Hybrid Subsupport Guessing: A New Hybrid Technique for the Rank Decoding Problem

Hugo Beeloo-Sauerbier Couvée[1], Antonia Wachter-Zeh[1], Violetta Weger[2]

[1] Department of Electrical and Computer Engineering, Technical University of Munich, Germany.
{hugo.sauerbier-couvee, antonia.wachter-zeh}@tum.de *
[2] Department of Mathematics, Technical University of Munich, Germany. violetta.weger@tum.de **

Keywords: Rank-metric Codes, Generic Decoding, Signature Schemes, Algebraic Decoders

Abstract. The Rank Decoding Problem (R-DP) has gained a lot of attention due to the competitive KEM proposals ROLLO and RQC, as well as the more recent signature scheme RYDE, the latter being a second-round candidate in the ongoing NIST post-quantum standardization process. While previous attacks on the R-DP are based on combinatorial methods, the seminal work of [Bardet et al., 2020] has shown the potential of attacks that use algebraic modelings, breaking the proposed parameters of ROLLO and RQC. These algebraic attacks model the R-DP as a large system of equations. For most parameter ranges, this system is underdetermined; hence, the algebraic attack first needs to perform several guessing steps to obtain a reduced instance for which the system of equations is overdetermined. These steps, in essence, guess a supersupport of the unknown error support, making this attack a hybrid approach between combinatorial and algebraic solvers.
In this paper, we present a novel type of guessing step based on searching a *sub*support of the error support. While supersupport guessing only reduces the length and dimension of the code, subsupport guessing instead reduces the length and the rank weight of the sought-after error vector. This introduces an additional method for instance reduction compatible with supersupport guessing. Both types of guessing step can be performed sequentially in hybrid attacks, and their numbers can be optimized to outperform current hybrid attacks. We provide experimentally supported comparisons of the attack complexities with and without the novel guessing technique. We measure the impact of our new hybrid attack on the RYDE parameters; for the NIST security category 5 parameters, we decrease the hybrid MaxMinors attack complexity from 301 bits to 272 bits, outperforming all other known rank decoders and tightening the margin above the 256 threshold. For the other security levels, we decrease the complexities to be on par with the best performing combinatorial decoders.

1 Introduction

Rank-metric codes are rapidly gaining attention lately due to their applications in post-quantum-secure digital signature schemes and Key Encapsulation Mechanisms (KEMs). Since the launch of NIST's post-quantum cryptography standardization process in 2016, the field has evolved significantly. The first call ended with the selection of two KEMs and three signature schemes for standardization. However, the standardization effort is far from complete; NIST opened an additional call for digital signatures in 2023, emphasizing the importance of novel schemes.

One leading approach in post-quantum cryptography uses hard problems from algebraic coding theory. Originating with McEliece's seminal work [23], code-based cryptography has grown into

* HBSC's and AWZ's work has been supported by the German Research Council (DFG) as an ANR-DFG project under Grant no. WA 3907/9-1.
** With the support of the Technical University of Munich – Institute for Advanced Study, funded by the German Excellence Initiative.

a major area of research. In particular, rank-based cryptosystems have drawn interest for their compact key and ciphertext sizes. The rank-based encryption schemes ROLLO [2] and RQC [3] were submitted to the first NIST call, and the digital signature schemes RYDE [5,4] and Mirath (merger of MIRA [6] and MiRitH [1]) have recently advanced to the second round of the additional NIST call. These proposals, and other schemes like LowMS [7], rely on the hardness of solving (variants of) the *Rank Decoding Problem* (R-DP) or the related *MinRank* problem.

This paper deals with reducing the cost of solving the R-DP: given a random generator matrix $\boldsymbol{G}$ of a code $\mathcal{C}$ over a finite field $\mathbb{F}_{q^m}$ of dimension k and length n, a received vector $\boldsymbol{y}$ and a rank weight r, find $\boldsymbol{c} \in \mathcal{C}$, such that $d_{\mathrm{R}}(\boldsymbol{c}, \boldsymbol{y}) \leq r$.

Several generic decoding algorithms for the rank metric have been proposed, either of combinatorial or algebraic nature. On the one hand, the combinatorial attacks proposed in [8,15,19,24] employ the enumeration of certain subspaces, bases, or matrices combined with solving systems of linear equations. On the other hand, recent algebraic attacks on R-DP and MinRank proposed in [9,11,12] exploit different algebraic modelings of the R-DP as a system of polynomial equations. These attacks have been celebrated as new benchmarks with lower costs in certain parameter regimes compared to the mentioned combinatorial attacks.

The algebraic modelings from [9,11,12] require that the system of equations is overdetermined. For most parameter ranges, this is not satisfied, and thus the authors suggest a hybrid approach, where they first guess certain entries of the error vector. This guessing step leads to a novel instance with reduced parameters, where the overdetermined condition is now satisfied. In more detail, given the parameters (q, m, n, k, r), the overdetermined condition is given by

$$m\binom{n-k-1}{r} \geq \binom{n}{r} - 1.$$

If this condition is not met, the authors propose a randomization step and guessing that a entries of $\boldsymbol{e}$ are zero. This requires approximately q^{ar} trials on average but reduces the parameters to $(q, m, n-a, k-a, r)$.

In this paper, we advance their idea by proposing a different guessing step. Instead of guessing zero positions of $\boldsymbol{e}$, which can be seen as guessing a *supersupport* of the unknown error support, we guess a random h-dimensional subspace $S = \ker(\boldsymbol{\Pi})$ of the unknown error support called a *subsupport*, for some $\boldsymbol{\Pi} \in \mathbb{F}_{q^m}^{n \times (n-h)}$. Multiplying our instance $(\boldsymbol{G}, \boldsymbol{y}, r)$ with $\boldsymbol{\Pi}$ results in a projection of the n-dimensional ambient space to an $(n-h)$-dimensional space. In fact, this reduces the parameters (q, m, n, k, r) to $(q, m, n-h, k, r-h)$.

Notably, both types of guesses, i.e., guessing supersupports and subsupports, can be performed sequentially and thus the number of guesses of each type can be optimized. This leads to a novel hybrid attack, able to outperform [9,11,12] for several parameter ranges such as the RYDE parameters. In particular, for the RYDE parameters corresponding to NIST security category 5, the novel attack reduces the complexity of the hybrid MaxMinors attack and outperforms the best rank-metric decoders known so far; see Table 1.

Category	q	m	n	k	r	AGHT	OJ1	Hybrid MM-$\mathbb{F}_q$	NEW hybrid MM-$\mathbb{F}_q$
1	2	53	53	45	4	148	153	169	152
3	2	61	61	51	5	217	224	245	**217**
5	2	61	57	44	7	301	287	301	**272**

Table 1. Comparison of the average time complexities of AGHT [8], OJ1 [24], hybrid MM-$\mathbb{F}_q$ [12], and our novel approach for the RYDE V2.1.0 parameters, where we take the linear algebra constant $\omega = 2$.

This paper is structured as follows. In Section 2, we provide all the necessary notation and definitions for the remainder of the paper. We also give a quick overview of the known generic decoders in the rank metric in Section 3. In Section 4 we present the novel guessing step and present the combined hybrid approach and its impact on different parameter ranges.

2 Preliminaries

Let q be a prime power and m be a positive integer. We denote by $\mathbb{F}_q$ the finite field with q elements and by $\mathbb{F}_{q^m}$ its extension field of extension degree m. For positive integers k, n we denote by $\mathbb{F}_q^{k\times n}$ the $k \times n$ matrices over $\mathbb{F}_q$. For $\boldsymbol{A} \in \mathbb{F}_q^{k\times n}$ we denote by $\boldsymbol{A}^\top$ its transpose matrix and by $\mathrm{rk}(\boldsymbol{A})$ its rank. For a vector $\boldsymbol{x} \in \mathbb{F}_{q^m}^n$, we denote by $\langle x_1, \ldots, x_n\rangle_{\mathbb{F}_q}$ the $\mathbb{F}_q$-linear subspace spanned by the entries of $\boldsymbol{x}$.

Finally, the q-binomial coefficient is defined as

$$\begin{bmatrix} n \\ k \end{bmatrix}_q = \prod_{i=0}^{k-1} \frac{q^n - q^i}{q^k - q^i} = \frac{\prod_{i=1}^{n}(1-q^i)}{\prod_{i=1}^{k}(1-q^i)\prod_{i=1}^{n-k}(1-q^i)}.$$

There are two types of rank-metric codes; the first one considers linear subspaces in $\mathbb{F}_q^{m\times n}$, thus codewords are $m \times n$ matrices, and the code is $\mathbb{F}_q$-linear. Such codes are referred to as $\mathbb{F}_q$-*linear codes*; note that in this case, the decoding problem coincides with the MinRank problem [13].

We are more interested in the second case, that is, in $\mathbb{F}_{q^m}$-*linear codes*, used in the cryptosystems ROLLO [2], RQC [3], RYDE [5,4] and LowMS [7]. An $[n, k]$ linear code $\mathcal{C} \subseteq \mathbb{F}_{q^m}^n$, for integers $k \leq n$, is a k-dimensional linear subspace of $\mathbb{F}_{q^m}^n$. The elements of a code are called *codewords* and n is called the *length* of the code.

We call $\boldsymbol{G} \in \mathbb{F}_{q^m}^{k\times n}$ a *generator matrix* of an $[n, k]$ linear code $\mathcal{C}$, if it has $\mathcal{C}$ as image, that is its rows span the subspace $\mathcal{C}$. Moreover, we call $\boldsymbol{H} \in \mathbb{F}_{q^m}^{(n-k)\times n}$ a *parity-check matrix* of $\mathcal{C}$, if it has $\mathcal{C}$ as kernel:

$$\mathcal{C} = \{\boldsymbol{x}\boldsymbol{G} \mid \boldsymbol{x} \in \mathbb{F}_{q^m}^k\} = \{\boldsymbol{c} \in \mathbb{F}_{q^m}^n \mid \boldsymbol{c}\boldsymbol{H}^\top = 0\}.$$

The *rate* of an $[n, k]$ linear code is given by $R = \frac{k}{n}$. For $\boldsymbol{x} \in \mathbb{F}_{q^m}^n$ and a parity-check matrix $\boldsymbol{H} \in \mathbb{F}_{q^m}^{(n-k)\times n}$ we call $\boldsymbol{s} = \boldsymbol{x}\boldsymbol{H}^\top \in \mathbb{F}_{q^m}^{n-k}$ a *syndrome.*

Definition 1. *Let $\boldsymbol{x} \in \mathbb{F}_{q^m}^n$. The rank weight of $\boldsymbol{x}$ is given by the $\mathbb{F}_q$-dimension of the $\mathbb{F}_q$-vector space spanned by its entries, i.e.,*

$$\mathrm{wt_R}(\boldsymbol{x}) = \dim_{\mathbb{F}_q}(\langle x_1, \ldots, x_n\rangle_{\mathbb{F}_q}).$$

Let $\boldsymbol{x} = (x_1, \ldots, x_n) \in \mathbb{F}_{q^m}^n$ and let $\Gamma = \{\gamma_1, \ldots, \gamma_m\}$ be a basis of $\mathbb{F}_{q^m}$ over $\mathbb{F}_q$. We define the *associated matrix* of $\boldsymbol{x}$ to be $\Gamma(\boldsymbol{x}) \in \mathbb{F}_q^{m\times n}$ with entries $\Gamma_{ij}(\boldsymbol{x})$, where

$$x_j = \sum_{i=1}^{m} \gamma_i \, \Gamma_{ij}(\boldsymbol{x}),$$

for all $j \in \{1, \ldots, n\}$. Note that, if we consider the vector $\gamma = (\gamma_1, \ldots, \gamma_m) \in \mathbb{F}_{q^m}^m$, we get $\boldsymbol{x} = \gamma\Gamma(\boldsymbol{x})$. Likewise, we define the *expansion* of an element $x \in \mathbb{F}_{q^m}$ to be the column vector $\Gamma(x) = (\Gamma_1(x), \ldots, \Gamma_m(x))^\top \in \mathbb{F}_q^m$, where

$$x = \sum_{i=1}^{m} \gamma_i \, \Gamma_i(x) = \gamma\Gamma(x).$$

The rank weight of $\boldsymbol{x}$ is the same as the rank of the associated matrix of $\boldsymbol{x}$, i.e.,

$$\mathrm{wt_R}(\boldsymbol{x}) = \mathrm{rk}(\Gamma(\boldsymbol{x})),$$

for any basis Γ of $\mathbb{F}_{q^m}$ over $\mathbb{F}_q$. We refine the notion of support in the rank

metric to three variants: for $\boldsymbol{x} = (x_1, \ldots, x_n) \in \mathbb{F}_{q^m}^n$ we define

- the *rank support* of $\boldsymbol{x}$ as $\mathrm{supp}_{\mathrm{R}}(\boldsymbol{x}) := \langle x_1, \ldots, x_n \rangle_{\mathbb{F}_q} \subset \mathbb{F}_{q^m}$;
- the *column (rank) support* of $\boldsymbol{x}$ as $\mathrm{supp}_{\mathrm{col}}(\boldsymbol{x}) := \mathrm{colsp}(\Gamma(\boldsymbol{x})) \subset \mathbb{F}_q^m$;
- the *row (rank) support* of $\boldsymbol{x}$ as $\mathrm{supp}_{\mathrm{row}}(\boldsymbol{x}) := \mathrm{rowsp}(\Gamma(\boldsymbol{x})) \subset \mathbb{F}_q^n$.

