Cryptanalysis of rank-metric schemes based on distorted Gabidulin codes

Pierre Briaud, Pierre Loidreau

Cryptanalysis of rank-metric schemes based on distorted Gabidulin codes

Pierre Briaud[1,2] and Pierre Loidreau[3]

[1] Sorbonne Universités, UPMC Univ Paris 06
[2] Inria, Team COSMIQ, Paris, France
pierre.briaud@inria.fr
[3] DGA and IRMAR, Univ. Rennes
pierre.loidreau@univ-rennes.fr

Abstract. In this work, we introduce a new attack for the Loidreau scheme [PQCrypto 2017] and its more recent variant LowMS. This attack is based on a constrained linear system for which we provide two solving approaches:

- the first one is an enumeration algorithm inspired from combinatorial attacks on the Rank Decoding (RD) Problem. While the attack technique remains very simple, it allows us to obtain the best known structural attack on the parameters of these two schemes.
- the second one is to rewrite it as a bilinear system over $\mathbb{F}_q$. Even if Gröbner basis techniques on this second system seem infeasible, we provide a detailed analysis of the first degree fall polynomials which arise when applying such algorithms.

1 Introduction

The idea of building rank-metric cryptography relying on Gabidulin codes is over 30 years old. It dates back to the seminal GPT scheme [15]. The initial goal of Gabidulin was to use the properties of the rank metric in order to propose a scheme with a public-key size one order of magnitude smaller than that of the original McEliece cryptosystem [20]. However, this proposal and following variants have suffered structural attacks [22] tending to show that masking these codes is difficult.

The Loidreau cryptosystem introduced in [19] is based on a different type of masking. Along with the LowMS variant [1], it is arguably one of the few reparations which resists cryptanalysis for well-chosen parameters.

On the one hand, this scheme offers nice features compared to other modern PKEs and especially to those proposed at the NIST post-quantum standardization process. First, decryption is deterministic. Second, regarding performance, the key is between one and two orders of magnitude smaller than that of non-structured Hamming-based cryptosystems. It even favorably compares with that of PKEs based on unstructured lattices. Similarly, the ciphertext is small compared to that of unstructured lattice proposals and it compares favourably with that of structured lattices.

On the other hand, its security analysis is not yet sufficiently stabilized. This is mainly due to the new type of masking, which calls for assessing the difficulty of distinguishing the public code from a random one. This code is a Gabidulin code distorted with a non-singular matrix with coefficients in a small-dimensional secret subspace of $\mathbb{F}_{q^m}$. A conjecture was made in [19] concerning the complexity of solving the problem, for parameters not impacted by the Coggia-Couvreur attack.

Contributions. First, we improve upon the enumeration approach of Loidreau presented as an extended abstract at the WCC 2022 conference. We adapt techniques from combinatorial attacks on RD [2] showing that it is more efficient to enumerate over vector spaces of larger dimension than that of the original secret subspace. This allows us to obtain the best complexity for this type of technique.

Second, we propose an algebraic approach to find a distinguisher by modeling the original problem as a bilinear system over $\mathbb{F}_q$. Even if the solving by Gröbner bases does not seem promising from our experiments, we manage to analyze precisely the first steps of the computation. In particular, we show that there exist degree falls of the same nature as in [4,6] due to the specific structure of the system.

2 Preliminaries on the rank metric

Rank-metric cryptography relies on codes which are $\mathbb{F}_{q^m}$-linear, where $\mathbb{F}_{q^m}$ is an extension of degree m over $\mathbb{F}_q$. In this context, the *rank* (or *weight*) of a vector $\mathbf{a} = (a_1, \ldots, a_n) \in \mathbb{F}_{q^m}^n$ denoted by $\mathrm{Rk}(\mathbf{a})$ is the dimension of the $\mathbb{F}_q$-subspace of $\mathbb{F}_{q^m}$ generated by the components of $\mathbf{a}$, *i.e.*,

$$\mathrm{Rk}(\mathbf{a}) \stackrel{def}{=} \dim\langle a_1, \ldots, a_n\rangle_{\mathbb{F}_q}.$$

Gabidulin codes were first constructed by Delsarte as extremal object in Bose-Mesner algebra [10]. Some years later, Gabidulin presented an algebraic theory as well as a polynomial-time decoding algorithm [14]. These codes can be viewed as analogues of Reed-Solomon codes in the rank metric, where polynomials are replaced by linearized polynomials.

Notation 1 *In the whole paper, we will denote by $(a_{i,j})_{1\leq i\leq n_r, 1\leq j\leq n_c}$ the $n_r \times n_c$ matrix whose entry in row i and column j is equal to $a_{i,j}$ for $i \in \{1..n_r\}$ and $j \in \{1..n_c\}$ or simply $(a_{i,j})$ when the sizes are already clear from the context.*

Definition 1 *For integers $k \leq n \leq m$, let $\mathbf{g} = (g_1, \ldots, g_m) \in \mathbb{F}_{q^m}^n$ such that $Rk(\mathbf{g}) = n$. The k-dimensional Gabidulin code with support vector $\mathbf{g}$, denoted $\mathcal{G}_k(\mathbf{g})$, is the $\mathbb{F}_{q^m}$-linear code generated by the matrix $(g_j^{[i-1]})_{1\leq i\leq k, 1\leq j\leq n}$, where $[i] \stackrel{def}{=} q^i$.*

Finally, the following proposition shows that the dual of a Gabidulin code is a Gabidulin code.

Proposition 1 ([14]) *Let $\mathcal{G}_k(\mathbf{g}) \subset \mathbb{F}_{q^m}^n$, then there exists $\mathbf{h} \in \mathbb{F}_{q^m}^n$ of rank n such that $\mathcal{G}_{n-k}(\mathbf{h}) = \mathcal{G}_k(\mathbf{g})^\perp$ for the usual scalar product in $\mathbb{F}_{q^m}$*

3 Loidreau cryptosystem

The Loidreau scheme was introduced in [19] with $q = 2$ but it can be be declined for any prime power q. For positive integers $m,\ n$ and an $\mathbb{F}_q$-vector space $\mathcal{A}$, let $\mathcal{M}_{m,n}(\mathcal{A})$ be the vector space of matrices of size $m \times n$ with entries in $\mathcal{A}$ and let $\mathrm{GL}_n(\mathbb{F}_{q^m})$ be the group of non-singular matrices of size n with entries in $\mathbb{F}_{q^m}$.

3.1 Description of the scheme

The parameters are integers $k \leq n \leq m$ related to the underlying Gabidulin code as well as $\lambda \in \mathbb{N}$ related to the masking. The value of λ is chosen such that $\lambda < \lfloor (n-k)/2 \rfloor$ for correctness and $\lambda \geq 3$ to avoid the polynomial attack of [8]. The three standard building blocks of a public encryption scheme are the following:

`KeyGen(`1^ν`)`

1. Construct $\mathcal{G} \subset \mathbb{F}_{q^m}^n$ a k-dimensional Gabidulin code.
2. Pick $\mathbf{G} \in \mathcal{M}_{k,n}(\mathbb{F}_{q^m})$ random in the set of full-rank generator matrices for $\mathcal{G}$. A usual way to do it is to choose a matrix under canonical form, say the one given by Definition 1 and then multiply on the left by a randomly chosen matrix in $\mathrm{GL}_k(\mathbb{F}_{q^m})$.
3. Pick $\mathcal{V} \subset \mathbb{F}_{q^m}$ a random λ-dimensional $\mathbb{F}_q$-subspace of $\mathbb{F}_{q^m}$.
4. Pick $\mathbf{P}$ a random element in $\mathrm{GL}_n(\mathbb{F}_{q^m}) \cap \mathcal{M}_{n,n}(\mathcal{V})$.
5. **return** $\mathbf{G}_{pub} = \mathbf{G}\mathbf{P}^{-1}$ and $\mathbf{sk} = (\mathbf{G}, \mathbf{P})$.

Let $\mathbf{p} \in \mathbb{F}_{q^m}^k$ be the plaintext to be encrypted.

`Encrypt(`$\mathbf{p}$`,`$\mathbf{G}_{pub}$`)`

1. Pick $\mathbf{e} \in \mathbb{F}_{q^m}^n$ such that $\mathrm{Rk}(\mathbf{e}) \leq \lfloor (n-k)/2\lambda \rfloor$.
2. **return** $\mathbf{c} = \mathbf{p}\mathbf{G}_{pub} + \mathbf{e}$.

`Decrypt(`$\mathbf{c}$`,`$\mathbf{sk}$`)`

– **return** `Decode(`$\mathbf{c}\mathbf{P}$`,`$\mathbf{G}$`)`, where `Decode(`$*$`,`$\mathbf{G}$`)` stands for any decoding algorithm for a Gabidulin code with generator matrix $\mathbf{G}$ decoding up to the error-correcting capability $\lfloor (n-k)/2 \rfloor$.

