# MinRank Gabidulin encryption scheme on matrix codes

Adrien Vinçotte

In collaboration with Nicolas Aragon, Alain Couvreur, Victor Dyseryn and Philippe Gaborit

# Historical background of the rank metric

Constructions on rank metric:

- Introduction of "arithmetic distance" over $\mathbb{F}_q^{n\times n}$: Hua51
- Matrix codes with rank distance: Del78
- Rank metric for vectors over an extension $\mathbb{F}_{q^m}$: Gab85
- GPT cryptosystem: GPT91
  Many variations proposed, successfully attacked: Ove08
- LRPC cryptosystem: GMRZ13
  Light masking of a LRPC code (small structure)
- Propositions to the NIST in 2017: RQC and ROLLO submissions
  RQC: the Gabidulin code is public
  ROLLO: relies on LRPC cryptosystem

# McEliece frame

Consists on masking a structured code used for both encryption and decryption.

- **Advantage:** Small ciphertexts (especially if the code has strong decoding capacity)
- **Drawback:** Structured secret code, very large public key

# Use of Gabidulin codes in the McEliece frame

Gabidulin codes have strong decoding capacity, which implies small parameters. However, their strong structure makes them easy to characterize.

Structural attack: consists on distinguish the structure of the masked code.

In case of Gabidulin codes: Overbeck and Ourivski-Johnson attacks, which use the $\mathbb{F}_{q^m}$-linear structure of the code.

# Our contribution

**New masking:** Turn a Gabidulin code into a matrix code $\mathcal{C}_{mat}$ with coefficients on the base field $\mathbb{F}_q$, which breaks the $\mathbb{F}_{q^m}$-linearity. After hiding $\mathcal{C}_{mat}$, use a McEliece-like encryption frame adapted to matrix codes.

**Decoding of matrix codes:** relies on the MinRank problem.

Concrete parameters: System with small ciphertexts, and public key smaller than the classic McEliece scheme.

# $\gamma$-expansion

Let $\gamma = (\gamma_1, \ldots, \gamma_m) \in \mathcal{B}(\mathbb{F}_{q^m})$.

For every $x \in \mathbb{F}_{q^m}$, there exists an only vector $(x_1, ..., x_m) \in \mathbb{F}_q^m$ such that $x = \sum_{i=1}^{m} x_i \gamma_i$.

We can define $\gamma$-expansion as an application:

$$\Psi_\gamma : x \in \mathbb{F}_{q^m} \mapsto \begin{pmatrix} x_1 \\ \vdots \\ x_m \end{pmatrix} \in \mathbb{F}_q^m$$

# From vectors to matrices

$\Psi_\gamma$ extends naturally to a vector $\boldsymbol{x} \in \mathbb{F}_{q^m}^n$ and turns it into a matrix $\Psi_\gamma(\boldsymbol{x}) \in \mathbb{F}_q^{m\times n}$:

$$\Psi_\gamma : \boldsymbol{x} = (x_1, ..., x_n) \longrightarrow \begin{pmatrix} \vdots & & \vdots \\ \vdots & & \vdots \\ \Psi_\gamma(x_1) & & \Psi_\gamma(x_n) \\ \vdots & & \vdots \\ \vdots & & \vdots \end{pmatrix} \in \mathbb{F}_q^{m\times n}$$

## Definition: Rank metric

The *support* of $\boldsymbol{x} \in \mathbb{F}_{q^m}^n$ is the the $\mathbb{F}_q$-vector space spanned by its coordinates. The *rank* of $\boldsymbol{x}$ is the dimension of its support.

$$\mathrm{Supp}(\boldsymbol{x}) \stackrel{\text{def}}{=} \langle x_1, ..., x_n \rangle_q$$

$$\|\boldsymbol{x}\| \stackrel{\text{def}}{=} \dim(\langle x_1, ..., x_n \rangle_q) = \mathrm{rank}(\Psi_\gamma(\boldsymbol{x}))$$

## Definition: Rank metric

The *support* of $\boldsymbol{x} \in \mathbb{F}_{q^m}^n$ is the the $\mathbb{F}_q$-vector space spanned by its coordinates. The *rank* of $\boldsymbol{x}$ is the dimension of its support.

$$\mathrm{Supp}(\boldsymbol{x}) \stackrel{\text{def}}{=} \langle x_1, ..., x_n \rangle_q$$

$$\|\boldsymbol{x}\| \stackrel{\text{def}}{=} \dim(\langle x_1, ..., x_n \rangle_q) = \mathrm{rank}(\Psi_\gamma(\boldsymbol{x}))$$

Weight of a vector: independent of the basis $\gamma$.

For two bases $\beta$ and $\gamma$, if we denote $\boldsymbol{P}$ the transition matrix between $\beta$ and $\gamma$, we get:

$$\Psi_\gamma(\boldsymbol{x}) = \boldsymbol{P}\, \Psi_\beta(\boldsymbol{x})$$

# Matrix codes

## Definition: Matrix code

A matrix code $\mathcal{C}_{mat}$ is an $\mathbb{F}_q$-subspace of $\mathbb{F}_q^{m \times n}$ endowed with the rank metric.

# Matrix codes

**Definition: Matrix code**

A matrix code $\mathcal{C}_{mat}$ is an $\mathbb{F}_q$-subspace of $\mathbb{F}_q^{m\times n}$ endowed with the rank metric.

