ON THE HARDNESS OF CODE EQUIVALENCE PROBLEMS IN RANK METRIC

by

Alain Couvreur, Thomas Debris–Alazard & Philippe Gaborit

Abstract. — In recent years, the notion of rank metric in the context of coding theory has known many interesting developments in terms of applications such as space time coding, network coding or public key cryptography. These applications raised the interest of the community for theoretical properties of this type of codes, such as the hardness of decoding in rank metric or better decoding algorithms. Among classical problems associated to codes for a given metric, the notion of code equivalence has always been of the greatest interest.

In this article, we discuss the hardness of the code equivalence problem in rank metric for $\mathbb{F}_{q^m}$–linear and general rank metric codes. In the $\mathbb{F}_{q^m}$–linear case, we reduce the underlying problem to another one called Matrix Codes Right Equivalence Problem (MCRE). We prove the latter problem to be either in $\mathcal{P}$ or in $\mathcal{ZPP}$ depending of the ground field size. This is obtained by designing an algorithm whose principal routines are linear algebra and factoring polynomials over finite fields. It turns out that the most difficult instances involve codes with non trivial stabilizer algebras. The resolution of the latter case will involve tools related to finite dimensional algebras and the so–called Wedderburn–Artin theory. It is interesting to note that 30 years ago, an important trend in theoretical computer science consisted to design algorithms making effective major results of this theory. These algorithmic results turn out to be particularly useful in the present article.

Finally, for general matrix codes, we prove that the equivalence problem (both left and right) is at least as hard as the well–studied Monomial Equivalence Problem for codes endowed with the Hamming metric.

Contents

Introduction ........................................................................................................... 2
1. Basic objects and tools .................................................................................... 3
2. About the code equivalence problem ............................................................. 4
3. Main results ..................................................................................................... 6
4. Solving the Matrix Codes Right Equivalence Problem (MCRE) ............... 11
5. The Code Equivalence Problem for $\mathbb{F}_{q^m}$–linear codes : an Easy Problem . 16
6. Reduction from the Code Equivalence Problem in Hamming Metric .......... 21
Conclusion ........................................................................................................... 24
References .......................................................................................................... 25
Introduction

Rank metric and its applications. — Besides the well known notions of Hamming distance for error-correcting codes and Euclidean distance for lattices, the concept of rank metric was introduced in 1951 by Loo-Keng Hu [Hua51] as “arithmetic distance” for matrices over a field $\mathbb{F}_q$. Given two $n \times n$ matrices $A$ and $B$ over a finite field $\mathbb{F}_q$, the rank distance between $A$ and $B$ is defined as $|A - B| = \text{Rank}(A - B)$. Later, in 1978, Delsarte defined [Del78] the notion of rank distance on the space of bilinear forms (a way to regard spaces of matrices) and proposed a construction of optimal matrix codes in bilinear form representation. A matrix code over $\mathbb{F}_q$ is defined as an $\mathbb{F}_q$–linear subspace of the space of $m \times n$ matrices over $\mathbb{F}_q$ endowed with the rank metric. Later, in 1985, Gabidulin introduced in [Gab85] the notion of rank metric codes in vector representation (as opposed to the matrix representation) over a finite extension field $\mathbb{F}_{q^m}$ of $\mathbb{F}_q$. In the same paper, Gabidulin also introduced an optimal class of rank metric codes (in vector representation): the so-called Gabidulin codes, which can be regarded as analogues of Reed-Solomon codes but in a rank metric context, where polynomials are replaced by so–called linearized polynomials as introduced by Ore in 1933 in [Ore33].

Over the years, the notion of rank metric has become a very central tool with various applications such as space–time coding, network coding, public key cryptography and more recently for distributed data storage or digital watermarking.

The code equivalence problem. — In Hamming metric, the code equivalence problem for binary codes consists in: given two equivalent codes finding a permutation matrix permitting to pass from a code to its equivalent form. Such problem is central in coding theory. The difficulty of the this problem has been studied in [PR97] on a theoretical point of view. On the other hand, algorithms have been proposed by Leon [Leo82] and Sendrier [Sen00]. Interestingly enough the approach by Sendrier shows that for the binary case the practical complexity of the problem is exponential in the dimension of the hull (the intersection of a code and its dual) but since the average dimension of the hull for random codes is very low [Sen97] it shows that, in average, for binary codes (and also over $\mathbb{F}_3$ and $\mathbb{F}_4$), equivalence is easy to decide. On the other hand, it remains difficult for $q \geq 5$ [SS13]. Note that, in Euclidean metric a notion of lattice isomorphism (an equivalent notion for lattices) exists and has been studied by Haviv and Regev in [HR14].

In rank metric, the notion of code equivalence has been introduced by Berger in [Ber03] (see also [Mor14], [Gor19]) and invariants with respect to this code equivalence are considered in [GZ19], [HTM17], [NPHT20]. However, contrary to the Hamming metric case, to our knowledge, neither the algorithmic resolution of the code equivalence problem in rank metric nor its theoretical hardness have ever been discussed in the literature. This is the purpose of the present paper.

Our contribution. — In this article, we discuss the code equivalences problems in rank metric from a theoretical and algorithmic perspective. Our contributions in this article are three–fold

1°) We show that the right equivalence problem is easy. Interestingly, this proof involves many theoretical and algorithmic developments from Wedderburn-Artin theory.

2°) The $\mathbb{F}_{q^m}$–linear case is proved to be reduced to the previous case and hence is easy.

3°) Finally, the general case is proved to be harder than the code equivalence problem in Hamming metric by providing a polynomial time reduction.

Outline of the article. — The present article is organised as follows. Section 1 recalls main objects and tools that we consider. Section 2 gives background on the code equivalence problem in Hamming metric and established various code equivalence problems in rank metric. Our main results are stated in Section 3 and their proofs are sketched; detailed on proofs are given in the following sections. In Section 4 we show how to solve the right equivalence problem. Section 5 considers the code equivalence problem for $\mathbb{F}_{q^m}$–linear codes and finally, in Section 6 we present a polynomial–time reduction from the code equivalence problem in Hamming metric to the (both left and right) equivalence problem in rank metric.
1. Basic objects and tools

Before explaining the main ideas behind our results, we need to recall basic objects and tools that we consider. In all the article, we will consider two different kinds of objects which we will refer to codes. Namely:

- vector codes are usual codes, i.e. vector subspaces of $\mathbb{F}_q^n$. Such codes are usually endowed with the Hamming metric. These codes are denoted with calligraphic letter and with the subscript "Vec" such as $\mathcal{C}_{Vec}$;
- matrix codes, are subspaces of $\mathcal{M}_{m,n}(\mathbb{F}_q)$ ($m \times n$ matrices whose coefficients belong to $\mathbb{F}_q$) endowed with the rank metric. They are denoted as $\mathcal{C}_{Mat}$.

Nevertheless, there is class of vector codes that we can equip with the rank distance. There are subspaces of $\mathbb{F}_q^n$. Such codes are referred to as rank metric $\mathbb{F}_q^m$–linear codes and there are a particular class of matrix codes which turn out to be the most commonly studied in the literature. Let us describe them more precisely. Given a vector $v \in \mathbb{F}_q^m$ its rank weight is defined as

$$|v| \overset{def}{=} \dim_{\mathbb{F}_q} \{ \lambda_1 v_1 + \cdots + \lambda_n v_n : \lambda_i \in \mathbb{F}_q \}. \quad (1)$$

In other words, $|v|$ is the dimension of the $\mathbb{F}_q$–linear subspace of $\mathbb{F}_q^m$ spanned by the entries of $v$. It is well–known that any $\mathbb{F}_q^m$–linear code is isometric to a matrix code as follows. Choose an $\mathbb{F}_q$–basis $B = (b_1, \ldots, b_m)$ of $\mathbb{F}_q^m$, then consider the map

$$M_B : \begin{cases} \mathbb{F}_q^m & \longrightarrow \mathcal{M}_{m,n}(\mathbb{F}_q) \\ (v_1, \ldots, v_n) & \longmapsto \begin{pmatrix} v_{11} & \cdots & v_{1n} \\ \vdots & \ddots & \vdots \\ v_{m1} & \cdots & v_{mn} \end{pmatrix} \end{cases} \quad (2)$$

where for any $j, v_{1j}, \ldots, v_{mj} \in \mathbb{F}_q$ denote the coefficients of $v_j$ in the basis $B$. That is : $v_j = v_{1j}b_1 + \cdots + v_{mj}b_m$. One can easily prove that for any choice of $\mathbb{F}_q$–basis of $\mathbb{F}_q^m$, the above map preserves the metric.