3.2 Security

Let $\mathcal{C}_{pub} \subset \mathbb{F}_{q^m}^n$ be the $\mathbb{F}_{q^m}$-linear code of dimension k generated by the public matrix $\mathbf{G}_{pub}$. The IND-CPA security of the scheme is related to the difficulty of solving the two following problems:

- Distinguish the code $\mathcal{C}_{pub}$ from a random $\mathbb{F}_{q^m}$-linear code with the same parameters.
- Solve a generic instance of the Rank Decoding problem whose parameters are $(m, n, k, t \stackrel{def}{=} \lfloor (n-k)/(2\lambda) \rfloor)$.

In addition to these assumptions, note that LowMS also relies on the Rank Support Learning problem [16].

We address the hardness of the first problem which is used in both [19] and [1]. We even go further since we provide an attack enabling to decrypt. For these schemes, our work also shows that the Gabidulin code itself can be considered as a parameter (meaning that $\mathbf{G}$ generating $\mathcal{G}$ is public) without security loss. This leads to a simplification of the key-generation procedure that can be rewritten as

`KeyGen()`

1. Pick $\mathcal{V} \subset \mathbb{F}_{q^m}$ a random λ-dimensional $\mathbb{F}_q$-subspace of $\mathbb{F}_{q^m}$.
2. Pick $\mathbf{P}$ randomly in $\mathrm{GL}_n(\mathbb{F}_{q^m}) \cap \mathcal{M}_{n,n}(\mathcal{V})$.
3. **return** $\mathbf{G}_{pub} = \mathbf{pk} = \mathbf{G}\mathbf{P}^{-1}$ and $\mathbf{sk} = \mathbf{P}$.

4 A constrained linear system for decryption

In this section, we introduce a constrained linear system (Proposition 3) whose solution allows to devise a polynomial time decryption algorithm for the public code $\mathcal{C}_{pub}$. Note that this trivially implies that one has designe a distinguisher for the public code. The issue of solving this system will be addressed in the next sections in two different ways.

Let $r \stackrel{def}{=} n - k$. In the following, we overline with a hat data known to an attacker. For instance, let $\widehat{\mathbf{H}}_{pub} \in \mathcal{M}_{r,n}(\mathbb{F}_{q^m})$ an arbitrary parity-check matrix for $\mathcal{C}_{pub}$ and for $\alpha \in \mathbb{F}_{q^m}$ a normal element, let $\widehat{\mathbf{H}}_{norm}$ be the matrix $(\alpha^{[i+j-2]})_{1 \le i \le r, 1 \le j \le m}$. Note that

$$\widehat{\mathcal{A}} \stackrel{def}{=} \{\alpha^{[i]},\ i = 0, \ldots, m-1\}$$

is a basis of $\mathbb{F}_{q^m}$ over $\mathbb{F}_q$. From Proposition 1, there exists a vector $\mathbf{h} \in \mathbb{F}_{q^m}^n$ such that $\mathbf{H} \stackrel{def}{=} \begin{pmatrix} \mathbf{h}^{[0]} \\ \vdots \\ \mathbf{h}^{[r-1]} \end{pmatrix} \in \mathcal{M}_{r,n}(\mathbb{F}_{q^m})$ is a parity-check matrix for the Gabidulin

code $\mathcal{G}$. Then, it is easy to see that there exists a unique $\mathbf{S} \in \mathrm{GL}_r(\mathbb{F}_{q^m})$ such that

$$\mathbf{S}\widehat{\mathbf{H}}_{pub} = \mathbf{H}\mathbf{P}^t. \tag{1}$$

We indeed have $\mathbf{H}\mathbf{P}^t\mathbf{G}_{pub}^t = \mathbf{H}\mathbf{P}^t(\mathbf{P}^t)^{-1}\mathbf{G}^t = \mathbf{H}\mathbf{G}^t = \mathbf{0}$, so that $\mathbf{H}\mathbf{P}^t$ is a parity-check matrix for $\mathcal{C}_{pub}$. Finally, any parity-check matrix and, a fortiori, $\widehat{\mathbf{H}}_{pub}$, is obtained with a basis transformation induced by a non-singular matrix over $\mathbb{F}_{q^m}$. Another straightforward proposition is

Proposition 2 *Let $\mathbf{H}$ be a parity check matrix for $\mathcal{G}$ under canonical form. There exists a q-ary matrix $\mathbf{M} \in \mathcal{M}_{m,n}(\mathbb{F}_q)$ of rank n such that*

$$\mathbf{H} = \widehat{\mathbf{H}}_{norm}\mathbf{M}.$$

Proof. Let $\mathbf{h} = (h_1, \ldots, h_n)$ be the first row of $\mathbf{H}$. We consider the matrix $\mathbf{M}$ whose i-th column corresponds to the m-dimensional q-ary vector formed by the coordinates of h_i in the basis $\widehat{\mathcal{A}}$, for $1 \leq i \leq n$. By construction we have $\mathbf{H} = \widehat{\mathbf{H}}_{norm}\mathbf{M}$ and moreover $h_1, \ldots, h_n$ are linearly independent over $\mathbb{F}_q$ by construction of Gabidulin codes. This shows that $\mathbf{M}$ has full rank.

Now Equation (1) can be rewritten as

$$\mathbf{S}\widehat{\mathbf{H}}_{pub} = \widehat{\mathbf{H}}_{norm}\mathbf{T}, \tag{2}$$

where the matrix $\mathbf{T} \stackrel{def}{=} \mathbf{M}\mathbf{P}^t$ is full rank in $\mathcal{V}^{m\times n}$ since $\mathbf{M}$ is a q-ary matrix of full rank n and since $\mathbf{P} \in \mathrm{GL}_n(\mathcal{V})$. Finally, the following proposition shows that any solution to the constrained linear system described by (2) indeed yields a polynomial-time decryption algorithm.

Proposition 3 *Let $r = n-k$ and let $\widehat{\mathbf{H}}_{pub}$ be a parity-check matrix for $\mathcal{C}_{pub}$. Let $\alpha \in \mathbb{F}_{q^m}$ be a normal element and let $\widehat{\mathbf{H}}_{norm}$ be the matrix $(\alpha^{[i+j-2]})_{1\leq i\leq r, 1\leq j\leq m}$. From the knowledge of any non-singular matrix $\mathbf{V} \in \mathcal{M}_{r\times r}(\mathbb{F}_{q^m})$ and $\mathbf{W} \in \mathcal{M}_{m\times n}(\mathcal{W})$ of rank n such that*

$$\mathbf{V}\widehat{\mathbf{H}}_{pub} = \widehat{\mathbf{H}}_{norm}\mathbf{W} \tag{3}$$

and where $\mathcal{W}$ is $\mathbb{F}_q$-vector subspace of $\mathbb{F}_{q^m}$ of dimension $\leq \lambda$, it is possible to decrypt any ciphertext in polynomial time.

Proof. Recall that a ciphertext is $\mathbf{c} = \mathbf{p}\cdot\mathbf{G}_{pub} + \mathbf{e} \in \mathbb{F}_{q^m}^n$, where $\mathrm{Rk}(\mathbf{e}) = \lfloor (n-k)/(2\lambda) \rfloor$. Thus $\widehat{\mathbf{H}}_{pub}\mathbf{c}^t = \widehat{\mathbf{H}}_{pub}\mathbf{e}^t$ and

$$\mathbf{V}\widehat{\mathbf{H}}_{pub}\mathbf{e}^t = \widehat{\mathbf{H}}_{norm}\underbrace{\mathbf{W}\mathbf{e}^t}_{\mathbf{e}'^t}.$$

Since $\mathcal{W}$ has dimension $\leq \lambda$, this implies that and $\mathrm{Rk}(\mathbf{e}') \leq \lambda\mathrm{Rk}(\mathbf{e}) \leq \lfloor (n-k)/2 \rfloor$. Therefore by decoding in the public Gabidulin code with parity-check matrix $\widehat{\mathbf{H}}_{norm}$, one recovers $\mathbf{e}'^t = \mathbf{W}\mathbf{e}^t$. Since $\mathbf{W}$ has rank $n \leq m$, $\mathbf{e} \mapsto \mathbf{W}^t\mathbf{e}$ is one-to-one and $\mathbf{e}$ can be uniquely recovered. The vector $\mathbf{p}$ such that $\mathbf{p}\cdot\mathbf{G}_{pub} = \mathbf{c} - \mathbf{e}$ can also be uniquely recovered. □

To conclude this section, note that a first naive solving approach would be to enumerate all solutions $(\mathbf{V}, \mathbf{W}) \in \mathcal{M}_{r\times r}(\mathbb{F}_{q^m}) \times \mathcal{M}_{m\times n}(\mathbb{F}_{q^m})$ to (3) and to test if they satisfy the constraint, *i.e.*, the $\mathbf{W}$ matrix has its entries in a small dimensional $\mathbb{F}_q$-vector subspace of $\mathbb{F}_{q^m}$. Even if one takes into account the fact that there may be multiple possibilities, this effort lies beyond the capacities of any computer even for moderate parameters. Indeed, a first difficulty is that the solution space to (3) is an $\mathbb{F}_{q^m}$-vector space of dimension at least $r^2 + (m-r)n$ without the imposed condition.