Let $\mathcal{C}_{vec}$ be an $\mathbb{F}_{q^m}$-linear vector code of parameters $[n,k]_{q^m}$.
Turn $\mathcal{C}_{vec}$ into a matrix code:

$$\mathcal{C}_{mat} \stackrel{\text{def}}{=} \Psi_\gamma(\mathcal{C}_{vec}) = \{\Psi_\gamma(\boldsymbol{x}) \,|\, \boldsymbol{x} \in \mathcal{C}_{vec}\}.$$

$\mathcal{C}_{mat}$ is a matrix code of parameters $[m \times n, mk]_q$

- **Size of matrices:** $m \times n$ by definition of $\Psi_\gamma$.
- **Dimension:** $\mathcal{C}_{vec}$ is $\mathbb{F}_{q^m}$-linear, then for every $\boldsymbol{x} \in \mathcal{C}_{vec}$ and $\alpha \in \mathbb{F}_{q^m}$, we have $\Psi_\gamma(\alpha \boldsymbol{x}) \in \mathcal{C}_{mat}$. Then $\mathcal{C}_{mat}$ has dimension $mk$.

# Encoding and Decoding in matrix codes

Let $\mathcal{C}_{mat}$ be a $[m \times n, K]_q$ matrix code of basis $(\boldsymbol{M}_1, ..., \boldsymbol{M}_K)$.
To encode $\boldsymbol{x} \in \mathbb{F}_q^K$, sample an matrix $\boldsymbol{E} \in \mathbb{F}_q^{m \times n}$ of rank at most $r$ and compute:

$$\boldsymbol{Y} = \sum_{i=1}^{K} x_i \boldsymbol{M}_i + \boldsymbol{E}$$

# Encoding and Decoding in matrix codes

Let $\mathcal{C}_{mat}$ be a $[m \times n, K]_q$ matrix code of basis $(\boldsymbol{M}_1, ..., \boldsymbol{M}_K)$.
To encode $\boldsymbol{x} \in \mathbb{F}_q^K$, sample an matrix $\boldsymbol{E} \in \mathbb{F}_q^{m \times n}$ of rank at most $r$ and compute:

$$\boldsymbol{Y} = \sum_{i=1}^{K} x_i \boldsymbol{M}_i + \boldsymbol{E}$$

The decoding problem is exactly the well-known MinRank problem.

## MinRank$(q, m, n, K, r)$ problem

Given as input matrices $\boldsymbol{Y}, \boldsymbol{M}_1, \ldots, \boldsymbol{M}_K \in \mathbb{F}_q^{m \times n}$, the problem asks to find $x_1, \ldots, x_K \in \mathbb{F}_q$ and $\boldsymbol{E} \in \mathbb{F}_q^{m \times n}$ with $\operatorname{rank} \boldsymbol{E} \leq r$ such that $\boldsymbol{Y} = \sum_{i=1}^{K} x_i \boldsymbol{M}_i + \boldsymbol{E}$.

# Folding

Fold: turns a vector to a matrix.

$$\text{Fold}: \quad \mathbb{F}_q^{mn} \longrightarrow \mathbb{F}_q^{m\times n}$$

$$(v_1^{(1)},...,v_m^{(1)},...,v_1^{(n)},...,v_m^{(n)}) \longrightarrow \begin{pmatrix} v_1^{(1)} & v_1^{(n)} \\ \vdots & \vdots \\ v_m^{(1)} & v_m^{(n)} \end{pmatrix}$$

Unfold: inverse map which turns a matrix into a vector

# Vectorial representation of a matrix code

Let $(\boldsymbol{M}_1, ..., \boldsymbol{M}_K)$ a basis of a $[m \times n, K]_q$ matrix code $\mathcal{C}_{mat}$. We can define $\mathcal{C}_{mat}$ with an only generator matrix:

$$\boldsymbol{G} = \begin{pmatrix} \cdots & \cdots & \mathsf{Unfold}(\boldsymbol{M}_1) & \cdots & \cdots \\ \cdots & \cdots & \mathsf{Unfold}(\boldsymbol{M}_2) & \cdots & \cdots \\ & & \vdots & & \\ \cdots & \cdots & \mathsf{Unfold}(\boldsymbol{M}_K) & \cdots & \cdots \end{pmatrix} \in \mathbb{F}_q^{K \times mn}.$$

Allows to compute a parity check-matrix $\boldsymbol{H} \in \mathbb{F}_q^{(mn-K) \times mn}$ and define the dual code $\mathcal{C}_{mat}^{\perp}$.

# Dual of matrix code

Note that for two matrices $\boldsymbol{A}, \boldsymbol{B} \in \mathbb{F}_q^{m \times n}$, then:

$$\langle \mathsf{Unfold}(\boldsymbol{A}), \mathsf{Unfold}(\boldsymbol{B}) \rangle = \mathrm{tr}(\boldsymbol{A}\boldsymbol{B}^t)$$

## Definition: Dual code

Let be $\mathcal{C}_{mat}$ a matrix code of size $m \times n$ and dimension $K$. Its dual is the matrix code of size $m \times n$ and dimension $mn - K$:

$$\mathcal{C}_{mat}^{\perp} = \left\{ \boldsymbol{Y} \in \mathbb{F}_q^{m \times n} \mid \forall \boldsymbol{X} \in \mathcal{C}_{mat} \ \ \mathrm{tr}(\boldsymbol{X}\boldsymbol{Y}^t) = 0 \right\}.$$

# Syndrome decoding of a matrix code

Let $\boldsymbol{H} \in \mathbb{F}_q^{(mn-K)\times mn}$ a parity check matrix of the dual code. We write $(\boldsymbol{h}_i)_{1\leq i\leq mn}$ its columns.