Here $\mathrm{rowsp}(\cdot)$ and $\mathrm{colsp}(\cdot)$ denote the row space and column space respectively. Note that the rank support and column support are equivalent as $\mathbb{F}_q$-linear subspaces via the isomorphism $\mathbb{F}_{q^m} \to \mathbb{F}_q^m$ corresponding to the basis Γ. The rank weight of $\boldsymbol{x}$ is also the $\mathbb{F}_q$-dimension of the supports, i.e.,

$$\mathrm{wt}_{\mathrm{R}}(\boldsymbol{x}) = \dim_{\mathbb{F}_q}(\mathrm{supp}_{\mathrm{R}}(\boldsymbol{x})) = \dim_{\mathbb{F}_q}(\mathrm{supp}_{\mathrm{col}}(\boldsymbol{x})) = \dim_{\mathbb{F}_q}(\mathrm{supp}_{\mathrm{row}}(\boldsymbol{x})).$$

The rank weight induces a distance: the rank distance between $\boldsymbol{x}$ and $\boldsymbol{y} \in \mathbb{F}_{q^m}^n$ is given by

$$d_{\mathrm{R}}(\boldsymbol{x}, \boldsymbol{y}) \coloneqq \mathrm{wt}_{\mathrm{R}}(\boldsymbol{x} - \boldsymbol{y}).$$

For a vector $\boldsymbol{e} \in \mathbb{F}_{q^m}^n$ of rank weight t, we can uniquely (up to a change of basis) decompose $\boldsymbol{e}$ into

$$\boldsymbol{e} = \boldsymbol{\alpha}\boldsymbol{A} = (\alpha_1, \ldots, \alpha_t) \begin{pmatrix} \boldsymbol{r}_1 \\ \vdots \\ \boldsymbol{r}_t \end{pmatrix},$$

where $\alpha_i \in \mathbb{F}_{q^m}$ are $\mathbb{F}_q$-linearly independent and $\boldsymbol{r}_i \in \mathbb{F}_q^n$ are also $\mathbb{F}_q$-linearly independent. The $\mathbb{F}_q$-span of the rows $\boldsymbol{r}_i$ coincides with the row support, that is

$$\mathrm{supp}_{\mathrm{row}}(\boldsymbol{e}) = \mathrm{rowsp}(\boldsymbol{A}) = \langle \boldsymbol{r}_1, \ldots, \boldsymbol{r}_t \rangle_{\mathbb{F}_q},$$

while the column span of $\Gamma(\boldsymbol{\alpha})$ coincides with the column support, i.e.,

$$\mathrm{supp}_{\mathrm{col}}(\boldsymbol{e}) = \mathrm{colsp}(\Gamma(\boldsymbol{\alpha})).$$

Finally, we can define the minimum rank distance of a code $\mathcal{C} \subseteq \mathbb{F}_{q^m}^n$, as

$$d_{\mathrm{R}}(\mathcal{C}) \coloneqq \min\{d_{\mathrm{R}}(\boldsymbol{x}, \boldsymbol{y}) \mid \boldsymbol{x}, \boldsymbol{y} \in \mathcal{C}, \boldsymbol{x} \neq \boldsymbol{y}\}.$$

In the rank-metric setting, there are analogous bound to the Gilbert-Varshamov bound, which we call the rank-metric Gilbert-Varshamov bound (GV). For this, we define the *relative minimum rank distance* to be $\delta = d_{\mathrm{R}}(\mathcal{C})/n$ and when considering the extension degree m as function in n, we can define $M = \lim_{n \to \infty} m(n)/n$.

Proposition 1 ([22, Thm 2]). *Let $\mathcal{C} \subseteq \mathbb{F}_{q^m}^n$ be a random $\mathbb{F}_{q^m}$-linear code of dimension $k = Rn$. For n large enough, we have that $\mathcal{C}$ has the relative minimum distance*

$$\delta = d_{\mathrm{R}}/n = M/2 + 1/2 - \sqrt{RM + (M-1)^2/4}$$

with high probability.

In particular, if $M = 1$, which will often be the case for applications, we get $\delta = 1 - \sqrt{R}$. The equation in Proposition 1 holds asymptotically as $n \to \infty$. For arbitrary n, the equation holds up to a small term depending on q, which for large n becomes negligible. Interestingly, this vanishing dependency on q is in contrast to the Hamming-metric GV bound, where the asymptotic relative minimum distance *does* depend on the field size q. Let us denote the minimum distance on the Gilbert-Varshamov bound by d_{GV}^{R}, that is

$$d_{GV}^{\mathrm{R}} = \left\lfloor n \left(M/2 + 1/2 - \sqrt{RM + (M-1)^2/4} \right) \right\rfloor.$$

Coding theory has found a crucial application in cryptography, by using the hardness of decoding a random linear code in the Hamming metric. One can formulate the same problem for rank-metric codes.

Problem 1 (Rank Decoding Problem (R-DP)). Given a generator matrix $\boldsymbol{G} \in \mathbb{F}_{q^m}^{k\times n}$ of a code $\mathcal{C}$, a vector $\boldsymbol{y} \in \mathbb{F}_{q^m}^n$ and $r \in \mathbb{N}$, find a codeword $\boldsymbol{c} \in \mathcal{C}$ such that $d_{\mathrm{R}}(\boldsymbol{c}, \boldsymbol{y}) \leq r$.

For $\mathbb{F}_{q^m}$-linear codes, it is not known whether the R-DP is NP-hard. However, in [20], the authors give a randomized reduction from the DP in the Hamming metric to the R-DP, which hints at the hardness of this problem.

Equivalently, we may formulate the R-DP using the the parity-check matrix, which results in the so-called *Rank Syndrome Decoding Problem*.

Problem 2 (Rank Syndrome Decoding Problem (R-SDP)). Given a parity-check matrix $\boldsymbol{H} \in \mathbb{F}_{q^m}^{(n-k)\times n}$, a syndrome $\boldsymbol{s} \in \mathbb{F}_{q^m}^{n-k}$ and $r \in \mathbb{N}$, find a vector $\boldsymbol{e} \in \mathbb{F}_{q^m}^n$ of rank weight $\mathrm{wt}_{\mathrm{R}}(\boldsymbol{e}) \leq r$ such that $\boldsymbol{e}\boldsymbol{H}^\top = \boldsymbol{s}$.

The equivalence between the R-SDP and R-DP is easy to show, but as we occasionally use instances of these problems interchangeably, we emphasize that for a given an R-SDP instance $(\boldsymbol{H}, \boldsymbol{s}, r)$, an equivalent R-DP instance is given by $(\boldsymbol{G}, \boldsymbol{y}, r)$ with $\boldsymbol{G}$ a corresponding generator matrix and $\boldsymbol{y} \in \mathbb{F}_{q^m}^n$ any vector (of unrestricted rank weight) satisfying $\boldsymbol{y}\boldsymbol{H}^\top = \boldsymbol{s}$. The R-SDP and R-DP are also strongly related to the MinRank problem, which has been shown to be NP-complete in [13].

Problem 3 ((Inhomogeneous) MinRank Problem). Let $\boldsymbol{G}_0, \boldsymbol{G}_1, \ldots, \boldsymbol{G}_K \in \mathbb{F}_q^{m\times n}$ and $r \leq \min\{m, n\}$ be a positive integer. Find a non-zero $(x_1, \ldots, x_K) \in \mathbb{F}_q^K$ such that

$$\mathrm{rk}(\boldsymbol{G}_0 + x_1\boldsymbol{G}_1 + \cdots + x_K\boldsymbol{G}_K) \leq r.$$

The MinRank problem coincides with the decoding problem for $\mathbb{F}_q$-linear codes and is thus equivalent to the respective syndrome decoding problem.

While any $\mathbb{F}_{q^m}$-linear code can be regarded as an $\mathbb{F}_q$-linear code, the opposite is not true. Hence, the R-DP is polynomial-time reducible to the MinRank problem; whether the other direction also holds true is still unknown.

3 Attacks on the R-DP

Recall Problem 1, i.e., given a generator matrix $\boldsymbol{G} \in \mathbb{F}_{q^m}^{k\times n}$ of the code $\mathcal{C} \subset \mathbb{F}_{q^m}^n$, a received vector $\boldsymbol{y} \in \mathbb{F}_{q^m}^n$ and a positive integer, called the *target weight* $r \leq \min\{n, m\}$, find $\boldsymbol{c} \in \mathcal{C}$ such that $d_{\mathrm{R}}(\boldsymbol{c}, \boldsymbol{y}) \leq r$.

For many cryptographic applications it is relevant to consider R-DP instances with *exactly one* solution, i.e., there exists a *unique* codeword $\boldsymbol{c}$ that satisfies the above conditions. Moreover, for instances with multiple solutions, existing decoders can directly be used to find at least one solution in the same or smaller average time complexity compared to instances with a unique solution. We will therefore also assume uniqueness unless stated otherwise. Uniqueness always holds if $r \leq \lfloor \frac{d_{\mathrm{R}}-1}{2} \rfloor$ with d_{R} the minimum rank distance of the code corresponding to $\boldsymbol{G}$. For r up to the rank-metric Gilbert-Varshamov bound, i.e., $r \leq d_{GV}^{\mathrm{R}}$, uniqueness also holds with high probability. Note that the uniqueness of $\boldsymbol{c}$ implies that there exists a unique error vector $\boldsymbol{e} \in \mathbb{F}_{q^m}^n$ of rank weight up to r, such that $\boldsymbol{y} = \boldsymbol{c} + \boldsymbol{e}$. For this unique $\boldsymbol{e}$, we denote its support by E (the results in Section 3 hold for both the row and column support, thus we do not need to specify one of the two types). Additionally, we assume without loss of generality the rank weight of the unique solution $\boldsymbol{e}$ to be exactly equal to r.

Attacks on the R-DP historically split into two categories; first, *combinatorial attacks* [8,15,19,24] with the strategy of guessing enough information about $\boldsymbol{e}$ so that the syndrome equations of the R-DP instance can be reduced to an easily solvable full-rank system of linear equations over $\mathbb{F}_q$. For an efficient decoder, these guesses should be well-adapted to error vectors of rank weight r:

the more rank-weight-r vectors satisfy these guesses, the fewer guesses we need to make, and the more efficient our algorithm. Researching what types of information require relatively few guesses to make and then iterating over these guesses is where combinatorics comes into play. Of course, these guesses should contain enough information to be usable to transform the syndrome equations, i.e., $\boldsymbol{e}\boldsymbol{H}^\top = \boldsymbol{s}$, for a parity-check matrix $\boldsymbol{H}$, into an easily solvable system. Existing combinatorial decoders involve guessing values of the error $\boldsymbol{e}$ or row/column support E [15,24] or guessing a *super-support* [8,19], i.e., a larger space containing E.

The second kind of attacks on the R-DP are *algebraic attacks* [18,21], wherein one tries to model the equality $\mathrm{wt_R}(\boldsymbol{e}) = r$ or $\dim(E) = r$ algebraically as a system of (polynomial) equations on $\boldsymbol{e}$ in combination with the syndrome equations. Existing algebraic decoders use linearized polynomials [19], or determinants of certain submatrices called minors [12,11,9]. Of our particular interest is the MaxMinors (MM-$\mathbb{F}_q$) modeling decoder [12]. When given an R-DP instance $(\boldsymbol{G}, \boldsymbol{y}, r)$ with parameters (q, m, n, k, r), the MM-$\mathbb{F}_q$ decoder constructs a certain system of equations (based on minors equaling 0) that consists of $m\binom{n-k-1}{r}$ linear equations in $\binom{n}{r}$ variables over $\mathbb{F}_q$. If the parameters (q, m, n, k, r) are such that the number of independent linear equations is greater or equal to the number of variables, a unique solution to the linear system is computed that directly leads to the error $\boldsymbol{e}$. In this case the R-DP is generally much faster solved than with purely combinatorial attacks. However, for many parameters the number of independent linear equations in the MM-$\mathbb{F}_q$ system is smaller than the number of variables, hence we have to resort to *hybrid attacks*. In such an attack, we first, like in combinatorial attacks, guess partial information about $\boldsymbol{e}$ to decrease the number of variables in an algebraic modeling, like MM-$\mathbb{F}_q$, which can subsequently be applied. In fact, by guessing sufficient information about $\boldsymbol{e}$, the number of variables becomes lower than the number of equations, so that a unique solution to the MM-$\mathbb{F}_q$ system can be computed.