5 Combinatorial approach

A first idea to take advantage of this extra information is to enumerate candidate bases $\boldsymbol{\mu} \in \mathbb{F}_{q^m}^{\lambda}$ for the secret vector space $\mathcal{V}$. Any such candidate is then completed into a basis of $\mathbb{F}_{q^m}$ in which we express the coefficients of $\boldsymbol{V}$ and $\boldsymbol{W}$ in order to write down the linear system (3) over $\mathbb{F}_q$. We assume that each entry in $\boldsymbol{W}$ belongs to the $\mathbb{F}_q$-vector space spanned by $\boldsymbol{\mu}$ and thus we introduce only λmn unknowns over $\mathbb{F}_q$ instead of m^2n for this matrix. Since we typically have $rmn \gg \lambda mn + mr^2$, this initial guess can be tested by solving the resulting linear equations over $\mathbb{F}_q$ to check if they have a non-zero solution. As is usual for this type of approach, the total cost contains two factors:

- an exponential one coming from enumerating the bases;
- a polynomial one which corresponds to the linear system solving over $\mathbb{F}_q$.

Proposed algorithm. We can in fact obtain a better exponential factor by relying on the same techniques as used in combinatorial attacks on the Rank Decoding problem [21,17,2]. The rationale is that it is enough to know (a basis for) a γ-dimensional vector space $\mathcal{U}$ which contains $\mathcal{V}$ for $\gamma \geq \lambda$ to apply the same algorithm, provided that γ is not too large. The advantage is that it is always easier to find such a $\mathcal{U}$ than to guess a basis of $\mathcal{V}$ directly, the extreme case being $\gamma = m$ for which we succeed with probability 1. Here, we even note that a vector space $\mathcal{U}$ which contains an arbitrary multiple $x\mathcal{V}$ for $x \in \mathbb{F}_{q^m}^*$ instead of simply $\mathcal{V}$ is enough for our purposes. This is because any pair $(x\boldsymbol{V}, x\boldsymbol{W})$ is a solution to the constrained linear system. The following Proposition 4 gives the condition on γ for our attack to succeed.

Proposition 4 *Assume that $\gamma \geq \lambda \in \mathbb{N}$ is such that*

$$rn \geq \gamma n + r^2. \tag{4}$$

If $\boldsymbol{\nu} \in \mathbb{F}_{q^m}^{\gamma}$ is a basis for a vector space $\mathcal{U}$ which contains a multiple $x\mathcal{V}$ for $x \in \mathbb{F}_{q^m}^*$, the linear system over $\mathbb{F}_q$ derived from (3) by writing the coefficients of the secret matrix $\boldsymbol{W}$ in the basis $\boldsymbol{\nu}$ is expected to have a solution space of dimension 1. If $\boldsymbol{\nu}$ does not correspond to such a basis, this linear system will not have a non-zero solution with overwhelming probability.

From this proposition, we can then use the same algorithm as sketched at the beginning of Section 5 with γ instead of λ provided that $\gamma \leq r(1 - r/n)$.

Estimated cost. The exponential factor of this approach is given by the inverse of the probability that a fixed subspace $\mathcal{U}$ of dimension γ contains a subspace of the form $x\mathcal{V}$ for some $x \in \mathbb{F}_{q^m}^*$. According to [2, B.], it can be estimated by $q^{m-\lambda(m-\gamma)} = q^{-(\lambda-1)m+\lambda\gamma}$. We assume that the optimal complexity corresponds to the optimal exponential factor and thus we consider the highest possible value for γ. By Equation (4), this leads to choose $\gamma \stackrel{def}{=} \lfloor r(1-r/n) \rfloor$.

The linear system solving step can be performed by applying Gaussian elimination on a matrix of size $rnm \times (\gamma n + r^2)m$ over $\mathbb{F}_q$. The corresponding cost in $\mathbb{F}_q$-operations can be estimated by $\mathcal{O}((\gamma n + r^2)m)^\omega)$, where ω is the linear algebra constant. However, checking that a linear system is consistent does not require to compute a row echelon form. We can actually apply the Wiedemann algorithm [9], which may offer an advantage since the input matrix is sparse. Indeed, equations have weight $m(r+\gamma)$ but they contain $m(r^2+\gamma n) \gg m(r+\gamma)$ unknowns. In particular, we lower bound the complexity of linear algebra by considering the cost of computing the kernel of a sparse square matrix of size $m(\gamma n + r^2)$ corresponding to the number of unknowns with a number of non-zero coefficients roughly equal to $m^2(r+\gamma)(\gamma n+r^2)$. An estimation of this lower bound is

$$m^3(r+\gamma)(\gamma n + r^2)^2 > m^3 r^5$$

q-ary operations. Recalling that $r = n - k$ and by introducing the code rate $R \stackrel{def}{=} k/n$, a lower bound of the overall complexity of this precise attack is then given by

$$W_{\text{Spec\_Inf}} = m^3(n-k)^5 q^{(\lambda-1)m-\lambda\lfloor n(1-R)R \rfloor}. \tag{5}$$

Application to some parameters. Finally, we instantiate our bound with the parameters of the WCC 2022 abstract and the ones of LowMS [1]. We believe that the comparison is fair since they have been obtained from the content the abstract. In Table 1, column *Lower bound* contains the value of the binary logarithm of the cost of Equation (5). Our results always improve the cost of the best structural attack. If it becomes below the one of attacks on RD, this might lead to re-evaluate parameters in [19] and [1].

(m, n, k, λ)	Security	Source	Lower bound	Former
$(128, 128, 20, 3)$	128	WCC 2022	263	311
$(128, 128, 44, 3)$	128	WCC 2022	225	308
$(59, 50, 25, 3)$	128	LowMS	**123**	158
$(67, 66, 33, 4)$	128	LowMS	180	244
$(83, 74, 37, 3)$	192	LowMS	**157**	211
$(79, 78, 39, 4)$	192	LowMS	206	282

Table 1. Cost estimate on former parameters.

Note however that we do not claim that the lower bound is on all possible algorithms which would solve the same problem. The lower bound in our case only deals with the linear algebra part when using Wiedemann's algorithm. It is a lower bound relatively to the state of the art of research in this field.

6 A bilinear system

Instead of guessing a basis for $\mathcal{V}$ or for a vector space which contains it as in Section 5, our second approach consists in solving a bilinear system over $\mathbb{F}_q$ (**System 1**). These former quantities attached to $\mathcal{V}$ still appear in the system as an unknown block of variables but we will not fix them in the first place.

Let $\widehat{\mathcal{B}}$ denote an arbitrary basis of $\mathbb{F}_{q^m}$ over $\mathbb{F}_q$. For an element $a \in \mathbb{F}_{q^m}$, we consider $\vec{a}$ the m-dimensional vector of its coordinates over $\widehat{\mathcal{B}}$, so that $\widehat{\mathcal{B}}\vec{a} = a$. For $\mu \in \mathbb{F}_{q^m}$, we also define $\mathbf{M}_\mu \in \mathcal{M}_{m\times m}(\mathbb{F}_q)$ the matrix of the multiplication by μ in the basis $\widehat{\mathcal{B}}$. This matrix is such that

$$\forall a,\ b \in \mathbb{F}_{q^m},\ b = \mu a, \text{ then } \vec{b} = \mathbf{M}_\mu \vec{a}.$$

The claimed bilinear system is as follows:

System 1 *Let $\widehat{\mathbf{H}}_{pub} = (\widehat{h}_{ij})$ and let $\widehat{\mathbf{H}}_{norm} = (\alpha^{[i+j-2]})$. We consider the bilinear system over $\mathbb{F}_q$ in the non-zero unknowns $\vec{v_{iu}}$, $b_{ij}^{(\ell)}$ and linearly independent $\vec{\mu_\ell} \in \mathbb{F}_q^m$, whose equations are given by*

$$\forall \begin{cases} i \in \{1..r\} \\ j \in \{1..n\} \end{cases}, \quad \sum_{u=1}^{r} \mathbf{M}_{\widehat{h}_{uj}} \vec{v_{iu}} = \sum_{u=1,\ell=1}^{m,\lambda} b_{uj}^{(\ell)} \mathbf{M}_{\alpha^{[i+u-2]}} \vec{\mu_\ell}. \tag{6}$$

System 1 contains mrn affine equations over $\mathbb{F}_q$. The linear parts involve mr^2 variables $\vec{v_{iu}}$ while the bilinear parts involve $\lambda mn + \lambda m$ variables $b_{uj}^{(\ell)}$ and $\vec{\mu_\ell}$ respectively. Proposition 5 states that the solutions to this system are actually equivalent to the ones of the constrained linear equations (3).

