mathbf{H}'$ be a parity check matrix of $\mathbf{G}'$. Then we have that

$$\mathbf{GQH}'^{\top} = \mathbf{0} \Leftrightarrow (\mathbf{G} \otimes \mathbf{H}')\mathrm{vec}(\mathbf{Q}) = \mathbf{0}$$

where $\mathrm{vec}(\mathbf{Q})$ is the column vector whose entries are the entries of $\mathbf{Q}$ row-by-row.

In particular, if $\mathbf{G} = (\mathbf{I}_k|\mathbf{M})$ and $\mathbf{G}' = (\mathbf{I}_k|\mathbf{M}')$, then we have

$$\left[\left(\begin{array}{c|c} \mathbf{I}_k & \mathbf{M} \end{array}\right) \otimes \left(\begin{array}{c|c} -\mathbf{M}'^{\top} & \mathbf{I}_{n-k} \end{array}\right)\right] \mathrm{vec}(\mathbf{Q}) = \mathbf{0}.$$

The idea is to *stack* systems derived from different samples until the rank is large enough to retrieve $\mathrm{vec}(\mathbf{Q})$ via Gaussian elimination. That is, one constructs the system $\mathbf{A} \cdot \mathrm{vec}(\mathbf{Q}) = \mathbf{0}$, where

$$\mathbf{A} = \begin{bmatrix} \left(\begin{array}{c|c} \mathbf{I}_k & \mathbf{M}_1 \end{array}\right) \otimes \left(\begin{array}{c|c} -\mathbf{M}_1'^{\top} & \mathbf{I}_{n-k} \end{array}\right) \\ \left(\begin{array}{c|c} \mathbf{I}_k & \mathbf{M}_2 \end{array}\right) \otimes \left(\begin{array}{c|c} -\mathbf{M}_2'^{\top} & \mathbf{I}_{n-k} \end{array}\right) \\ \cdots \\ \left(\begin{array}{c|c} \mathbf{I}_k & \mathbf{M}_t \end{array}\right) \otimes \left(\begin{array}{c|c} -\mathbf{M}_t'^{\top} & \mathbf{I}_{n-k} \end{array}\right) \end{bmatrix}. \tag{1}$$

Sample complexity

Technology Innovation Institute

Lemma (LCE Sample Complexity - informal)

For $t \geq \left\lfloor \frac{n^2}{k(n-k)} \right\rfloor + 1$, then t-LCE is solvable with non-negligible probability in time $O(n^{2\omega})$ for some constant $\omega \in [2, 3]$.

For $k = \frac{n}{2}$, we have $t \geq 5$.

Lemma (MCE Sample Complexity - informal)

For $t \geq \left\lfloor \frac{m^2 r^2}{k(mr-k)} \right\rfloor + 1$, then t-MCE is solvable with overwhelming probability in time $O((mr)^{2\omega})$ for some constant $\omega \in [2, 3]$.

For $m = r = k$, we have $t \geq \lfloor \frac{k^2}{k-1} \rfloor + 1 \geq k + 1$.

Sample complexity

Lemma (LCE Sample Complexity - informal)

For $t \geq \left\lfloor \frac{n^2}{k(n-k)} \right\rfloor + 1$, then t-LCE is solvable with non-negligible probability in time $O(n^{2\omega})$ for some constant $\omega \in [2, 3]$.

For $k = \frac{n}{2}$, we have $t \geq 5$.

Lemma (MCE Sample Complexity - informal)

For $t \geq \left\lfloor \frac{m^2 r^2}{k(mr-k)} \right\rfloor + 1$, then t-MCE is solvable with overwhelming probability in time $O((mr)^{2\omega})$ for some constant $\omega \in [2, 3]$.

For $m = r = k$, we have $t \geq \lfloor \frac{k^2}{k-1} \rfloor + 1 \geq k + 1$.

With only two samples, one obtains an underdetermined linear system; hence, the secret matrix $\mathbf{Q}$ cannot be recovered via Gaussian elimination.

