Research Article

Antonio J. Di Scala and Carlo Sanna*

Smaller public keys for MinRank-based schemes

https://doi.org/10.1515/jmc-2024-0008
received February 14, 2024; accepted October 15, 2024

Abstract: MinRank is an NP-complete problem in linear algebra whose characteristics make it attractive to build post-quantum cryptographic primitives. Several MinRank-based digital signature schemes have been proposed. In particular, two of them, MIRA and MiRitH, have been submitted to the NIST post-quantum cryptography standardization process. In this article, we propose a key-generation algorithm for MinRank-based schemes that reduces the size of the public key to about 50% of the size of the public key generated by the previous best (in terms of public-key size) algorithm. Precisely, the size of the public key generated by our algorithm sits in the range of 328–676 bits for security levels of 128–256 bits. We also prove that our algorithm is as secure as the previous ones.

Keywords: digital signatures, key generation, MinRank problem, post-quantum cryptography, public key, zero-knowledge proof of knowledge

MSC 2020: 11T71, 15A99, 94A60, 94A62

1 Introduction

MinRank is a problem in linear algebra that was first introduced by Buss et al. [1]. Roughly speaking, given $k+1$ matrices $M_0, \ldots, M_k$ of size $m \times n$ over a finite field $\mathbb{F}_q$, the decisional version of MinRank asks to determine if there exists a non-trivial linear combination of $M_0, \ldots, M_k$ whose rank does not exceed a fixed parameter r. The search version of MinRank, which is the one we will be focusing on hereafter, asks to find such a linear combination.

For several reasons, MinRank is an attractive candidate to build post-quantum cryptographic primitives. First, MinRank is completely based on simple linear algebra operations, which can be implemented easily and efficiently. Second, the hardness of MinRank is supported by a long line of research: MinRank is an NP-complete problem [1] and, due to its relevance in cryptanalysis [2–4], algorithms for solving it have been extensively studied, to the extent that random instances of MinRank are expected to be hard [5–12]. Finally, there are no known quantum algorithms to solve MinRank that go beyond straightforward quantum search applications.

Several digital signature schemes based on MinRank have been proposed, namely: a scheme due to Courtois (2001) [13], MR-DSS (2022) [14], MIRA (2023) [15] (see also [16]), and MiRitH (2023) [17] (see also [18]). In particular, MIRA and MiRitH have been submitted to the NIST post-quantum cryptography standardization process.

*** Corresponding author: Carlo Sanna**, Department of Mathematical Sciences, Politecnico di Torino, Corso Duca degli Abruzzi 24, Torino, 10129, Italy, e-mail: carlo.sanna@polito.it
Antonio J. Di Scala: Department of Mathematical Sciences, Politecnico di Torino, Corso Duca degli Abruzzi 24, Torino, 10129, Italy, e-mail: antonio.discala@polito.it
ORCID: Antonio J. Di Scala 0000-0003-0758-7062; Carlo Sanna 0000-0002-2111-7596

Table 1: Comparison of the sizes of the public keys, for the parameter sets proposed for MiRitH [17, Table 1]

	Parameters					Public key (bits)		
λ	q	m	n	k	r	KeyGen1	KeyGen2	KeyGen3
128	16	15	15	78	6	1,028	716	356
128	16	16	16	142	4	1,152	584	328
192	16	19	19	109	8	1,636	1,200	592
192	16	19	19	167	6	1,636	968	512
256	16	21	21	189	7	2,020	1,264	676
256	16	22	22	254	6	2,192	1,176	648

In all these schemes, the public key is a random instance of MinRank, the secret key is the solution of such an instance, and the signing and verification algorithms together are a non-interactive zero-knowledge proof of knowledge of the solution. While the secret key can be easily compressed as a seed of λ bits, where λ is the security parameter, compressing the public key is less obvious.

Courtois [13, Section 5.1] proposed an algorithm, which we call KeyGen1, that compresses the public key in $\lambda + mn \log q$ bits, where log is the logarithm in base 2. This method was improved in MR-DSS [14, Section 4.4] by reducing the compressed public key to $\lambda + (mn - k)\log q$ bits. This improvement, which we call KeyGen2, is employed by MIRA [15, Section 2.4.1], while MiRitH uses KeyGen1 [17, Section 3.2].

We propose a new key-generation algorithm for MinRank-based schemes, which we call KeyGen3, with a compressed public key of $\lambda + (m(n - r) - k)\log q$ bits. (Note that $k < m(n - r)$. In fact, all parameter sets satisfy the stronger inequality $k < (m - r)(n - r)$, in order to make the MinRank problem *overdetermined*, see Section 2.2.)

Table 1 provides a comparison of the sizes of the public keys[1] of the three key-generation algorithms, for the parameter sets proposed for MiRitH [17, Table 1]. As it can be seen, the public-key size of KeyGen3 is about 50% of that of KeyGen2 and sits in the range of 328–676 bits for security levels of 128–256 bits.

The next theorem reduces the security of KeyGen3 to that of KeyGen1. For every $x > 0$, let $\tau(x) \coloneqq \min(0.72, 2.1x)$.

Theorem 1. *Assume that $k < m(n - r)$ and $n > 2r$ (which are satisfied in practice*, Table 1), *and that no attacker has a non-negligible advantage against the pseudorandom generators employed by* KeyGen1 *and* KeyGen3. *Let $\mathcal{A}_3$ be an attacker that, given a random public key generated by* KeyGen3, *can retrieve in time t_3 the corresponding secret key with probability p_3. Then, there exists an attacker $\mathcal{A}_1$ that, given a random public key generated by* KeyGen1, *can retrieve in time t_1 the corresponding secret key with probability p_1, where*

$$t_1 \coloneqq t_3 + \mathsf{poly}(q, m, n, k) \quad \textit{and} \quad p_1 > (1 - \tau(q^{-1}))^4 p_3.$$

Note that if we take $q = 16$ as in Table 1, then $(1 - \tau(q^{-1}))^4 > 0.56$. Roughly speaking, Theorem 1 says that the set of keys generated by KeyGen3 is equivalent (via an efficient transformation) to a large subset of the keys generated by KeyGen1, where, for $q = 16$, "large" means more than 56% of the total. Since the MinRank problem is supposed to be hard to solve on average, considering a large subset of all the possible instances remains hard to solve on average. More precisely, KeyGen3 has a security loss of less than $\log_2(1/0.56) \approx 0.836$ bits compared to KeyGen1.

The structure of the article is as follows: First, in Section 2, we provide the necessary notation (Section 2.1), the formal definition of the MinRank problem (Section 2.2), and we recall the key-generation algorithm KeyGen1 of Courtois (Section 2.3). Second, in Section 3, we describe our new key-generation algorithm KeyGen3. To simplify the exposition, we show first a partial (less efficient) version of the algorithm (Section 3.1), and then, after recalling a canonical form for MinRank instances (Section 3.2), we show the complete algorithm (Section 3.3). Finally, in Section 4, we prove Theorem 1.