Syndrome associated to a matrix word $\boldsymbol{Y}$:

$$\begin{aligned}\boldsymbol{s} &= \mathsf{Unfold}(\boldsymbol{Y})\boldsymbol{H}^t \\ &= \sum_{i=1}^{K} x_i \mathsf{Unfold}(\boldsymbol{M}_i)\boldsymbol{H}^t + \mathsf{Unfold}(\boldsymbol{E})\boldsymbol{H}^t \\ &= \mathsf{Unfold}(\boldsymbol{E})\boldsymbol{H}^t \\ &= \sum_{i=1}^{mn} e_i \boldsymbol{h}_i^t\end{aligned}$$

Retrieve the error $\boldsymbol{E}$ from $\boldsymbol{s}$ and $\boldsymbol{H}$: equivalent to solve the MinRank problem.

**Definition: MinRank-Syndrome problem**

Given as input vectors $\boldsymbol{s}, \boldsymbol{v}_1, \ldots, \boldsymbol{v}_{mn} \in \mathbb{F}_q^{mn-K}$, the MinRank-Syndrome$(q, m, n, K, r)$ problem asks to find $(e_1, \ldots, e_{mn}) \in \mathbb{F}_q^{mn}$ with $\operatorname{rank} \mathsf{Fold}(\boldsymbol{e}) \leq r$ such that $\boldsymbol{s} = \sum_{i=1}^{mn} e_i \boldsymbol{v}_i$.

# $q$-polynomials

Let $x \in \mathbb{F}_{q^m}$. We define: $x^{[i]} = x^{q^i}$.

## Definition: q-polynomial

q-polynomial of q-degree r:

$$P(X) = \sum_{i=0}^{r} p_i X^{[i]} \in \mathbb{F}_{q^m}[X] \quad \text{with } p_r \neq 0$$

We denote $q$-degree by $\deg_q$.

# Gabidulin codes

## Definition: Gabidulin code

Let $k, m, n \in \mathbb{N}$, such that $k \leq n \leq m$. Let $\boldsymbol{g} = (g_1, \ldots, g_n) \in \mathbb{F}_{q^m}^n$ a vector of $\mathbb{F}_q$-linearly independent elements of $\mathbb{F}_{q^m}$. The Gabidulin code $\mathcal{G}_{\boldsymbol{g}}(n, k, m)$ is the vector code of parameters $[n, k]_{q^m}$ defined by:

$$\mathcal{G}_{\boldsymbol{g}}(n, k, m) = \left\{ P(\boldsymbol{g}) |\ \deg_q P < k \right\},$$

where $P(\boldsymbol{g}) = (P(g_1), \ldots, P(g_n))$ and $P$ is a $q$-polynomial.

# Gabidulin codes

## Definition: Gabidulin code

Let $k, m, n \in \mathbb{N}$, such that $k \leq n \leq m$. Let $\boldsymbol{g} = (g_1, \ldots, g_n) \in \mathbb{F}_{q^m}^n$ a vector of $\mathbb{F}_q$-linearly independent elements of $\mathbb{F}_{q^m}$. The Gabidulin code $\mathcal{G}_{\boldsymbol{g}}(n, k, m)$ is the vector code of parameters $[n, k]_{q^m}$ defined by:

$$\mathcal{G}_{\boldsymbol{g}}(n, k, m) = \left\{ P(\boldsymbol{g}) | \ \deg_q P < k \right\},$$

where $P(\boldsymbol{g}) = (P(g_1), \ldots, P(g_n))$ and $P$ is a $q$-polynomial.

Decoding capacity $= \left\lfloor \frac{n-k}{2} \right\rfloor$

Generator matrix: $\boldsymbol{G} = \begin{pmatrix} g_1 & g_2 & \cdots & g_n \\ g_1^{[1]} & g_2^{[1]} & \cdots & g_n^{[1]} \\ \vdots & \vdots & & \vdots \\ g_1^{[k-1]} & g_2^{[k-1]} & \cdots & g_n^{[k-1]} \end{pmatrix} \in \mathbb{F}_{q^m}^{k \times n}$

# GPT cryptosystem: hide the structure of the code

Principle: take a generator matrix $\boldsymbol{G}$ of a $[n,k]_{q^m}$ Gabidulin code, scrambling it and then publishing the scrambled form.

Public key: Add to $\boldsymbol{G}$ a matrix $\boldsymbol{X}$ of random coefficients, and multiply by invertible matrices $\boldsymbol{S}$ and $\boldsymbol{P}$.

$$\boldsymbol{G}_{pub} = \boldsymbol{S}(\boldsymbol{X}|\boldsymbol{G})\boldsymbol{P} \in \mathbb{F}_{q^m}^{k\times(n+\ell)}$$

$$\mathsf{sk} = (\boldsymbol{G}, \boldsymbol{S}, \boldsymbol{P})$$

# GPT cryptosystem: encryption and decryption

Encryption of $\boldsymbol{m} \in \mathbb{F}_{q^m}^k$: sample a vector error $\boldsymbol{e}$ of small rank, and return $\boldsymbol{c} = \boldsymbol{m}\boldsymbol{G}_{pub} + \boldsymbol{e}$.

Decryption of $\boldsymbol{c}$:
$\boldsymbol{c}\boldsymbol{P}^{-1} = \boldsymbol{m}\boldsymbol{S}(\boldsymbol{X}|\boldsymbol{G}) + \boldsymbol{e}\boldsymbol{P}^{-1}$.
Truncate the $\ell$ first coefficients and apply the decoding algorithm on $\boldsymbol{c}\boldsymbol{P}^{-1}$ allows to retrieve $\mu = \boldsymbol{m}\boldsymbol{S}$.

# An attack by Overbeck against the GPT scheme

Let $\boldsymbol{x} = (x_1, \ldots, x_n) \in \mathbb{F}_{q^m}^n$. We define: $\boldsymbol{x}^{[i]} = (x_1^{q^i}, \ldots, x_n^{q^i})$.