Solving 2-LCE and ILCE for $k = n/2$

With only two samples, one obtains an underdetermined linear system; hence, the secret matrix **Q** cannot be recovered via Gaussian elimination.

We propose an algorithm for solving 2-ILCE, which takes inspiration from Saeed's work [11].

- Guess some unknown variables $\mathbf{Q}_{ij}$ by exploiting the monomial structure.
- Check whether the obtained reduced system accepts (or not) a solution.
- Retrieve the remaining variables using Gaussian elimination.

Eliminate variables

Solving 2-LCE and ILCE for $k = n/2$

With only two samples, one obtains an underdetermined linear system; hence, the secret matrix $\mathbf{Q}$ cannot be recovered via Gaussian elimination.

We propose an algorithm for solving 2-ILCE, which takes inspiration from Saeed's work [11].

$$\left.\begin{array}{l}\bullet\ \text{Guess some unknown variables } \mathbf{Q}_{ij} \text{ by exploiting the monomial structure.}\\ \bullet\ \text{Check whether the obtained reduced system accepts (or not) a solution.}\end{array}\right\} \text{Eliminate variables}$$

- Retrieve the remaining variables using Gaussian elimination.

Solving 2-LCE and ILCE for $k = n/2$

With only two samples, one obtains an underdetermined linear system; hence, the secret matrix $\mathbf{Q}$ cannot be recovered via Gaussian elimination.

We propose an algorithm for solving 2-ILCE, which takes inspiration from Saeed's work [11].

$$\left.\begin{array}{l} \bullet \text{ Guess some unknown variables } \mathbf{Q}_{ij} \text{ by exploiting the monomial structure.} \\ \bullet \text{ Check whether the obtained reduced system accepts (or not) a solution.} \end{array}\right\} \text{Eliminate variables}$$

- Retrieve the remaining variables using Gaussian elimination.

Core idea of the algorithm

→ Guessing a non-zero entry of the monomial **Q** corresponds to eliminating $2n-1$ (specific) columns from **A** (and variables from vec(**Q**)).

$$\begin{bmatrix} 0 & 0 & 0 & 0 & a & 0 & 0 \\ b & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & c & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & d \\ 0 & 0 & 0 & 0 & 0 & e & 0 \\ 0 & 0 & 0 & f & 0 & 0 & 0 \\ 0 & g & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

→ From the guessing at entry (i, j), one obtains a *reduced* linear system

$$\mathbf{A}_{ij} \cdot \mathrm{vec}(\mathbf{Q}') = \mathbf{b}_{ij}$$

where $\mathbf{Q}'$ is the $(n-1) \times (n-1)$ resulting secret matrix. The vector $\mathbf{b}_{ij}$ corresponds with the column of **A** determined by the non-zero entry, and $\mathbf{A}_{ij}$ has $\frac{n^2}{2}$ rows and $(n-1)^2$ columns.

Core idea of the algorithm

→ Guessing a non-zero entry of the monomial **Q** corresponds to eliminating $2n-1$ (specific) columns from **A** (and variables from vec(**Q**)).

$$\begin{bmatrix} 0 & 0 & 0 & 0 & a & 0 & 0 \\ b & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & c & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & d \\ 0 & 0 & 0 & 0 & 0 & e & 0 \\ 0 & 0 & 0 & f & 0 & 0 & 0 \\ 0 & g & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

→ From the guessing at entry (i, j), one obtains a *reduced* linear system

$$\mathbf{A}_{ij} \cdot \mathrm{vec}(\mathbf{Q}') = \mathbf{b}_{ij}$$

where $\mathbf{Q}'$ is the $(n-1) \times (n-1)$ resulting secret matrix. The vector $\mathbf{b}_{ij}$ corresponds with the column of **A** determined by the non-zero entry, and $\mathbf{A}_{ij}$ has $\frac{n^2}{2}$ rows and $(n-1)^2$ columns.