1 Hereafter, we will say "public key," respectively "secret key," instead of "compressed public key," respectively "compressed secret key," since the difference will be always clear from the context.

2 Preliminaries

2.1 Notation

Let $\mathbb{F}_q$ be a finite field of q elements. For all positive integers m, n, and $r \leq \min(m, n)$, let $\mathbb{F}_q^{m\times n}$ be the vector space of $m \times n$ matrices over $\mathbb{F}_q$, and let $\mathbb{F}_q^{m\times n,r}$ be the set of $m \times n$ matrices over $\mathbb{F}_q$ having rank equal to r. For every $A \in \mathbb{F}_q^{m\times n}$, let $A^{\mathsf{T}} \in \mathbb{F}_q^{n\times m}$ be the transpose of A. Moreover, let $A^{\mathrm{L}} \in \mathbb{F}_q^{m\times(n-r)}$, respectively, $A^{\mathrm{R}} \in \mathbb{F}_q^{m\times r}$, denote the matrix consisting of the first $n - r$, respectively, the last r, columns of A, so that $A = (A^{\mathrm{L}}|A^{\mathrm{R}})$. Note that r is omitted in the notation A^{L} and A^{R}, but it will be always clear from the context. Let $\langle A\rangle \in \mathbb{F}_q^{1\times mn}$ denote the row vector consisting of the entries of A in column-major order, that is, the entries of $\langle A\rangle$ are, in order, the entries of the first column of A, followed by the entries of the second column of A, etc. Let $\langle A\rangle_i$ be the ith entry of $\langle A\rangle$. Let I_s, or just I when the dimension is clear from the context, be the identity matrix of $\mathbb{F}_q^{s\times s}$. With a slight abuse of notation, let 0 denote the zero matrix of $\mathbb{F}_q^{s\times t}$, the dimension $s \times t$ being always clear from the context. Finally, let $\delta_{i,j}$ be the Kronecker delta, let $\#S$ be the cardinality of the finite set S, and let $|\mathrm{obj}|$ be the size in bits of the object obj.

2.2 MinRank

The search version of MinRank is formally defined as follows.

Definition 1. (MinRank) Let q, m, n, k, r be positive integers, with q being a prime power and $m \geq n > r$. Given $k + 1$ matrices $M_0, \ldots, M_k \in \mathbb{F}_q^{m\times n}$, the MinRank problem asks to find $\alpha_1, \ldots, \alpha_k \in \mathbb{F}_q$ (if they exist) such that

$$E \coloneqq M_0 + \sum_{i=1}^{k} \alpha_i M_i \tag{1}$$

has rank at most r.

In MinRank-based schemes, the parameters q, m, n, k, r are selected so that: Every known algorithm to find a solution of MinRank with $\mathrm{rank}(E) = r$ requires on average at least 2^λ operations; and random instances of MinRank are expected to have exactly one solution with overwhelming probability. Consequently, the schemes have to construct the solution so that $\mathrm{rank}(E) = r$. Furthermore, to enforce the uniqueness of the solution, it is required that MinRank is *overdetermined*, that is, $k < (m - r)(n - r)$ [19, p. 33]. For details on the algorithms to solve MinRank, and consequentially on the selection of the parameters of MinRank-based schemes, see, for example, the documentation of MiRitH [17, Sections 4 and 5].

2.3 The key-generation algorithm of Courtois

We begin by briefly reviewing the algorithms proposed by Courtois [13, Section 5.1] to generate and decompress the public key and the secret key (Figure 1)[2]. It is clear that KeyGen1 in Figure 1 generates a random uniformly distributed instance of MinRank, and that the public key has a size of $|\mathrm{seed}_{\mathrm{pk}}| + |M_0| = \lambda + mn \log q$ bits. The most computationally expensive step (not taking into account the cost of running the PRG) is the

2 Actually, the key-generation algorithm in [13, Section 5.1] is slightly different from that of Figure 1 (M_k plays the role of M_0, and consequently a division by α_k is necessary). However, this makes no difference in later arguments. We stated the key-generation algorithm this way only to uniformize it with the other algorithms.

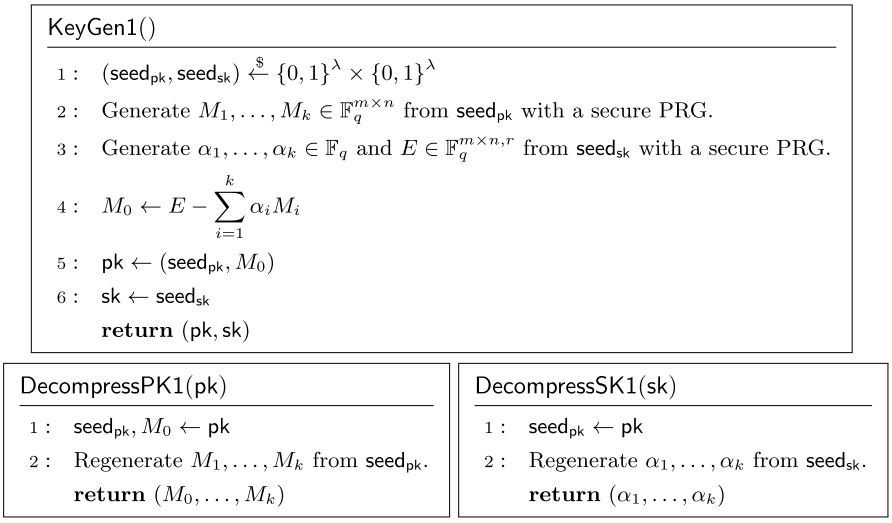

Figure 1: The algorithms of Courtois to generate and decompress the keys.

generation of E, which Courtois suggested to compute as $E = SLT$, where $L \in \mathbb{F}_q^{m \times n, r}$ is a fixed matrix and $S \in \mathbb{F}_q^{m \times m}$ and $T \in \mathbb{F}_q^{n \times n}$ are pseudorandom invertible matrices.

3 New key-generation algorithm

3.1 A first improvement

To simplify the exposition, we provide first a key-generation algorithm with a public key of $\lambda + m(n - r)\log q$ bits.