3.1 Combinatorial decoders

We now give a brief overview of the best-known combinatorial decoders solving the R-DP. The main idea of the most recent combinatorial decoders in [8,19] (denoted by GRS1, GRS2, GRS3, AGHT in Table 2) is to search for a space called a *supersupport* S of suitable dimension containing the error support E.

For a chosen supersupport S, we use its basis over $\mathbb{F}_q$ to write $\boldsymbol{e}$ over $\mathbb{F}_q$ and then try to solve the resulting syndrome equations over $\mathbb{F}_q$. If the found solution does not have the correct rank weight, the supersupport S is discarded and the process is repeated for a different supersupport chosen at random. The cost of these algorithms is mainly given by the number of iterations required to find a supersupport S with $E \subseteq S$.

The time complexity formulas in Table 2 provide an easy way of comparing the various algorithms. We note that AGHT always outperforms GRS2 and GRS3, so we only need to consider the decoders CS/OJ2, OJ1, GRS1, GRS4 and AGHT. For a rough comparison, we omit the polynomial factors and define

$$M \coloneqq \frac{m}{n} \in \mathbb{R}_{>0}, \quad R \coloneqq \frac{k}{n} \in (0, 1).$$

For $r \leq n - k + 1$, and in particular if r is on or close to the rank-metric GV-bound

$$\frac{r}{n} \approx \frac{M+1}{2} - \sqrt{RM + \frac{(M-1)^2}{4}} \leq 1 - R$$

(as commonly chosen in many cryptographic schemes, e.g. in RYDE [5,4]), we find that the following

combinatorial decoders are generally the fastest in the given parameter regimes:

- AGHT for $M \leq 1$ and all R;
- OJ1 for $M > 1$ and $R \geq \frac{1}{2}$;
- GRS4 for $M > 1$ and $R < \frac{1}{2}$.

Authors	Abbreviation	Main idea	Average time complexity in $\mathbb{F}_q$-operations
Chabaud-Stern(1996) [15]	CS	Basis enumeration	$\mathcal{O}\left((nr+m)^{\omega} q^{(r-1)(m-r)}\right)$
Ourivski-Johansson (2002) [24]	OJ1*	Row supp. enumeration	$\mathcal{O}\left((mr)^{\omega} q^{(r-1)(k+\lceil \frac{k+1}{m-1} \rceil)}\right)$
	OJ2*	Column supp. enumeration	$\mathcal{O}\left((k+r)^{\omega} r^{\omega} q^{(r-1)(m-r)}\right)$
Gaborit-Ruatta-Schrek (2016) [19]	GRS1	Super(row)support	$\mathcal{O}\left((n-k)^{\omega} m^{\omega} q^{rk}\right)$
	GRS2	Super(column)support	$\mathcal{O}\left((n-k)^{\omega} m^{\omega} q^{r\lceil \frac{km}{n} \rceil}\right)$
	GRS3	Super(column)support	$\mathcal{O}\left((n-k)^{\omega} m^{\omega} q^{(r-1)\lceil \frac{(k+1)m}{n} \rceil}\right)$
	GRS4*	Annulator polynomial	$\mathcal{O}\left((mrk)^{\omega} q^{r\lceil \frac{(r+1)(k+1)-(n+1)}{r} \rceil}\right)$
Aragon-Gaborit-Hauteville-Tillich(2018) [8]	AGHT	Super(column)support	$\mathcal{O}\left((n-k)^{\omega} m^{\omega} q^{r\lceil \frac{(k+1)m}{n} \rceil - m}\right)$

Table 2. Overview of combinatorial generic rank-metric decoders with main idea behind the algorithm and time complexity. The number $2 \leq \omega \leq 3$ is the *matrix multiplication exponent*, also called *linear algebra constant*. For cryptographic analysis it is common to underestimate the time complexities by assuming $\omega = 2$. The algorithms marked with * make partial use of modeling the R-DP as systems of polynomial equations, i.e., they are also considered *algebraic* to some extent.

For completeness, we also note that AGHT and GRS4 could be slightly improved with *guessing-enhancement* [16]. When applied to AGHT, the resulting decoder is a combination of an exhaustive search of t values in $\mathbb{F}_q$ (with success probability q^{-t} for each guess) and AGHT as subroutine. As t values are guessed, an internal parameter concerning the supersupport can be adjusted slightly within the AGHT algorithm, resulting in a reduced complexity $\mathcal{O}\left(q^{r\lceil \frac{(k+1)m-t}{n} \rceil - m}\right)$ (up to polynomial factors). When optimized for the guessing parameter t, the guessing-enhanced AGHT decoder has average time complexity

$$\mathcal{O}\left(\min_{0 \leq t \leq mn} ((n-k)m+t)^{\omega} q^{t+r\lceil \frac{(k+1)m-t}{n} \rceil - m}\right),$$

with $t = 0$ resulting in plain AGHT. Note that a larger $t > 0$ only gives an improvement if the remainder $(k+1)m \pmod n$ is between 1 and $r-1$.

3.2 Algebraic attacks

The most recent algebraic attacks were developed in [10,12,9,11] and are celebrated as new benchmark in the cost of solving the R-DP and MinRank problem. For this paper, two of these algorithms are of main importance and are considered the fastest decoders for most parameters: the MaxMinors (MM-$\mathbb{F}_q$) modeling decoder [12] and the SupportMinors (SM-$\mathbb{F}_{q^m}^+$) modeling decoder [11].

MaxMinors modeling decoder The MM-$\mathbb{F}_q$ decoder in [12] uses the modeling idea of [10], where the authors reformulate the R-DP problem as a bilinear system of equations with coefficient variables and support variables. They observe that due to $\mathrm{wt}_{\mathrm{R}}(\boldsymbol{e}) = r$, an $r \times (n-k-1)$ matrix in the modeling consisting of coefficient variables is rank-deficient, and hence its maximal minors are all zero, leading to a new system of polynomial equations. With the help of the Cauchy-Binet formula and by setting certain maximal minors as new variables, this system is then transformed into a system of $m\binom{n-k-1}{r}$ linear equations in $\binom{n}{r}$ variables over $\mathbb{F}_q$. Inherent to solving this system is the requirement

$$m\binom{n-k-1}{r} \geq \binom{n}{r} - 1 \tag{3.1}$$

which we call the *overdetermined case*. Further, this system needs to be of near full rank, which was found to hold true with high probability after running many experiments [12,11].

Heuristic 1 *The system of $m\binom{n-k-1}{r}$ linear equations in the MaxMinors modeling of a random R-DP instance is with high probability of rank* $\min\{m\binom{n-k-1}{r}, \binom{n}{r} - 1\}$.

Assuming this heuristic, the MaxMinors modeling suffices to find a one-dimensional space of solutions in the new variables, from which the sought-after error $\boldsymbol{e}$ is easily obtained. The final cost of this MM-$\mathbb{F}_q$ decoder is mainly given by the cost of solving the associated system of linear equations, i.e.,

$$\mathcal{O}\left(m\binom{n-k-1}{r}\binom{n}{r}^{\omega-1}\right),$$

where again it is common to assume $\omega = 2$, taking into account that any structure of the linear system could be exploited.

In case the number of equations is much larger than the number of variables, called the *superdetermined case*, a puncturing argument can additionally be applied to reduce the parameters of the system, specifically replacing the length n by $n-p$. Using this technique, the new cost is then given by

$$\mathcal{O}\left(m\binom{n-p-k-1}{r}\binom{n-p}{r}^{\omega-1}\right) \tag{3.2}$$

where

$$p := \max\left\{i \in \{0, ..., n-k-1-r\} \mid m\binom{n-i-k-1}{r} \geq \binom{n-i}{r} - 1\right\}. \tag{3.3}$$

Hybrid technique In [12] and further explained in [11], the authors propose a workaround when the overdetermined condition is not satisfied, that is, we are in the so-called *underdetermined case* $m\binom{n-k-1}{r} < \binom{n}{r} - 1$. Their solution lies in guessing a entries of $\boldsymbol{e}$ to be zero for some $a \geq 1$ (within the modeling, this comes down to guessing ar many $\mathbb{F}_q$-variables to be zero, with success probability q^{-ar}), combined with a re-randomization method. Importantly, we note that this is equivalent to guessing a *supersupport* analogous to the GRS and AGHT decoders. Namely, if we denote by A the subspace in $\mathbb{F}_q^n$ consisting of all vectors with zeros at these a positions, we guess that after applying a random isometry U, the transformed error support $U(E)$ is in contained A, where $E = \mathrm{supp}_{\mathrm{row}}(\boldsymbol{e})$. This is equivalent to guessing that E is contained in the random superspace $U^{-1}(A)$.

As alternative perspective, guessing a super(row)support is equivalent to shortening the code, or dually, puncturing the dual code. To see this, suppose that $E = \mathrm{supp}_{\mathrm{row}}(\boldsymbol{e})$ is contained in a supersupport $A \subseteq \mathbb{F}_q^n$ of dimension $n-a$. Then the error vector can be decomposed as $\boldsymbol{e} = \boldsymbol{e}' A_{\mathrm{mat}}$, with $\boldsymbol{e}' \in \mathbb{F}_{q^m}^{n-a}$ of rank weight r and $A_{\mathrm{mat}} \in \mathbb{F}_q^{(n-a)\times a}$ a full rank matrix whose rows form a basis

of A. Now, the syndrome equation $\boldsymbol{e}\boldsymbol{H}^\top = \boldsymbol{s}$ implies that $\boldsymbol{e}'\boldsymbol{H}'^\top = \boldsymbol{s}$, where $\boldsymbol{H}' = \boldsymbol{H}A_{\text{mat}}^\top \in \mathbb{F}_{q^m}^{(n-k)\times(n-a)}$. Multiplying $\boldsymbol{H}$ by $A_{\text{mat}}^\top$ corresponds to the rank-metric equivalent of puncturing the dual code (see also [14, Def. 9]), and results in a shortened code of length $n-a$ and dimension $k-a$.

This hybrid technique generalizes to reducing the R-DP with instance parameters (q,m,n,k,r) to instance parameters $(q,m,n-a,k-a,r)$, with success probability q^{-ar}. We call this the *zero-reduction with parameter* a, denoted $\text{Red}_{\text{zero},\,a}$. Applied to a generic rank-metric decoder $\mathcal{A}$ gives the algorithm $\text{Red}_{\text{zero},\,a}(\mathcal{A})$ with description:

1. Set ar many $\mathbb{F}_q$-variables to zero, obtain reduced instance;
2. Try solving the reduced instance with decoder $\mathcal{A}$, obtain candidate solution $\hat{\boldsymbol{e}}$;
3. If no solution or if $\text{wt}_{\text{R}}(\hat{\boldsymbol{e}}) \neq r$, re-randoomize and go back to 1. Else, if $\text{wt}_{\text{R}}(\hat{\boldsymbol{e}}) = r$, then $\boldsymbol{e} = \hat{\boldsymbol{e}}$ and we are done.

Proposition 2. *[12,11] Given a random R-DP instance with unique solution and parameters (q,m,n,k,r), the average time complexity of a generic rank-metric decoder $\mathcal{A}$ with zero-reduction with parameter $a \geq 0$ solving the instance is given by*

$$TC_{\text{Red}_{\text{zero},\,a}(\mathcal{A})}(q,m,n,k,r) \;=\; q^{ar}\cdot TC_{\mathcal{A}}(q,m,n-a,k-a,r).$$

Here $TC_{\mathcal{A}}(q,m,n,k,r)$ denotes the average time complexity in $\mathbb{F}_q$-operations of a generic decoder $\mathcal{A}$ solving an R-DP instance with parameters (q,m,n,k,r) with *unique* solution. Naturally we can set a to minimize $TC_{\text{Red}_{\text{zero},\,a}(\mathcal{A})}(q,m,n,k,r)$.