An $\mathbb{F}_q^m$–linear code of dimension $k$ is through $M_B$ a $km$–dimensional matrix code of $\mathcal{M}_{m,n}(\mathbb{F}_q)$. However, it requires only $k$ vectors in $\mathbb{F}_q^m$ or equivalently $k$ matrices in $\mathcal{M}_{m,n}(\mathbb{F}_q)$ to be represented while a general $km$–dimensional matrix code of $\mathcal{M}_{m,n}(\mathbb{F}_q)$ is represented by a basis of $km$ matrices of size $m \times n$. Thus, the representation of an $\mathbb{F}_q^m$–linear code requires $m$ times less memory size compared to that of a general matrix code of the same $\mathbb{F}_q$–dimension. This gain is particularly interesting for cryptographic applications. This is basically what explains why in general McEliece cryptosystems based on rank metric codes have a smaller key size than McEliece cryptosystems based on the Hamming metric. All of these proposals (see for instance [GPT91, GO01, Gab08, GMRZ13, GRSZ14, ABD+19, AAB+17, ABG+19]) are actually built from matrix codes over $\mathbb{F}_q$ obtained from $\mathbb{F}_q^m$–linear codes.

On the other hand, vector codes which are $\mathbb{F}_q^m$–linear can be viewed as structured matrix codes with some “extra” algebraic structure in the same way as, for instance, cyclic linear codes can be viewed as structured versions of linear codes. In the latter case, the code is globally invariant by a linear isometric transform on the codewords corresponding to shifts of a certain length. Let us describe what happens in the case of $\mathbb{F}_q^m$–linear codes. Let $P = \sum_{i=0}^{m-1} a_i X^i + X^m \in \mathbb{F}_q[X]$ be a monic irreducible polynomial of degree $m$ and $x \in \mathbb{F}_q$ be a root of $P$. Then $\mathcal{B} = \{1, x, \ldots, x^{m-1}\}$ is an $\mathbb{F}_q$–basis of $\mathbb{F}_q^m$. Let $\mathcal{C}_{Vec}$ be an $\mathbb{F}_q^m$–linear code. The $\mathbb{F}_q^m$–linearity with the definition of $\mathcal{B}$ means that:

$$\forall c \in \mathcal{C}_{Vec}, \forall Q \in \mathbb{F}_q[X], Q(x) \cdot c \in \mathcal{C}_{Vec}.$$
In terms of matrices this stability can be expressed as follows. It is readily seen that (see (2) for the meaning of $M_B(\cdot)$)

$$M_B(P(x)c) = P(C_x)M_B(c) \quad \text{where} \quad C_x \overset{\text{def}}{=} \begin{pmatrix}
0 & 0 & \cdots & \cdots & -a_0 \\
1 & \ddots & \cdots & \cdots & -a_1 \\
0 & \ddots & \ddots & \cdots & \vdots \\
\vdots & \ddots & \ddots & \ddots & \vdots \\
0 & \cdots & \cdots & 1 & -a_{m-1}
\end{pmatrix} \in \mathbb{F}_q^{m \times m}$$

i.e. $C_x$ is the companion matrix of $P$ which represents the multiplication by $x$ in the basis $B$. In other words, the matrix code $M_B(C_{\text{Vec}}) \overset{\text{def}}{=} \{ M_B(c) : c \in C_{\text{Vec}} \}$ associated to $C_{\text{Vec}}$ is stable by left multiplication by an element of the algebra generated by the companion matrix $C_x$ of $x$.

This property of $\mathbb{F}_q$-linear codes make them as special a case of matrix codes: they have a large left stabilizer algebra (see Definition 11). As we will see in Section 5, the easiness of code equivalence problems for $\mathbb{F}_q$-linear codes comes, roughly speaking, from the fact that they have a particular stabilizer algebra.

2. About the code equivalence problem

We arrive now to the core of our paper: the code equivalence problem. We first recall here the definition of this problem (and some of its variants) for vector codes endowed with the Hamming metric.

2.1. In Hamming Metric. — In Hamming metric, the group of linear isometries of $\mathbb{F}_q^n$ is the subgroup of $GL_n(\mathbb{F}_q)$ (invertible $n \times n$ matrices over $\mathbb{F}_q$) spanned by permutation matrices and nonsingular diagonal matrices. Any element of this group can be represented as a product $DP$ where $D$ is diagonal and $P$ is a permutation matrix. Such a matrix is usually called a monomial matrix. It is a matrix having exactly one non-zero entry per row and per column.

Remark 1. — Note that over $\mathbb{F}_2$, monomial matrices are nothing but permutation matrices.

If two codes are image by each other under a permutation, they are said to be permutation equivalent. If they are image by a monomial matrix, they are said to be monomially equivalent.

2.1.1. Statement of the problems. — In Hamming metric, one usually considers two code equivalence problems. Given two codes $C_{\text{Vec}}, D_{\text{Vec}}$ endowed with the Hamming metric, there are two natural questions:

– are they permutation equivalent?
– are they monomially equivalent?

More formally, it leads to the following problems.

Problem 1 (Permutation Equivalence Problem (PE)). —

– Instance: $C_{\text{Vec}}, D_{\text{Vec}} \subseteq \mathbb{F}_q^n$;
– Decision: It exists a permutation matrix $P$ such that

$$C_{\text{Vec}}P = D_{\text{Vec}}.$$ 

Problem 2 (Monomial Equivalence Problem (ME)). —

– Instance: $C_{\text{Vec}}, D_{\text{Vec}} \subseteq \mathbb{F}_q^n$;
– Decision: It exists a permutation matrix $P$ and a nonsingular diagonal matrix $D$ such that

$$C_{\text{Vec}}DP = D_{\text{Vec}}.$$ 

Note that, according to Remark 1 when $q = 2$, i.e. for binary codes, the two questions are the same.
Remark 2. — Both problems are stated as decision problems, i.e. problems whose answer should be “yes” or “no”. Of course, one could be interested in the search version of these problem which would consist if the answer is “yes” to return the permutation $P$ (resp. the monomial matrix $DP$). We preferred introducing the decision versions which are more suitable when discussing reductions. On the other hand any of the considered algorithms actually aim at solving the search version of the considered problem.

Some comments. —

1. Note that we only consider linear isometries here. Some references in the literature discuss semi-linear isometries, i.e. the composition of a monomial transform and some iteration of a component wise Frobenius map (assuming that the ground field $\mathbb{F}_q$ is not prime). This would lead to an alternative problem but that can be solved from any solver of $ME$ in essentially the same time. Indeed if one can solve $ME$ in polynomial time, then it is enough to brute–force any iterate of the Frobenius.

2. More general problems are sometimes discussed in the literature. For instance deciding whether a given code $C_{Vec}$ is permutation equivalent to a sub-code of a given code $D_{Vec}$ (which has been proved $NP$–complete in $BGK17$):

Problem 3 (Permutation sub-code Equivalence Problem ($PsE$))

- **Instance:** $C_{Vec}, D_{Vec} \subseteq \mathbb{F}_q^n$.
- **Decision:** It exists a permutation matrix $P \in S_n$ such that $C_{Vec}P \subseteq D_{Vec}$.

3. Note that MacWilliams equivalence theorem $Mac62, HP03$ asserts that if there is a linear map $C_{Vec} \rightarrow D_{Vec}$ which is an isometry, then it extends to the whole ambient space. Therefore, solving $ME$ is equivalent to decide whether two codes are image of each other by a linear isometry with respect to the Hamming distance.

2.1.2. Algorithms. — Several algorithms appeared in the literature in view to solve $PE$ and $ME$.

- The first work is probably due to Leon $Leo82$, whose algorithm permits to solve both $PE$ and $ME$ at the cost of the calculation of the full list of minimum weight codewords of both $C_{Vec}$ and $D_{Vec}$, which requires an exponential running time for almost any code.

- Later, Sendrier $Sen00$ introduced the so–called support–splitting algorithm to solve $PE$. Its approach consists in comparing the weight enumerators of the hulls of the codes, i.e. the codes $C_{Vec} \cap C_{Vec}^\perp$ and $D_{Vec} \cap D_{Vec}^\perp$. The cost of this algorithm is heuristically of $O(2^{\dim C_{Vec} \cap C_{Vec}^\perp} \cdot \omega)$, where $\omega$ denotes the complexity exponent of linear algebraic operations. In particular, the algorithm solves easily the problem if the hull of the codes has small dimension, which typically holds $Sen97$.

Sendrier’s algorithm does not extends to $ME$ unless $q = 3$ and $q = 4$ if one replaces the hull by its counterpart with respect to the Hermitian inner product.

- In $Pen09$ is proposed a notion of normal form of a class of codes under the action of the monomial group. No mention of complexity appears in this reference but the computation of this normal form seems to require an exponential running time. In addition, a significant speed-up is possible for some input codes using tree–cutting dynamical programming methods.

- More recently, it was proposed in $ST17$ §3.2.4 to solve $PE$ by solving a polynomial system with Gröbner bases techniques. Actually this approach permits to solve $PsE$ and can be extended to the resolution of monomial equivalence. However, with this use of Gröbner bases, the average-time complexity analysis is unknown.