This algorithm employs the facts that: If $E \in \mathbb{F}_q^{m \times n, r}$ is taken at random with uniform distribution, then $E^{\mathrm{R}} \in \mathbb{F}_q^{m \times r, r}$ with significant probability (Lemma 6); and, in such a case, there exists a unique matrix $K \in \mathbb{F}_q^{r \times (n-r)}$ such that $E^{\mathrm{L}} = E^{\mathrm{R}}K$ (Lemma 5). Then, assuming that $E^{\mathrm{L}} = E^{\mathrm{R}}K$, it follows from (1) that

$$M_0^{\mathrm{L}} = E^{\mathrm{R}}K - \sum_{i=1}^{k} \alpha_i M_i^{\mathrm{L}}. \tag{2}$$

Hence, we can generate pseudorandom M_0^{R}, M_1,..., M_k, and K, compute

$$E^{\mathrm{R}} = M_0^{\mathrm{R}} + \sum_{i=1}^{k} \alpha_i M_i^{\mathrm{R}} \tag{3}$$

and M_0^{L} via (2), and finally pack M_0^{L} into the public key (see Figure 2 for the details). In this way, the size in bits of the public key is equal to

$$|\mathsf{seed}_{\mathrm{pk}}| + |M_0^{\mathrm{L}}| = \lambda + m(n - r)\log q.$$

Note that we cannot be sure that the matrix E^{R} computed by (3) has full rank (this, by $E^{\mathrm{L}} = E^{\mathrm{R}}K$, is equivalent to $\mathrm{rank}(E) = r$). Therefore, we have to test if $\mathrm{rank}(E^{\mathrm{R}}) < r$ (step 5 of KeyGen in Figure 2). Since E^{R} is a uniformly distributed random matrix in $\mathbb{F}_q^{m \times r}$, the probability that E^{R} is not full-rank is very small (less than $2^{-38.9}$ for the parameters in Table 1), see Lemma 3. Hence, the test has to be repeated only for a few times before finding a matrix E^{R} of full-rank.

Furthermore, note that checking if $\mathrm{rank}(E^{\mathrm{R}}) < r$ must be done in way that prevents timing attacks, so either by a constant-time algorithm (see [20] for constant-time Gaussian elimination), or by a non-constant

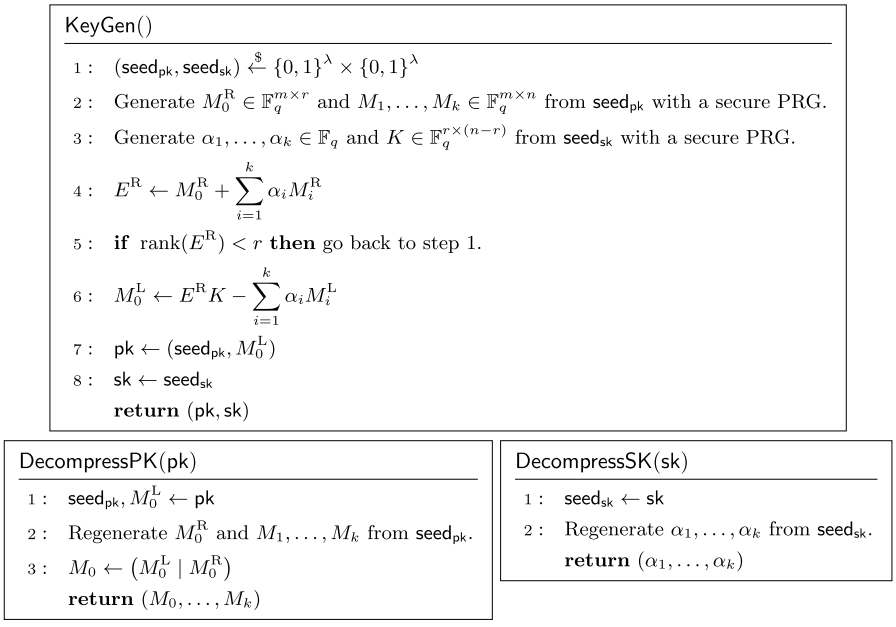

Figure 2: First version of the improved key-generation algorithm.

time algorithm that do not leak information about E^{R}. For instance, one can multiply E^{R} on the left and on the right by random invertible matrices and then check if the resulting product has rank less than r, so that the no information on E^{R} is leaked from the execution time.

3.2 Canonical form of MinRank instances

In this section, we recall a canonical form of MinRank instances that was first introduced in [14, Section 4.4].

Given a MinRank instance $\mathcal{M} = (M_0, \ldots, M_k) \in (\mathbb{F}_q^{m\times n})^{k+1}$, let $L \in \mathbb{F}_q^{(k+1)\times mn}$ be the matrix whose rows are $\langle M_1\rangle, \ldots, \langle M_k\rangle$ and $\langle M_0\rangle$, in this order. Furthermore, write

$$L = \begin{pmatrix} L_1 & L_2 \\ \ell_1 & \ell_2 \end{pmatrix},$$

where $L_1 \in \mathbb{F}_q^{k\times k}$, $L_2 \in \mathbb{F}_q^{k\times(mn-k)}$, $\ell_1 \in \mathbb{F}_q^{1\times k}$, and $\ell_2 \in \mathbb{F}_q^{1\times(mn-k)}$.

If L_1 is invertible, then we say that $\mathcal{M}$ is *reducible to canonical form* and that the *canonical form* of $\mathcal{M}$ is $\mathcal{M}' = (M_0', \ldots, M_k') \in (\mathbb{F}_q^{m\times n})^{k+1}$, where $\langle M_1'\rangle, \ldots, \langle M_k'\rangle$ and $\langle M_0'\rangle$ are the rows, in this order, of the matrix

$$L' = \begin{pmatrix} L_1^{-1} & 0 \\ -\ell_1 L_1^{-1} & 1 \end{pmatrix} L = \begin{pmatrix} I_k & L_1^{-1} L_2 \\ 0 & \ell_2 - \ell_1 L_1^{-1} L_2 \end{pmatrix}.$$

In particular, we have that $(M_0', \ldots, M_k') \in C_0 \times C_1$, where

$$C_0 = \{N \in \mathbb{F}_q^{m\times n} : \langle N\rangle_i = 0 \text{ for } i \in \{1, \ldots, k\}\}$$

and

$$C_1 = \{(N_1, \ldots, N_k) \in (\mathbb{F}_q^{m\times n})^k : \langle N_i\rangle_j = \delta_{i,j} \text{ for } i, j \in \{1, \ldots, k\}\}.$$

In general, we say that MinRank instances belonging to $C_0 \times C_1$ are in *canonical form*. If $\mathcal{M}$ is reducible to the canonical form $\mathcal{M}'$, then an easy computation shows that (1) is equivalent to

$$E \coloneqq M'_0 + \sum_{i=1}^{k} \alpha'_i M'_i,$$

where

$$(\alpha'_1 \ \ldots \ \alpha'_k) = (\alpha_1 \ \ldots \ \alpha_k)L_1 + \ell_1. \tag{4}$$

Consequently, finding a solution to the instance $\mathcal{M}$ is equivalent to finding a solution to the instance $\mathcal{M}'$.

3.3 The complete algorithm

Now, we can provide the key-generation algorithm with a public key of $\lambda + (m(n-r)-k)\log q$ bits.

The idea is to generate $M_0, \ldots, M_k$ so that they are in canonical form. In this way, the first k entries of $\langle M_0^{\mathrm{L}} \rangle$ are equal to 0, and there is no need to pack them into the public key. Thus, the size of the public key is reduced to $\lambda + (m(n-r)-k)\log q$ bits.