Sketch of the algorithm

$$\mathbf{A} \cdot \mathrm{vec}(\mathbf{Q}) = \mathbf{0}$$ $\leftarrow$ initial linear system

$$\Downarrow$$ $\leftarrow$ make guess on entry(i, j)

$$\mathbf{A_{ij}} \cdot \mathrm{vec}(\mathbf{Q}') = \mathbf{b_{ij}}$$ $\leftarrow$ reduced linear system

$$\mathrm{rank}(\mathbf{A_{ij}}) \stackrel{?}{=} \mathrm{rank}(\mathbf{A_{ij}}|\mathbf{b_{ij}})$$ $\leftarrow$ Rouché-Capelli test

Sketch of the algorithm

$$\mathbf{A} \cdot \mathrm{vec}(\mathbf{Q}) = \mathbf{0}$$ ← initial linear system

$$\mathrm{rank}(\mathbf{A}) = \frac{n^2}{2}$$

$$\Downarrow$$ ← make guess on entry(i, j)

$$\mathbf{A}_{\mathbf{ij}} \cdot \mathrm{vec}(\mathbf{Q}') = \mathbf{b}_{\mathbf{ij}}$$ ← reduced linear system

$$\mathrm{rank}(\mathbf{A}_{\mathbf{ij}}) \stackrel{?}{=} \mathrm{rank}(\mathbf{A}_{\mathbf{ij}} | \mathbf{b}_{\mathbf{ij}})$$ ← Rouché-Capelli test

$$\mathrm{rank}(\mathbf{A}_{\mathbf{ij}}) = \frac{n^2}{2} - 1$$

$$\mathrm{rank}(\mathbf{A}_{\mathbf{ij}} | \mathbf{b}_{\mathbf{ij}}) = \begin{cases} \frac{n^2}{2} - 1 & \text{accept} \\ \frac{n^2}{2} & \text{reject} \end{cases}$$

→ The probability of accepting a *wrong* guess is $\approx \frac{1}{q}$.

→ There are n *correct* guesses that always pass the Rouché-Capelli test.

→ There are $n^2 - n$ wrong guesses that might pass the Rouché-Capelli test.

→ Therefore, the expected number of survivals (missing unknown variables) is

$$n + (n^2 - n)\frac{1}{q}.$$

→ Consequently, we can recover the secret monomial matrix $\mathbf{Q}$ when

$$n + (n^2 - n)\frac{1}{q} \leq \frac{n^2}{2} \Rightarrow q \geq \frac{2(n-1)}{n-2}.$$

→ Complexity:

- Rouché-Capelli test takes $O(n^{2\omega})$ field operations.
- We need to perform n^2 guesses.

The total complexity is then $O(n^{2+2\omega})$ for some constant $\omega \in [2, 3]$.

→ The probability of accepting a *wrong* guess is $\approx \frac{1}{q}$.
→ There are n *correct* guesses that always pass the Rouché-Capelli test.
→ There are $n^2 - n$ wrong guesses that might pass the Rouché-Capelli test.
→ Therefore, the expected number of survivals (missing unknown variables) is

$$n + (n^2 - n)\frac{1}{q}.$$

→ Consequently, we can recover the secret monomial matrix **Q** when

$$n + (n^2 - n)\frac{1}{q} \leq \frac{n^2}{2} \Rightarrow q \geq \frac{2(n-1)}{n-2}.$$

→ Complexity:

- Rouché-Capelli test takes $O(n^{2\omega})$ field operations.
- We need to perform n^2 guesses.

The total complexity is then $O(n^{2+2\omega})$ for some constant $\omega \in [2, 3]$.