2.1.3. Theoretical hardness and known reductions. — From a more theoretical point of view, what can we say about the hardness of these problems? We already mentioned that sub-code equivalence problem $PsE$ is $NP$–complete. The hardness of $PE$ and $ME$ is much less clear. Here is what is known:

\footnote{In his article, Leon only considers the case of binary case, however, his approach is suitable for solving $ME$.}
It has been proved in [PR97] that the graph isomorphism problem reduces polynomially to \( \text{PE} \), it is also proved in the same reference that \( \text{PE} \) is not \( \text{NP} \)--complete unless the polynomial-time hierarchy collapses;

- Recently, in [BOS19] it was proved that \( \text{PE} \) restricted to instances with zero hull reduces to the graph isomorphism problem;

- Finally, from [SS13], \( \text{ME} \) reduces polynomially to \( \text{PE} \) when the field cardinality \( q \) is polynomial in \( n \). However, it should be noticed that the reduction sends any instance of \( \text{ME} \) into an instance of \( \text{PE} \) with a large hull, hence in the set of instances which seem to be the hardest ones.

The reductions are summarized in Figure 1.

![Figure 1. Reductions between various problems. The notation “A \( \longrightarrow \) B” means that “Problem A reduces to Problem B in polynomial time”. It can be translated as “If one can solve Problem B in polynomial time, then one can solve Problem A in polynomial time.”](image)

2.2. In Rank Metric. — In rank metric, the following linear automorphism of \( M_{m,n}(\mathbb{F}_q) \) are well-known to preserve the rank:

- left multiplication by an element \( A \) of \( \text{GL}_m(\mathbb{F}_q) \): \( X \mapsto AX \);
- right multiplication by an element \( B \) of \( \text{GL}_n(\mathbb{F}_q) \): \( X \mapsto XB \);
- the transposition map \( X \mapsto X^\top \) (only when \( m = n \)).

A classical result [Hua51] of linear algebra asserts that any rank-preservation linear automorphism of \( M_{m,n}(\mathbb{F}_q) \) is a composition of these three kinds of maps.

**Definition 3.** Two matrix codes \( C_{\text{Mat}}, D_{\text{Mat}} \subseteq M_{m,n}(\mathbb{F}_q) \) are said to be equivalent if there are matrices \( A \in \text{GL}_m(\mathbb{F}_q) \) and \( B \in \text{GL}_n(\mathbb{F}_q) \) such that \( C_{\text{Mat}} = A D_{\text{Mat}} B \). When \( A = I_m \) (resp. \( B = I_m \)) codes are said right (resp. left) equivalent.

**Remark 4.** Note that the previous definition does not involve the possibility of a transposition. Hence two square matrix codes \( C_{\text{Mat}}, D_{\text{Mat}} \subseteq M_n(\mathbb{F}_q) \) are equivalent according to our definition if and only if \( C_{\text{Mat}} \) is the image by a rank preserving linear map of either \( D_{\text{Mat}} \) or \( D_{\text{Mat}}^\top \). In practice, if we benefit from an algorithm which decides whether two square matrix codes are equivalent (according to our definition), then by applying the algorithm successively on the pairs \((C_{\text{Mat}}, D_{\text{Mat}})\) and \((C_{\text{Mat}}, D_{\text{Mat}}^\top)\), we can decide whether the codes are image of each other by a rank preserving automorphism. In short, considering a possible transposition will only multiply by 2 the algorithm complexity.

**Remark 5.** Contrary to the Hamming case there does not seem to exist a rank metric McWilliams equivalence theorem [BG13, Example 2.9]. Two codes may be image of each other by a linear isometry without being equivalent.
In this way, the code equivalence problem for matrix codes endowed with the rank metric can be stated as follows.

**Problem 4 (Matrix Code Equivalence Problem (MCE)).**

- Instance: two matrix codes $\mathcal{C}_{\text{Mat}}, \mathcal{D}_{\text{Mat}} \subseteq \mathcal{M}_{m,n}(\mathbb{F}_q)$;
- Decision: there exist matrices $A \in \text{GL}_m(\mathbb{F}_q)$ and $B \in \text{GL}_m(\mathbb{F}_q)$ such that
  \[ \mathcal{C}_{\text{Mat}} = A \mathcal{D}_{\text{Mat}} B. \]

In addition to this problem, one can be interested in the equivalence of the most commonly used rank metric codes, namely $\mathbb{F}_q$–linear codes. If codes are represented with vectors, there is no longer equivalence by left multiplying by a non-singular matrix but right multiplication is still possible.

**Problem 5 (Vector Matrix Code Equivalence Problem (V-MCE))**

- Instance: two vector codes $\mathcal{C}_{\text{Vec}}, \mathcal{D}_{\text{Vec}} \subseteq \mathbb{F}_q^n$;
- Decision: there exists a matrix $Q \in \text{GL}_m(\mathbb{F}_q)$ such that
  \[ \mathcal{C}_{\text{Vec}} = \mathcal{D}_{\text{Vec}} Q. \]

**Remark 6.** It is worth noting the difference of defining fields of the objects involved in V-MCE: the matrix $Q$ has entries in $\mathbb{F}_q$ while $\mathcal{C}_{\text{Vec}}, \mathcal{D}_{\text{Vec}}$ are spaces of vectors with entries in $\mathbb{F}_q^n$.

When regarding vector codes as matrix codes, using the expansion operation $M_B(\cdot)$ defined in (2), then V-MCE can be regarded as a restriction of the following problem to instances corresponding to matrix representations of $\mathbb{F}_q$–linear codes.

**Problem 6 (Matrix Codes Right Equivalence Problem (MCRE))**

- Instance: two matrix codes $\mathcal{C}_{\text{Mat}}, \mathcal{D}_{\text{Mat}} \subseteq \mathcal{M}_{m,n}(\mathbb{F}_q)$;
- Decision: there exists a matrix $Q \in \text{GL}_m(\mathbb{F}_q)$ such that
  \[ \mathcal{C}_{\text{Mat}} = \mathcal{D}_{\text{Mat}} Q. \]

Finally, let us introduce a last problem which could be more of cryptographic nature. Suppose given two vector codes $\mathcal{C}_{\text{Vec}}, \mathcal{D}_{\text{Vec}} \subseteq \mathbb{F}_q^n$ and two $\mathbb{F}_q$–bases $B, B'$ of $\mathbb{F}_q$–linear codes and consider the matrix codes $M_B(\mathcal{C}_{\text{Vec}})$ and $M_{B'}(\mathcal{D}_{\text{Vec}}).$ From the data of these matrix codes and without knowing the bases $B, B'$, is deciding equivalence easier? This leads to the following problem which is nothing but a restriction of MCE to the subset of instances of matrix codes arising from $\mathbb{F}_q$–linear codes.

**Problem 7 (Hidden Vector Matrix Code Equivalence Problem (HV-MCE))**

- Instance: $\mathcal{C}_{\text{Mat}}, \mathcal{D}_{\text{Mat}} \subseteq \mathcal{M}_{m,n}(\mathbb{F}_q)$ be two spaces of matrices representing $\mathbb{F}_q$–linear codes
- Decision: it exists $(S, P) \in \text{GL}_m(\mathbb{F}_q) \times \text{GL}_n(\mathbb{F}_q)$ such that
  \[ \mathcal{C}_{\text{Mat}} = S \mathcal{D}_{\text{Mat}} P. \]

3. Main results

Our contributions in this article are three-fold

1°) We propose a polynomial time algorithm to solve MCRE. This algorithm is deterministic if $q$ is polynomial in $mn$ and Las Vegas for a larger $q$.

2°) We prove that HV-MCE is easier than MCE and in particular that it naturally reduces to MCRE in deterministic polynomial time when $q$ is polynomial in $mn$ and via a polynomial time Las Vegas algorithm for larger $q$.

3°) Finally, we prove that the general problem MCE is at least as hard as ME in Hamming metric by providing a polynomial time reduction.

This leads to the following statements.

**Theorem 7.** MCRE and HV-MCE are in $\mathcal{P}$ if $q = (mn)^O(1)$ and in $\mathcal{ZPP}$ in the general case.

**Theorem 8.** ME reduces in polynomial time to MCE.
The why of $\mathcal{P}$ v.s. $\mathcal{ZPP}$. — In the sequel, we first propose an algorithm to solve MCRE. Then, we give a second algorithm which proves that HV-MCE reduces to MCRE. The major tools of these algorithms are the resolution of linear systems and factorization of univariate polynomials over a finite field. Linear systems are solvable in polynomial time while for factorization of univariate polynomials, Berlekamp algorithm [Ber68] is deterministic polynomial only when the ground field cardinality $q$ is polynomial in the degree. For larger $q$, one should use a Las Vegas algorithm such as Cantor Zassenhaus algorithm [Ber70, CZ81]. This is the reason why, for a large $q$, we cannot assert that the problem is in $\mathcal{P}$ but it is in $\mathcal{ZPP}$.

In the sequel we sketch the proofs of these theorems. The detailed proofs appear in the following sections as follows. In Section 4 we show how to solve MCRE efficiently. Section 5 gives an efficient algorithm which reduced HV-MCE to MCRE and therefore proves 7. Finally, Section 6 is devoted to our reduction, namely the proof of Theorem 8.