Proposition 5 *Let $\widetilde{\mathbf{V}} = (v_{ij}) \in \mathcal{M}_{r\times r}(\mathbb{F}_{q^m})$ and $\widetilde{\mathbf{W}} = (w_{ij}) \in \mathcal{M}_{m\times n}(\mathbb{F}_{q^m})$ which satisfy the constrained linear equations (3) and let $\mathcal{W}$ a[4] λ-dimensional subspace of $\mathbb{F}_{q^m}$ which contains the entries of $\widetilde{\mathbf{W}}$. Let $(\mu_1, \ldots, \mu_\lambda) \in \mathbb{F}_{q^m}^\lambda$ be a basis for $\mathcal{W}$ and*

$$w_{ij} \stackrel{def}{=} \sum_{\ell=1}^{\lambda} b_{ij}^{(\ell)} \mu_\ell \tag{7}$$

*be the unique decomposision of w_{ij} in this basis. Then $\vec{v_{iu}} \in \mathbb{F}_q^m$, $b_{ij}^{(\ell)}$ and $\vec{\mu_\ell}$ are a solution to **System 1**. Conversely, any solution $\vec{v_{iu}}$, $b_{ij}^{(\ell)}$, $\vec{\mu_\ell}$ to **System 1** gives a pair of matrices $\widetilde{\mathbf{V}} = (v_{ij})$, $\widetilde{\mathbf{W}} = (w_{ij})$ solution to the constrained linear equations (3), where w_{ij} is defined by Equation (7).*

[4] concretely, "the"

If $(\boldsymbol{V}, \boldsymbol{W})$ stands for the genuine couple of matrices which is implicit from the description of the scheme, we have already mentioned that any $(\widetilde{\boldsymbol{V}}, \widetilde{\boldsymbol{W}}) \stackrel{def}{=} (x\boldsymbol{V}, x\boldsymbol{W})$ for $x \in \mathbb{F}_{q^m}^*$ allows to decrypt. Concretely, to reduce the number of solutions to **System 1**, we will thus:

- fix μ_1 to 1 and choose a basis $\widehat{\mathcal{B}}$ such that $\hat{b}_1 = 1$;
- target a basis in systematic form, *i.e.*,

$$(1, \mu_2, \ldots, \mu_\lambda)^\mathsf{T} \stackrel{def}{=} \begin{pmatrix} & \mathbf{0}_{1\times(m-\lambda)} & \\ \boldsymbol{I}_\lambda & & \\ & \boldsymbol{R}' & \end{pmatrix} \widehat{\mathcal{B}}^\mathsf{T}, \tag{8}$$

  where $\boldsymbol{R}' \in \mathcal{M}_{(\lambda-1)\times(m-\lambda)}(\mathbb{F}_q)$. We cannot always guarantee to have a solution in this way but the success probability is constant.

Note that similar strategies to fix variables had already been suggested in previous works, see for instance [7, §3.4] or [21, §3.1].

Solving by Gröbner bases. To solve **System 1**, one may be tempted to use Gröbner basis techniques [11,12,13]. However, our practical experiments for this method were not conclusive. A reason is that there is a great imbalance between the two blocks of variables $\vec{\mu_\ell}$ and $b_{ij}^{(\ell)}$ since $(\lambda-1)(m-\lambda) \ll mn\lambda$. This also explains why it was quite natural in Section 5 to proceed by enumeration on the smallest block $\vec{\mu_\ell}$ (corresponding to an unknown basis for $\mathcal{V}$) in order to obtain linear equations.

7 Tools to analyze System 1

Even if the Gröbner basis approach seems infeasible, this section gives some background to partially explain the early steps of such an algorithm. More specifically, in Section 8, we will characterize the first degree fall polynomials (see Definition 2) which arise in the computation.

Gröbner basis solvers [11,12,13] had already been analyzed by [13] in the context of generic bilinear systems. However, in our case, we need to use the fact that **System 1** admits a much stronger structure than being merely bilinear. It turns out that its analysis is much closer to the one performed in [4,23] on bilinear modelings of MinRank and of the Rank Decoding problem. Indeed, a common feature in such systems is that the equations can be viewed as the entries of a matrix $\boldsymbol{M} = \boldsymbol{AXY}$, where $\boldsymbol{A}$ is a matrix of scalars and where $\boldsymbol{X}$ and $\boldsymbol{Y}$ are matrices of unknowns $\boldsymbol{x}$ and $\boldsymbol{y}$ respectively. It is easy to see that our equations exhibit a similar shape. Using the notation from **System 1**, we can indeed write each column $\boldsymbol{w}_j = (w_{1,j}, \ldots, w_{m,j}) \in \mathbb{F}_{q^m}^m$ of the unknown $\boldsymbol{W}$ as $\boldsymbol{w}_j^\mathsf{T} = \boldsymbol{C}_j(\mu_1, \ldots, \mu_\lambda)^\mathsf{T} = \boldsymbol{C}_j\boldsymbol{R}\widehat{\mathcal{B}}^\mathsf{T}$, where $\boldsymbol{C}_j \stackrel{def}{=} (b_{i,j}^{(\ell)})_{1\le i\le m, 1\le \ell\le \lambda}$ and where the rows of $\boldsymbol{R} \in \mathcal{M}_{\lambda\times m}(\mathbb{F}_q)$ are the $\vec{\mu_\ell}$'s for $1 \le \ell \le \lambda$. We then obtain

System 0 *For $j \in \{1..n\}$, let $\widehat{\boldsymbol{h}_j} \in \mathbb{F}_{q^m}^r$ denote the j-th column in $\widehat{\mathbf{H}}_{pub}$. There are r bilinear equations in the entries of $\widetilde{\boldsymbol{V}}$, $\boldsymbol{R}$ and $\boldsymbol{C}_j$ from the equality*

$$\widetilde{\boldsymbol{V}}\widehat{\boldsymbol{h}_j}^{\mathsf{T}} = \widehat{\mathbf{H}}_{norm}\boldsymbol{C}_j\boldsymbol{R}\widehat{\mathcal{B}}^{\mathsf{T}}. \tag{9}$$

By considering all columns, we obtain an affine bilinear system with

- *rn equations over $\mathbb{F}_{q^m}$.*
- *r^2 unknowns v_{ij} over $\mathbb{F}_{q^m}$ and $\lambda mn + \lambda m$ unknowns over $\mathbb{F}_q$.*

Note that **System 1** captures exactly the same information as the system over $\mathbb{F}_q$ obtained from **System 0** by taking as unknowns the $\vec{v_{iu}}$'s instead of the v_{ij}'s and then by projecting over the base field. We may adopt the latter for the theoretical analysis since it is more convenient.

7.1 Algebraic background

Let us start with some necessary facts on Gröbner bases techniques applied to bilinear systems.

Syzygies and degree falls. For a polynomial sequence $\mathcal{F} = (f_1, \ldots, f_M)$, a *syzygy* is a polynomial combination $\sum_{i=1}^M g_i f_i = 0$. Its degree is defined by $\max_{i=1}^M (\deg(g_i f_i))$. In our systems, recall that any polynomial is of the form $f_i = b_i + l_i$ where b_i is bilinear and l_i is linear. In particular, a syzygy $\sum_{i=1}^M g_i b_i = 0$ of degree d for $(b_1, \ldots, b_M)$ typically yields an equation $\sum_{i=1}^m g_i f_i = \sum_{i=1}^M g_i l_i = 0$ of degree $d-1$ in the ideal. This is a particular case of

Definition 2 (Degree fall polynomial) *A degree fall polynomial for a sequence $\mathcal{F} = (f_1, \ldots, f_M)$ is a non-zero polynomial combination $\sum_{i=1}^M g_i f_i$ whose degree δ is strictly less than $d \stackrel{def}{=} max_{i=1}^M (\deg(g_i f_i))$. We may also refer to it as a degree fall from degree d to degree δ.*

Such an equation will be meaningful if and only if it is not a linear combination between previously considered equations of degree $\leq d-1$. Some actually prefer to include this extra constraint already in Definition 2. Degree fall polynomials for affine systems play a similar role to that of syzygies for homogeneous equations. Their study is thus instrumental to understand the complexity of solving such affine equations.