The KeyGen algorithm of Figure 2 can be easily modified to generate $(M_1, \ldots, M_k) \in C_1$. However, the way in which M_0^{L} is computed does not guarantee that $M_0, \ldots, M_k$ are in canonical form, i.e., that $M_0 \in C_0$. To achieve that, we have to choose $\alpha_1, \ldots, \alpha_k$ so that the first k entries of $\langle M_0^{\mathrm{L}} \rangle$ are equal to 0. Since

$$M_0^{\mathrm{L}} = \left(M_0^{\mathrm{R}} + \sum_{j=1}^{k} \alpha_j M_j^{\mathrm{R}} \right) K - \sum_{j=1}^{k} \alpha_j M_j^{\mathrm{L}}$$

and $\langle M_i^{\mathrm{L}} \rangle_j = \delta_{i,j}$ for $i, j \in \{1, \ldots, k\}$ (note that $k < m(n-r)$), this amount to solving the linear system

$$\sum_{j=1}^{k} (\delta_{i,j} - \langle M_j^{\mathrm{R}} K \rangle_i) \alpha_j = \langle M_0^{\mathrm{R}} K \rangle_i \quad (i = 1, \ldots, k). \tag{$\star$}$$

We will prove that ($\star$) has a unique solution with high probability (Lemma 8). The algorithms for the generation of the keys and their decompression are given in Figure 3.

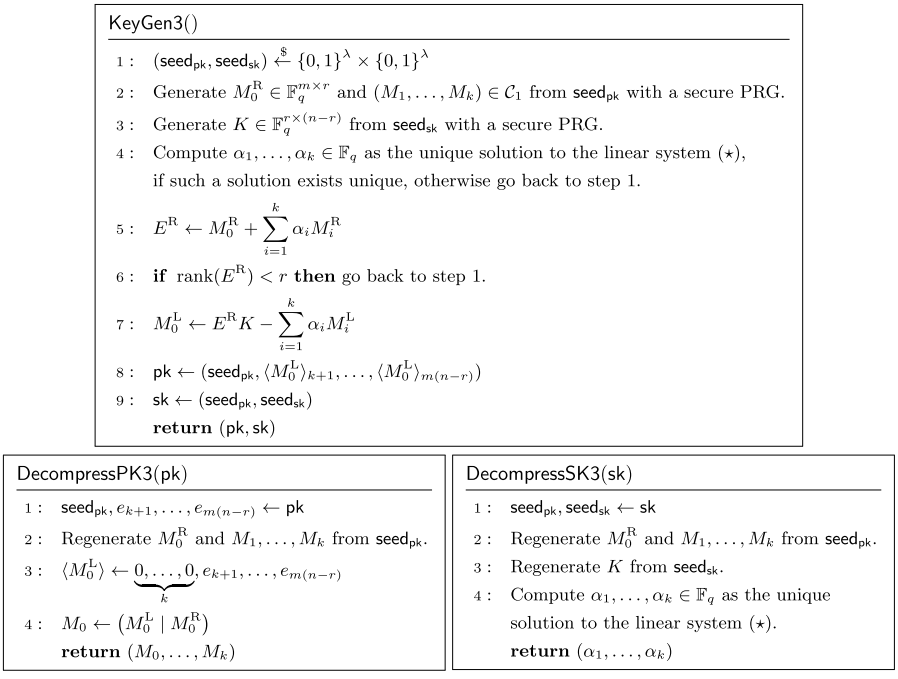

Figure 3: The proposed key-generation algorithm.

Note that solving (⋆) must be done in constant time, in order to protect the secret $\alpha_1, \ldots, \alpha_k$ from timing attacks. Furthermore, note that this construction requires to store $\mathrm{seed}_{\mathrm{pk}}$ into the secret key (see Remark 1 for a way to avoid that). However, this should not be an issue since, usually, whoever has the secret key also has the public key.

Remark 1. One of the referees pointed out that, instead of storing $\mathrm{seed}_{\mathrm{pk}}$ into the secret key, one could derive $\mathrm{seed}_{\mathrm{pk}}$ from $\mathrm{seed}_{\mathrm{sk}}$ For instance, one can set $\mathrm{seed}_{\mathrm{pk}} \leftarrow \mathrm{Hash}(\mathrm{seed}_{\mathrm{sk}})$, where Hash is a cryptographically secure hash function.

4 Proof of Theorem 1

4.1 Preliminaries

In this section, we collect some preliminary lemmas. We begin with the following inequality.

Lemma 1. *We have that*

$$\prod_{j=s}^{\infty}(1 - q^{-j}) > 1 - \tau(q^{-s}) \tag{5}$$

for all integers $s \geq 1$.

Proof. Let $P_s(q)$ denote the product in (5). First, suppose that $q^{s+1} \geq 8$. Since the logarithm is concave, we have that $\ln(1 - x) \geq -c_0 x$, for all $x_0 \in (0, 1)$ and $x \in [0, x_0]$, where

$$c_0 = c_0(x_0) \coloneqq -\frac{\ln(1 - x_0)}{x_0} > 0.$$

Hence, taking $x_0 = q^{-(s+1)}$, we obtain that

$$P_{s+1}(q) \geq \exp\left(-c_0 \sum_{j=s+1}^{\infty} q^{-j}\right) = \exp\left(-\frac{c_0 q^{-(s+1)}}{1 - q^{-1}}\right) > 1 - \frac{c_0 q^{-(s+1)}}{1 - q^{-1}},$$

where we also used the fact that $\exp(-x) > 1 - x$ for all $x > 0$. Therefore, we obtain that

$$P_s(q) > (1 - q^{-s})\left(1 - \frac{c_0 q^{-(s+1)}}{1 - q^{-1}}\right) > 1 - \left(1 + \frac{c_0}{q - 1}\right)q^{-s}.$$

Since $c_0(x_0)$ is an increasing function of x_0, it follows that $c_0(x_0) \leq c_0(1/8) < 1.1$. Hence, we obtain that

$$P_s(q) > 1 - (1 + c_0)q^{-s} > 1 - 2.1q^{-s} = 1 - \tau(q^{-s}),$$

since $2.1q^{-s} < 0.72$.

Now, suppose that $q^{s+1} < 8$. Then, $q = 2$ and $s = 1$. Moreover, we obtain that

$$\begin{aligned} P_s(q) &= (1 - 2^{-1})(1 - 2^{-2})(1 - 2^{-3})P_4(2) \\ &> (1 - 2^{-1})(1 - 2^{-2})(1 - 2^{-3})(1 - 2.1 \cdot 2^{-4}) \\ &> 1 - 0.72 \\ &= 1 - \tau(q^{-s}), \end{aligned}$$

since $2.1q^{-s} > 0.72$. The proof is complete. □

The next lemma provides a formula for the number of $m \times n$ matrices of rank r over $\mathbb{F}_q$.

Lemma 2. *We have that*

$$\#\mathbb{F}_q^{m\times n,r} = \prod_{i=0}^{r-1} \frac{(q^m - q^i)(q^n - q^i)}{q^r - q^i}.$$

Proof. See, e.g., [21]. □

The next three results are well known (more or less in these forms), but we include their proofs for completeness.