Specifically when applied to the MM-$\mathbb{F}_q$ decoder, the zero-reduction can reduce an *underdetermined instance* to an *overdetermined instance*, which can then be solved with the plain MM-$\mathbb{F}_q$ decoder. The average time complexity is then usually minimized for a around

$$a_{min} := \min\left\{ i \geq 0 \;\mid\; m\binom{n-k-1}{r} \geq \binom{n-i}{r} - 1 \right\}. \tag{3.4}$$

$$\begin{aligned}
TC_{\text{Hybrid MM-}\mathbb{F}_q}(q,m,n,k,r) \;&=\; \min_{a\geq a_{min}} \; TC_{\text{Red}_{\text{zero},\,a}(\text{MM-}\mathbb{F}_q)}(q,m,n,k,r) \\
&=\; \min_{a\geq a_{min}} \; q^{ar}\cdot TC_{\text{MM-}\mathbb{F}_q}(q,m,n-a,k-a,r) \\
&= \mathcal{O}\left(\min_{a\geq a_{min}} \; m\, q^{ar}\binom{n-k-1}{r}\binom{n-a}{r}^{\omega-1} \right) \qquad (3.5)\\
&\approx \mathcal{O}\left(m\, q^{a_{min}r}\binom{n-k-1}{r}\binom{n-a_{min}}{r}^{\omega-1} \right). \qquad (3.6)
\end{aligned}$$

SupportMinors modeling decoder In [11], a similar approach to the MaxMinors decoder is taken but applied to a different matrix in the modeling of the R-DP. The resulting SupportMinors modeling decoder SM-$\mathbb{F}_{q^m}^+$ then solves, for a parameter $b \geq 1$, a system over $\mathbb{F}_q$ of $\mathcal{N}_b(m,n,k,r)$ linear independent equations in $\mathcal{M}_b(m,n,k,r)$ variables, with

$$\begin{aligned}
\mathcal{N}_b(m,n,k,r) \;:=\; &\sum_{i=1}^{k}\binom{n-i}{r}\binom{k+b-1-i}{b-1} - \binom{n-k-1}{r}\binom{k+b-1}{b} \\
&- (m-1)\sum_{i=1}^{b}(-1)^{i+1}\binom{k+b-i-1}{b-i}\binom{n-k-1}{r+i}
\end{aligned}$$

and

$$\mathcal{M}_b(m,n,k,r) := \binom{k+b-1}{b}\left(\binom{n}{r} - m\binom{n-k-1}{r}\right).$$

Like for MM-$\mathbb{F}_q$, we again require the overdetermined condition

$$\mathcal{N}_b(m,n,k,r) \geq \mathcal{M}_b(m,n,k,r) - 1$$

to solve the system. If this is not satisfied we apply zero-reduction with parameter $a \geq a_{min}(b) := \min\{i \geq 0 \mid \mathcal{N}_b(m,n-i,k-i,r) \geq \mathcal{M}_b(m,n-i,k-i,r)-1\}$ and optimize for a and b. This gives us the final cost $TC_{\text{Hybrid SM-}\mathbb{F}_{q^m}^+}(q,m,n,k,r)$ of the hybrid SM-$\mathbb{F}_{q^m}^+$ decoder:

$$\mathcal{O}\left(\min_{b\geq 1}\min_{a\geq a_{min}(b)} m^2\, \mathcal{N}_b(m,n-a,k-a,r)\, \mathcal{M}_b(m,n-a,k-a,r)^{\omega-1}\right) \tag{3.7}$$

where we can optimistically set $\omega = 2$.

4 Novel Hybrid Attack Technique on R-DP

Recall that for an R-DP instance $(\boldsymbol{G},\boldsymbol{y},r)$ with parameters (q,m,n,k,r) we are given a generator matrix $\boldsymbol{G} \in \mathbb{F}_{q^m}^{k\times n}$, a received vector $\boldsymbol{y} \in \mathbb{F}_{q^m}^n$ and a target rank weight r. For $\boldsymbol{y} \notin \mathcal{C}$ we assume that there exists a unique solution $\boldsymbol{c}$ such that $d_{\mathrm{R}}(\boldsymbol{y},\boldsymbol{c}) = r$; for $\boldsymbol{y} \in \mathcal{C}$ we assume a 1-dimensional space of solutions.

4.1 Subsupport-reduction: Idea

Consider for some parameter $h < r$ a random full-rank matrix $\boldsymbol{\Pi} \in \mathbb{F}_q^{n\times(n-h)}$ and its h-dimensional left kernel $S := \ker(\boldsymbol{\Pi}) \subset \mathbb{F}_q^n$.

If S is fully contained in $E := \mathrm{supp}_{\mathrm{row}}(\boldsymbol{e})$, i.e., S is a subsupport, there exists a representation $\boldsymbol{e} = \lambda_1\boldsymbol{v}_1 + \cdots + \lambda_r\boldsymbol{v}_r$ with non-zero scalars $\lambda_i \in \mathbb{F}_{q^m}$ and linearly independent vectors $\boldsymbol{v}_i \in \mathbb{F}_q^n$, such that $\{\boldsymbol{v}_1,\ldots,\boldsymbol{v}_h\} \subset S$, and $\boldsymbol{v}_i \notin S$ for all $i \in \{h+1,\ldots,r\}$. Thus, by multiplying $\boldsymbol{e}$ by $\boldsymbol{\Pi}$ we reduce the rank weight to $\mathrm{wt}_{\mathrm{R}}(\boldsymbol{e\Pi}) = r-h$, as $\boldsymbol{e\Pi} = \lambda_{h+1}\boldsymbol{v}_{h+1}\boldsymbol{\Pi} + \cdots + \lambda_r\boldsymbol{v}_r\boldsymbol{\Pi}$.

This gives us the following idea for a new decoding strategy: under the assumption that $S \subset E$, we apply the matrix $\boldsymbol{\Pi}$ to the R-DP instance by defining $\boldsymbol{G'} := \boldsymbol{G\Pi}$ and $\boldsymbol{y'} := \boldsymbol{y\Pi}$. Multiplying by $\boldsymbol{\Pi}$ on the right can be thought of as a rank-metric analog of puncturing (see also [14, Def. 9]), as $\boldsymbol{\Pi}$ is equal to the composition $\boldsymbol{\Pi} = \boldsymbol{M}\cdot\binom{\mathrm{Id}_{n-h}}{\mathbf{0}}$ for some invertible $\boldsymbol{M} \in \mathbb{F}_q^{n\times n}$ (i.e., a rank-metric isometry) and an $n\times(n-h)$ puncturing matrix $\binom{\mathrm{Id}_{n-h}}{\mathbf{0}}$. Note that this is dual to supersupport guessing, where the puncturing was applied to the parity-check matrix instead (see Section 3.2).

The vector $\boldsymbol{c'} = \boldsymbol{c\Pi}$ in the code $\mathcal{C}' \subset \mathbb{F}_{q^m}^{n-h}$ generated by $\boldsymbol{G'}$ lies at distance $r-h$ from $\boldsymbol{y'}$, since

$$d_{\mathrm{R}}(\boldsymbol{c'},\boldsymbol{y'}) = \mathrm{wt}_{\mathrm{R}}(\boldsymbol{y\Pi} - \boldsymbol{c\Pi}) = \mathrm{wt}_{\mathrm{R}}(\boldsymbol{e\Pi}) = r-h.$$

Thus, solving the reduced R-DP instance $(\boldsymbol{G'},\boldsymbol{y'},r')$ with $r' := r-h$ using the fastest known decoder gives us the 'punctured' solution $\boldsymbol{c\Pi}$, from which it is straightforward to recover $\boldsymbol{c}$ under common assumptions regarding the uniqueness of the solution.

In summary, the above strategy reduces an instance with parameters (q,m,n,k,r) to one with parameters $(q,m,n-h,k,r-h)$ if $S \subset E$. We call this reduction the *subsupport-reduction with parameter* h denoted by $\mathrm{Red}_{\mathrm{sub},h}$. Algorithmically, we will generate $\boldsymbol{\Pi}$ randomly and solve the reduced instance with the fastest known decoder $\mathcal{A}$ for these parameters. The solution to this

instance can then easily be used to recover $\boldsymbol{c}$, but only if the random h-dimensional subspace S is actually contained in E, which happens with success probability

$$\frac{\begin{bmatrix} r \\ h \end{bmatrix}_q}{\begin{bmatrix} n \\ h \end{bmatrix}_q} \approx q^{-h(n-r)}. \tag{4.1}$$

When S is not contained in E, the reduced instance will not produce a solution and we try again with a new randomly generated $\boldsymbol{\Pi}$. We note that the approximation in Eq. (4.1) is good for our parameter region of interest. In particular, for the parameters $(q, n, r, h) = (2, 57, 7, 5)$ that we will use later in Table 5, we have $\begin{bmatrix} r \\ h \end{bmatrix}_q / \begin{bmatrix} n \\ h \end{bmatrix}_q \approx 2^{-250.37} \approx 0.78 \cdot q^{-h(n-r)}$.

4.2 Algorithm description

We will now formally describe hybrid decoding algorithms that use subsupport-reduction. Let $\mathcal{A}$ be a generic rank-metric decoder. Applying subsupport-reduction with parameter h to $\mathcal{A}$ gives the algorithm $\mathrm{Red}_{\mathrm{sub},\,h}(\mathcal{A})$ with the following description:

1. Choose a random $\boldsymbol{\Pi} \in \mathbb{F}_q^{n\times(n-h)}$ of full rank $n-h$. Compute $\boldsymbol{G}' = \boldsymbol{G}\boldsymbol{\Pi}$ and $\boldsymbol{y}' = \boldsymbol{y}\boldsymbol{\Pi}$ and set $r' = r - h$;
2. Try solving the reduced instance $(\boldsymbol{G}', \boldsymbol{y}', r')$ with decoder $\mathcal{A}$, obtain candidate solution $\boldsymbol{c}'$;
3. If no solution or if $d_{\mathrm{R}}(\boldsymbol{c}', \boldsymbol{y}') \neq r'$, go back to 1. Else, if $d_{\mathrm{R}}(\boldsymbol{c}', \boldsymbol{y}') = r'$, then solve for $\boldsymbol{c} \in \mathcal{C}$ satisfying $\boldsymbol{c}\boldsymbol{\Pi} = \boldsymbol{c}'$ and we are done.

Algorithm 1 Hybrid decoder with subsupport-reduction $\mathrm{Red}_{\mathrm{sub},\,h}$ with $0 \leq h \leq r-1$

Input: Generic decoder $\mathcal{A}$, R-DP instance $(\boldsymbol{G}, \boldsymbol{y}, r)$.

Output: $\boldsymbol{c} \in \mathcal{C}$ with $d_{\mathrm{R}}(\boldsymbol{c}, \boldsymbol{y}) = r$.

1: $\boldsymbol{\Pi} \xleftarrow{\$} \mathbb{F}_q^{n\times(n-h)}$ until $\mathrm{rk}_{\mathbb{F}_q}(\boldsymbol{\Pi}) = n-h$ (happens w.h.p.)	// sample random full rank $(n \times (n-h))$-matrix over $\mathbb{F}_q$
2: $\boldsymbol{G}' \leftarrow \boldsymbol{G}\boldsymbol{\Pi}$	//multiply $\boldsymbol{\Pi}$ from the right to $\boldsymbol{G}$
3: $\boldsymbol{y}' \leftarrow \boldsymbol{y}\boldsymbol{\Pi}$	// multiply $\boldsymbol{\Pi}$ from the right to $\boldsymbol{y}$
4: $\boldsymbol{c}' \leftarrow \mathcal{A}(\boldsymbol{G}', \boldsymbol{y}', r' = r-h)$	// solve new reduced R-DP instance with $\mathcal{A}$
5: If no $\boldsymbol{c}'$ or if $d_{\mathrm{R}}(\boldsymbol{c}', \boldsymbol{y}') \neq r-h$, go back to Step 1.	
6: Else, return $\boldsymbol{c}$ such that $\boldsymbol{c}\boldsymbol{\Pi} = \boldsymbol{c}'$.	// solution $\boldsymbol{c}$ is found

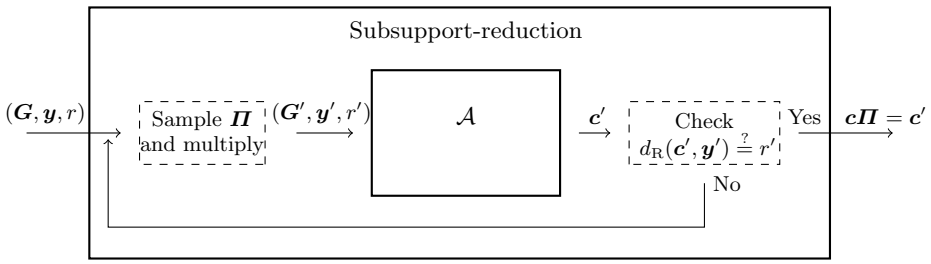


Fig. 1. Diagrammatic representation of hybrid decoder with subsupport-reduction

4.3 Completeness and heuristics

Let us now show rigorously why this reduction works. We start with investigating the effect of multiplying a vector of certain rank weight with $\boldsymbol{\Pi}$, and first prove for completeness the following lemma.