- The probability of accepting a *wrong* guess is $\approx \frac{1}{q}$.
- There are n *correct* guesses that always pass the Rouché-Capelli test.
- There are $n^2 - n$ wrong guesses that might pass the Rouché-Capelli test.
- Therefore, the expected number of survivals (missing unknown variables) is

$$n + (n^2 - n)\frac{1}{q}.$$

- Consequently, we can recover the secret monomial matrix $\mathbf{Q}$ when

$$n + (n^2 - n)\frac{1}{q} \leq \frac{n^2}{2} \Rightarrow q \geq \frac{2(n-1)}{n-2}.$$

- Complexity:
  - Rouché-Capelli test takes $O(n^{2\omega})$ field operations.
  - We need to perform n^2 guesses.

  The total complexity is then $O(n^{2+2\omega})$ for some constant $\omega \in [2, 3]$.

Experiments on solving 2-LCE and ILCE for $k = n/2$

$q \backslash n$		16	24	32	40	$1 - \frac{1}{q}$
7	2-LCE	0.81	0.84	0.81	0.86	0.86
	ILCE	0.87	0.82	0.86	0.85	
11	2-LCE	0.92	0.87	0.93	0.87	0.91
	ILCE	0.91	0.93	0.89	0.90	
17	2-LCE	0.95	0.95	0.93	0.92	0.94
	ILCE	0.96	0.94	0.96	0.96	
31	2-LCE	0.96	0.99	0.96	0.95	0.97
	ILCE	0.94	0.96	0.98	0.98	

Table: The data corresponds to the number of solved instances divided by the total number of experiments (which is 100). The last column reports the expected success probability from our analysis, that is, the matrix **A** has full rank. In all the experiments, we have $k = n/2$.

n	q	Corresponding LCE bit security	Expected survival vars.	Measured survival vars.	Memory (GB)	Runtime	Ratio
64	17	35	305	288	16.96	07m 08s	17/20
	23	37	242	240	16.96	07m 00s	17/20
	31	38	196	191	16.96	07m 06s	20/20
	127	44	96	97	16.97	07m 02s	20/20
72	19	39	345	343	27.19	13m 27s	20/20
	23	40	297	291	27.19	13m 58s	17/20
	37	42	212	212	27.20	12m 50s	18/20
	127	47	113	113	27.21	13m 08s	20/20
80	19	41	416	417	41.48	21m 40s	18/20
	29	44	301	302	41.50	21m 48s	20/20
	41	46	236	228	41.49	18m 37s	18/20
	127	51	130	132	41.50	18m 09s	20/20
96	23	48	496	499	86.10	01h 04m	20/20
	31	51	393	392	86.10	01h 04m	19/20
	47	54	292	284	86.10	01h 04m	20/20
	127	58	169	169	86.09	01h 08m	20/20
128	31	63	656	639	272.06	06h 02m	20/20
	43	66	509	519	272.07	06h 02m	19/20
	61	69	397	397	272.06	05h 51m	19/20
	127	73	257	252	272.10	04h 39m	20/20

Table: Average of 20 iterations.

Direct implications of our work

- We reply to the question raised in [2]: ILCE is not secure $\Rightarrow$ no linkability in LCE-based ring signature.
- The distributed key-generation in the threshold-group action signature GRASS [4] instantiated with LCE is not secure. (The authors revised their work dropping the dependency on 2-LCE [3]).

	ID scheme / signature	Commitment	Linkable ring signature from [6, 2, 7]	Pseudo random function from [1]	Updatable encryption from [10]
LCE	✓	✓	✗	✗	✗
MCE	✓	✓	✓(?)	✗	✗

Table: Overview of the secure and insecure known instantiations of primitives constructed from LCE and MCE group actions. The symbols ✗ and ✓ denote that the corresponding primitive is insecure or remains secure. The symbol ✓(?) denotes that no specific attacks are known, but we suggest further investigation. The third column in the LCE setting concerns the cryptographic scenario when the code length doubles the code dimension.

- We reply to the question raised in [2]: ILCE is not secure $\Rightarrow$ no linkability in LCE-based ring signature.
- The distributed key-generation in the threshold-group action signature GRASS [4] instantiated with LCE is not secure. (The authors revised their work dropping the dependency on 2-LCE [3]).