## $f$-th Frobenius sum

Let $\mathcal{C}$ be an $[n, k]_{q^m}$ linear vector code. We define the $f$-th Frobenius sum of $\mathcal{C}$ as:

$$\Lambda_f(\mathcal{C}) = \mathcal{C} + \mathcal{C}^{[1]} + \cdots + \mathcal{C}^{[f]}.$$

$$\Lambda_f(\boldsymbol{G}) = \begin{pmatrix} \boldsymbol{G} \\ \boldsymbol{G}^{[1]} \\ \vdots \\ \boldsymbol{G}^{[f]} \end{pmatrix} \in \mathbb{F}_{q^m}^{(f+1)k \times n}.$$

# An attack by Overbeck against the GPT scheme

If $\mathcal{C}$ is random $[n,k]_{q^m}$ linear code, for $f \geq 0$, with high probability:

$$\dim \Lambda_f(\mathcal{C}) = \min\{n, k(f+1)\}.$$

If $\mathcal{G}$ is an $[n,k]_{q^m}$ Gabidulin code, for $f \geq 0$:

$$\dim \Lambda_f(\mathcal{G}) = \min\{n, k+f\}.$$

# MinRank-McEliece frame: Keygen

KeyGen $(1^\lambda)$:

- Select a matrix code $\mathcal{C}_{mat}$ of size $m \times n$ and dimension $K$ on $\mathbb{F}_q$, with an efficient algorithm capable of decoding up to $r$ errors.
- Let $\mathcal{T}$ a transformation which turns a matrix code into an other one with a trapdoor. Define the code $\mathcal{C}'_{mat} = \mathcal{T}(\mathcal{C}_{mat})$.
- Compute $\mathcal{B} = (\boldsymbol{M}_1, ..., \boldsymbol{M}_K)$ a basis of $\mathcal{C}'_{mat}$.
- Return $\mathsf{pk} = \mathcal{B}$ and $\mathsf{sk} = \left(\mathcal{C}_{mat}, \mathcal{T}^{-1}\right)$.

Figure: MinRank-McEliece encryption frame - Keygen algorithm

# MinRank-McEliece frame: Encryption and Decryption

$\mathsf{Encrypt}(\mathsf{pk}, \mu)$:

*Input:* $\mathsf{pk} = (\boldsymbol{M}_1, ..., \boldsymbol{M}_K)$, $\mu \in \mathbb{F}_q^K$.

- Sample uniformly at random a matrix $\boldsymbol{E} \in \mathbb{F}_q^{m \times n}$ such that $\operatorname{rank} \boldsymbol{E} \leq r$.
- Return $\boldsymbol{Y} = \sum_{i=1}^{K} \mu_i \boldsymbol{M}_i + \boldsymbol{E}$.

Figure: MinRank-McEliece encryption frame - Encryption algorithm

# MinRank-McEliece frame: Encryption and Decryption

Encrypt(pk, $\mu$):

*Input:* pk $= (\boldsymbol{M}_1, ..., \boldsymbol{M}_K)$, $\mu \in \mathbb{F}_q^K$.

- Sample uniformly at random a matrix $\boldsymbol{E} \in \mathbb{F}_q^{m \times n}$ such that rank $\boldsymbol{E} \leq r$.
- Return $\boldsymbol{Y} = \sum_{i=1}^{K} \mu_i \boldsymbol{M}_i + \boldsymbol{E}$.

Figure: MinRank-McEliece encryption frame - Encryption algorithm

Decrypt(sk, $\boldsymbol{Y}$):

- Compute $\tilde{\boldsymbol{Y}} = \mathcal{T}^{-1}(\boldsymbol{Y})$.
- Apply the decoding algorithm of $\mathcal{C}_{mat}$ on the matrix $\tilde{\boldsymbol{Y}}$ to retrieve the message $\mu$.

Figure: MinRank-McEliece encryption frame - Decryption algorithm

# MinRank-Niederreiter frame: Keygen

KeyGen $(1^\lambda)$:

- Select a matrix code $\mathcal{C}_{mat}$ of size $m \times n$ and dimension $K$ on $\mathbb{F}_q$, with an efficient algorithm capable of decoding up to $r$ errors.
- Let $\mathcal{T}$ a transformation which turns a matrix code into an other one with a trapdoor. Define $\mathcal{C}'_{mat} = \mathcal{T}(\mathcal{C}_{mat})$.
- Compute $\bar{\boldsymbol{H}} \in \mathcal{M}_{(mn-K)\times mn}(\mathbb{F}_q)$ a parity check matrix of $\mathcal{C}'_{mat}$.
- Return $\mathbf{pk} = \bar{\boldsymbol{H}}$ and $\mathsf{sk} = \left(\mathcal{C}_{mat}, \mathcal{T}^{-1}\right)$.

Figure: MinRank-McEliece encryption frame - Keygen algorithm

# MinRank-Niederreiter frame: Encryption and Decryption

$\mathsf{Encrypt}(\mathsf{pk}, \mu)$:

*Input:* $\mathsf{pk} = \bar{\boldsymbol{H}}$, a message $\mu \in \mathbb{F}_q^{nm}$ such that $\operatorname{rank} \mathsf{Fold}(\mu) \leq r$.

- For every integer $i$ from 1 to $nm$, let $\boldsymbol{h}_i$ the i-th column of $\bar{\boldsymbol{H}}$.
- Return $\boldsymbol{c} = \sum_{i=1}^{nm} \mu_i \boldsymbol{h}_i^t$.

Figure: MinRank-Niederreiter encryption frame - Encryption algorithm

# MinRank-Niederreiter frame: Encryption and Decryption

$\mathsf{Encrypt}(\mathsf{pk}, \mu)$:

*Input:* $\mathsf{pk} = \bar{\boldsymbol{H}}$, a message $\mu \in \mathbb{F}_q^{nm}$ such that $\operatorname{rank} \mathsf{Fold}(\mu) \leq r$.