Lemma 1. *Let $\boldsymbol{x} \in \mathbb{F}_{q^m}^n$ and $\boldsymbol{\Pi} \in \mathbb{F}_q^{n\times(n-h)}$ of full rank. Then*

$$\mathrm{wt_R}(\boldsymbol{x}\boldsymbol{\Pi}) = \mathrm{wt_R}(\boldsymbol{x}) - \dim(\ker(\boldsymbol{\Pi}) \cap \mathrm{supp_{row}}(\boldsymbol{x})) \geq \mathrm{wt_R}(\boldsymbol{x}) - h.$$

Proof. First, $\boldsymbol{\Pi}$ naturally represents a surjective linear map $\mathbb{F}_q^n \to \mathbb{F}_q^{n-h}$ whose kernel coincides with the left kernel of the matrix $\boldsymbol{\Pi}$. Let $V = \mathrm{supp_{row}}(\boldsymbol{x}) \subset \mathbb{F}_q^n$ (for some fixed $\mathbb{F}_q$-basis Γ of $\mathbb{F}_{q^m}$), and consider the restricted map $\boldsymbol{\Pi}|_V : V \to \mathbb{F}_q^{n-h}$. Then by the rank-nullity theorem we have

$$\begin{aligned}\mathrm{wt_R}(\boldsymbol{x}\boldsymbol{\Pi}) &= \dim(\mathrm{supp_{row}}(\boldsymbol{x}\boldsymbol{\Pi})) = \dim(\mathrm{im}(\boldsymbol{\Pi}|_V)) \\ &= \dim(V) - \dim(\ker(\boldsymbol{\Pi}|_V)) \\ &= \mathrm{wt_R}(\boldsymbol{x}) - \dim(\ker(\boldsymbol{\Pi}) \cap V) \\ &\geq \mathrm{wt_R}(\boldsymbol{x}) - \dim(\ker(\boldsymbol{\Pi})) = \mathrm{wt_R}(\boldsymbol{x}) - h.\end{aligned}$$

□

This implies that the new instance $\boldsymbol{G}' = \boldsymbol{G}\boldsymbol{\Pi}$ will still have full rank k if h is chosen appropriately.

Corollary 1. *Let $\boldsymbol{G} \in \mathbb{F}_{q^m}^{k\times n}$ be a (rank k) generator matrix of a code $\mathcal{C}$. If $\boldsymbol{\Pi} \in \mathbb{F}_q^{n\times(n-h)}$ is a full rank matrix with $h < d_{\mathrm{R}}(\mathcal{C})$, the minimum rank distance of $\mathcal{C}$, then $\boldsymbol{G}\boldsymbol{\Pi} \in \mathbb{F}_{q^m}^{k\times(n-h)}$ has full rank k.*

Proof. If $\boldsymbol{c} \in \mathcal{C}$ is a non-zero codeword, then by Lemma 1 we have $\mathrm{wt_R}(\boldsymbol{c}\boldsymbol{\Pi}) \geq \mathrm{wt_R}(\boldsymbol{c}) - h \geq d_{\mathrm{R}}(\mathcal{C}) - h > 0$ and so $\boldsymbol{c}\boldsymbol{\Pi} \neq 0$. This means that $\dim_{\mathbb{F}_{q^m}}(\mathcal{C}\boldsymbol{\Pi}) = \dim_{\mathbb{F}_{q^m}}(\mathcal{C}) = k$. □

Now we have the tools to show completeness of decoders with subsupport-reduction. Concretely, we show that solving the reduced instance $(\boldsymbol{G}', \boldsymbol{y}', r')$ leads to the correct solution of the original instance $(\boldsymbol{G}, \boldsymbol{y}, r)$ when the left kernel of the randomly sampled $\boldsymbol{\Pi}$ is contained in the unknown error support.

Theorem 1. *Let $(\boldsymbol{G}, \boldsymbol{y}, r)$ be an R-DP instance (assuming $\boldsymbol{y} \notin \mathcal{C}$) with unique solution $\boldsymbol{c} \in \mathcal{C}$, i.e., $d_{\mathrm{R}}(\boldsymbol{c}, \boldsymbol{y}) = r \leq d_{\mathrm{R}}(\mathcal{C})$ and $d_{\mathrm{R}}(\boldsymbol{x}, \boldsymbol{y}) > r$ for all other codewords $\boldsymbol{x} \in \mathcal{C} \setminus \{\boldsymbol{c}\}$. Let $\boldsymbol{\Pi} \in \mathbb{F}_q^{n\times(n-h)}$ be a full rank matrix with $h < r$ and denote $\boldsymbol{e} := \boldsymbol{y} - \boldsymbol{c}$. Suppose that $\ker(\boldsymbol{\Pi}) \subset \mathrm{supp_{row}}(\boldsymbol{e})$. Then $\boldsymbol{c}' := \boldsymbol{c}\boldsymbol{\Pi}$ is the unique solution to the R-DP instance $(\boldsymbol{G}', \boldsymbol{y}', r')$, where $\boldsymbol{G}' := \boldsymbol{G}\boldsymbol{\Pi}$, $\boldsymbol{y}' := \boldsymbol{y}\boldsymbol{\Pi}$, and $r' := r - h$.*

Proof. First we show that $\boldsymbol{c}'$ is a solution. We have $\boldsymbol{y}' - \boldsymbol{c}' = (\boldsymbol{y} - \boldsymbol{c})\boldsymbol{\Pi} = \boldsymbol{e}\boldsymbol{\Pi}$, and using Lemma 1 we obtain

$$d_{\mathrm{R}}(\boldsymbol{c}', \boldsymbol{y}') = \mathrm{wt_R}(\boldsymbol{y}' - \boldsymbol{c}') = \mathrm{wt_R}(\boldsymbol{e}\boldsymbol{\Pi}) = \mathrm{wt_R}(\boldsymbol{e}) - \dim(\ker(\boldsymbol{\Pi})) = r - h = r'$$

as desired.

Secondly, to show uniqueness, let $\hat{\boldsymbol{c}}'$ be any solution to the R-DP instance $(\boldsymbol{G}', \boldsymbol{y}', r')$. By Corollary 1, as $\hat{\boldsymbol{c}}'$ is in the code $\mathcal{C}'$ generated by $\boldsymbol{G}\boldsymbol{\Pi}$, there is a unique $\hat{\boldsymbol{c}} \in \mathcal{C}$ such that $\hat{\boldsymbol{c}}\boldsymbol{\Pi} = \hat{\boldsymbol{c}}'$. Again using Lemma 1 we get

$$d_{\mathrm{R}}(\hat{\boldsymbol{c}}, \boldsymbol{y}) = \mathrm{wt_R}(\boldsymbol{y} - \hat{\boldsymbol{c}}) \leq \mathrm{wt_R}((\boldsymbol{y} - \hat{\boldsymbol{c}})\boldsymbol{\Pi}) + h = \mathrm{wt_R}(\boldsymbol{y}' - \hat{\boldsymbol{c}}') + h \leq (r - h) + h = r,$$

so by uniqueness $\hat{\boldsymbol{c}} = \boldsymbol{c}$ and $\hat{\boldsymbol{c}}' = \boldsymbol{c}'$. □

We will see in Section 4.4 that choosing the hybrid MM-$\mathbb{F}_q$ decoder for $\mathcal{A}$ in Algorithm 1 leads to the fastest attack for several RYDE parameters. Since the MM-$\mathbb{F}_q$ decoder relies on Heuristic 1, we need to justify that this heuristic also holds on reduced instances $(\boldsymbol{G}', \boldsymbol{y}', r')$ used in Algorithm 1.

For this, let us denote by $U(\cdot)$ the uniform distribution on a space of matrices. If the original instance $\boldsymbol{G}$ was chosen uniformly at random, then $\boldsymbol{G}'$ is also uniformly distributed.

Proposition 3. *Let $\boldsymbol{G} \sim U(\mathbb{F}_{q^m}^{k\times n})$ and $\boldsymbol{\Pi} \in \mathbb{F}_q^{n\times(n-h)}$ a full rank matrix. Then $\boldsymbol{G\Pi} \sim U(\mathbb{F}_{q^m}^{k\times(n-h)})$.*

Proof. Note that $\boldsymbol{\Pi} = \boldsymbol{M} \cdot \binom{\mathrm{Id}_{n-h}}{\mathbf{0}}$ for some invertible $\boldsymbol{M} \in \mathbb{F}_q^{n\times n}$ and $\binom{\mathrm{Id}_{n-h}}{\mathbf{0}}$ an $n \times (n-h)$ puncturing matrix. Naturally $\boldsymbol{GM} \sim U(\mathbb{F}_{q^m}^{k\times n})$ as $\boldsymbol{M}$ acts bijectively on $\mathbb{F}_{q^m}^{k\times n}$ (it is even a rank-metric isometry), and since puncturing the last h coordinates does not affect the uniform randomness of the remaining $n-h$ columns, we have proven the claim. □

When $\boldsymbol{G}$ is uniformly at random sampled from $\mathbb{F}_{q^m}^{k\times n}$, it is with high probability a full rank generator matrix of a code $\mathcal{C}$ whose minimum rank distance $d_{\mathrm{R}}(\mathcal{C})$ attains the rank Gilbert-Varshamov bound d_{GV}^{R} (for dimension k and length n codes). As corollary of Proposition 3, when $h < d_{GV}^{\mathrm{R}}$, $\boldsymbol{G\Pi} \in \mathbb{F}_{q^m}^{k\times(n-h)}$ is also with high probability a full rank generator matrix of a code $\mathcal{C}'$ whose minimum rank distance $d_{\mathrm{R}}(\mathcal{C}')$ attains the rank Gilbert-Varshamov bound d_{GV}^{R} (for dimension k and length $n-h$ codes).

Combining subsupport-reduction with parameter h and zero-reduction with parameter a reduces an instance with parameters (q, m, n, k, r) to an instance with parameters $(q, m, n-h-a, k-a, r-h)$. The reasoning above, together with the heuristics used in [12,11] implies that the MaxMinors modeling of the newly reduced instance is with high probability of maximum rank.

Heuristic 2 *The system of $m\binom{n-h-k-1}{r-h}$ linear equations in the MaxMinors modeling of a random R-DP instance after the novel subsupport-reduction is with high probability of rank* $\min\{m\binom{n-h-k-1}{r-h}, \binom{n-h-a}{r-h} - 1\}$.

We validated this heuristic by performing similar experiments as [12,11]. We adapted the code from https://github.com/mbardet/Rank-Decoding-tools that check Heuristic 1 to include tests on Heuristic 2 for the subsupport reduced instances. The code can be found at https://github.com/HBeelooSauerbierCouvee/generic-rank-decoding.

4.4 Time complexity and application to the hybrid MM-$\mathbb{F}_q$ decoder

Recall that $TC_{\mathcal{A}}(q, m, n, k, r)$ denotes the average time complexity in $\mathbb{F}_q$-operations of a generic decoder $\mathcal{A}$ solving an R-DP instance with parameters (q, m, n, k, r) with *unique* solution.