3.1. A polynomial–time algorithm to solve MCRE. — The resolution of this problem starts with the computation of an object called the conductor of $\mathcal{C}_{\text{Mat}}$ into $\mathcal{D}_{\text{Mat}}$ and

Definition 9 (Conductor of matrix codes). — Let $\mathcal{C}_{\text{Mat}}, \mathcal{D}_{\text{Mat}} \subseteq M_{m,n}(\mathbb{F}_q)$ and suppose they have the same $\mathbb{F}_q$–dimension \(^{(2)}\) and define the conductor of $\mathcal{C}_{\text{Mat}}$ into $\mathcal{D}_{\text{Mat}}$ as the $n \times n$ matrices space:

$$\text{Cond}(\mathcal{C}_{\text{Mat}}, \mathcal{D}_{\text{Mat}}) \overset{\text{def}}{=} \{ M \in M_{n}(\mathbb{F}_q) \mid \mathcal{C}_{\text{Mat}}M \subseteq \mathcal{D}_{\text{Mat}} \}.$$

Proposition 10. — If $\mathcal{C}_{\text{Mat}}Q = \mathcal{D}_{\text{Mat}}$ for some $Q \in GL_n(\mathbb{F}_q)$, that is to say, if $\mathcal{C}_{\text{Mat}}$ and $\mathcal{D}_{\text{Mat}}$ are right equivalent, then $Q \in \text{Cond}(\mathcal{C}_{\text{Mat}}, \mathcal{D}_{\text{Mat}})$. Conversely, if $\text{Cond}(\mathcal{C}_{\text{Mat}}, \mathcal{D}_{\text{Mat}})$ contains a nonsingular matrix $Q$, then $\mathcal{C}_{\text{Mat}}$ and $\mathcal{D}_{\text{Mat}}$ are right equivalent.

The conductor can be computed by solving a linear system with $n^2$ unknowns and $K(mn - K)$ equations, where $K = \dim(\mathcal{C}_{\text{Mat}}) = \dim(\mathcal{D}_{\text{Mat}})$. Indeed, our unknowns are matrices $M \in \text{Cond}(\mathcal{C}_{\text{Mat}}, \mathcal{D}_{\text{Mat}})$ whose entries provide $n^2$ unknowns in $\mathbb{F}_q$. For the equations, let $C_1, \ldots, C_K$ be a basis of $\mathcal{C}_{\text{Mat}}$ and $D_1, \ldots, D_{mn - K}$ be a basis of $\mathcal{D}_{\text{Mat}}$. Then, our equations with unknown $M$ are

$$\text{Tr}(C_i MD_j^T) = 0, \quad i \in \{1, \ldots, K\}, \quad j \in \{1, \ldots, N - K\}.$$  

In many classical situations, $m \approx n$, $K$ is linear in $mn$ and then the number of equations is of $O(n^3)$ while that of variables is of $O(n^2)$. Thus, the system is over constraint and one can expect that one of the two following situations appear:

- either $\text{Cond}(\mathcal{C}_{\text{Mat}}, \mathcal{D}_{\text{Mat}}) = \{0\}$ and one can assert that the codes are not right equivalent;
- or $\dim(\text{Cond}(\mathcal{C}_{\text{Mat}}, \mathcal{D}_{\text{Mat}})) = 1$ and, it suffices to pick out one non-zero matrix $M$ in the conductor and check if it is non singular. If it is non singular, we conclude that they are right equivalent, and provide a matrix $M$ which realizes the equivalence. If $M$ is singular, since the conductor has dimension 1, any other element of the conductor will be singular too and we conclude that the two codes are not equivalent.

In both situations, which are the ones that will “typically” happen, MCRE can be solved in polynomial time.

There remains to treat the case where $\text{Cond}(\mathcal{C}_{\text{Mat}}, \mathcal{D}_{\text{Mat}})$ has a larger dimension and may contain singular and non singular elements. In this situation, how to find non singular elements? In the other direction, how to assert that any element in this space is singular? Answering these questions requires the introduction of a fundamental notion: the stabilizer algebras of a matrix code.

Definition 11 (Stabilizer Algebra). — Let $\mathcal{C}_{\text{Mat}} \subseteq M_{m,n}(\mathbb{F}_q)$ be a matrix code. Its right stabilizer algebra is defined as:

$$\text{Stab}_{\text{right}}(\mathcal{C}_{\text{Mat}}) \overset{\text{def}}{=} \{ M \in M_{n}(\mathbb{F}_q) \mid \mathcal{C}_{\text{Mat}}M \subseteq \mathcal{C}_{\text{Mat}} \}.$$  

\(^{(2)}\) If they do not, then they cannot be equivalent.
while its left stabilizer algebra is defined as:

\[ \text{Stab}_{\text{left}}(C_{\text{Mat}}) \overset{def}{=} \{ M \in M_{m,n}(\mathbb{F}_q) \mid M C_{\text{Mat}} \subseteq C_{\text{Mat}} \}. \]

**Remark 12.** — The right stabilizer algebra is nothing but the conductor of \( C_{\text{Mat}} \) into itself, namely \( \text{Cond}(C_{\text{Mat}}, C_{\text{Mat}}) \). In particular, it can be computed by solving a system of linear equations.

It is easily verified that \( \text{Stab}_{\text{left}}(C_{\text{Mat}}) \) and \( \text{Stab}_{\text{right}}(C_{\text{Mat}}) \) are sub-algebras of \( M_{m,n}(\mathbb{F}_q) \) and \( M_{m,n}(\mathbb{F}_q) \) respectively. In particular, for a given code \( C_{\text{Mat}} \), its right stabilizer algebra contains scalar matrices, i.e. matrices of the form \( \lambda I_n \). If the stabilizer does not contain other matrices, it is said to be trivial. Note that most of the matrix codes have trivial stabilizer algebras. The following statement gives a first motivation for the introduction of these algebras.

**Proposition 13.** — Let \( C_{\text{Mat}}, D_{\text{Mat}} \subseteq M_{m,n}(\mathbb{F}_q) \) be two right equivalent codes; i.e. there exists \( Q \in \text{GL}_n(\mathbb{F}_q) \) such that \( C_{\text{Mat}} Q = D_{\text{Mat}} \). Suppose also that \( \dim(\text{Cond}(C_{\text{Mat}}, D_{\text{Mat}})) \geq 2 \). Then, \( C_{\text{Mat}} \) and \( D_{\text{Mat}} \) have non trivial right stabilizer algebra.

**Proof.** — By assumption, there exists \( M \in \text{Cond}(C_{\text{Mat}}, D_{\text{Mat}}) \) which is not a scalar multiple of \( Q \) and hence \( MQ^{-1} \in \text{Stab}_{\text{right}}(C_{\text{Mat}}) \setminus \{\lambda I_n \mid \lambda \in \mathbb{F}_q\} \).

While we observed that, when the conductor has dimension \( \leq 1 \), then solving MCRE is easy to solve. The previous statement asserts that, the cases where the conductor has a higher dimension while the codes are right equivalent are precisely the cases where stabilizer algebras are non-trivial.

In such case, the problem is more technical to prove and requires a further study of the stabilizer algebra using Wedderburn–Artin Theory. We refer the reader to [DK94] for the mathematical aspects of this theory and to the works of Friedl and Rónyai [FR85, Rón90] for the computational aspects. The results of the previous references assert the existence of decompositions

\[ C_{\text{Mat}} = C_{\text{Mat}} E_1 \oplus \cdots \oplus C_{\text{Mat}} E_s, \quad D_{\text{Mat}} = C_{\text{Mat}} F_1 \oplus \cdots \oplus C_{\text{Mat}} F_t \]

where the \( E_i \)'s (resp. the \( F_i \)'s) are projectors matrices onto subspaces in direct sum. Moreover, these decompositions are unique up to conjugation and deciding the right equivalence reduces to decide equivalences of these small pieces of codes and then deduce a global equivalence. This yields a recursive algorithm whose stopping case is the case where \( s = t = 1 \), corresponding to the case where the right stabilizer algebra is said to be local (Definition 17). This stopping case will turn out to be easy to solve as detailed in Section 3.2.2.

### 3.2. Equivalence of \( F_q^m \)-linear codes.

— Clearly, an instance of V-MCE is a particular instance of MCRE and hence, according to the previous discussion, can be solved in polynomial time. On the other hand, instances of HV-MCE are particular instances of MCE. Thus, solving HV-MCE requires \( \text{à priori} \) to look for left and right equivalence at the same time. However, it turns out, that the \( F_q^m \)-linear structure permits to treat these problems separately.