- For every integer $i$ from 1 to $nm$, let $\boldsymbol{h}_i$ the i-th column of $\bar{\boldsymbol{H}}$.
- Return $\boldsymbol{c} = \sum_{i=1}^{nm} \mu_i \boldsymbol{h}_i^t$.

Figure: MinRank-Niederreiter encryption frame - Encryption algorithm

$\mathsf{Decrypt}(\mathsf{sk}, \boldsymbol{c})$:

*Input:* $\mathsf{sk} = (\mathcal{C}_{mat}, \boldsymbol{P}, \boldsymbol{Q})$, $\boldsymbol{c} \in \mathbb{F}_{q^m}^{nm-K}$.

- Find any $\boldsymbol{y} \in \mathbb{F}_q^{nm}$ such that $\boldsymbol{c} = \sum_{i=1}^{nm} \bar{y}_i \boldsymbol{h}_i^t$.
- Let $\boldsymbol{Y} = \mathcal{T}^{-1}(\mathsf{Fold}(\boldsymbol{y}))$. Apply the decoding algorithm of $\mathcal{C}_{mat}$ on the matrix $\boldsymbol{Y}$ to retrieve the error $\mu$.

Figure: MinRank-Niederreiter encryption frame - Decryption algorithm

# Our masking: Random Rows and Columns Matrix Code transformation

Let $\mathcal{B} = (\boldsymbol{A}_1, ..., \boldsymbol{A}_K)$ a basis a matrix $\mathcal{C}_{mat}$ of size $m \times n$ and dimension $K$. How we propose to hide $\mathcal{C}_{mat}$:

- Add $\ell_1$ rows and $\ell_2$ columns of random coefficients: represented by matrices $\boldsymbol{R}_i$, $\boldsymbol{R}'_i$ and $\boldsymbol{R}''_i$.
- Scrambler matrices: multiply by invertible matrices $\boldsymbol{P}$ and $\boldsymbol{Q}$.

$$\mathcal{B}' = \left( \boldsymbol{P} \left( \begin{array}{c|c} \boldsymbol{A}_1 & \boldsymbol{R}_1 \\ \hline \boldsymbol{R}'_1 & \boldsymbol{R}''_1 \end{array} \right) \boldsymbol{Q}, \ldots, \boldsymbol{P} \left( \begin{array}{c|c} \boldsymbol{A}_K & \boldsymbol{R}_K \\ \hline \boldsymbol{R}'_K & \boldsymbol{R}''_K \end{array} \right) \boldsymbol{Q} \right).$$

Trapdoor: relies on MinRank and Code Equivalence problems.

# Enhanced Gabidulin matrix code

Let $\mathcal{G}_{\boldsymbol{g}}$ a Gabidulin code $[n, k, r]$ on $\mathbb{F}_{q^m}$, $\gamma$ a $\mathbb{F}_q$-basis of $\mathbb{F}_{q^m}$.

Enhanced Gabidulin code: matrix code $\Psi_\gamma(\mathcal{G}_{\boldsymbol{g}})$ on which we apply the Random Rows and Columns matrix code transformation.

It follows $\mathcal{EG}_{\boldsymbol{g}}(n, k, m, \ell_1, \ell_2)$: a matrix code of size $(m + \ell_1) \times (n + \ell_2)$ and dimension $km$.

# Application of the MinRank-McEliece frame with our masking

We apply the MinRank-McEliece frame to matrix Gabidulin codes, using the RRCMC previously defined.

KeyGen $(1^\lambda)$:

- Select an $[m,k]_{q^m}$ Gabidulin code $\mathcal{G}$, capable of decoding up to $r = \left\lfloor \frac{m-k}{2} \right\rfloor$ errors.
- Sample a basis $\gamma \xleftarrow{\$} \mathcal{B}(\mathbb{F}_{q^m})$ and compute a basis of the code $\mathcal{C}_{mat} = \Psi_\gamma(\mathcal{G})$.
- Apply the RRCMC transformation to $\Psi_\gamma(\mathcal{G})$, by sampling random matrices $\boldsymbol{R}_i, \boldsymbol{R}'_i, \boldsymbol{R}''_i$, and invertible matrices $\boldsymbol{P}, \boldsymbol{Q}$. Let be $\mathcal{C}'_{mat}$ the resulting matrix code.
- **Return:** $\mathsf{pk} = \mathcal{B}$ a basis of $\mathcal{C}'_{mat}$, $\mathsf{sk} = (\mathcal{G}, \gamma, \boldsymbol{P}, \boldsymbol{Q})$

Figure: EGMC-McEliece encryption scheme: KeyGen

# EGMC-McEliece encryption scheme: Encryption

The encryption relies on coding the message $\mu$ with the public code $\mathcal{C}_{mat}$.

Takes in input: $\mathsf{pk} = (\boldsymbol{M}_1, ..., \boldsymbol{M}_{km})$ a basis of $\mathcal{C}'_{mat}$, $\mu \in \mathbb{F}_q^{km}$.

Sample uniformly at random a matrix $\boldsymbol{E} \in \mathbb{F}_q^{(m+\ell_1)\times(m+\ell_2)}$ such that $\operatorname{rank} \boldsymbol{E} \leq r$.