Lemma 3. *Let s, t be positive integers, and let $A \in \mathbb{F}_q^{s\times t}$ be a random matrix taken with uniform distribution. Then, the probability that* $\operatorname{rank}(A) = \min(s, t)$ *is greater than* $1 - \tau(q^{-|s-t|-1})$.

Proof. Since $\operatorname{rank}(A^{\mathrm{T}}) = \operatorname{rank}(A)$, we can assume that $s \geq t$. Hence, the probability that $\operatorname{rank}(A) = \min(s, t)$ is equal to the probability that $A \in \mathbb{F}_q^{s\times t,t}$. In turn, by Lemma 2, such a probability is equal to

$$\frac{\#\mathbb{F}_q^{s\times t,t}}{\#\mathbb{F}_q^{s\times t}} = \left(\prod_{i=0}^{t-1}(q^s - q^i)\right)\cdot q^{-st} = \prod_{i=0}^{t-1}(1 - q^{i-s}) > \prod_{j=s-t+1}^{\infty}(1 - q^{-j}),$$

and the claim follows from Lemma 1. □

Corollary 1. *Let s be a positive integer and let $A \in \mathbb{F}_q^{s\times s}$ be a random matrix taken with uniform probability. Then, the probability that A is invertible is greater than* $1 - \tau(q^{-1})$.

Lemma 4. *Let $A \in \mathbb{F}_q^{s\times s,s}$ be a random matrix with an arbitrary probability distribution, and let $B \in \mathbb{F}_q^{s\times t}$ (respectively, $C \in \mathbb{F}_q^{t\times s}$) be a random uniformly distributed matrix independent from A. Then, the matrix AB (respectively, CA) is uniformly distributed in $\mathbb{F}_q^{s\times t}$ (respectively $\mathbb{F}_q^{t\times s}$) and independent from A.*

Proof. It suffices to prove the claim for B. Then, the claim for C follows by matrix transposition. For each $D \in \mathbb{F}_q^{s\times t}$, we have that

$$\begin{aligned}\Pr[AB = D] &= \sum_{A_0\in\mathbb{F}_q^{s\times s,s}} \Pr[A = A_0]\Pr[B = A_0^{-1}D] \\ &= \sum_{A_0\in\mathbb{F}_q^{s\times s,s}} \Pr[A = A_0]\frac{1}{\#\mathbb{F}_q^{s\times t}} = \frac{1}{\#\mathbb{F}_q^{s\times t}}.\end{aligned}$$

Hence, we obtain that AB is uniformly distributed in $\mathbb{F}_q^{s\times t}$.

For each $E \in \mathbb{F}_q^{s\times s,s}$, we have

$$\Pr[AB = D|A = E] = \Pr[EB = D] = \Pr[B = E^{-1}D] = \frac{1}{\#\mathbb{F}_q^{s\times t}} = \Pr[AB = D],$$

since B and AB are uniformly distributed. □

Let $\mathcal{E}$ be the set of $E \in \mathbb{F}_q^{m\times n,r}$ such that $E^{\mathrm{R}} \in \mathbb{F}_q^{m\times r,r}$.

Lemma 5. *Let $E \in \mathbb{F}_q^{m\times n,r}$. Then, $E \in \mathcal{E}$ if and only if $E^{\mathrm{L}} = E^{\mathrm{R}}K$ for some $K \in \mathbb{F}_q^{r\times(n-r)}$. In such a case, we have that K is unique.*

Proof. First, suppose that $E \in \mathcal{E}$. Then, the columns of E^{R} generate the column-space of E. Consequently, the columns of E^{L} are a linear combination of those of E^{R}, that is, $E^{\mathrm{L}} = E^{\mathrm{R}}K$ for some $K \in \mathbb{F}_q^{r\times(n-r)}$. Moreover, the matrix K is unique, since the columns of E^{R} are linearly independent. Vice versa, if $E^{\mathrm{L}} = E^{\mathrm{R}}K$

for some $K \in \mathbb{F}_q^{r\times(n-r)}$, then the column-space of E is generated by the columns of E^{R}. Since E has rank r, it follows that $E^{\mathrm{R}} \in \mathbb{F}_q^{m\times r,r}$, that is $E \in \mathcal{E}$. □

Lemma 6. *Let $E \in \mathbb{F}_q^{m\times n,r}$ be a random matrix taken with uniform distribution. Then, $E \in \mathcal{E}$ with probability greater that $1 - \tau(q^{-1})$. In such a case, the unique matrix $K \in \mathbb{F}_q^{r\times(n-r)}$ such that $E^{\mathrm{L}} = E^{\mathrm{R}}K$* (Lemma 5) *is uniformly distributed in $\mathbb{F}_q^{r\times(n-r)}$.*

Proof. By Lemma 5, the map Φ that sends each $E \in \mathcal{E}$ to (E^{R}, K), where $K \in \mathbb{F}_q^{r\times(n-r)}$ is the unique matrix such that $E^{\mathrm{L}} = E^{\mathrm{R}}K$, is a bijection

$$\mathcal{E} \to \mathbb{F}_q^{m\times r,r} \times \mathbb{F}_q^{r\times(n-r)}.$$

Hence, by Lemma 2, the probability that $E \in \mathcal{E}$ is equal to

$$\begin{aligned}\frac{\#\mathbb{F}_q^{m\times r,r} \cdot \#\mathbb{F}_q^{r\times(n-r)}}{\#\mathbb{F}_q^{m\times n,r}} &= \left(\prod_{i=0}^{r-1}(q^m - q^i)\right) \cdot q^{r(n-r)} \cdot \left(\prod_{i=0}^{r-1}\frac{(q^m - q^i)(q^n - q^i)}{q^r - q^i}\right)^{-1} \\ &= \prod_{i=0}^{r-1}\frac{(q^r - q^i)q^{n-r}}{q^n - q^i} = \prod_{i=0}^{r-1}\frac{1 - q^{i-r}}{1 - q^{i-n}} > \prod_{i=0}^{r-1}(1 - q^{i-r}) \\ &> \prod_{j=1}^{\infty}(1 - q^{-j}) > 1 - \tau(q^{-1}),\end{aligned}$$

where the last inequality follows from Lemma 1.

Furthermore, again since Φ is a bijection, we obtain that K is uniformly distributed in $\mathbb{F}_q^{r\times(n-r)}$. □

The next lemma regards the probability that a MinRank instance can be reduced to canonical form, and the distributions of its canonical form and the corresponding solution.