Basically, we proceed as follows. We start from a pair of matrix codes \( C_{\text{Mat}}, D_{\text{Mat}} \subseteq M_{m,n}(\mathbb{F}_q) \) obtained by expanding vector codes \( C_{\text{Vec}}, D_{\text{Vec}} \subseteq F_q^m \) in (possibly distinct) \( F_q \)-bases of \( F_q^m \). The left stabilizer algebras of these codes can be computed and both are either isomorphic to \( F_q^m \) or contain a subalgebra isomorphic to \( F_q^m \). The latter case is rather more technical to study (see Section 5.2.3), thus let us suppose we are in the former case. Classical results of linear algebra permit to prove that two sub-algebras \( \text{Stab}_{\text{left}}(C_{\text{Mat}}), \text{Stab}_{\text{left}}(D_{\text{Mat}}) \subseteq M_{m,n}(\mathbb{F}_q) \) which are both isomorphic to \( F_q^m \) are conjugated and that a matrix \( P \in \text{GL}_{m,n}(\mathbb{F}_q) \), that can be efficiently computed, such that

\[ \text{Stab}_{\text{left}}(D_{\text{Mat}}) = P^{-1}\text{Stab}_{\text{left}}(C_{\text{Mat}})P. \]

Next, replacing \( D_{\text{Mat}} \) by \( P D_{\text{Mat}} \), the two codes turn out to have the same left stabilizer matrix and then, up to some technical details related to the action of the Frobenius map (see Section 5.2), we are then reduced to decide right equivalence on the two codes.
3.3. Reduction from MCE to ME. — We prove that the general equivalence problem for matrix codes, MCE, is at least as hard as the monomial equivalence problem ME. We obtain such a statement by providing a polynomial time reduction from ME to MCE.

To make our reduction we have to start with an instance \((\mathcal{C}_{\text{Vec}}, \mathcal{D}_{\text{Vec}})\) of ME. Our aim is then to efficiently map it into an instance of matrix codes \((\mathcal{C}_{\text{Mat}}, \mathcal{D}_{\text{Mat}})\) that will be equivalent (in the sense of Definition 3) if and only if \(\mathcal{C}_{\text{Vec}}\) and \(\mathcal{D}_{\text{Vec}}\) are monomially equivalent. Basically, the rationale behind our reduction relies on the following observation. Suppose that \(\mathcal{C}_{\text{Vec}}\) and \(\mathcal{D}_{\text{Vec}}\) are monomially equivalent, then, by definition, there exist \(P\) invertible and \(D\) a diagonal matrix such that:

\[
\mathcal{C}_{\text{Vec}} = \mathcal{D}_{\text{Vec}}DP
\]

In other words for any generator matrix \(A\) of \(\mathcal{C}_{\text{Vec}}\) and \(B\) of \(\mathcal{D}_{\text{Vec}}\), there exists an invertible matrix \(S\) such that

\[
A = SBDP
\]

Here the action of \(P\) and \(D\) is on the columns of \(B\). In this way, let us introduce \(a_1^\top, \ldots, a_n^\top, b_1^\top, \ldots, b_n^\top \in F_q^k\) which denote the columns of \(A\) and \(B\) respectively (vectors are in row notation). The key observation is then that (3) implies:

\[
a_i^\top = \alpha_j S b_j^\top \quad \text{and} \quad a_i^\top a_i = \alpha_j^2 S b_j^\top b_j S^\top,
\]

where \(j\) is image by \(i\) of the permutation \(P\) while \(\alpha_i\) is the \(i\)-th diagonal coefficient of \(D\). Therefore, the matrix codes

\[
\mathcal{C}_{\text{Mat}} \overset{\text{def}}{=} \left\{ \sum_{i=1}^n \lambda_i a_i^\top a_i : \lambda_i \in F_q \right\} \quad \text{and} \quad \mathcal{D}_{\text{Mat}} \overset{\text{def}}{=} \left\{ \sum_{i=1}^n \lambda_i b_i^\top b_i : \lambda_i \in F_q \right\}
\]

are equivalent in the following manner

\[
\mathcal{C}_{\text{Mat}} = S \mathcal{D}_{\text{Mat}} S^\top.
\]

Consequently, by considering matrix codes instead of vector codes in this way, roughly speaking we “erase” the action of the permutation and the diagonal matrix over the coordinates of the code. This brings back the decision to a matrix equivalence problem.

The technical part of this reduction is to show that any equivalent matrix code like \(\mathcal{C}_{\text{Mat}}\) and \(\mathcal{D}_{\text{Mat}}\) will lead to associate vector codes \(\mathcal{C}_{\text{Vec}}\) and \(\mathcal{D}_{\text{Vec}}\) which are monomially equivalent. A fundamental observation for is the fact that \(\mathcal{C}_{\text{Mat}}\) and \(\mathcal{D}_{\text{Mat}}\) are generated by matrices of rank one. Thus, if \(\mathcal{C}_{\text{Mat}} = U \mathcal{D}_{\text{Mat}} V\) for some invertible matrices \(U\) and \(V\) we could typically expect that for any \(i\),

\[
a_i = \lambda_j b_j V
\]

for some \(\lambda_j \in F_q\) and \(j\). Indeed, this would untypical that a matrix of rank 1 \(a_i^\top a_i\) can be written as a linear combination of \(b_j^\top b_j\). But this may happen. This is a technical issue can be avoided by changing a little bit the construction of \(\mathcal{C}_{\text{Mat}}\) and \(\mathcal{D}_{\text{Mat}}\) and we refer to Section 6 for further details. In any case, Equation (5) leads to

\[
\mathcal{C}_{\text{Vec}} = \mathcal{D}_{\text{Vec}}DP
\]

where \(D\) is diagonal of coefficients the \(\lambda_j\)’s and \(P\) is a permutation matrix which maps \(i\) to \(j\).
4. Solving the Matrix Codes Right Equivalence Problem (MCRE)

This section is devoted to the resolution of MCRE. As we already noticed in Section 3.1, most of the time, the resolution of the problem is simple since the conductor \( \text{Cond}(\mathcal{C}_{\text{Mat}}, \mathcal{D}_{\text{Mat}}) \) has typically dimension \( \leq 1 \). If it does not, while the codes are right equivalent, then, according to Proposition 13, the codes \( \mathcal{C}_{\text{Mat}} \) and \( \mathcal{D}_{\text{Mat}} \) should have non trivial stabilizer algebras. The following statements provide further relations between the structure of the conductor with that of the stabilizer algebras, which will be helpful in what follows.

**Proposition 14.** — If two codes \( \mathcal{C}_{\text{Mat}} \) and \( \mathcal{D}_{\text{Mat}} \) are right equivalent, i.e. \( \mathcal{C}_{\text{Mat}} Q = \mathcal{D}_{\text{Mat}} \) for some \( Q \in \text{GL}_n(\mathbb{F}_q) \), then their right stabilizer algebras are conjugated under \( Q \):

\[
\text{Stab}_{\text{right}}(\mathcal{C}_{\text{Mat}}) = Q \cdot \text{Stab}_{\text{right}}(\mathcal{D}_{\text{Mat}}) \cdot Q^{-1}.
\]

**Proof.** — Let \( D \in \text{Stab}_{\text{right}}(\mathcal{D}_{\text{Mat}}) \). By definition we have the following computation,

\[
\mathcal{C}_{\text{Mat}} Q D Q^{-1} = \mathcal{D}_{\text{Mat}} D Q^{-1} \subseteq \mathcal{D}_{\text{Mat}} Q^{-1} = \mathcal{C}_{\text{Mat}}
\]

which shows that \( Q D Q^{-1} \in \text{Stab}_{\text{right}}(\mathcal{C}_{\text{Mat}}) \) and easily concludes the proof. \( \square \)

**Proposition 15.** — Let \( \mathcal{C}_{\text{Mat}}, \mathcal{D}_{\text{Mat}} \subseteq \text{M}_{m,n}(\mathbb{F}_q) \) be matrix codes and \( Q \in \text{GL}_n(\mathbb{F}_q) \) such that \( \mathcal{C}_{\text{Mat}} Q = \mathcal{D}_{\text{Mat}} \). Then,

\[
\text{Cond}(\mathcal{C}_{\text{Mat}}, \mathcal{D}_{\text{Mat}}) = \text{Stab}_{\text{right}}(\mathcal{C}_{\text{Mat}}) \cdot Q = Q \cdot \text{Stab}_{\text{right}}(\mathcal{D}_{\text{Mat}}).
\]

**Proof.** — Let \( S \in \text{Stab}_{\text{right}}(\mathcal{C}_{\text{Mat}}) \), then

\[
\mathcal{C}_{\text{Mat}} S Q \subseteq \mathcal{C}_{\text{Mat}} Q = \mathcal{D}_{\text{Mat}}.
\]

Therefore, \( S Q \in \text{Cond}(\mathcal{C}_{\text{Mat}}, \mathcal{D}_{\text{Mat}}) \) and hence \( \text{Stab}_{\text{right}}(\mathcal{C}_{\text{Mat}}) Q \subseteq \text{Cond}(\mathcal{C}_{\text{Mat}}, \mathcal{D}_{\text{Mat}}) \). Conversely, let \( C \in \text{Cond}(\mathcal{C}_{\text{Mat}}, \mathcal{D}_{\text{Mat}}) \), then

\[
\mathcal{C}_{\text{Mat}} C Q^{-1} \subseteq \mathcal{D}_{\text{Mat}} Q^{-1} = \mathcal{C}_{\text{Mat}}
\]