Return the ciphertext:

$$\boldsymbol{Y} = \sum_{i=1}^{km} \mu_i \boldsymbol{M}_i + \boldsymbol{E}$$

# EGMC-McEliece encryption scheme: Decryption

Compute:

$$\boldsymbol{P}^{-1}\boldsymbol{Y}\boldsymbol{Q}^{-1} = \sum_{i=1}^{km} \mu_i \left( \begin{array}{c|c} \boldsymbol{A}_i & \boldsymbol{R}_i \\ \hline \boldsymbol{R}'_i & \boldsymbol{R}''_i \end{array} \right) + \boldsymbol{P}^{-1}\boldsymbol{E}\boldsymbol{Q}^{-1}$$

Truncate the $\ell_1$ last rows and $\ell_2$ last columns, be $\boldsymbol{M} \in \mathbb{F}_q^{m \times m}$ the resulting matrix.

The first $m$ coordinates of $\Psi_\gamma^{-1}(\boldsymbol{M}) \in \mathbb{F}_{q^m}^m$ form a noisy codeword of $\mathcal{G}$. Its decoding algorithm allow to retrieve the vector error $\boldsymbol{e}$.

By computing $\Psi_\gamma(\boldsymbol{e})$, we can consider the system $\boldsymbol{Y} = \sum_{i=1}^{km} \mu_i \boldsymbol{M}_i + \boldsymbol{E}$, whose unknowns are the $(\mu_i)$ and some coefficients of $\boldsymbol{E}$.

# OW-CPA security

Under the assumption that:

- there exists no PPT algorithm to solve the MinRank problem with non negligible probability
- there exists no PPT distinguisher for the problem which consists on distinguish a valid public key and a random matrix code with non negligible advantage

then the scheme is OW-CPA.

# EGMC-Niederreiter encryption scheme: KeyGen

The key generation is identical to that of the EGMC-McEliece encryption scheme: we compute $\mathcal{C}'_{mat}$ the RRCMC transformation of a code $\Psi_\gamma(\mathcal{G})$.

One difference: rather than a basis of $\mathcal{C}'_{mat}$, the public key is a parity check matrix $\bar{\boldsymbol{H}}$.

The secret key is still $(\gamma, \mathcal{G}, \boldsymbol{P}, \boldsymbol{Q})$.

# EGMC-Niederreiter encryption scheme: Encryption

As for classic Niederreiter, the message is a vector of small weight, and the ciphertext is the associated syndrome.

Message: $\boldsymbol{m} \in \mathbb{F}_q^{(m+\ell_1)(m+\ell_2)}$ such that $\operatorname{rank} \mathsf{Fold}(\boldsymbol{m}) \leq r$.

Let $\boldsymbol{h}_i$ the i-th column of $\bar{\boldsymbol{H}}$. Return:

$$\boldsymbol{c} = \sum_{i=1}^{(m+\ell_1)(m+\ell_2)} m_i \boldsymbol{h}_i^t$$

# EGMC-Niederreiter encryption scheme: Decryption

Begin to find any $\boldsymbol{y} \in \mathbb{F}_q^{(m+\ell_1)(m+\ell_2)}$ such that:

$$\boldsymbol{c} = \sum_{i=1}^{(m+\ell_1)(m+\ell_2)} y_i \boldsymbol{h}_i^t$$

We can apply the same algorithm than the EGMC-McEliece encryption scheme, the only difference being that the message is not the word, but the error.

# Attacks on the message

From $\boldsymbol{Y} = \sum_{i=1}^{km} \mu_i \boldsymbol{M}_i + \boldsymbol{E}$, with $\operatorname{rank} \boldsymbol{E} \leq r$, and $\mathsf{pk} = (\boldsymbol{M}_1, ..., \boldsymbol{M}_{km})$, retrieve $\mu$.

Equivalent to solve a generic $\mathrm{MinRank}(q, m + \ell_1, m + \ell_2, km, r)$ instance.

# Attacks on the MinRank problem

Main attacks on an instance MinRank$(q, m, n, K, r)$:

- Hybrid attack: solving smaller instances. Complexity:

$$\min_a q^{ar} \mathcal{A}(q, m, n-a, K-am, r)$$

- Kernel attack: combinatorial attack which consists on sampling vectors, hoping they are in the kernel of $\boldsymbol{E}$, and deducing a linear system of equations. Complexity:

$$O(q^{r\lceil \frac{K}{m} \rceil} K^{\omega})$$

- Support minors attack: rank $\left(\boldsymbol{Y} - \sum_{i=1}^{K} \mu_i \boldsymbol{M}_i\right) \leq r$. All the minors of size more than $r$ are equal to zero. We deduce a system of equations whose unknowns are the $(\mu_i)$.

# Attacks on the key

**Indistinguishability problem**

**Instance:** A matrix code $\mathcal{C}$ sampled from the $\mathsf{EGMC}(k, m, n, \ell_1, \ell_2)$ distribution, or sampled uniformly at random.

**Problem:** Guess from which distribution $\mathcal{C}$ has been sampled.

# Stabilizer algebra

## Left Stabilizer algebra

$$\mathrm{Stab}_L(\mathcal{C}_{mat}) \stackrel{\text{def}}{=} \left\{ \boldsymbol{P} \in \mathbb{F}_q^{m \times m} \mid \boldsymbol{P}\mathcal{C}_{mat} \subseteq \mathcal{C}_{mat} \right\}$$

We similarly define the Right Stabilizer algebra.