Lemma 7. *Assume that $M_1, \ldots, M_k \in \mathbb{F}_q^{m\times n}$, $\alpha_1, \ldots, \alpha_k \in \mathbb{F}_q$, and $E \in \mathbb{F}_q^{m\times n,r}$ are independent and uniformly distributed in their respective spaces. Set*

$$M_0 \coloneqq E - \sum_{i=1}^{k}\alpha_i M_i.$$

Then, $M_0, \ldots, M_k$ can be reduced to canonical form with probability greater than $1 - \tau(q^{-1})$. In such a case, letting $M_0', \ldots, M_k'$ be the canonical form of $M_0, \ldots, M_k$, and letting $\alpha_1', \ldots, \alpha_k'$ be given by (4)*, we have that $(M_1', \ldots, M_k')$ and $(\alpha_1', \ldots, \alpha_k')$ are independent and uniformly distributed in C_1 and $\mathbb{F}_q^k$, respectively.*

Proof. With the notation of Section 3.2, we have that

$$\begin{pmatrix}\langle M_1\rangle \\ \vdots \\ \langle M_k\rangle\end{pmatrix} = (L_1 \quad L_2).$$

Hence, it follows that $L_1 \in \mathbb{F}_q^{k\times k}$ and $L_2 \in \mathbb{F}_q^{k\times(mn-k)}$ are independent and uniformly distributed. Since $M_0, \ldots, M_k$ can be reduced to canonical form exactly when the matrix L_1 is invertible, it follows from Corollary 1 that the probability that the reduction is possible is greater than $1 - \tau(q^{-1})$. Furthermore, if L_1 is invertible, we have that

$$\begin{pmatrix}\langle M_1'\rangle \\ \vdots \\ \langle M_k'\rangle\end{pmatrix} = (I_k \quad L_1^{-1}L_2),$$

and the claim about the distribution of $(M_1', \ldots, M_k')$ and $(\alpha_1', \ldots, \alpha_k')$ follows from Lemma 4. □

We conclude with a lemma concerning the invertibility of a certain matrix.

Lemma 8. *Let $N_1, \ldots, N_k \in \mathbb{F}_q^{m\times r}$ and $K \in \mathbb{F}_q^{r\times(n-r)}$ be random matrices that are independent and uniformly distributed in their respective spaces. Let $X \in \mathbb{F}_q^{k\times k}$ be the matrix whose entry of the ith row and jth column is equal to $\langle N_j K\rangle_i$. Then*

$$\Pr[I - X \in \mathbb{F}_q^{k\times k,k}] > (1 - \tau(q^{-1}))^2.$$

Proof. Let $\rho(s)$ be the probability that a uniformly distributed random matrix in $\mathbb{F}_q^{s\times s}$ is invertible. Write $K = (K_1|K_2)$, where $K_1 \in \mathbb{F}_q^{r\times r}$ and $K_2 \in \mathbb{F}_q^{r\times(n-2r)}$ (recall that $n > 2r$). Note that

$$\begin{aligned}\Pr[I - X \in \mathbb{F}_q^{k\times k,k}] &\geq \Pr[I - X \in \mathbb{F}_q^{k\times k,k} \text{ and } K_1 \in \mathbb{F}_q^{r\times r,r}]\\ &= \Pr[I - X \in \mathbb{F}_q^{k\times k,k} | K_1 \in \mathbb{F}_q^{r\times r,r}]\ \Pr[K_1 \in \mathbb{F}_q^{r\times r,r}]\\ &= \Pr[I - X \in \mathbb{F}_q^{k\times k,k} | K_1 \in \mathbb{F}_q^{r\times r,r}]\ \rho(r).\end{aligned} \tag{6}$$

Therefore, it suffices to prove that the conditional probability in (6) is equal to $\rho(\min(mr, k))$, and then the claim follows from Corollary 1.

Hereafter, assume that K_1 is invertible. Let $N'_j \coloneqq N_j K_1$ for each $j \in \{1, \ldots, k\}$. By Lemma 4, we have that $N'_1, \ldots, N'_k$ are independent, uniformly distributed in $\mathbb{F}_q^{m\times r}$ and independent of K_1. Moreover, we have that $N_j K = (N'_j | N'_j K_1^{-1} K_2)$ for each $j \in \{1, \ldots, k\}$. Consequently, we obtain that $\langle N_j K\rangle_i = \langle N'_j\rangle_i$ for all positive integers $i \leq mr$ and $j \leq k$.

If $mr \geq k$, then it follows that $\langle N_j K\rangle_i = \langle N'_j\rangle_i$ for each $i, j \in \{1, \ldots, k\}$. Hence, X is uniformly distributed in $\mathbb{F}_q^{k\times k}$. Thus, the conditional probability in (6) is equal to $\rho(k)$, as desired.

Assume that $mr < k$. It follows easily that there exists a matrix $H \in \mathbb{F}_q^{mr\times(k-mr)}$, which is completely determined by K, such that $X = (I_{mr}|H)^{\mathsf{T}} J$, where

$$J \coloneqq (\langle N'_1\rangle^{\mathsf{T}} \ \ldots \ \langle N'_k\rangle^{\mathsf{T}})$$

is uniformly distributed in $\mathbb{F}_q^{mr\times k}$.

Note that the matrix

$$P \coloneqq \begin{pmatrix} I_{mr} & 0 \\ H^{\mathsf{T}} & -I_{k-mr} \end{pmatrix}$$

satisfies $P(I_{mr}|H)^{\mathsf{T}} = (I_{mr}|0)^{\mathsf{T}}$ and $P^2 = I$. In particular, P is invertible. Hence, by Lemma 4, we have that $J' \coloneqq JP$ is uniformly distributed in $\mathbb{F}_q^{mr\times k}$. Write $J' = (J'_1|J'_2)$, where $J'_1 \in \mathbb{F}_q^{mr\times mr}$ and $J'_2 \in \mathbb{F}_q^{mr\times(k-mr)}$ are independent and uniformly distributed. Then, we have that

$$\begin{aligned}P(I - X)P &= P^2 - PXP = I - P(I_{mr}|H)^{\mathsf{T}} JP\\ &= I - (I_{mr}|0)^{\mathsf{T}} J' = \begin{pmatrix} I_{mr} - J'_1 & -J'_2 \\ 0 & I_{k-mr} \end{pmatrix}.\end{aligned}$$

Consequently, we obtain that $I - X$ is invertible if and only if $I - J'_1$ is invertible. Therefore, the conditional probability in (6) is equal to $\rho(mr)$, as desired. □

4.2 Proof of Theorem 1

Our strategy to prove Theorem 1 is the following. First, we provide an algorithm $\mathcal{R}$ that takes as input a random instance of MinRank $\mathcal{M}$ generated by KeyGen1 and, with probability greater than $(1 - \tau(q^{-1}))^4$, returns as output the canonical form $\mathcal{M}'$ of $\mathcal{M}$ together with the matrices L_1, ℓ_1 described in Section 3.2.

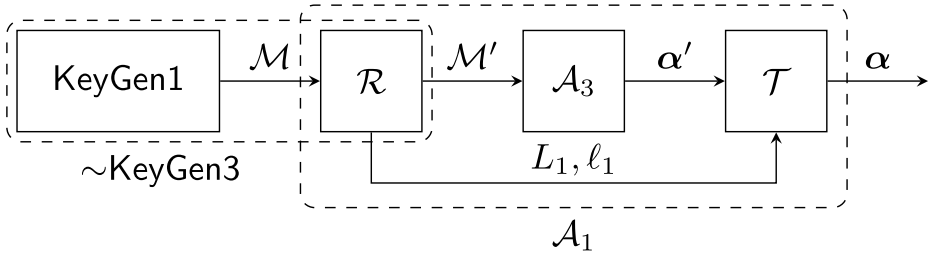

Figure 4: An illustration of the strategy of the proof of Theorem 1. The block labeled ~KeyGen3 returns a random MinRank instance $\mathcal{M}'$ having the same probability distribution of the output of KeyGen3.