Therefore, \( C Q^{-1} \in \text{Stab}_{\text{right}}(\mathcal{C}_{\text{Mat}}) \) or equivalently \( C \in \text{Stab}_{\text{right}}(\mathcal{C}_{\text{Mat}}) Q \). This yields

\[
\text{Cond}(\mathcal{C}_{\text{Mat}}, \mathcal{D}_{\text{Mat}}) \subseteq \text{Stab}_{\text{right}}(\mathcal{D}_{\text{Mat}}) Q.
\]

The equality \( \text{Cond}(\mathcal{C}_{\text{Mat}}, \mathcal{D}_{\text{Mat}}) = Q \text{Stab}_{\text{right}}(\mathcal{D}_{\text{Mat}}) \) is a consequence of the previous equality together with Proposition 13. \( \square \)

4.1. Finite dimensional algebras. — As said earlier, the general resolution of MCRE involves results on finite dimensional algebras and the Wedderburn–Artin theory. Here we recall some known results in the literature that we will apply to the right stabilizer algebra of our codes. The theoretical facts can be found for instance in [DK94], while for the constructive and algorithmic aspects, we refer the reader to [FR85, Rön90].

A finite dimensional algebra \( \mathcal{A} \) is finite dimensional vector space over \( \mathbb{F}_q \) which is also a ring (possibly non commutative) with unit \( 1_{\mathcal{A}} \). A way to represent such an algebra is to use a basis \( a_1, \ldots, a_\ell \) together with the collection of so-called structure constants with respect to this basis, which are a sequence \( (\lambda_{ijk})_{i,j,k} \) such that:

\[
\forall i,j \in \{1, \ldots, \ell\}, \quad a_i a_j = \sum_{k=1}^\ell \lambda_{ijk} a_k.
\]

In addition, it is well-known that any such ring can always be represented as a subring of \( \mathcal{M}_t(\mathbb{F}_q) \) as follows. To each element \( a \in \mathcal{A} \), we associate the matrix representation in basis \( a_1, \ldots, a_\ell \) of the map \( x \mapsto ax \). Note that if \( a \neq 0 \), then this right multiplication map cannot be identically zero on \( \mathcal{A} \) since \( \mathcal{A} \) contains a unit \( 1_{\mathcal{A}} \) whose image by left multiplication by \( a \) is \( a \). Therefore, this matrix representation is faithful (i.e. the corresponding map \( \mathcal{A} \to \mathcal{M}_t(\mathbb{F}_q) \) is injective).

A finite dimensional algebra is simple if its only two-sided ideals are \( \{0\} \) and the whole algebra itself. From Wedderburn–Artin Theorem [DK94, Cor. 2.4.5] together with Wedderburn Theorem
4.2.2. When the stabilizer algebras are local

4.2.1. Context

4.2. General resolution of MCRE

To solve this problem, we are going to proceed in the following way:

1. handle the case where \( \text{Stab}_{\text{right}}(\mathcal{C}_{\text{Mat}}) \) is local;
2. give a recursive algorithm whose base case is 1.

4.2.2. When the stabilizer algebras are local

It is easy to decide if \( \mathcal{C}_{\text{Mat}} \) and \( \mathcal{D}_{\text{Mat}} \) are right equivalent when \( \text{Stab}_{\text{right}}(\mathcal{C}_{\text{Mat}}) \) is a local algebra. Basically it relies on the following lemma.

Lemma 19. — Let \( \mathcal{C}_{\text{Mat}}, \mathcal{D}_{\text{Mat}} \subseteq \mathcal{M}_{m,n}(\mathbb{F}_q) \) be two matrix codes that are right equivalent and such that \( \text{Stab}_{\text{right}}(\mathcal{C}_{\text{Mat}}) \) is local. Then any element of

\[
\text{Cond}(\mathcal{C}_{\text{Mat}}, \mathcal{D}_{\text{Mat}}) \setminus (\text{Cond}(\mathcal{C}_{\text{Mat}}, \mathcal{D}_{\text{Mat}}) \cap \text{Cond}(\mathcal{C}_{\text{Mat}}, \mathcal{D}_{\text{Mat}}))
\]

is nonsingular.

Proof. — By hypothesis, \( \mathcal{C}_{\text{Mat}}Q = \mathcal{D}_{\text{Mat}} \) for some \( Q \in \text{GL}_n(\mathbb{F}_q) \). From Proposition 15, \( \text{Cond}(\mathcal{C}_{\text{Mat}}, \mathcal{D}_{\text{Mat}}) = \text{Stab}_{\text{right}}(\mathcal{C}_{\text{Mat}})Q \) and hence any element of \( \text{Cond}(\mathcal{C}_{\text{Mat}}, \mathcal{D}_{\text{Mat}}) \) can be written \( CQ \) where \( C \in \text{Stab}_{\text{right}}(\mathcal{C}_{\text{Mat}}) \). But now as \( Q \) is invertible, \( CQ \) is singular if and only if \( C \in \text{Stab}_{\text{right}}(\mathcal{C}_{\text{Mat}}) \) is singular. Therefore, as \( \text{Stab}_{\text{right}}(\mathcal{C}_{\text{Mat}}) \) is local, by Theorem 15 \( \mathcal{C}_{\text{Mat}} \) is singular if and only if \( C \in \mathcal{R} \).

This situation yields to Algorithm 1 which decides the right-equivalence problem in polynomial time when right stabilizer algebras are local.
Algorithm 1: An algorithm to decide the right equivalence problem when stabilizer algebras are local

**Input**: Two codes $C_{\text{Mat}}, D_{\text{Mat}} \subseteq \text{Mat}_{m,n}(\mathbb{F}_q)$

**Output**: Codes are right-equivalent or not

Compute $\text{Cond}(C_{\text{Mat}}, D_{\text{Mat}})$, $R \overset{\text{def}}{=} \text{Rad}(\text{Stab}_{\text{right}}(C_{\text{Mat}}))$ and $R_{\text{Cond}}(C_{\text{Mat}}, D_{\text{Mat}})$

Pick $A \in \text{Cond}(C_{\text{Mat}}, D_{\text{Mat}}) \setminus R_{\text{Cond}}(C_{\text{Mat}}, D_{\text{Mat}})$

if $A$ non singular then

\[ \text{return} \text{ Codes are right-equivalent} \]

else

\[ \text{return} \text{ Codes are not right-equivalent} \]

**Theorem 20**. — Let $C_{\text{Mat}}, D_{\text{Mat}}$ be matrix codes with local stabilizer algebras. Algorithm 1 succeeds to decide if they are right equivalent in polynomial time.

**Proof**. — Algorithm 1 is clearly polynomial as conductor and Jacobson radical can be computed in polynomial time (see Theorem 18). Now, the correctness of the algorithm is a direct consequence of Lemma 19.

4.2.3. The general case. — We discuss here the algorithm to solve MCRE in the general case. Let us begin by a little discussion about puncturing a matrix code. It will be helpful to understand why our algorithm which solves MCRE terminates.

**Puncturing a Matrix Code.** Recall that row-support of $C_{\text{Mat}} \subseteq \text{Mat}_{m,n}(\mathbb{F}_q)$ is defined as the sum of the row-spaces of its elements. We say that $C_{\text{Mat}}$ is of full support if its row-support is equal to $\mathbb{F}_n^q$. When a code is not of full support we can always bring it back to a full-support one. For this we use a operation of puncturing (see for instance [BR17, She19]).

**Definition 21** (Puncturing). — Let $C_{\text{Mat}} \subseteq \text{Mat}_{m,n}(\mathbb{F}_q)$ be a matrix code and $r \in \{1, \ldots, n\}$. Let $M \in \text{Mat}_{n,r}(\mathbb{F}_q)$. The puncturing of $C_{\text{Mat}}$ with respect to $M$ is the code:

$C_{\text{Mat}}M \subseteq \text{Mat}_{m,r}(\mathbb{F}_q)$

Let $R$ be the support of $C_{\text{Mat}}$ and $r \overset{\text{def}}{=} \dim R$. Then we can efficiently compute a projection matrix $M \in \text{Mat}_{n,r}(\mathbb{F}_q)$ (whose row image is $R$) such that puncturing $C_{\text{Mat}}$ according to it gives a matrix-code $\subseteq \text{Mat}_{m,r}(\mathbb{F}_q)$ which is of full support.

Note now that if two codes $C_{\text{Mat}}$ and $D_{\text{Mat}}$ are right equivalent then their row-support are isomorphic. Suppose that these codes don’t have a full-support then we can puncture them to get full-support codes which are still right-equivalent. It is why in the sequel we suppose that instances of MCRE are matrix codes with full support.