For every $\mathbb{F}_{q^m}$-linear $\mathcal{C}_{vec} \subseteq \mathbb{F}_{q^m}^n$, the code $\Psi_\gamma(\mathcal{C}_{vec})$ has a non trivial stabilizer algebra:

$$\dim \mathrm{Stab}_L(\Psi_\gamma(\mathcal{C}_{vec})) \geq m$$

# Distinguisher for matrix Gabidulin codes

## Proposition

Suppose that $n = m$, $\boldsymbol{g} = (g_1, ..., g_m) \in \mathbb{F}_{q^m}^m$, $\gamma$ an $\mathbb{F}_q$–basis of $\mathbb{F}_{q^m}$. Let $\mathcal{G} = \mathcal{G}_{\boldsymbol{g}}(m, k, m)$. Then:

$$\dim \mathrm{Stab}_R(\Psi_\gamma(\mathcal{G})) \geq m$$

For $P = p_0 X + p_1 X^q + \cdots + p_{k-1} X^{q^{k-1}}$, we have

$$P \circ \alpha X = p_0 \alpha X + p_1 \alpha^q X^q + \cdots + p_{k-1} \alpha^{q^{k-1}} X^{q^{k-1}}$$

Then $\Psi_\gamma(\mathcal{G}) = \Psi_\gamma(P(g_1), \cdots, P(g_m))$ is stabilized on the right by the matrix representing the multiplication by $\alpha$ in the basis $\boldsymbol{g}$.

# Combinatorial distinguisher against the $\mathbb{F}_{q^m}$-linear structure

Non scrambled version of the code, denoted $\mathcal{C}_0$, spanned by the basis:

$$\mathcal{B}_0 = \left( \begin{pmatrix} \boldsymbol{A}_1 & \boldsymbol{R}_1 \\ \boldsymbol{R}'_1 & \boldsymbol{R}''_1 \end{pmatrix}, \ldots, \begin{pmatrix} \boldsymbol{A}_{km} & \boldsymbol{R}_{km} \\ \boldsymbol{R}'_{km} & \boldsymbol{R}''_{km} \end{pmatrix} \right)$$

where $(\boldsymbol{A}_i)_i$ is a $\mathbb{F}_q$-basis of $\Psi_\gamma(\mathcal{G}_{\boldsymbol{g}}(n,k,m))$.

Idea: apply a projection map on both the row and columns spaces of $\mathcal{C}_{pub}$ in order to get rid of the contributions of the matrices $\boldsymbol{R}_i, \boldsymbol{R}'_i$ and $\boldsymbol{R}''_i$.

Choose two matrices:

- $\boldsymbol{U} \in \mathbb{F}_q^{m \times (m+\ell_1)}$ such that $\boldsymbol{U} = (\boldsymbol{U}_0 \mid \mathbf{0})$, with $\boldsymbol{U}_0 \in \mathbf{GL}_m(\mathbb{F}_q)$
- $\boldsymbol{V} \in \mathbb{F}_q^{(n+\ell_2) \times n'}$ such that
  $\boldsymbol{V} = \begin{pmatrix} \boldsymbol{V}_0 \\ \mathbf{0} \end{pmatrix}$, with $\boldsymbol{V}_0 \in \mathbb{F}_q^{n \times n'}$ of full rank and $k < n' \leq n$

Observation: the code $\boldsymbol{U}\mathcal{C}_0\boldsymbol{V}$ spanned by the $(\boldsymbol{U}_0\boldsymbol{A}_i\boldsymbol{V}_0)_i$.

Consequently: $\boldsymbol{U}\mathcal{C}_0\boldsymbol{V} = \Psi_{\gamma \boldsymbol{U}_0}(\mathcal{G}_{\boldsymbol{g}\boldsymbol{V}_0}(n', k, m))$

Number of choices for $\boldsymbol{U}, \boldsymbol{V}$ is $\approx q^{m^2+nn'}$. Being minimal when $n' = k + 1$.

The public code $\mathcal{C}_{pub}$ is spanned by:

$$\mathcal{B}' = \left( \boldsymbol{P} \begin{pmatrix} \boldsymbol{A}_1 & \boldsymbol{R}_1 \\ \boldsymbol{R}'_1 & \boldsymbol{R}''_1 \end{pmatrix} \boldsymbol{Q}, \ldots, \boldsymbol{P} \begin{pmatrix} \boldsymbol{A}_{km} & \boldsymbol{R}_{km} \\ \boldsymbol{R}'_{km} & \boldsymbol{R}''_{km} \end{pmatrix} \boldsymbol{Q} \right) = \boldsymbol{P}\mathcal{B}_0\boldsymbol{Q}$$

The same reasoning can be made replacing $\boldsymbol{U}$ by $\boldsymbol{U}' \stackrel{\text{def}}{=} \boldsymbol{U}\boldsymbol{P}^{-1}$ and $\boldsymbol{V}$ by $\boldsymbol{V}' \stackrel{\text{def}}{=} \boldsymbol{Q}^{-1}\boldsymbol{V}$.

The number of choices for $\boldsymbol{U}', \boldsymbol{V}'$ is still $\approx q^{m^2+n(k+1)}$.

The distinguisher consists in:

- Guess the pair $\boldsymbol{U}', \boldsymbol{V}'$ with $\boldsymbol{U}' \in \mathbb{F}_q^{m \times (m+\ell_1)}$ and $\boldsymbol{V}' \in \mathbb{F}_q^{(n+\ell_2)\times(k+1)}$,
- Compute the left stabilizer algebra of $\boldsymbol{U}'\mathcal{C}_{pub}\boldsymbol{V}'$,

until get a stabilizer algebra of dimension $\geq m$.

Probability of finding a valid pair $\boldsymbol{U}', \boldsymbol{V}'$ is

$$\mathbb{P} \approx \frac{q^{m^2+n(k+1)}}{q^{m(m+\ell_1)+(n+\ell_2)(k+1)}} = q^{-(m\ell_1+(k+1)\ell_2)}$$

which yields a complexity of $\widetilde{O}(q^{m\ell_1+(k+1)\ell_2})$.

# An Overbeck-like distinguisher

Let remember that for $\mathcal{G}$ a Gabidulin vector code, $\mathcal{G} + \mathcal{G}^{[1]} + \cdots + \mathcal{G}^{[t]}$ is small compared to the random case.