Second, we show that $\mathcal{M}'$ follows the same probability distribution of a random MinRank instance generated by KeyGen3. Let $\mathcal{A}_1$ be an attacker built from $\mathcal{R}$, $\mathcal{A}_3$, and $\mathcal{T}$ as in Figure 4, where $\mathcal{T}$ is the algorithm computing

$$(\alpha_1 \dots \alpha_k) = ((\alpha_1' \dots \alpha_k') - \ell_1)L_1^{-1},$$

in light of (4). Since the attacker $\mathcal{A}_3$ can solve $\mathcal{M}'$ with probability p_3, we obtain that $\mathcal{A}_1$ can solve $\mathcal{M}$ with probability $p_1 > (1 - \tau(q^{-1}))^4 p_3$, as desired. Moreover, it will be clear that the algorithms $\mathcal{R}$ and $\mathcal{T}$ have complexities that are polynomial in q, m, n, k. Hence, we obtain that $t_1 = t_3 + \text{poly}(q, m, n, k)$, as claimed.

Let $M_0, \dots, M_k \in \mathbb{F}_q^{m \times n}$, $E \in \mathbb{F}_q^{m \times n, r}$, and $\alpha_1, \dots, \alpha_k \in \mathbb{F}_q$ be generated by KeyGen1. In particular, we have that $M_1, \dots, M_k$, E, and $\alpha_1, \dots, \alpha_k$ are independent and uniformly distributed in their respective spaces.

The steps of the algorithm $\mathcal{R}$ are the following.

(1) The algorithm $\mathcal{R}$ takes as input $M_0, \dots, M_k$.
(2) If $M_0, \dots, M_k$ cannot be reduced to canonical form, then stop. Otherwise, if $M_0, \dots, M_k$ can be reduced to canonical form, then compute the canonical form $M_0', \dots, M_k'$ and the conversion matrices L_1, ℓ_1, as described in Section 3.2.
(3) Return $M_0', \dots, M_k'$ and L_1, ℓ_1.

For the sake of the analysis of $\mathcal{R}$, we define the following events and objects.

(1) Event O_1 occurs if $M_0, \dots, M_k$ can be reduced to canonical form. In such a case, let $\alpha_1', \dots, \alpha_k'$ be given by (4). Note that, by Lemma 7, event O_1 happens with probability greater than $1 - \tau(q^{-1})$, while $(M_1', \dots, M_k')$ and $(\alpha_1', \dots, \alpha_k')$ are independent and uniformly distributed in C_1 and $\mathbb{F}_q^k$, respectively. Furthermore, since $k < m(n - r)$, we have that $M_1'^{\mathrm{R}}, \dots, M_k'^{\mathrm{R}}$ are independent and uniformly distributed in $\mathbb{F}_q^{m \times r}$.
(2) Event O_2 occurs if $E^{\mathrm{R}} \in \mathbb{F}_q^{m \times r, r}$. In such a case, in light of Lemma 5, let $K \in \mathbb{F}_q^{r \times (n-r)}$ be the unique matrix such that $E^{\mathrm{L}} = E^{\mathrm{R}}K$. Note that, by Lemma 6, event O_1 happens with probability greater than $1 - \tau(q^{-1})$, and K is uniformly distributed in $\mathbb{F}_q^{r \times (n-r)}$.
(3) Event O_3 occurs if both O_1 and O_2 occur and the matrix $I - X$ is invertible, where $X \in \mathbb{F}_q^{k \times k}$ is the matrix having the entry of the ith row and jth column equal to $\langle M_j'^{\mathrm{R}} K \rangle_i$. Note that, by Lemma 8, the probability that $I - X$ is invertible is greater than $(1 - \tau(q^{-1}))^2$.

By construction, we have that event O_3 happens with probability greater than $(1 - \tau(q^{-1}))^4$. Therefore, the algorithm $\mathcal{R}$ returns as output $\mathcal{M}'$ and L_1, ℓ_1 with probability greater than $(1 - \tau(q^{-1}))^4$, as desired.

It remains to prove that $\mathcal{M}'$ follows the same probability distribution of a MinRank instance generated by KeyGen3.

Let $\mathcal{S}$ be the set of

$$(M_0^*, \dots, M_k^*, E^*, \alpha_1^*, \dots, \alpha_k^*) \in C_0 \times C_1 \times \mathcal{E} \times \mathbb{F}_q^k$$

such that

(i) $E^* = M_0^* + \sum_{i=1}^k \alpha_i^* M_i^*$;

(ii) $\alpha_1^*, \ldots, \alpha_k^*$ is the unique solution to the linear system

$$\sum_{j=1}^{k} (\delta_{i,j} - \langle M_j^{*\mathrm{R}} K^* \rangle_i) x_i = \langle M_0^{*\mathrm{R}} K^* \rangle_i \quad (i = 1, \ldots, k),$$

where $K^* \in \mathbb{F}_q^{r \times (n-r)}$ is the unique matrix such that $E^{*\mathrm{L}} = E^{*\mathrm{R}} K^*$, by Lemma 5.

Note that each element of $\mathcal{S}$ is completely determined by either

(a) $M_1^*, \ldots, M_k^*, E^*$, and $\alpha_1^*, \ldots, \alpha_k^*$, since using (i) one can retrieve M_0^* from the former matrices and scalars; or

(b) $M_0^{*\mathrm{R}}, M_1^*, \ldots, M_k^*$, and K^*. In fact, given such matrices, one can retrieve $\alpha_1^*, \ldots, \alpha_k^*$ by using (ii). Then, using (i), one obtains that

$$E^{*\mathrm{R}} = M_0^{*\mathrm{R}} + \sum_{i=1}^{k} \alpha_i^* M_i^{*\mathrm{R}}.$$

Finally, one has that $E^* = (E^{*\mathrm{R}} K^* | E^{*\mathrm{R}})$.

Recall that $M_0, \ldots, M_k \in \mathbb{F}_q^{m \times n}$, $E \in \mathbb{F}_q^{m \times n, r}$, and $\alpha_1, \ldots, \alpha_k \in \mathbb{F}_q$ are generated by KeyGen1. If O_1 occurs, let

$$S' \coloneqq (M_0', \ldots, M_k', E, \alpha_1', \ldots, \alpha_k').$$

Note that $(M_1', \ldots, M_k')$, E, and $(\alpha_1', \ldots, \alpha_k')$ are independent and uniformly distributed in C_1, $\mathbb{F}_q^{m \times n, r}$, and $\mathbb{F}_q^k$, respectively. It follows easily that the event O_3 happens if and only if $S' \in \mathcal{S}$. Hence, thanks to (a), we obtain that, conditionally to the event O_3, the random variable S' is uniformly distributed in $\mathcal{S}$.