Bilinear systems [13]. Let $\mathcal{B} = (b_1, \ldots, b_M) \subset \mathbb{F}[\boldsymbol{x}, \boldsymbol{y}]$ be the homogeneous bilinear sequence in two blocks of variables $\boldsymbol{x}$ and $\boldsymbol{y}$ over a field $\mathbb{F}$ which contains the degree 2 parts of an affine bilinear sequence $\mathcal{F}$. As we have just said, degree fall polynomials for $\mathcal{F}$ are directly related to syzygies for $\mathcal{B}$. Let us now consider the Jacobian matrices which are defined by

$$\mathrm{Jac}_{\boldsymbol{x}}(\mathcal{S}) \stackrel{def}{=} \left(\frac{\partial b_i}{\partial x_j}\right)_{1\leq i\leq M,\ 1\leq j\leq n_{\boldsymbol{x}}}$$

and

$$\mathrm{Jac}_{\boldsymbol{y}}(\mathcal{S}) \stackrel{def}{=} \left(\frac{\partial b_i}{\partial y_j}\right)_{1\leq i\leq M,\ 1\leq j\leq n_{\boldsymbol{y}}}.$$

Their entries are linear forms in $\mathbb{F}[\boldsymbol{y}]$ and $\mathbb{F}[\boldsymbol{x}]$ respectively. The study of these Jacobians is motivated by the following Lemma 1, which states that generic syzygies for $\mathcal{S}$ are provided by vectors in the left kernel of these matrices.

Lemma 1 *Let $\mathcal{S} \stackrel{def}{=} (b_1, \ldots, b_M) \subset \mathbb{F}[\boldsymbol{x}, \boldsymbol{y}]$ be a homogeneous bilinear sequence and let $\mathcal{G} \stackrel{def}{=} (g_1, \ldots, g_M) \subset \mathbb{F}[\boldsymbol{y}]^M$ be a polynomial sequence. We have $\sum_{i=1}^M g_i b_i = 0$ if and only if $\mathcal{G}$ belongs to the left kernel of $Jac_{\boldsymbol{x}}(\mathcal{S})$.*

Proof. Let $\mathcal{G} = (g_1, \ldots, g_M)$ be an arbitrary polynomial sequence. Since we have

$$\mathcal{G}\mathrm{Jac}_{\boldsymbol{x}}(\mathcal{S})\boldsymbol{x}^{\mathsf{T}} = \textstyle\sum_{i=1}^M g_i b_i,$$

we obtain a syzygy from any kernel vector of $\mathrm{Jac}_{\boldsymbol{x}}(\mathcal{S})$. The converse statement is only valid for $\mathcal{G} \subset \mathbb{F}[\boldsymbol{y}]^M$. For such a vector of polynomials, the product by $\mathrm{Jac}_{\boldsymbol{x}}(\mathcal{S})$ is still a row vector of elements in $\mathbb{F}[\boldsymbol{y}]$. The only possibility for it to be 0 when multiplied by $\boldsymbol{x}^{\mathsf{T}}$ is that it is already 0, *i.e.*, $\mathcal{G} \in \ker\left(\mathrm{Jac}_{\boldsymbol{x}}(\mathcal{S})\right)$. □

The following Lemma 2 gives kernel vectors for these Jacobian matrices regardless of their structure.

Lemma 2 (Lemma 3.1 in [13]) *Let $\boldsymbol{M} \in \mathcal{M}_{M\times t}(\mathbb{F}[\boldsymbol{y}])$ be a matrix whose entries are linear forms with $t < M$. Let*

$$\boldsymbol{V}_J \stackrel{def}{=} (\ldots, \underbrace{0}_{j\notin J}, \ldots, \underbrace{(-1)^{\ell+1}|\boldsymbol{M}|_{J\setminus j_\ell,*}}_{j=j_\ell}, \ldots),$$

where $J = \{j_1 < \cdots < j_{t+1}\} \subset \{1..M\}$. These vectors are such that $\boldsymbol{V}_J\boldsymbol{M} = 0$.

Generically, the vectors $\boldsymbol{V}_J$ generate the left kernel of such a matrix $\boldsymbol{M}$, see for instance [13, Conjecture 4.1]. Also, for a bilinear random $\mathcal{S}$, the entries of the matrix $\mathrm{Jac}_{\boldsymbol{x}}(\mathcal{S})$ are random linear forms in $\mathbb{F}[\boldsymbol{y}]$. Lemma 2 was thus used in [13] to have a complete description of its left kernel. Based on this result, they show that the degree of regularity of a generic bilinear system is such that $d_{\mathrm{reg}} \leq \min(n_{\boldsymbol{x}} + 1, n_{\boldsymbol{y}} + 1)$.

However, the bilinear equations relevant to us are not generic and we will have to analyze the structure of the Jacobians.

A useful lemma. Consider a matrix equation $\boldsymbol{M} = \boldsymbol{AXY} \in \mathcal{M}_{p\times n}(\mathbb{F}[\boldsymbol{x}, \boldsymbol{y}])$ as in the beginning of this section, where $\boldsymbol{A} \in \mathcal{M}_{p\times m}(\mathbb{F})$ is a matrix of scalars and where $\boldsymbol{X} \in \mathcal{M}_{m\times r}(\mathbb{F}[\boldsymbol{x}])$ and $\boldsymbol{Y} \in \mathcal{M}_{r\times n}(\mathbb{F}[\boldsymbol{y}])$ are matrices of unknowns $\boldsymbol{x}$ and $\boldsymbol{y}$ respectively. Let us define the row vector $\left(\boldsymbol{M}_{\{1\},*} \ldots \boldsymbol{M}_{\{m\},*}\right)$ formed by the concatenation of the rows of $\boldsymbol{M}$ and similarly $\mathrm{col}(\boldsymbol{M}) \stackrel{def}{=} \mathrm{row}(\boldsymbol{M}^\mathsf{T})$. Then we have the following lemma

Lemma 3 *The Jacobian matrix of a system $\boldsymbol{AXY} = \boldsymbol{0}_{p\times n}$ with respect to the $\boldsymbol{x}$ variables is given by*

$$\mathrm{Jac}_{row(\boldsymbol{X})}\left(row(\boldsymbol{AXY})\right) = \boldsymbol{A} \otimes \boldsymbol{Y}^\mathsf{T} \in \mathcal{M}_{np\times mr}(\mathbb{F}[\boldsymbol{y}]),$$
$$\mathrm{Jac}_{col(\boldsymbol{X})}\left(col(\boldsymbol{AXY})\right) = \boldsymbol{Y}^\mathsf{T} \otimes \boldsymbol{A} \in \mathcal{M}_{np\times mr}(\mathbb{F}[\boldsymbol{y}]).$$

Proof. See [4, Lemma 1]. □

7.2 Understanding the projection over $\mathbb{F}_q$

In addition to the matrix product structure, another particularity comes from the extension field. Indeed, recall that **System 1** can be seen as the projection over $\mathbb{F}_q$ of **System 0** whose equations have coefficients in $\mathbb{F}_{q^m}$ but where the variables involved in the bilinear parts belong to $\mathbb{F}_q$. In that respect, this system is obtained in the exact same manner as in [3,4,6] which aim at solving the Rank Decoding Problem and the Rank Support Learning Problem.

In the case of [3,4,6], the analysis of the full system over $\mathbb{F}_q$ can be boiled down to the one of the initial system over $\mathbb{F}_{q^m}$. On our side, however, the situation is less simple. For instance, it is not sufficient to analyze **System 0** to understand the computation on **System 1** over $\mathbb{F}_q$. This might be due to the following simple fact: by choosing $\widehat{\mathcal{B}} = \mathcal{A} = \{1, \alpha^{[1]}, \ldots, \alpha^{[m-1]}\}$ to express **System 0**, we actually recover the first row of $\widehat{\mathbf{H}}_{norm}$. This gives another interesting property which was not present in [3,4,6].

As explained after the definition of **System 0**, projecting this system yields equations over $\mathbb{F}_q$ which generate the same system as **System 1**. Let us denote by $\{b_1, \ldots, b_m\}$ the set of m equations over $\mathbb{F}_q$ obtained by projecting the bilinear part b of an equation of **System 0**. Also, let us extend the Frobenius map to polynomials by reducing modulo the field equations of the small field since all variables belong to $\mathbb{F}_q$, namely $b^{[\ell]} \stackrel{def}{=} b^{q^\ell} \bmod \langle x_i^q - x_i\rangle_i$. For a matrix $\boldsymbol{M} = (m_{ij})$ over $\mathbb{F}_{q^m}$ (or over a polynomial ring with base field $\mathbb{F}_{q^m}$), we also denote by $\boldsymbol{M}^{[\ell]}$ the matrix $\left(m_{ij}^{[\ell]}\right)$. Finally, in our analysis, we will use the fact that the algebraic properties[5] of both sequences $(b_1, \ldots, b_m)$ and $(b, \ldots, b^{[m-1]})$ are the same. In particular, it will be relevant to consider the following **System 2** which is equivalent to **System 1**.