Proposition 4. *Given a random R-DP instance with unique solution and parameters (q, m, n, k, r), the average time complexity of a generic rank-metric decoder $\mathcal{A}$ with subsupport-reduction with parameter $0 \le h \le r-1$ solving the instance is given by*

$$TC_{\mathrm{Red}_{\mathrm{sub},\,h}(\mathcal{A})}(q, m, n, k, r) \;= \mathcal{O}\left(q^{h(n-r)} \cdot TC_{\mathcal{A}}(q, m, n-h, k, r-h)\right).$$

Proof. As $\ker(\boldsymbol{\Pi}) \subset \mathrm{supp}_{\mathrm{row}}(\boldsymbol{e})$ is satisfied with probability $\approx q^{h(r-n)}$ by equation (4.1), we need to loop back from Step 5 to Step 1 on average approximately $q^{h(n-r)}$ times. For each of these iterations, we need to sample $\boldsymbol{\Pi}$ (Step 1), compute the new instance $(\boldsymbol{G\Pi}, \boldsymbol{y\Pi}, r-h)$ (Steps 2-3), and decode with $\mathcal{A}$ with average time complexity $TC_{\mathcal{A}}(q, m, n-h, k, r-h)$ (Steps 4-5). While Steps 1-3 cost naively around $mkn(n-h)$ $\mathbb{F}_q$-operations to perform matrix multiplication, we can do a lot faster on average by realizing that for every iteration, we are using the same $\boldsymbol{G}$ and $\boldsymbol{y}$, and use a new random $\boldsymbol{\Pi}$. Since multiplication of $\boldsymbol{G}$ by any column of $\boldsymbol{\Pi}$ is equivalent to

taking a random $\mathbb{F}_q$-linear combination of columns of $\boldsymbol{G}$, we can benefit by (pre)computing and storing linear combinations of columns of $\boldsymbol{G}$ in lookup tables, or using a similar method. Then each subsequent iteration can use previously calculated linear combinations to greatly reduce the cost to compute $\boldsymbol{G\Pi}$ (similarly for $\boldsymbol{y\Pi}$). If the average time complexity $TC_{\mathcal{A}}(q, m, n-h, k, r-h)$ dominates the cost $mkn(n-h)$ of matrix multiplication, there is no need for any precomputations. On the other hand, if $TC_{\mathcal{A}}(q, m, n-h, k, r-h)$ is smaller than the cost $mkn(n-h)$, this implies for cryptographically significant parameters that h is large. Since for large h the number of iterations $q^{h(n-r)}$ dominates the number q^n of $\mathbb{F}_q$-linear combinations, the cost of doing precomputations will be negligible compared to the other costs of the attack. In this case, for each iteration in the algorithm, we replace the expensive matrix multiplication by performing random sampling of the precomputed linear combinations. Lastly, for every iteration, $\boldsymbol{G\Pi}$ has with high probability full $\mathbb{F}_q$-rank $(n-h)$, so this does not meaningfully affect the overall complexity. □

Specifically, when we want to use the MM-$\mathbb{F}_q$ decoder for $\mathcal{A}$, the subsupport-reduction in combination with zero-reduction can reduce an *underdetermined instance* to an *overdetermined instance*, which can then be solved with the plain MM-$\mathbb{F}_q$ decoder. For this we introduce the set $A(m, n, k, r)$ of admissible pairs of parameters (a, h) where, after combined subsupport- and zero-reduction, the reduced instance is overdetermined, defined as

$$\left\{(a,h) \in \{0,\dots,k\} \times \{0,\dots,r-1\} \mid m\binom{n-h-k-1}{r-h} \geq \binom{n-h-a}{r-h} - 1\right\}. \tag{4.2}$$

Note that this set is always non-empty: pairs of the form $(a, r-1)$ are admissible for any a satisfying $m(n-r-k) \geq n-r-a$.

Now we have our new hybrid MM-$\mathbb{F}_q$ decoder $\mathrm{Red}_{\mathrm{sub},\,h}(\mathrm{Red}_{\mathrm{zero},\,a}(\text{MM-}\mathbb{F}_q))$, where we can naturally minimize the time complexity over all admissible pairs $(a, h) \in A(m, n, k, r)$.

$$\begin{aligned}
&TC_{\text{New hybrid MM-}\mathbb{F}_q}(q, m, n, k, r) \\
&\quad = \min_{(a,h)\in A(m,n,k,r)} TC_{\mathrm{Red}_{\mathrm{sub},\,h}(\mathrm{Red}_{\mathrm{zero},\,a}(\text{MM-}\mathbb{F}_q))}(q, m, n, k, r) \\
&\quad = \min_{(a,h)\in A(m,n,k,r)} q^{h(n-r)+a(r-h)} \cdot TC_{\text{MM-}\mathbb{F}_q}(q, m, n-h-a, k-a, r-h) \\
&\quad = \mathcal{O}\left(\min_{(a,h)\in A(m,n,k,r)} m\, q^{h(n-r)+a(r-h)} \binom{n-h-k-1}{r-h}\binom{n-h-a}{r-h}^{\omega-1}\right)
\end{aligned} \tag{4.3}$$

where $2 \leq \omega \leq 3$ is the matrix multiplication exponent.

We first observe that choosing h non-zero brings with it the additional factor $q^{h(n-r)}$ from the inverse success probability. This implies a total *reduction cost* of $q^{h(n-r)+a(r-h)}$. On the other hand, the cost factor $\binom{n-h-k-1}{r-h}\binom{n-h-a}{r-h}^{\omega-1}$ of solving the new reduced instance with MM-$\mathbb{F}_q$ generally decreases with larger h.

Let us now have a look at the impact of the new decoder for some proposed systems. We will mainly investigate the RYDE signature scheme [4], which is a second round candidate in the additional NIST standardization process. In the following tables we display the RYDE parameters (field size q, extension degree m, code length n, code dimension k, rank weight r) for Round 2, V2.1.0, NIST security categories $1, 3, 5$ and V2.0.0, NIST security category 5.

We compare the estimated **cost** in bits (i.e., the logarithm of average time complexities $\log_2(TC_{\mathcal{A}}(q, m, n, k, r))$) of the algorithms

- OJ1
- AGHT

- Hybrid MM-$\mathbb{F}_q$ with only zero-reduction ($h = 0$ and optimal parameters a and p, see Eq. (3.5),(3.2))
- Hybrid MM-$\mathbb{F}_q$ with additionally subsupport reduction (various $h > 0$ and optimal parameters a and p, see Eq. (4.3)),

where we assume the linear algebra constant to be $\omega = 2$. The lowest cost among these algorithms will be marked in bold. The estimated cost is split into two parts: **reduction cost**, i.e., the logarithm of the inverse success probability $\log_2\left(q^{h(n-r)+a(r-h)}\right)$ and the estimated **cost of solving the reduced instance** in bits. We use the previously established notation $n' = n-h-a$, $k' = k-a$, $r' = r-h$. Note that when $m = n$, we get no improvement from adding guessing enhancement to AGHT (so $t = 0$), and generally, puncturing with parameter p (Eq.(3.3)) does not yield an improvement.

Additionally, in order to confirm the claimed complexity in Proposition 4, we suppose a (suboptimal) precomputation method of dividing the columns of $\boldsymbol{G}$ into two halves, generating and storing a list of all linear combinations of each halve $\{\boldsymbol{G}(\boldsymbol{x}_1 \mid 0)^\top : \boldsymbol{x}_1 \in \mathbb{F}_q^{n/2}\}$ and $\{\boldsymbol{G}(0 \mid \boldsymbol{x}_2)^\top : \boldsymbol{x}_2 \in \mathbb{F}_q^{n/2}\}$, with approximate time complexity of $TC_{\text{precomp}} = \mathcal{O}(2mkn\,q^{n/2})$ $\mathbb{F}_q$-operations. Then each iteration only requires $n-h$ times the addition of a random vector from each list. We verify that for the RYDE parameters, this cost is negligible compared to that in Eq. (4.3). Hence $TC_{\text{precomp}} \ll TC_{\text{New hybrid MM-}\mathbb{F}_q}(q,m,n,k,r)$ and thus $\log_2\left(TC_{\text{precomp}} + TC_{\text{New hybrid MM-}\mathbb{F}_q}(q,m,n,k,r)\right) \approx \log_2\left(TC_{\text{New hybrid MM-}\mathbb{F}_q}(q,m,n,k,r)\right)$.

We used Sage to compute the costs of the decoders in the tables below. The code is available here https://github.com/HBeelooSauerbierCouvee/generic-rank-decoding. The computations are verified with CryptographicEstimators v2.8.4 [17] at https://estimators.crypto.tii.ae/configuration?id=RankSDEstimator.

RYDE-1, Version 2.1.0
$(q,m,n,k,r) = (2,53,53,45,4)$

R-DP Algorithm	Cost ($\omega=2$)	Reduction params. h	a	p	Reduction cost	Reduced instance params. (q,m,n',k',r') and cost	
OJ1	153	-	-	-	-	-	-
AGHT ($t=0$)	**148**	-	-	-	-	-	-
MM-$\mathbb{F}_q$ ($h=0$)	169	0	37	0	148	$(2,53,16,8,4)$	21
MM-$\mathbb{F}_q$ ($h=1$)	167	1	33	0	148	$(2,53,19,12,3)$	19
MM-$\mathbb{F}_q$ ($h=2$)	152	2	18	0	134	$(2,53,33,27,2)$	18

Table 3. RYDE Round 2 parameters for NIST security category 1.
Cost (in bits) of precomputing linear combinations: $\log_2(2mkn\,q^{n/2}) \approx 45 \ll 152$.

RYDE-3, Version 2.1.0
$(q,m,n,k,r) = (2,61,61,51,5)$

R-DP Algorithm	Cost ($\omega=2$)	Reduction params. h	a	p	Reduction cost	Reduced instance params. (q,m,n',k',r') and cost	
OJ1	224	-	-	-	-	-	-
AGHT ($t=0$)	**217**	-	-	-	-	-	-
MM-$\mathbb{F}_q$ ($h=0$)	245	0	44	0	220	$(2,61,17,7,5)$	25
MM-$\mathbb{F}_q$ ($h=1$)	243	1	41	0	220	$(2,61,19,10,4)$	23
MM-$\mathbb{F}_q$ ($h=2$)	239	2	35	0	217	$(2,61,24,16,3)$	22
MM-$\mathbb{F}_q$ ($h=3$)	**217**	3	15	0	198	$(2,61,43,36,2)$	19

Table 4. RYDE Round 2 parameters for NIST security category 3.
Cost (in bits) of precomputing linear combinations: $\log_2(2mkn\,q^{n/2}) \approx 49 \ll 217$.

RYDE-5, Version 2.1.0
$(q, m, n, k, r) = (2, 61, 57, 44, 7)$

R-DP Algorithm	Cost ($\omega = 2$)	Reduction params. h	a	p	Reduction cost	Reduced instance params. (q, m, n', k', r') and cost	
OJ1	287	-	-	-	-	-	-
AGHT ($t = 0$)	301	-	-	-	-	-	-
MM-$\mathbb{F}_q$ ($h = 0$)	301	0	39	1	273	$(2, 61, 18, 5, 7)$	28
MM-$\mathbb{F}_q$ ($h = 1$)	301	1	37	0	272	$(2, 61, 19, 7, 6)$	29
MM-$\mathbb{F}_q$ ($h = 4$)	298	4	25	0	275	$(2, 61, 28, 19, 3)$	23
MM-$\mathbb{F}_q$ ($h = 5$)	**272**	5	1	0	252	$(2, 61, 51, 43, 2)$	20

Table 5. RYDE Round 2 parameters for NIST security category 5.
Cost (in bits) of precomputing linear combinations: $\log_2(2mkn\, q^{n/2}) \approx 47 \ll 272$.

RYDE-5, Version **2.0.0**
$(q, m, n, k, r) = (2, 67, 67, 55, 6)$

R-DP Algorithm	Cost ($\omega = 2$)	Reduction params. h	a	p	Reduction cost	Reduced instance params. (q, m, n', k', r') and cost	
OJ1	297	-	-	-	-	-	-
AGHT ($t = 0$)	288	-	-	-	-	-	-
MM-$\mathbb{F}_q$ ($h = 0$)	317	0	48	0	288	$(2, 67, 19, 7, 6)$	29
MM-$\mathbb{F}_q$ ($h = 3$)	311	3	35	0	288	$(2, 67, 29, 20, 3)$	23
MM-$\mathbb{F}_q$ ($h = 4$)	**284**	4	10	0	264	$(2, 67, 53, 45, 2)$	20

Table 6. RYDE Round 2 parameters for Version 2.0.0 and NIST security category 5.
Cost (in bits) of precomputing linear combinations: $\log_2(2mkn\, q^{n/2}) \approx 53 \ll 284$.

We observe from the tables above that for all RYDE parameters, the subsupport guessing with $h > 0$ decreases the cost for the hybrid MM-$\mathbb{F}_q$ decoder, and provides the new best attack for RYDE-5, Version 2.1.0.

Assuming the linear algebra constant to be $\omega = 2$ is the most optimistic case when the structure of the involved matrices can be exploited for multiplication; on the other hand, most practical matrix multiplication algorithms have an exponent not far below $\omega = 3$, such as the well-known Strassen algorithm with $\omega = \log_2(7) \approx 2.8$. To get a more realistic cost estimate, we also present the costs of our hybrid attack (with the best choice of h) for this choice of ω.

We note that for both $\omega = 2$ and $\omega = \log_2(7)$, the optimal reduction parameters h, a, p are the same, and that differences in costs between the different decoders are comparable. This points at a strong qualitative similarity in the cost behavior among different values of ω.