Let $M_0^\circ, \ldots, M_k^\circ, K^\circ, E^{\circ\mathrm{R}}$, and $\alpha_1^\circ, \ldots, \alpha_k^\circ$ be the matrices and the scalars generated by KeyGen3. Also, put $E^\circ \coloneqq (E^{\circ\mathrm{R}} K^\circ | E^{\circ\mathrm{R}})$ and

$$S^\circ \coloneqq (M_0^\circ, \ldots, M_k^\circ, E^\circ, \alpha_1^\circ, \ldots, \alpha_k^\circ).$$

It follows easily that KeyGen3 generates ${M_0^\circ}^{\mathrm{R}}$, $(M_1^\circ, \ldots, M_k^\circ)$, and K independently and with uniform distribution in $\mathbb{F}_q^{m \times r, r}$, C_1, and $\mathbb{F}_q^{r \times (n-r)}$, respectively, until the condition $S^\circ \in \mathcal{S}$ is satisfied. Hence, by (b), we obtain that S° is uniformly distributed in $\mathcal{S}$.

The proof is complete.

Acknowledgements: The authors are members of GNSAGA of INdAM and of CrypTO, the group of Cryptography and Number Theory of the Politecnico di Torino. This work was partially supported by project SERICS (PE00000014) under the MUR National Recovery and Resilience Plan funded by the European Union – NextGenerationEU. These results have been accepted for presentation at CIFRIS24, the Italian congress of De Cifris (www.decifris.it/cifris24). The authors would like to thank the anonymous referees for providing insightful comments that improved the quality of the article.

Funding information: Authors state no funding involved.

Author contributions: All authors have accepted responsibility for the entire content of this manuscript and approved its submission.

Conflict of interest: The authors state no conflict of interest.

References

[1] Buss JF, Frandsen GS, Shallit JO. The computational complexity of some problems of linear algebra. J. Comput. System Sci. 1999;58(3):572–96.

[2] Beullens W. Improved cryptanalysis of UOV and Rainbow. In: Advances in cryptology-EUROCRYPT 2021. Part I, volume 12696 of Lecture Notes in Comput. Sci., Cham: Springer; 2021. p. 348–73.

[3] Gaborit P, Ruatta O, Schrek J. On the complexity of the rank syndrome decoding problem. IEEE Trans. Inform. Theory, 2016;62(2):1006–19.

[4] Tao C, Petzoldt A, Ding J. Efficient key recovery for all HFE signature variants. In: Advances in cryptology-CRYPTO 2021. Part I, volume 12825 of Lecture Notes in Comput. Sci., Cham: Springer; 2021. p. 70–93.

[5] Bardet M, Bertin M. Improvement of algebraic attacks for solving superdetermined MinRank instances. Lecture Notes Comput Sci. 2022;13512:107–23.

[6] 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.

[7] Bardet M, Bros M, Cabarcas D, Gaborit P, Perlner R, Smith-Tone D, et al. Improvements of algebraic attacks for solving the rank decoding and MinRank problems. In: Advances in cryptology-ASIACRYPT 2020. Part I, volume 12491 of Lecture Notes in Comput. Sci., Cham: Springer; 2020. p. 507–36.

[8] Bettale L, Faugère J-C, Perret L. Cryptanalysis of HFE, multi-HFE and variants for odd and even characteristic. Des. Codes Cryptogr. 2013;69(1):1–52.

[9] Faugère J-C, Levy-dit Vehel F, Perret L. Cryptanalysis of MinRank. In: Advances in cryptology-CRYPTO 2008, volume 5157 of Lecture Notes in Comput. Sci., Berlin: Springer; 2008. p. 280–96.

[10] Faugère J-C, Safey El Din M, Spaenlehauer P-J. Computing loci of rank defects of linear matrices using Gröbner bases and applications to cryptology. In: ISSAC 2010-Proceedings of the 2010 International Symposium on Symbolic and Algebraic Computation, New York: ACM; 2010. p. 257–64.

[11] Kipnis A, Shamir A. Cryptanalysis of the HFE public key cryptosystem by relinearization. In: Advances in cryptology-CRYPTO ’99 (Santa Barbara, CA), volume 1666 of Lecture Notes in Comput. Sci., Berlin: Springer; 1999. p. 19–30.

[12] Verbel J, Baena J, Cabarcas D, Perlner R, Smith-Tone D. On the complexity of “superdetermined” MinRank instances. In: Post-quantum cryptography, volume 11505 of Lecture Notes in Comput. Sci., Cham: Springer; 2019. p. 167–86.

[13] Courtois NT. Efficient zero-knowledge authentication based on a linear algebra problem MinRank. In: Advances in cryptology-ASIACRYPT 2001 (Gold Coast), volume 2248 of Lecture Notes in Comput. Sci., Berlin: Springer; 2001. p. 402–21.

[14] Bellini E, Esser A, Sanna C, Verbel J. M-DSS-Smaller MinRank-based (ring-)signatures. In: Post-quantum cryptography, volume 13512 of Lecture Notes in Comput. Sci., Cham: Springer; 2022. p. 144–69.

[15] Aragon N, Bidoux L, Chi-Domínguez J-J, Feneuil T, Gaborit P, Neveu R, et al. MIRA: a Digital Signature Scheme based on the MinRank problem and the MPC-in-the-Head paradigm. 2023. https://arxiv.org/abs/2307.08575.

[16] Feneuil T. Building MPCitH-based Signatures from MQ, MinRank, Rank SD and PKP. Cryptology ePrint Archive, Paper 2022/1512, 2022. https://eprint.iacr.org/2022/1512.

[17] Adj G, Barbero S, Bellini E, Esser A, Rivera-Zamarripa L, Sanna C, et al. MiRitH: MinRank in the Head. Submission to NIST, 2023. https://csrc.nist.gov/csrc/media/Projects/pqc-dig-sig/documents/round-1/spec-files/MiRitH_spec-web.pdf. see also https://pqc-mirith.org.

[18] Adj G, Rivera-Zamarripa L, Verbel J. MinRank in the head. In: El Mrabet N, De Feo L, Duquesne S, editors, Progress in Cryptology - AFRICACRYPT 2023, Cham: Springer Nature Switzerland; 2023. p. 3–27.

[19] Faugère J-C, Safey El Din M, Spaenlehauer P-J. On the complexity of the generalized MinRank problem. J Symbolic Comput. 2013;55:30–58.

[20] Bernstein DJ, Chou T, Schwabe P. McBits: fast constant-time code-based cryptography. In: Bertoni G, Coron J-S, editors, Cryptographic Hardware and Embedded Systems - CHES 2013, Berlin, Heidelberg: Springer; 2013. p. 250–72.

[21] Fisher SD, Alexander MN. Classroom notes: matrices over a finite field. Amer Math Monthly. 1966;73(6):639–41.