[5] syzygies, etc.

System 2 *For $j \in \{1..n\}$, let $\widehat{\boldsymbol{h}_j} \in \mathbb{F}_{q^m}^r$ denote the j-th column in $\widehat{\mathbf{H}}_{pub}$. For any $0 \leq \ell \leq m-1$, we consider the r equations obtained by applying the Frobenius ℓ times on Equation* (9)*. They are given by*

$$\boldsymbol{V}^{[\ell]} \left(\widehat{\boldsymbol{h}_j^{[\ell]}} \right)^{\mathsf{T}} = \widehat{\mathbf{H}}_{norm}^{[\ell]} \boldsymbol{C}_j \boldsymbol{R} \left(\widehat{\mathcal{B}^{[\ell]}} \right)^{\mathsf{T}}. \tag{10}$$

We stress that **System 2** has essentially a theoretical value. In particular, it would not be suitable to solve it by using naive Gröbner basis algorithms since the equations have very high degree in the $v_{i,j}$ variables.

8 Degree fall polynomials from Jacobians

This final section aims at studying the Jacobians associated to the bilinear parts of our equations. We will show in Lemma 4 and Lemma 5 that their kernels provide two types syzygies in degree $\lambda + 2$, hence degree fall polynomials of degree $\lambda + 1$ for the original affine equations. Moreover, our experiments suggest that these are the only ones at this degree and that these are the *first*, *i.e.*, no one appear at a lower degree.

Interestingly enough, we do not need to wait the degree $\lambda+2$ step of a Gröbner basis algorithm on **System 0** or **System 1** for a graded order to obtain these polynomials. They can indeed be pre-computed as maximal minors of public matrices of linear forms. In that respect, the situation is quite similar to the one of algebraic attacks on the Rank Decoding problem [21,4,6,5]. For instance, the so-called MaxMinors equations introduced in [4] were originally obtained as degree fall polynomials for the former bilinear modeling of Ourivski-Johansson [21] but they can also be computed directly.

For the sake of simplicity, we give the results for the non-specialized version of our systems. They can be easily adapted if we fix μ_1 to 1 and if we choose a matrix $\boldsymbol{R}$ in systematic form as presented above.

8.1 Jacobian with respect to the $\boldsymbol{R}$ variables

We start from the Jacobian matrices with respect to the block of $\boldsymbol{R}$ variables. We will see that their structure is similar to the one encountered in [4, §5.1]. As in their work, we also observed that all degree falls over $\mathbb{F}_q$ from these matrices were obtained by projecting over $\mathbb{F}_q$ degree fall polynomials whose coefficients are in $\mathbb{F}_{q^m}$. This means that we can focus on **System 0** rather than on **System 1** for this part of the analysis, the situation being different in Section 8.2.

If we restrict ourselves to the bilinear parts in **System 0**, a direct application of Lemma 3 for $1 \leq j \leq n$ with $\boldsymbol{X} \stackrel{def}{=} \boldsymbol{R}$, $\boldsymbol{A} \stackrel{def}{=} \widehat{\mathbf{H}}_{norm} \boldsymbol{C}_j$ and $\boldsymbol{Y} \stackrel{def}{=} \widehat{\mathcal{B}}^{\mathsf{T}}$ yields

$$\mathrm{Jac}_{\mathrm{row}(\boldsymbol{R})}(\mathrm{row}(\widehat{\mathbf{H}}_{norm} \boldsymbol{C}_j \boldsymbol{R} \widehat{\mathcal{B}}^{\mathsf{T}})) = \widehat{\mathbf{H}}_{norm} \boldsymbol{C}_j \otimes \widehat{\mathcal{B}}. \tag{11}$$

The full system can also be viewed as the following matrix product

$$\left(\boldsymbol{I}_n \otimes \widehat{\mathbf{H}}_{norm}\right) \begin{pmatrix} \boldsymbol{C}_1 \\ \vdots \\ \boldsymbol{C}_n \end{pmatrix} \boldsymbol{R}\widehat{\mathcal{B}}^{\mathsf{T}},$$

and thus we obtain in the same manner

$$\begin{aligned} \mathrm{Jac}_{\mathrm{row}(\boldsymbol{R})} \left(\left(\boldsymbol{I}_n \otimes \widehat{\mathbf{H}}_{norm}\right) \begin{pmatrix} \boldsymbol{C}_1 \\ \vdots \\ \boldsymbol{C}_n \end{pmatrix} \boldsymbol{R}\widehat{\mathcal{B}}^{\mathsf{T}} \right) \\ = \begin{pmatrix} \widehat{\mathbf{H}}_{norm}\boldsymbol{C}_1 \\ \vdots \\ \widehat{\mathbf{H}}_{norm}\boldsymbol{C}_n \end{pmatrix} \otimes \widehat{\mathcal{B}}. \end{aligned} \tag{12}$$

Recall from Lemma 1 that the kernel of such Jacobians provides syzygies for the bilinear parts whose coefficients are polynomials in the $\boldsymbol{C}_j$ variables. In our case, we can obtain

Lemma 4 *In **System 0**, there are at least $\binom{nr}{\lambda+1}$ degree falls from degree $\lambda+2$ to $\lambda+1$. Indeed, some of them are already given by the maximal minors of the matrix*

$$\mathcal{M} \stackrel{def}{=} \begin{pmatrix} \widetilde{\boldsymbol{V}}\widehat{\boldsymbol{h}_1}^{\mathsf{T}} & \widehat{\mathbf{H}}_{norm}\boldsymbol{C}_1 \\ & \vdots \\ \widetilde{\boldsymbol{V}}\widehat{\boldsymbol{h}_n}^{\mathsf{T}} & \widehat{\mathbf{H}}_{norm}\boldsymbol{C}_n \end{pmatrix}. \tag{13}$$

Among these equations, we may find in particular the maximal minors of the matrix

$$\mathcal{M}_j \stackrel{def}{=} \mathcal{M}_{\{1+r(j-1)..rj\},*} = \left(\widetilde{\boldsymbol{V}}\widehat{\boldsymbol{h}_j}^{\mathsf{T}} \;\; \widehat{\mathbf{H}}_{norm}\boldsymbol{C}_j\right), \tag{14}$$

for $1 \leq j \leq n$.

Even before giving the proof of Lemma 4, it is easy to see from Equation (9) that all $\mathcal{M}_j$ matrices are not full-rank (a fortiori, $\mathcal{M}$) if and only if $(\widetilde{\boldsymbol{V}}, \boldsymbol{C}_1, \ldots, \boldsymbol{C}_n)$ are components of a solution to **System 0**.

Proof. (Analogous to [4]). We do the proof for a single matrix $\mathcal{M}_j$. Using Equation (11), it is sufficient to look at the left kernel of $\widehat{\mathbf{H}}_{norm}\boldsymbol{C}_j$. We then compute the kernel vectors $\boldsymbol{V}_J$ of Lemma 2 for this matrix of linear forms, namely

$$\boldsymbol{V}_J \stackrel{def}{=} \left(\underbrace{0}_{j\notin J}, \ldots, \underbrace{(-1)^{\ell+1} \left|\widehat{\mathbf{H}}_{norm}\boldsymbol{C}_j\right|_{J\setminus\{j\},*}}_{j=j_\ell\in J}, \ldots \right), \; \#J = \lambda+1, \; J \subset \{1..r\}.$$

The degree falls are then obtained by multiplying these vectors by the linear parts of the equations, *i.e.* $(\boldsymbol{V}_J) \cdot \widetilde{\boldsymbol{V}}\widehat{\boldsymbol{h}_j}^{\mathsf{T}}$. Finally, the latter actually coincides with the maximal minor $|\mathcal{M}_j|_{J,*}$ using Laplace expansion along the first column. The reasoning is similar for $\mathcal{M}$ if we replace Equation (11) by Equation (12). □

Bilinear structure. The degree fall polynomials given by Lemma 4 have degree $\lambda + 1$. Moreover, Laplace expansion along the first column of $\mathcal{M}$ in Equation (13) shows that these equations are *bilinear* in the entries of $\widetilde{\boldsymbol{V}}$ (which belong to $\mathbb{F}_{q^m}$) and in the maximal minors of the matrix

$$\boldsymbol{D} \stackrel{def}{=} \begin{pmatrix} \boldsymbol{C}_1 \\ \vdots \\ \boldsymbol{C}_n \end{pmatrix},$$

whose coefficients are in $\mathbb{F}_q$. Similarly, the maximal minors of $\mathcal{M}_j$ are simply bilinear in the entries of $\widetilde{\boldsymbol{V}}$ and in the $\binom{m}{\lambda}$ maximal minors of $\boldsymbol{C}_j$. Such a structure had already been encountered in the bilinear systems of [5,6] to attack the Rank Decoding Problem and MinRank. In particular, note that the newly introduced bilinear modeling of [5] ([5, Modeling 4]) has exactly the same shape as it involves a block of *linear variables* over the extension field $\mathbb{F}_{q^m}$ and a block of *minor variables* over $\mathbb{F}_q$.