RYDE-1, Version 2.1.0
$(q, m, n, k, r) = (2, 53, 53, 45, 4)$

R-DP Algorithm	Cost ($\omega \approx 2.8$)	Reduction params. h	a	p	Reduction cost	Reduced instance params. (q, m, n', k', r') and cost	
OJ1	159	-	-	-	-	-	-
AGHT ($t = 0$)	**155**	-	-	-	-	-	-
MM-$\mathbb{F}_q$ ($h = 0$)	178	0	37	0	148	$(2, 53, 16, 8, 4)$	30
MM-$\mathbb{F}_q$ ($h = 2$)	159	2	18	0	134	$(2, 53, 33, 27, 2)$	25

Table 7. RYDE Round 2 parameters for NIST security category 1 with $\omega = \log_2(7)$.

RYDE-3, Version 2.1.0
$(q, m, n, k, r) = (2, 61, 61, 51, 5)$

R-DP Algorithm	Cost ($\omega \approx 2.8$)	Reduction params. h	a	p	Reduction cost	Reduced instance params. (q, m, n', k', r') and cost	
OJ1	231	-	-	-	-	-	-
AGHT ($t = 0$)	**224**	-	-	-	-	-	-
MM-$\mathbb{F}_q$ ($h = 0$)	255	0	44	0	220	$(2, 61, 17, 7, 5)$	35
MM-$\mathbb{F}_q$ ($h = 3$)	225	3	15	0	198	$(2, 61, 43, 36, 2)$	27

Table 8. RYDE Round 2 parameters for NIST security category 3 with $\omega = \log_2(7)$.

RYDE-5, Version 2.1.0
$(q, m, n, k, r) = (2, 61, 57, 44, 7)$

R-DP Algorithm	Cost ($\omega \approx 2.8$)	Reduction params. h	a	p	Reduction cost	Reduced instance params. (q, m, n', k', r') and cost	
OJ1	294	-	-	-	-	-	-
AGHT ($t = 0$)	309	-	-	-	-	-	-
MM-$\mathbb{F}_q$ ($h = 0$)	313	0	39	1	273	$(2, 61, 18, 5, 7)$	40
MM-$\mathbb{F}_q$ ($h = 5$)	**280**	5	1	0	252	$(2, 61, 51, 43, 2)$	28

Table 9. RYDE Round 2 parameters for NIST security category 5 with $\omega = \log_2(7)$.

RYDE-5, Version **2.0.0**
$(q, m, n, k, r) = (2, 67, 67, 55, 6)$

R-DP Algorithm	Cost ($\omega \approx 2.8$)	Reduction params. h	a	p	Reduction cost	Reduced instance params. (q, m, n', k', r') and cost	
OJ1	304	-	-	-	-	-	-
AGHT ($t = 0$)	296	-	-	-	-	-	-
MM-$\mathbb{F}_q$ ($h = 0$)	329	0	48	0	288	$(2, 67, 19, 7, 6)$	41
MM-$\mathbb{F}_q$ ($h = 4$)	**293**	4	10	0	264	$(2, 67, 53, 45, 2)$	29

Table 10. RYDE Round 2 parameters for Version 2.0.0 and NIST security category 5 with $\omega = \log_2(7)$.

As a concluding remark, we stress that subsupport reduction with $h > 0$ is only preferable for attacking the RYDE parameters in combination with the hybrid MM-$\mathbb{F}_q$ decoder, as using the hybrid SM-$\mathbb{F}_{q^m}^+$ decoder will result in a larger cost compared to $h = 0$, according to our calculations.

For more general parameter sets lying on the rank GV-bound, subsupport reduction with $h > 0$ seems empirically only beneficial for high rate (large k, small r) regimes. Hence for schemes that use codes with rate $\frac{1}{2}$ such as LowMS, subsupport reduction cannot be directly applied in the same manner as with RYDE to obtain a lower attack cost.

4.5 Generalization to Support-Intersecting Spaces

We can generalize this approach to guessing a support-intersecting space S of the unknown error support, that is a subspace of dimension $h + \varepsilon$ which only intersects $E = \mathrm{supp}_{\mathrm{row}}(\boldsymbol{e})$ in dimension h.

More precisely, we choose the parameters h, ε, such that $0 \leq h < r$ and $0 \leq \varepsilon < n - h - k$. We select a matrix $\boldsymbol{\Pi} \in \mathbb{F}_q^{n \times (n-h-\varepsilon)}$ of full rank and define its left kernel to be $S = \ker(\boldsymbol{\Pi}) \subseteq \mathbb{F}_q^n$.

We again multiply our instance $\boldsymbol{G}, \boldsymbol{y}$ with $\boldsymbol{\Pi}$, to get $\boldsymbol{G}' = \boldsymbol{G}\boldsymbol{\Pi}$ and $\boldsymbol{y}' = \boldsymbol{y}\boldsymbol{\Pi}$ and we still have the property that $\boldsymbol{c}\boldsymbol{\Pi} \in \langle \boldsymbol{G}' \rangle$ is at distance $r - h$ from $\boldsymbol{y}'$. However, we now reduced the parameters from (q, m, n, k, r) to $(q, m, n - h - \varepsilon, k, r - h)$ and denote this *subsupport-reduction with parameters* h, ε by $\mathrm{Red}_{\mathrm{sub},\, h,\varepsilon}$.

We may now solve the reduced instance with the fastest known decoder $\mathcal{A}$ for these parameters. The solution to this instance can then easily be used to recover $\boldsymbol{c}$, but only if the random $(h+\varepsilon)$-dimensional subspace S satisfies $\dim(S \cap E) = h$, which happens with success probability (see [25, Lem. 2.1])

$$\frac{q^{\varepsilon(r-h)} \begin{bmatrix} n-r \\ \varepsilon \end{bmatrix}_q \begin{bmatrix} r \\ h \end{bmatrix}_q}{\begin{bmatrix} n \\ h+\varepsilon \end{bmatrix}_q} \approx q^{-h(n-r-\varepsilon)}.$$

When $\dim(S \cap E) \neq h$, the reduced instance will not produce a solution and we try again with a new randomly generated $\boldsymbol{\Pi}$.

The only difference to the previous results, are the following:

Lemma 2. *Let $\boldsymbol{x} \in \mathbb{F}_{q^m}^n$ and $\boldsymbol{\Pi} \in \mathbb{F}_q^{n \times (n-h-\varepsilon)}$ of full rank. Then*

$$\mathrm{wt_R}(\boldsymbol{x}\boldsymbol{\Pi}) = \mathrm{wt_R}(\boldsymbol{x}) - \dim(\ker(\boldsymbol{\Pi}) \cap \mathrm{supp_{row}}(\boldsymbol{x})) \geq \mathrm{wt_R}(\boldsymbol{x}) - h - \varepsilon.$$

As the proof is similar to Lemma 1, we omit it.

This implies that the new instance $\boldsymbol{G}' = \boldsymbol{G}\boldsymbol{\Pi}$ will still have full rank k if h and ε are chosen appropriately.

Corollary 2. *Let $\boldsymbol{G} \in \mathbb{F}_{q^m}^{k \times n}$ be a (rank k) generator matrix of a code $\mathcal{C}$. If $\boldsymbol{\Pi} \in \mathbb{F}_q^{n \times (n-h-\varepsilon)}$ is a full rank matrix with $h + \varepsilon < d_{\mathrm{R}}(\mathcal{C})$, the minimum rank distance of $\mathcal{C}$, then $\boldsymbol{G}\boldsymbol{\Pi} \in \mathbb{F}_{q^m}^{k \times (n-h-\varepsilon)}$ has full rank k.*

When $h + \varepsilon$ is larger and lies between $d_{\mathrm{R}}(\mathcal{C})$ and $n - k - 1$, the above result still holds with very high probability.

Proposition 5. *Let $\boldsymbol{G} \in \mathbb{F}_{q^m}^{k \times n}$ be a randomly sampled (rank k) generator matrix of a code $\mathcal{C}$. If $\boldsymbol{\Pi} \in \mathbb{F}_q^{n \times (n-h-\varepsilon)}$ is a full rank matrix with $h + \varepsilon < n - k$, then $\boldsymbol{G}\boldsymbol{\Pi} \in \mathbb{F}_{q^m}^{k \times (n-h-\varepsilon)}$ has full rank k with overwhelming probability.*

Proof. Let $S = \ker(\boldsymbol{\Pi})$, and let $\mathcal{C}, \mathcal{C}'$ be the codes with generator matrices $\boldsymbol{G}$ and $\boldsymbol{G}\boldsymbol{\Pi}$ respectively. For $j > 0$, the expected number of codewords $\boldsymbol{c} \in \mathcal{C}$ with $\mathrm{wt_R}(\boldsymbol{c}) = j$ is

$$\mathbb{E}[W_j(\mathcal{C})] = \frac{\#\ (m \times n)\text{-matrices of rank } j}{\#\ \text{syndromes}} = \frac{\prod_{i=0}^{j-1}(q^m - q^i)\begin{bmatrix} n \\ j \end{bmatrix}_q}{q^{m(n-k)}} \approx \frac{q^{j(m+n-j)}}{q^{m(n-k)}}.$$

Next, the probability of $\boldsymbol{G}\boldsymbol{\Pi}$ being rank-deficient is given by

$$\begin{aligned}
&\mathrm{Prob}(\exists \boldsymbol{c} \in \mathcal{C} \setminus \{0\} : \boldsymbol{c}\boldsymbol{\Pi} = 0) \\
&\quad \leq \sum_{\substack{c \in \mathcal{C} \setminus \{0\} \\ \mathrm{wt_R}(\boldsymbol{c}) \leq h+\varepsilon}} \mathrm{Prob}(\mathrm{supp_{row}}(\boldsymbol{c}) \subseteq S) = \sum_{j=d_R(\mathcal{C})}^{h+\varepsilon} \mathbb{E}[W_j(\mathcal{C})] \frac{\begin{bmatrix} h+\varepsilon \\ j \end{bmatrix}_q}{\begin{bmatrix} n \\ j \end{bmatrix}_q} \\
&\quad \approx \sum_{j=d_R(\mathcal{C})}^{h+\varepsilon} q^{j(m+n-j)-m(n-k)} \ \frac{q^{j(h+\varepsilon-j)}}{q^{j(n-j)}} \\
&\quad = \sum_{j=d_R(\mathcal{C})}^{h+\varepsilon} q^{j(m+h+\varepsilon-j)-m(n-k)} \approx q^{-m(n-k-h-\varepsilon)} \ll 1.
\end{aligned}$$

Hence, $\mathrm{Prob}(\boldsymbol{G}\boldsymbol{\Pi}$ has rank $k) = 1 - \mathrm{Prob}(\exists \boldsymbol{c} \in \mathcal{C} \setminus \{0\} : \boldsymbol{c}\boldsymbol{\Pi} = 0) \approx 1$. □

Theorem 2. *Let $(\boldsymbol{G}, \boldsymbol{y}, r)$ be an R-DP instance (assuming $\boldsymbol{y} \notin \mathcal{C}$) with unique solution $\boldsymbol{c} \in \mathcal{C}$, i.e., $d_{\mathrm{R}}(\boldsymbol{c}, \boldsymbol{y}) = r \leq d_{\mathrm{R}}(\mathcal{C})$ and $d_{\mathrm{R}}(\boldsymbol{x}, \boldsymbol{y}) > d_{\mathrm{R}}(\mathcal{C})$ for all other codewords $\boldsymbol{x} \in \mathcal{C} \setminus \{\boldsymbol{c}\}$. Let $\boldsymbol{\Pi} \in \mathbb{F}_q^{n \times (n-h-\varepsilon)}$ be a full rank matrix with $h < r$, $\varepsilon \leq d_{\mathrm{R}}(\mathcal{C}) - r$ and denote $\boldsymbol{e} := \boldsymbol{y} - \boldsymbol{c}$. Suppose that $\dim(\ker(\boldsymbol{\Pi}) \cap \mathrm{supp}_{\mathrm{row}}(\boldsymbol{e})) = h$. Then $\boldsymbol{c}' := \boldsymbol{c}\boldsymbol{\Pi}$ is the unique solution to the R-DP instance $(\boldsymbol{G}', \boldsymbol{y}', r')$, where $\boldsymbol{G}' := \boldsymbol{G}\boldsymbol{\Pi}$, $\boldsymbol{y}' := \boldsymbol{y}\boldsymbol{\Pi}$, and $r' := r - h$.*

Proof. First we show that $\boldsymbol{c}'$ is a solution. We have $\boldsymbol{y}' - \boldsymbol{c}' = (\boldsymbol{y} - \boldsymbol{c})\boldsymbol{\Pi} = \boldsymbol{e}\boldsymbol{\Pi}$, and using Lemma 2 we obtain

$$d_{\mathrm{R}}(\boldsymbol{c}', \boldsymbol{y}') = \mathrm{wt}_{\mathrm{R}}(\boldsymbol{e}\boldsymbol{\Pi}) = \mathrm{wt}_{\mathrm{R}}(\boldsymbol{e}) - \dim(\ker(\boldsymbol{\Pi}) \cap \mathrm{supp}_{\mathrm{row}}(\boldsymbol{e})) = r - h = r'$$

as desired.