Difficulty for matrix codes: we don't have access to the Frobenius map if we do not know the basis $\gamma$ in which $\mathcal{G}$ has been expanded.

# An Overbeck-like distinguisher

Let remember that for $\mathcal{G}$ a Gabidulin vector code, $\mathcal{G} + \mathcal{G}^{[1]} + \cdots + \mathcal{G}^{[t]}$ is small compared to the random case.

Difficulty for matrix codes: we don't have access to the Frobenius map if we do not know the basis $\gamma$ in which $\mathcal{G}$ has been expanded.

**Matrix version of the Overbeck's distinguisher**

Let $b \in \mathbb{N}$, $\boldsymbol{M} \in \Psi_\gamma(\mathcal{G}_\gamma(m, b, m))$ and $\boldsymbol{C} \in \Psi_\gamma(\mathcal{G}_{\boldsymbol{g}}(m, k, n))$. Then,

$$\boldsymbol{MC} \in \Psi_\gamma(\mathcal{G}_{\boldsymbol{g}}(m, k + b - 1, n)).$$

From this property, we can construct a code of high dimension $m(n-k)+\ell_1(m+\ell_1)$:

$$\mathcal{D} \stackrel{\text{def}}{=} \left\{ \begin{pmatrix} \boldsymbol{B} & \boldsymbol{0} \\ \boldsymbol{T}_1 & \boldsymbol{T}_2 \end{pmatrix} \;\middle|\; \boldsymbol{B} \in \Psi_\gamma(\mathcal{G}_\gamma(m, n-k, m)),\ \boldsymbol{T}_1 \in \mathbb{F}_q^{\ell_1 \times m},\ \boldsymbol{T}_2 \in \mathbb{F}_q^{\ell_1 \times \ell_1} \right\}$$

such that the code:

$$\mathrm{Span}_{\mathbb{F}_q} \{ \boldsymbol{DC} \mid \boldsymbol{D} \in \mathcal{D},\ \boldsymbol{C} \in \mathcal{C}_0 \}$$

does not fill the ambient space.

Problem: the code $\mathcal{D}$ is unknown, and we still not consider the scrambling matrices $\boldsymbol{P}$ and $\boldsymbol{Q}$. Although the problem can be put into equations, the number of variables is far too large. We claim that our system remains beyond the reach of such a distinguisher.

# Attack the dual

Given an basis $\gamma$ for $\mathbb{F}_{q^m}$, denoting by $\gamma'$ the dual basis with respect to the trace inner product in $\mathbb{F}_{q^m}$, then:

$$\Psi_\gamma(\mathcal{G})^\perp = \Psi_{\gamma'}(\mathcal{G}^\perp)$$

Then:

$$\mathcal{U}_{n-k} \stackrel{\text{def}}{=} \left\{ (\boldsymbol{P}^t)^{-1} \begin{pmatrix} \boldsymbol{R} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{pmatrix} (\boldsymbol{Q}^t)^{-1} \middle| \boldsymbol{R} \in \Psi_{\gamma'}(\mathcal{G}^\perp) \right\} \subset \mathcal{C}_{mat}^\perp$$

Then, there exists $\mathcal{W}$ a subspace of dimension $n\ell_1 + m\ell_2 + \ell_1\ell_2$ such that:

$$\mathcal{C}_{mat}^{\perp} = \mathcal{U}_{n-k} \oplus \mathcal{W}$$

Reasoning on the dual implies to get rid of the contribution of $\mathcal{W}$. This is equivalent to guess the contributions of added random rows and columns to the basis of $\Psi_\gamma(\mathcal{G})$, what we did in the previous combinatorial attack.

# Choice of parameters

To summarize, there are two types of attacks to guard against:

- Attack on the message, which consists on solving a generic $\text{MinRank}(q, m+\ell_1, m+\ell_2, km, r)$ instance
- Attack on the key, which consists on distinguish a valid key from a random matrix code.

Once $m$ and $k$ have been chosen to resist the message attack, we choose $\ell_1$ and $\ell_2$ to resist to the key attack.

# Resulting parameters

| Sec. | $q$ | $k$ | $m$ | $\ell_1$ | $\ell_2$ | $r$ | pk | ct |
|---|---|---|---|---|---|---|---|---|
| 128 | 2 | 17 | 37 | 3 | 3 | 10 | 76 kB | 121 B |
| | 2 | 25 | 37 | 3 | 3 | 6 | 78 kB | 84 B |
| | 2 | 35 | 43 | 2 | 2 | 4 | 98 kB | 65 B |
| | 2 | 47 | 53 | 2 | 2 | 3 | 166 kB | 66 B |
| 192 | 2 | 51 | 59 | 2 | 2 | 4 | 268 kB | 89 B |
| 256 | 2 | 23 | 47 | 3 | 3 | 12 | 191 kB | 177 B |
| | 2 | 37 | 53 | 3 | 2 | 8 | 274 kB | 139 B |
| | 2 | 71 | 79 | 2 | 2 | 4 | 667 kB | 119 B |

Figure: Reference parameters for the EGMC-Niederreiter encryption scheme

# Comparison with other schemes

| Scheme | pk | ct |
|---|---|---|
| **EGMC-Niederreiter** | 98 kB | 65 B |
| Classic McEliece | 261 kB | 96 B |
| ROLLO I | 696 B | 696 B |
| KYBER | 800 B | 768 B |
| RQC-Block-NH-MS-AG | 312 B | 1118 B |
| BIKE | 1540 B | 1572 B |
| RQC-NH-MS-AG | 422 B | 2288 B |
| RQC | 1834 B | 3652 B |
| HQC | 2249 B | 4481 B |

Figure: Comparison of different schemes for 128 bits of security

# Conclusion and perspectives

# Thank you for your attention