**Solving MCRE.** In the general case to solve MCRE we cannot make any hypothesis on the structure of the right stabilizer algebras. Our basic idea to solve MCRE is to work in order to reduce the decision to the case where stabilizer algebras of codes are local, case that we can solve with Algorithm 1. For this we are going “to decompose” right stabilizer algebras. We will use the following proposition.

**Proposition 22**. — Let $A$ be a $n$-dimensional algebra over $\mathbb{F}_q$. Then we can compute, in deterministic polynomial time if $q = n^{O(1)}$ or with a polynomial time Las Vegas algorithm for larger $q$, a decomposition of the identity of $A$ as a sum of minimal orthogonal idempotents:

$1_A = e_1 + \cdots + e_s$. 
Furthermore this conjugation is unique to conjugation, namely if:

\[ 1_A = f_1 + \cdots + f_t \]

we have \( s = t \) and there exists \( a \in A^\times \) such that, up to re-indexing the \( f_i \)'s, we have \( e_i = af_ia^{-1} \) for any \( i \in \{1, \ldots, s\} \).

Proof. — We proceed as follows.

– First, using Rónyai’s algorithm, one computes the Jacobson Radical \( \text{Rad}(A) \) of \( A \) in polynomial time;

– We compute the algebra \( S \triangleq A/\text{Rad}(A) \) which is semi-simple (its radical is \( \{0\} \)) [DK94, Proposition 3.1.13]. According to [Rón90, Theorem 3.1] we can compute, in deterministic polynomial time if \( q = n^{O(1)} \) or with a polynomial time Las Vegas method for larger \( q \), bases of some algebras \( S_i \)'s that are also ideals of \( S \) such that (the so-called Wedderburn decomposition):

\[ S = S_1 \oplus \cdots \oplus S_r. \]

Therefore we can decompose the identity of \( S \) as:

\[ 1_S = s_1 + \cdots + s_r \]

where each \( s_i \in S_i \). Note that if \( x \in S_i \) then \( xs_j = s_jx = 0 \) for \( j \neq i \) as the \( S_j \) are ideals in direct sum. By using (6) we conclude that \( xs_i = x \) for any \( x \in S_i \). Therefore, \( s_i = 1_{S_i} \) for any \( i \). In this way we have a decomposition of the identity as sum of orthogonal idempotents:

\[ 1_S = 1_{S_1} + \cdots + 1_{S_r}. \]

Now by definition of the Wedderburn decomposition [DK94, Thm. 2.4.1] the \( S_i \)'s are isomorphic to matrix algebras over some finite extension of \( \mathbb{F}_q \). In addition, using [Rón90, § 5.1], these isomorphisms can be explicitly computed. By this manner, we can compute for each \( 1_{S_i} \) a decomposition of the identity as sum of orthogonal idempotent that are minimal. Indeed, since \( S_i \) is isomorphic to an algebra of matrices, it is enough to decompose the identity matrix as a sum of matrices with only one non-zero element which is on the diagonal.

Therefore this leads to a decomposition of the identity of \( S \) as a sum of minimal orthogonal idempotents:

\[ 1_S = e_1 + \cdots + e_s \quad \text{where} \quad s \geq r. \]

– Since the ground field is finite, from Wedderburn–Malcev Theorem [DK94, Thm. 6.2.1], there exists an injective morphism of algebras \( S \to A \) and the image of this morphism is unique up to conjugation by a unipotent element (i.e. an element \( x \in A \) such that \( x \equiv 1 \mod \text{Rad}(A) \); such an element is known to be invertible). Moreover, such a lift of \( S \) can be computed by linear algebra as explained in [Bre11, § 1.12]. Using this lift and since we know it is unique up to unipotent conjugation, we deduce a decomposition of the identity of \( A \) by minimal orthogonal idempotents:

\[ 1_A = a_1 + \cdots + a_s, \]

and this decomposition is unique up to conjugation which concludes the proof.

Remark 23. — The case \( s = 1 \) in the decomposition corresponds to a local algebra (also referred to as indecomposable algebra in the literature). This is a consequence of [DK94, Theorem 3.2.2].

We are now equipped to give the algorithm to solve MCRE. Given \( S_{\text{Mat}} \) and \( S_{\text{Mat}} \) of full-support, let

\[ S_{\text{Mat}} \triangleq \text{Stab}_{\text{right}}(S_{\text{Mat}}) \quad \text{and} \quad S_{\text{Mat}} \triangleq \text{Stab}_{\text{right}}(S_{\text{Mat}}) \]

their right stabilizer algebras. To decide if \( S_{\text{Mat}} \) and \( S_{\text{Mat}} \) are right equivalent we proceed as follows.
First from Proposition 22 we can compute efficiently a decomposition of the identity $1_{\mathcal{S}_{\text{Mat}}} = I_n = E_1 \oplus \cdots \oplus E_\ell$ in $\mathcal{S}_{\text{Mat}}$. We deduce a decomposition of $\mathcal{C}_{\text{Mat}}$ as a direct sum of codes:

$$\mathcal{C}_{\text{Mat}} = \mathcal{C}_{\text{Mat}} E_1 \oplus \cdots \oplus \mathcal{C}_{\text{Mat}} E_\ell.$$  

We perform the very same operations on $\mathcal{D}_{\text{Mat}}$ and $\mathcal{S}_{\mathcal{D}_{\text{Mat}}}$ and get a minimal decomposition:

$$\mathcal{D}_{\text{Mat}} = \mathcal{D}_{\text{Mat}} F_1 \oplus \cdots \oplus \mathcal{D}_{\text{Mat}} F_\ell.$$  

Note now if $\mathcal{C}_{\text{Mat}} Q = \mathcal{D}_{\text{Mat}}$ for some $Q \in \text{GL}_n(\mathbb{F}_q)$, then, from Proposition 13 the stabilizer algebras of the codes are conjugated under $Q$ and by the uniqueness of the minimal decompositions of the identity, $\ell = \ell'$. If $\ell \neq \ell'$ we can directly deduce that codes are not right equivalent. If not, by uniqueness of the decomposition, there exists a matrix $A \in \text{Stab}_{\text{right}}(\mathcal{D}_{\text{Mat}})$, such that, up to re-indexing, we have $E_i = QAF_i A^{-1}Q^{-1}$. Therefore, we have the following right equivalences:

$$\forall i \in \{1, \ldots, \ell\}, \quad \mathcal{C}_{\text{Mat}} E_i = \mathcal{C}_{\text{Mat}} Q A F_i A^{-1} Q^{-1} = \mathcal{D}_{\text{Mat}} A F_i A^{-1} Q^{-1}.$$  

Next, since $A \in \text{Stab}_{\text{right}}(\mathcal{D}_{\text{Mat}})$ we deduce that

$$\mathcal{C}_{\text{Mat}} E_i = \mathcal{D}_{\text{Mat}} F_i A^{-1} Q^{-1}.$$  

Thus, up to re-indexing, the codes $\mathcal{C}_{\text{Mat}} E_i$ and $\mathcal{D}_{\text{Mat}} F_i$ are right equivalent. Here is the core of the algorithm. For any $i \in \{1, \ldots, \ell\}$, consider the code $\mathcal{C}_{\text{Mat}} E_i$ and check whether it is equivalent with one of the codes $\mathcal{D}_{\text{Mat}} F_j$. If $\ell = 1$ (we cannot decompose the identity), we can do it with Algorithm I as from Remark 22, stabilizer algebras are local. The case $\ell = 1$ corresponds to a base case. Otherwise we will make a recursive call to the previous procedure (after some puncturing that we describe in the next paragraph). From this “piecewise” equivalence, one can deduce a “global” equivalence as we will describe further.

It is worth noting that this recursive algorithm will terminate after a polynomial number of steps. It relies on the following observation. Suppose that in the decomposition of the identity we have $\ell > 1$, then row-support of $\mathcal{C}_{\text{Mat}} E_i$ (resp. $\mathcal{D}_{\text{Mat}} F_j$) will be strictly included in the row support of $\mathcal{C}_{\text{Mat}}$ (resp. $\mathcal{D}_{\text{Mat}}$). Indeed the rank of $E_i$ (resp. $F_j$) is $< n$ (the only idempotent of full rank is the identity). But as we make a recursive call to our procedure where we supposed that codes have a full support, we puncture them. It leads to a recursive call where matrix codes are $\subseteq \mathcal{M}_{m,n'}(\mathbb{F}_q)$ with $n' < n$. To summarize, in any recursive call we finish by stopping the algorithm (the base case $\ell = 1$ or $\ell \neq \ell'$) or we strictly decrease the number of columns of the matrix codes. Therefore, the last case to consider is when $n = 1$. In this case, stabilizer algebras are $\mathbb{F}_q$ which is a local algebra.