Projection over $\mathbb{F}_q$. In **System 1**, we observed $m\binom{nr}{\lambda+1}$ (linearly independent) degree falls from degree $\lambda + 2$ to degree $\lambda + 1$ which involve these variables[6] in our experiments. Clearly, they should coincide with the projection over $\mathbb{F}_q$ of the degree fall polynomials described in Lemma 4 for **System 0**. To project the equations, note that we also have to express the entries of $\widetilde{\boldsymbol{V}}$ over $\mathbb{F}_q$. This yields r^2m variables $\vec{v_{iu}}$ and thus $r^2m\binom{mn}{\lambda}$ degree 2 monomials among these degree fall polynomials (but only $r^2m\binom{m}{\lambda}$ if we restrict ourselves to one matrix $\boldsymbol{C}_j$).

8.2 Jacobian with respect to the $\boldsymbol{C}_j$ variables

Contrary to the systems of [4,5,6] to solve the Rank Decoding Problem, a specificity of **System 1** is that the Jacobian with respect to the other block of variables also yields degree fall polynomials of low degree, for instance $\lambda + 1$. One cannot grasp them by studying **System 0** only.

Absence of degree falls for System 0. First, let us explain why we do not expect degree fall polynomials of small degree coming from this Jacobian for **System 0**. Note that the set $\mathcal{S}$ of bilinear parts of the equations from this system can be written as $\mathcal{S} \stackrel{def}{=} \cup_{j=1}^{n} \mathcal{S}_j$, where the polynomials in $\mathcal{S}_j$ are defined

[6] Section 8.2 will give another type of degree falls in the same degree

as the entries of the matrix $\widehat{\mathbf{H}}_{norm}\boldsymbol{C}_j\boldsymbol{R}\widehat{\boldsymbol{\mathcal{B}}}^{\mathsf{T}}$ at the right hand side of Equation (9). The $\boldsymbol{R}\widehat{\boldsymbol{\mathcal{B}}}^{\mathsf{T}}$ part being independent of j, we already obtain $\mathrm{Jac}_{\mathrm{row}(\boldsymbol{C}_j)}(\mathrm{row}(\mathcal{S}_j)) = \mathrm{Jac}_{\mathrm{row}(\boldsymbol{C}_1)}(\mathrm{row}(\mathcal{S}_1))$ for any j and thus

$$\mathrm{Jac}_{\mathrm{row}(\boldsymbol{C})}(\mathrm{row}(\mathcal{S})) = \boldsymbol{I}_n \otimes \mathrm{Jac}_{\mathrm{row}(\boldsymbol{C}_1)}(\mathrm{row}(\mathcal{S}_1)).$$

Then, to compute $\mathrm{Jac}_{\mathrm{row}(\boldsymbol{C}_1)}(\mathrm{row}(\mathcal{S}_1))$, we apply Lemma 3 once again this time with $\boldsymbol{X} \stackrel{def}{=} \widehat{\mathbf{H}}_{norm}$, $\boldsymbol{A} \stackrel{def}{=} \boldsymbol{C}_1$ and $\boldsymbol{Y} \stackrel{def}{=} \boldsymbol{R}\widehat{\boldsymbol{\mathcal{B}}}^{\mathsf{T}}$. This yields

$$\mathrm{Jac}_{\mathrm{row}(\boldsymbol{C}_1)}(\mathrm{row}(\mathcal{S}_1)) = \widehat{\mathbf{H}}_{norm} \otimes \widehat{\boldsymbol{\mathcal{B}}}\boldsymbol{R}^{\mathsf{T}}. \tag{15}$$

This matrix is of size $r \times m\lambda$ and its entries are linear forms in the $\boldsymbol{R}$ variables. However, we cannot apply Lemma 2 since $r < m\lambda$. We expect a trivial left kernel for this matrix.

Additional degree falls for System 1. We analyze the situation over $\mathbb{F}_q$ by studying the **System 2** introduced in Section 7.2, which contains the same information as **System 1**. From now on we fix $\widehat{\boldsymbol{\mathcal{B}}} = \mathcal{A}$. As in the previous section, we can clearly reason in a similar way for all indexes $1 \leq j \leq n$. For $1 \leq j \leq n$ and $0 \leq \ell \leq m-1$, let us consider Equation (10) and for $1 \leq u \leq r$, let us denote by $g_{u,\ell,j}$ the bilinear polynomial

$$g_{u,\ell,j} \stackrel{def}{=} \left(\widehat{\mathbf{H}}_{norm}^{[\ell]}\right)_{u,*} \boldsymbol{C}_j\boldsymbol{R}\left(\mathcal{A}^{[\ell]}\right)^{\mathsf{T}} = \left(\mathcal{A}^{[\ell+u-1]}\right)\boldsymbol{C}_j\boldsymbol{R}\left(\mathcal{A}^{[\ell]}\right)^{\mathsf{T}}.$$

We also keep track of the corresponding linear part $L_{u,\ell,j} \stackrel{def}{=} \boldsymbol{V}_{u,*}^{[\ell]}\left(\widehat{\boldsymbol{h}_j^{[\ell]}}\right)^{\mathsf{T}}$, so that the whole equation reads $g_{u,\ell,j} - L_{u,\ell,j} = 0$. We then group the equations according to the value of $v \stackrel{def}{=} u+\ell-1 \bmod m$. We obtain the following equality, where all ℓ indexes are modulo m,

$$\begin{aligned}
\left(L_{1,v,j}\ L_{2,v-1,j}\ \ldots\ L_{r,v-r+1,j}\right) &= \left(g_{1,v,j}\ \ldots\ g_{r,v-r+1,j}\right)\\
\widehat{\boldsymbol{h}_j^{[\ell]}}\left(\boldsymbol{V}^{[\ell]}\right)^{\mathsf{T}} &= \mathcal{A}^{[v]}\boldsymbol{C}_j\boldsymbol{R}\left(\left(\mathcal{A}^{[v]}\right)^{\mathsf{T}} \vdots \left(\mathcal{A}^{[v-r+1]}\right)^{\mathsf{T}}\right)\\
&= \mathcal{A}^{[v]}\boldsymbol{C}_j\boldsymbol{R}\left(\widehat{\mathbf{H}}_{inv}^{[v]}\right)^{\mathsf{T}},
\end{aligned}$$

and where

$$\widehat{\mathbf{H}}_{inv} \stackrel{def}{=} \begin{pmatrix} \mathcal{A} \\ \ldots \\ \mathcal{A}^{-[r-1]} \end{pmatrix} \in \mathcal{M}_{m,r}(\mathbb{F}_{q^m}).$$

Using Lemma 3, we then compute the Jacobian matrix of these equations with respect to the $\boldsymbol{C}_j$ variables with $\boldsymbol{A} \stackrel{def}{=} \mathcal{A}^{[v]}$, $\boldsymbol{X} \stackrel{def}{=} \boldsymbol{C}_j$ and $\boldsymbol{Y} \stackrel{def}{=} \boldsymbol{R}\left(\widehat{\mathbf{H}}_{inv}^{[v]}\right)^{\mathsf{T}}$. This gives

$$\mathrm{Jac}_{\mathrm{row}(\boldsymbol{C}_j)}\left(g_{1,\ell_1,j}\ \ldots\ g_{r,\ell_r,j}\right) = \mathcal{A}^{[v]} \otimes \widehat{\mathbf{H}}_{inv}^{[v]}\boldsymbol{R}^{\mathsf{T}}.$$

We can continue as above to obtain Lemma 5, whose proof is analogous to the one of Lemma 4.