	ID scheme / signature	Commitment	Linkable ring signature from [6, 2, 7]	Pseudo random function from [1]	Updatable encryption from [10]
LCE	✓	✓	✗	✗	✗
MCE	✓	✓	✓(?)	✗	✗

Table: Overview of the secure and insecure known instantiations of primitives constructed from LCE and MCE group actions. The symbols ✗ and ✓ denote that the corresponding primitive is insecure or remains secure. The symbol ✓(?) denotes that no specific attacks are known, but we suggest further investigation. The third column in the LCE setting concerns the cryptographic scenario when the code length doubles the code dimension.

Questions?

Thanks for attending!

[1] Navid Alamati, Luca De Feo, Hart Montgomery, and Sikhar Patranabis.
Cryptographic group actions and applications.
In Shiho Moriai and Huaxiong Wang, editors, *ASIACRYPT 2020, Part II*, volume 12492 of *LNCS*, pages 411–439. Springer, Cham, December 2020.

[2] Alessandro Barenghi, Jean-François Biasse, Tran Ngo, Edoardo Persichetti, and Paolo Santini.
Advanced signature functionalities from the code equivalence problem.
Int. J. Comput. Math. Comput. Syst. Theory, 7(2):112–128, 2022.

[3] Michele Battagliola, Giacomo Borin, Alessio Meneghetti, and Edoardo Persichetti.
Cutting the GRASS: Threshold GRoup action signature schemes.
Cryptology ePrint Archive, Report 2023/859, 2023.

[4] Michele Battagliola, Giacomo Borin, Alessio Meneghetti, and Edoardo Persichetti.
Cutting the GRASS: Threshold GRoup action signature schemes.
In Elisabeth Oswald, editor, *CT-RSA 2024*, volume 14643 of *LNCS*, pages 460–489. Springer, Cham, May 2024.

[5] Benjamin Benčina, Alessandro Budroni, Jesús-Javier Chi-Domínguez, and Mukul Kulkarni.
Properties of lattice isomorphism as a cryptographic group action.
In Markku-Juhani Saarinen and Daniel Smith-Tone, editors, *Post-Quantum Cryptography - 15th International Workshop, PQCrypto 2024, Part I*, pages 170–201. Springer, Cham, June 2024.

[6] Ward Beullens, Shuichi Katsumata, and Federico Pintore.
Calamari and Falafl: Logarithmic (linkable) ring signatures from isogenies and lattices.
In Shiho Moriai and Huaxiong Wang, editors, *ASIACRYPT 2020, Part II*, volume 12492 of *LNCS*, pages 464–492. Springer, Cham, December 2020.

[7] Tung Chou, Ruben Niederhagen, Edoardo Persichetti, Tovohery Hajatiana Randrianarisoa, Krijn Reijnders, Simona Samardjiska, and Monika Trimoska.
Take your MEDS: Digital signatures from matrix code equivalence.
In Nadia El Mrabet, Luca De Feo, and Sylvain Duquesne, editors, *AFRICACRYPT 23*, volume 14064 of *LNCS*, pages 28–52. Springer, Cham, July 2023.

[8] Jean-Marc Couveignes.
Hard homogeneous spaces.
Cryptology ePrint Archive, Report 2006/291, 2006.

[9] Giuseppe D'Alconzo and Antonio J. Di Scala.
Representations of group actions and their applications in cryptography.
Cryptology ePrint Archive, Report 2023/1247, 2023.

[10] Antonin Leroux and Maxime Roméas.
Updatable encryption from group actions.
In Markku-Juhani Saarinen and Daniel Smith-Tone, editors, *Post-Quantum Cryptography - 15th International Workshop, PQCrypto 2024, Part II*, pages 20–53. Springer, Cham, June 2024.

References III

[11] Mohamed Ahmed Saeed.
Algebraic Approach for Code Equivalence.
PhD thesis, Normandie Université, University of Khartoum, 2017.
Available at `https://theses.hal.science/tel-01678829v2`.