Secondly, to show uniqueness, let $\hat{\boldsymbol{c}}'$ be any solution to the R-DP instance $(\boldsymbol{G}', \boldsymbol{y}', r')$. By Corollary 2, as $\hat{\boldsymbol{c}}'$ is in the code $\mathcal{C}'$ generated by $\boldsymbol{G}\boldsymbol{\Pi}$, there is a unique $\hat{\boldsymbol{c}} \in \mathcal{C}$ such that $\hat{\boldsymbol{c}}\boldsymbol{\Pi} = \hat{\boldsymbol{c}}'$. Again using Lemma 2 we get

$$\begin{aligned} d_{\mathrm{R}}(\hat{\boldsymbol{c}}, \boldsymbol{y}) &= \mathrm{wt}_{\mathrm{R}}(\boldsymbol{y} - \hat{\boldsymbol{c}}) \leq \mathrm{wt}_{\mathrm{R}}((\boldsymbol{y} - \hat{\boldsymbol{c}})\boldsymbol{\Pi}) + h + \varepsilon \\ &= \mathrm{wt}_{\mathrm{R}}(\boldsymbol{y}' - \hat{\boldsymbol{c}}') + h + \varepsilon \leq (r - h) + h + \varepsilon = r + \varepsilon \leq d_{\mathrm{R}}(\mathcal{C}), \end{aligned}$$

so by uniqueness $\hat{\boldsymbol{c}} = \boldsymbol{c}$ and $\hat{\boldsymbol{c}}' = \boldsymbol{c}'$. □

As before, since $\boldsymbol{G}$ was chosen uniformly at random, also $\boldsymbol{G}\boldsymbol{\Pi}$ is distributed uniformly at random.

Combining the subsupport-reduction with parameters h, ε and zero-reduction with parameter a reduces an instance with parameters (q, m, n, k, r) to an instance with parameters $(q, m, n - h - \varepsilon - a, k - a, r - h)$. As before, we may again use the MaxMinors modeling of the newly reduced instance.

Heuristic 3 *The system of $m\binom{n-h-\varepsilon-k-1}{r-h}$ linear equations in the MaxMinors modeling of a random R-DP instance after the novel subsupport-reduction is with high probability of rank* $\min\{m\binom{n-h-\varepsilon-k-1}{r-h}, \binom{n-h-\varepsilon-a}{r-h} - 1\}$.

Finally, we come to the time complexity of the subsupport-reduction with parameters h, ε. Recall that $TC_{\mathcal{A}}(q, m, n, k, r)$ denotes the average time complexity in $\mathbb{F}_q$-operations of a generic decoder $\mathcal{A}$ solving an R-DP instance with parameters (q, m, n, k, r) with *unique* solution.

Proposition 6. *Let $(\boldsymbol{G}, \boldsymbol{y}, r)$ be a random R-DP instance with unique solution and parameters (q, m, n, k, r). Let h, ε be parameters satisfying $0 \leq h < r$ and $0 \leq \varepsilon \leq d_{\mathrm{R}}(\mathcal{C}) - r < n - h - k$, with $\mathcal{C} = \langle \boldsymbol{G} \rangle$. Then the average time complexity of a generic rank-metric decoder $\mathcal{A}$ with subsupport-reduction $\mathrm{Red}_{\mathrm{sub},\, h,\varepsilon}$ solving this instance is given by*

$$TC_{\mathrm{Red}_{\mathrm{sub},\, h,\varepsilon}(\mathcal{A})}(q, m, n, k, r) \;= \mathcal{O}\left(q^{h(n-r-\varepsilon)} \cdot TC_{\mathcal{A}}(q, m, n - h - \varepsilon, k, r - h)\right).$$

The proof works exactly the same way as for Proposition 4.

As a working example, consider a random R-DP instance with unique solution and parameters $(q, m, n, k, r) = (2, 53, 52, 39, 6)$. By the rank GV-bound, a random code with these parameters will have $d_{\mathrm{R}}(\mathcal{C}) = d_{GV}^{\mathrm{R}} = 7$ with high probability. Applying subsupport reduction with $h = 4$, $\varepsilon = 1$ and $\mathcal{A} = \text{MM-}\mathbb{F}_q$ yields an average time complexity ($\omega = 2$) of

$$2^{4 \cdot 45} \cdot TC_{\text{MM-}\mathbb{F}_q}(2, 53, 47, 39, 2) = 2^{208},$$

while subsupport reduction with only $h = 4$, and $\varepsilon = 0$ and MM-$\mathbb{F}_q$ decoder yields an average time complexity of $2^{4 \cdot 46} \cdot TC_{\text{MM-}\mathbb{F}_q}(2, 53, 48, 39, 2) = 2^{212}$. Since $TC_{\mathrm{AGHT}}(2, 53, 52, 39, 6) = 221$ and $TC_{\mathrm{OJ1}}(2, 53, 52, 39, 6) = 229$, this shows that subsupport reduction with $\varepsilon > 0$ and $h > 0$ can give the best cost for certain parameters. For the RYDE parameters, the optimal choice for subsupport reduction is with $\varepsilon = 0$, and thus the generalization of subsupports to support-intersecting spaces does not yield an apparent improvement.

5 Conclusion

In this paper we introduced a new guessing step for the rank decoding problem based on guessing a subsupport of the error. The new technique allowed us to decrease the attack complexity for all RYDE parameters, in particular obtaining the best attack for NIST security level 5. While the focus of this paper was on $\mathbb{F}_{q^m}$-linear rank-metric codes and the parameters proposed by RYDE, such approach could also benefit the solvers for the MinRank problem, which is equivalent to an $\mathbb{F}_q$-linear rank-metric decoding problem, as used in Mirath.

References

1. Gora Adj, Luis Rivera-Zamarripa, Javier Verbel, Emanuele Bellini, Stefano Barbero, Andre Esser, Carlo Sanna, and Floyd Zweydinger. MiRitH. In *First Round Submission to the additional NIST Postquantum Cryptography Call*, 2023.
2. Carlos Aguilar Melchor, Nicolas Aragon, Magali Bardet, Slim Bettaieb, Loic Bidoux, Olivier Blazy, Jean-Christophe Deneuville, Philippe Gaborit, Adrien Hauteville, Ayoub Otmani, Olivier Ruatta, Jean-Pierre Tillich, and Gilles Zémor. ROLLO- Rank-Ouroboros, LAKE & LOCKER. *NIST PQC Call for Proposals*, 2020. Round 2 Submission.
3. Carlos Aguilar Melchor, Nicolas Aragon, Slim Bettaieb, Loic Bidoux, Olivier Blazy, Maxime Bros, Alain Couvreur, Jean-Christophe Deneuville, Philippe Gaborit, Adrien Hauteville, and Gilles Zémor. Rank Quasi-Cyclic (RQC). *NIST PQC Call for Proposals*, 2020. Round 2 Submission.
4. Nicolas Aragon, Magali Bardet, Loıc Bidoux, Jesús-Javier Chi-Domınguez, Victor Dyseryn, Thibauld Feneuil, Philippe Gaborit, Antoine Joux, Romaric Neveu, Matthieu Rivain, Jean-Pierre Tillich, and Adrien Vincotte. RYDE Signature Scheme. In *Second Round Submission to the additional NIST Postquantum Cryptography Call*, 2025.
5. Nicolas Aragon, Magali Bardet, Loïc Bidoux, Jesús-Javier Chi-Domínguez, Victor Dyseryn, Thibauld Feneuil, Philippe Gaborit, Antoine Joux, Matthieu Rivain, Jean-Pierre Tillich, and Adrien Vin***c***cotte. RYDE. In *First Round Submission to the additional NIST Postquantum Cryptography Call*, 2023.
6. Nicolas Aragon, Magali Bardet, Loïc Bidoux, Jesús-Javier Chi-Domínguez, Victor Dyseryn, Thibauld Feneuil, Philippe Gaborit, Romaric Neveu, Matthieu Rivain, and Jean-Pierre Tillich. MIRA. In *First Round Submission to the additional NIST Postquantum Cryptography Call*, 2023.
7. Nicolas Aragon, Victor Dyseryn, Philippe Gaborit, Pierre Loidreau, Julian Renner, and Antonia Wachter-Zeh. LowMS: a new rank metric code-based kem without ideal structure. *Designs, Codes and Cryptography*, 92(4):1075–1093, 2024.
8. Nicolas Aragon, Philippe Gaborit, Adrien Hauteville, and Jean-Pierre Tillich. A new algorithm for solving the rank syndrome decoding problem. In *2018 IEEE International Symposium on Information Theory (ISIT)*, pages 2421–2425, 2018.
9. Magali Bardet and Manon Bertin. Improvement of algebraic attacks for solving superdetermined minrank instances. In Jung Hee Cheon and Thomas Johansson, editors, *Post-Quantum Cryptography*, pages 107–123, Cham, 2022. Springer International Publishing.
10. Magali Bardet, Pierre Briaud, Maxime Bros, Philippe Gaborit, Vincent Neiger, Olivier Ruatta, and Jean-Pierre Tillich. An algebraic attack on rank metric code-based cryptosystems. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 64–93. Springer, 2020.
11. Magali Bardet, Pierre Briaud, Maxime Bros, Philippe Gaborit, and Jean-Pierre Tillich. Revisiting algebraic attacks on MinRank and on the rank decoding problem. *Designs, Codes and Cryptography*, 91(11):3671–3707, Nov 2023.
12. Magali Bardet, Maxime Bros, Daniel Cabarcas, Philippe Gaborit, Ray Perlner, Daniel Smith-Tone, Jean-Pierre Tillich, and Javier Verbel. Improvements of algebraic attacks for solving the rank decoding and MinRank problems. In Shiho Moriai and Huaxiong Wang, editors, *Advances in Cryptology – ASIACRYPT 2020*, pages 507–536, Cham, 2020. Springer International Publishing.
13. Jonathan F Buss, Gudmund S Frandsen, and Jeffrey O Shallit. The computational complexity of some problems of linear algebra. *Journal of Computer and System Sciences*, 58(3):572–596, 1999.
14. Eimear Byrne and Alberto Ravagnani. Covering radius of matrix codes endowed with the rank metric, 2016.

15. Florent Chabaud and Jacques Stern. The cryptographic security of the syndrome decoding problem for rank distance codes. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 368–381. Springer, 1996.
16. Giuseppe D'Alconzo, Andre Esser, Andrea Gangemi, and Carlo Sanna. Sneaking up the ranks: Partial key exposure attacks on rank-based schemes. *Cryptology ePrint Archive*, 2024.
17. Andre Esser, Javier A. Verbel, Floyd Zweydinger, and Emanuele Bellini. {SoK: CryptographicEstimators - a Software Library for Cryptographic Hardness Estimation}. In *AsiaCCS*. {ACM}, 2024.
18. Jean-Charles Faugere, Francoise Levy-dit Vehel, and Ludovic Perret. Cryptanalysis of MinRank. In *Annual International Cryptology Conference*, pages 280–296. Springer, 2008.
19. Philippe Gaborit, Olivier Ruatta, and Julien Schrek. On the complexity of the rank syndrome decoding problem. *IEEE Transactions on Information Theory*, 62(2):1006–1019, 2015.
20. Philippe Gaborit and Gilles Zémor. On the hardness of the decoding and the minimum distance problems for rank codes. *IEEE Transactions on Information Theory*, 62(12):7245–7252, 2016.
21. Francoise Levy-dit Vehel and L Perret. Algebraic decoding of rank metric codes. *Proceedings of YACC*, pages 142–152, 2006.
22. Pierre Loidreau. Properties of codes in rank metric. *arXiv preprint cs/0610057*, 2006.
23. Robert J. McEliece. A public-key cryptosystem based on algebraic coding theory. Technical report, DSN Progress report, Jet Propulsion Laboratory, Pasadena, 1978.
24. Alexei V Ourivski and Thomas Johansson. New technique for decoding codes in the rank metric and its cryptography applications. *Problems of Information Transmission*, 38:237–246, 2002.
25. Kaishun Wang, Jun Guo, and Fenggao Li. Association schemes based on attenuated spaces. *European Journal of Combinatorics*, 31(1):297–305, 2010.