Lemma 5 *For any fixed column $\boldsymbol{h}_j$ in $\boldsymbol{H}_{pub}$, for $0 \leq \ell \leq m-1$ and for a modulus $0 \leq v \leq m-1$, there are $\binom{r}{\lambda+1}$ degree falls from degree $\lambda+2$ to $\lambda+1$ which are given by the maximal minors of the matrix*

$$\mathcal{N}_{j,\ell,v} \stackrel{def}{=} \begin{pmatrix} \widehat{\boldsymbol{h}_j^{[\ell]}} \left(\boldsymbol{V}^{[\ell]}\right)^{\mathsf{T}} \\ \boldsymbol{R} \left(\widehat{\mathbf{H}}_{inv}^{[v]}\right)^{\mathsf{T}} \end{pmatrix}, \tag{16}$$

where

$$\widehat{\mathbf{H}}_{inv} \stackrel{def}{=} \begin{pmatrix} \mathcal{A} \\ \cdots \\ \mathcal{A}^{-[r-1]} \end{pmatrix}.$$

By gathering the equations for all columns $\boldsymbol{h}_j$ in $\boldsymbol{H}_{\text{pub}}$, all indexes ℓ and all moduli v, we obtain a system of $nm^2\binom{r}{\lambda+1}$ polynomials of degree $\lambda+1$. Similarly to the above, these polynomials have a bilinear structure: they are bilinear in the entries of the $\boldsymbol{V}^{[\ell]}$'s and in the maximal minors r_T of $\boldsymbol{R}$. Coming back to the **System 1** over $\mathbb{F}_q$ that we want to solve, this will correspond to an extra set of $nm^2\binom{r}{\lambda+1}$ polynomials of degree $\lambda+1$ which are produced in degree $\lambda+2$ by the computation. They can also be seen as an affine bilinear system involving $mr^2\binom{m}{\lambda}$ quadratic monomials.

8.3 Approach based on degree fall polynomials

Instead of simply solving the original bilinear system, our results suggest another method by focusing on a system of degree fall polynomials of degree $\lambda+1$. It would consist of the one given by Lemma 4, Lemma 5 or a subset of such equations. As we have just seen, this approach would benefit from the compactness of these polynomials since they have a specific bilinear structure. Its analysis is left for future work, including the study of linear dependencies and the possibility of using hybrid techniques.

In the case of the Rank Decoding Problem, solving the system given by the MaxMinors equations [6] lead to a significant improvement compared to attacks based on Ourivski-Johansson [21,4]. In our case, however, the same will not necessarily hold. First, the ratio between equations and variables in Lemma 4 or Lemma 5 seems less favorable than in [6]. Second, our experiments suggest that the degree falls polynomials in degree $\lambda+2$ do not mark the end of the computation on the original system in general, whereas it was often the case for the Rank Decoding Problem [18,4].

9 Conclusion

In the paper we presented two different approaches to *distinguish* a public-key from random.

The combinatorial approach seems to have reached its limits as is the case for the problem of decoding in rank metric and we do not expect significant gain (say non-polynomial improvements on the complexity) from further improvements, except if there is a major theoretical breakthrough, but who would probably also extend to the problem of decoding in rank metric.

Concerning the algebraic approach, it is more difficult to ascertain that no significant improvements are to be expected. Namely, as is the case for solving non-linear system, a smarter approach to rewrite the system could lead to major improvements. Anyway it is certainly an interesting field of research to obtain a finer analysis of system solving.

References

1. Aragon, N., Dyseryn, V., Gaborit, P., Loidreau, P., Renner, J., Wachter-Zeh, A.: LowMS: a new rank metric code-based KEM without ideal structure. Cryptology ePrint Archive, Paper 2022/1596 (2022), https://eprint.iacr.org/2022/1596, https://eprint.iacr.org/2022/1596
2. Aragon, N., Gaborit, P., Hauteville, A., Tillich, J.P.: A new algorithm for solving the rank syndrome decoding problem. In: 2018 IEEE International Symposium on Information Theory, ISIT 2018, Vail, CO, USA, June 17-22, 2018. pp. 2421–2425. IEEE (2018), https://doi.org/10.1109/ISIT.2018.8437464
3. Bardet, M., Briaud, P.: An algebraic approach to the Rank Support Learning problem. In: Cheon, J.H., Tillich, J.P. (eds.) PQCrypto. lncs, Springer International Publishing (2021)
4. Bardet, M., Briaud, P., Bros, M., Gaborit, P., Neiger, V., Ruatta, O., Tillich, J.: An Algebraic Attack on Rank Metric Code-Based Cryptosystems. In: Advances in Cryptology - EUROCRYPT 2020 - 30th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, May 10-14, 2020. Proceedings (2020), https://doi.org/10.1007/978-3-030-45727-3_3
5. Bardet, M., Briaud, P., Bros, M., Gaborit, P., Tillich, J.P.: Revisiting Algebraic Attacks on MinRank and on the Rank Decoding Problem. Cryptology ePrint Archive, Paper 2022/1031 (2022), https://eprint.iacr.org/2022/1031, https://eprint.iacr.org/2022/1031
6. Bardet, M., Bros, M., Cabarcas, D., Gaborit, P., Perlner, R., Smith-Tone, D., Tillich, J.P., Verbel, J.: Improvements of Algebraic Attacks for solving the Rank Decoding and MinRank problems. In: ASIACRYPT 2020, International Conference on the Theory and Application of Cryptology and Information Security, 2020. Proceedings. pp. 507–536 (2020), https://doi.org/10.1007/978-3-030-64837-4_17
7. Chabaud, F., Stern, J.: The Cryptographic Security of the Syndrome Decoding Problem for Rank Distance Codes. In: Advances in Cryptology - ASIACRYPT 1996. LNCS, vol. 1163, pp. 368–381. Springer, Kyongju, Korea (Nov 1996)
8. Coggia, D., Couvreur, A.: On the security of a Loidreau rank metric code based encryption scheme. Des. Codes Cryptogr. **88**(9), 1941–1957 (2020)
9. Coppersmith, D.: Solving homogeneous linear equations over GF(2) via block Wiedemann algorithm. Mathematics of Computation **62**, 333–350 (1994)
10. Delsarte, P.: Bilinear Forms over a Finite Field, with Applications to Coding Theory. J. Comb. Theory, Ser. A **25**(3), 226–241 (1978)

11. Faugère, J.C.: A New Efficient Algorithm for Computing Gröbner Bases (F4). J. Pure Appl. Algebra **139**(1-3), 61–88 (1999)
12. Faugère, J.C.: A New Efficient Algorithm for Computing Gröbner Bases without Reduction to Zero: F5. In: Proceedings ISSAC'02. pp. 75–83. ACM press (2002)
13. Faugère, J.C., Safey El Din, M., Spacodecrenlehauer, P.J.: Gröbner bases of bihomogeneous ideals generated by polynomials of bidegree (1,1): Algorithms and complexity. J. Symbolic Comput. **46**(4), 406–437 (2011)
14. Gabidulin, E.M.: Theory of codes with maximum rank distance. Problemy Peredachi Informatsii **21**(1), 3–16 (1985)
15. Gabidulin, E.M., Paramonov, A.V., Tretjakov, O.V.: Ideals over a non-commutative ring and their applications to cryptography. In: Advances in Cryptology - EUROCRYPT'91. pp. 482–489. No. 547 in LNCS, Brighton (Apr 1991)
16. Gaborit, P., Hauteville, A., Phan, D.H., Tillich, J.: Identity-based Encryption from Rank Metric. In: Advances in Cryptology - CRYPTO2017. LNCS, vol. 10403, pp. 194–226. Springer, Santa Barbara, CA, USA (Aug 2017), https://doi.org/10.1007/978-3-319-63697-9_7
17. Gaborit, P., Ruatta, O., Schrek, J.: On the Complexity of the Rank Syndrome Decoding Problem. IEEE Trans. Information Theory **62**(2), 1006–1019 (2016)
18. Levy-dit-Vehel, F., Perret, L.: Algebraic decoding of rank metric codes. Talk at the Special Semester on Gröbner Bases - Workshop D1 pp. 1–19 (2006), https://ricamwww.ricam.oeaw.ac.at/specsem/srs/groeb/download/Levy.pdf
19. Loidreau, P.: A new rank metric codes based encryption scheme. In: Post-Quantum Cryptography 2017. LNCS, vol. 10346, pp. 3–17. Springer (2017)
20. McEliece, R.J.: A Public-Key System Based on Algebraic Coding Theory, pp. 114–116. Jet Propulsion Lab (1978), dSN Progress Report 44
21. Ourivski, A.V., Johansson, T.: New Technique for Decoding Codes in the Rank Metric and Its Cryptography Applications. Problems of Information Transmission **38**(3), 237–246 (2002). https://doi.org/10.1023/A:1020369320078
22. Overbeck, R.: A New Structural Attack for GPT and Variants. In: Mycrypt. LNCS, vol. 3715, pp. 50–63 (2005)
23. Verbel, J., Baena, J., Cabarcas, D., Perlner, R., Smith-Tone, D.: On the Complexity of "Superdetermined" Minrank Instances. In: Post-Quantum Cryptography 2019. LNCS, vol. 11505, pp. 167–186. Springer, Chongqing, China (May 2019). https://doi.org/10.1007/978-3-030-25510-7_10, https://doi.org/10.1007/978-3-030-25510-7_10