Suppose now that recursive calls are terminated. Then, either codes are not right equivalent or we succeeded to find $s_i \times s_i$ nonsingular matrices $B_1, \ldots, B_\ell$ such that

$$\forall i \in \{1, \ldots, \ell\}, \quad \mathcal{C}_{\text{Mat}}^{(i)} B_i = \mathcal{D}_{\text{Mat}}^{(i)},$$  

where $\mathcal{C}_{\text{Mat}}^{(i)}, \mathcal{D}_{\text{Mat}}^{(i)}$ are respective puncturing of $\mathcal{C}_{\text{Mat}} E_i$ and $\mathcal{D}_{\text{Mat}} F_i$.

Then, after a suitable change of bases (i.e. after replacing $\mathcal{C}_{\text{Mat}}, \mathcal{D}_{\text{Mat}}$ by right equivalent codes), the matrix

$$Q = \begin{pmatrix} B_1 & & \\ & B_2 & \\ & & \ddots \\ & & & B_\ell \end{pmatrix}$$  

satisfies $\mathcal{C}_{\text{Mat}} Q = \mathcal{D}_{\text{Mat}}$. This concludes the algorithm.

The algorithm can be summarized as

- Compute minimal decompositions of the identity in $\text{Stab}_{\text{right}}(\mathcal{C}_{\text{Mat}})$ and $\mathcal{D}_{\text{Mat}}$ and deduce decompositions

$$\mathcal{C}_{\text{Mat}} = \mathcal{C}_{\text{Mat}} E_1 \oplus \cdots \oplus \mathcal{C}_{\text{Mat}} E_\ell \quad \text{ and } \quad \mathcal{D}_{\text{Mat}} = \mathcal{D}_{\text{Mat}} F_1 \oplus \cdots \oplus \mathcal{D}_{\text{Mat}} F_\ell.$$  

- Set $I \overset{\text{def}}{=} \{1, \ldots, \ell\}$ and $J \overset{\text{def}}{=} \{1, \ldots, \ell\}$
for $i \in \{1, \ldots, \ell \}$ and $j \in \{1, \ldots, \ell \}$ check if $e_{\text{Mat}}E_i$ and $D_{\text{Mat}}F_j$ are right equivalent (recursive call). If they are, then $J \leftarrow I \setminus \{i\}$ and $J \leftarrow J \setminus \{j\}$.

- If we succeeded to associate the $e_{\text{Mat}}E_i$’s to equivalent $D_{\text{Mat}}F_j$’s, then we deduce the codes are equivalent and get the matrix $Q$. Else, the codes are not equivalent.

All our discussion about this algorithm leads to the following statement.

**Theorem 24.** — MCRE is in $\mathcal{P}$ if $q = (mn)^{O(1)}$ and in $\mathcal{ZP\mathcal{P}}$ in the general case.

5. The Code Equivalence Problem for $\mathbb{F}_q^m$–linear codes : an Easy Problem

When described as matrix codes, $\mathbb{F}_q^m$–linear codes are nothing but codes $\mathcal{C}_{\text{Mat}} \subseteq M_{m,n}(\mathbb{F}_q)$ whose left stabilizer algebra $\text{Stab}_{\text{left}}(\mathcal{C}_{\text{Mat}}) \subseteq M_n(\mathbb{F}_q)$ contains a subalgebra isomorphic to $\mathbb{F}_q^m$ as in the following definition.

**Definition 25 (Representation of $\mathbb{F}_q^m$).** — A sub-algebra $\mathcal{F} \subseteq M_m(\mathbb{F}_q)$ is a representation of $\mathbb{F}_q^m$ if it satisfies one of the following equivalent conditions:

- It is isomorphic to $\mathbb{F}_q^m$ as an $\mathbb{F}_q$–algebra;
- It spanned (as an algebra) by a matrix $A \in M_m(\mathbb{F}_q)$ whose characteristic polynomial is $\mathbb{F}_q$–irreducible.

In particular, taking any monic irreducible polynomial $P \in \mathbb{F}_q[X]$ and $C_P$ be its companion matrix, then $F_q[C_P]$ is a representation of $\mathbb{F}_q^m$.

5.1. Statement of equivalence problems for $\mathbb{F}_q^m$–linear codes. — For $\mathbb{F}_q^m$–linear codes, the code equivalence problem can be stated in two different manners, either by considering codewords represented by vectors of $\mathbb{F}_q^m$, or by $m \times n$ matrices with entries in $\mathbb{F}_q$. The major difference between these two formulations is that if the codes are given in the matrix representation, the choice of the $\mathbb{F}_q$–basis of $\mathbb{F}_q^m$ permitting to move from the vector to the matrix representation is not supposed to be known, in addition, the basis could be different for the two matrix codes for which we wish to decide the equivalence. Recall that it leads to the definition of these two problems.

**Problem 5 (Vector Matrix Code Equivalence Problem (V-MCE))**

- **Instance:** two vector codes $\mathcal{C}_{\text{Vec}}, \mathcal{D}_{\text{Vec}} \subseteq \mathbb{F}_q^m$;
- **Decision:** there exists a matrix $Q \in \text{GL}_m(\mathbb{F}_q)$ such that $\mathcal{C}_{\text{Vec}} = \mathcal{D}_{\text{Vec}}Q$.

**Problem 7 (Hidden Vector Matrix Code Equivalence Problem (HV-MCE))**

- **Instance:** $\mathcal{C}_{\text{Mat}}, \mathcal{D}_{\text{Mat}} \subseteq M_{m,n}(\mathbb{F}_q)$ be two spaces of matrices representing $\mathbb{F}_q^m$–linear codes
- **Decision:** It exists $(S, P) \in \text{GL}_m(\mathbb{F}_q) \times \text{GL}_n(\mathbb{F}_q)$ such that $\mathcal{C}_{\text{Mat}} = S\mathcal{D}_{\text{Mat}}P$.

**Caution.** When saying representing $\mathbb{F}_q^m$–linear codes, we mean that $\mathcal{C}_{\text{Mat}}, \mathcal{D}_{\text{Mat}}$ represent $\mathbb{F}_q^m$–linear codes $\mathcal{C}_{\text{Vec}}, \mathcal{D}_{\text{Vec}}$ for a given choice of an $\mathbb{F}_q$–basis of $\mathbb{F}_q^m$. We emphasize the two following facts:

(a) The $\mathbb{F}_q$–bases of $\mathbb{F}_q^m$ chosen to represent $\mathcal{C}_{\text{Vec}}$ and $\mathcal{D}_{\text{Vec}}$ as $\mathcal{C}_{\text{Mat}}, \mathcal{D}_{\text{Mat}}$ may be distinct;
(b) these bases are unknown.

Therefore, an instance of V-MCE is nothing but a particular instance of MCRE which has already been proved to be easy in the worst case. On the other hand, an instance of HV-MCE is an instance of MCE. However, as shown in the sequel, the $\mathbb{F}_q^m$–linear structure of the considered codes permits to solve the problem in polynomial time and actually to reduce to an instance of MCRE. In the sequel we focus on HV-MCE.

We start from two matrix codes $\mathcal{C}_{\text{Mat}}, \mathcal{D}_{\text{Mat}} \subseteq M_{m,n}(\mathbb{F}_q)$ of which are known to be matrix representations of $\mathbb{F}_q^m$–linear codes but possibly represented in different bases. To reduce to an instance of MCRE, our objective is to find a matrix $S \in \text{GL}_m(\mathbb{F}_q)$ such that $S\mathcal{C}_{\text{Mat}}$ is right
equivalent to $\mathcal{D}_{\text{Mat}}$. Such an $S$ will be deduced from the left stabilizer algebras of the codes. Recall that these left stabilizer algebras $\text{Stab}_{\text{left}}(\mathcal{C}_{\text{Mat}}), \text{Stab}_{\text{left}}(\mathcal{D}_{\text{Mat}})$ can be computed by solving a linear system in a very similar manner as the computation of right stabilizer algebras detailed in Section [4] (see Remark [12]). In addition, similarly to Proposition [13] we have that

$$\text{Stab}_{\text{left}}(\mathcal{C}_{\text{Mat}}) = S^{-1}\text{Stab}_{\text{left}}(\mathcal{D}_{\text{Mat}})S.$$ 

Thus, our objective is, from the knowledge of the stabilizer algebras $\text{Stab}_{\text{left}}(\mathcal{C}_{\text{Mat}})$ and $\text{Stab}_{\text{left}}(\mathcal{D}_{\text{Mat}})$, to solve an algebra conjugacy problem. It will be possible since these algebras contain a representation of $\mathbb{F}_{q^m}$, and hence are non-trivial. We treat the problem by considering separately two cases:

(i) The left stabilizer algebras of $\mathcal{C}_{\text{Mat}}, \mathcal{D}_{\text{Mat}}$ are representations of $\mathbb{F}_{q^m}$;

(ii) The left stabilizer algebras of $\mathcal{C}_{\text{Mat}}, \mathcal{D}_{\text{Mat}}$ contain representations of $\mathbb{F}_{q^m}$ as proper sub-algebras.

5.2. When the left stabilizer algebras are representations of $\mathbb{F}_{q^m}$. — In the sequel, we need some very classical results in linear algebra and representations of $\mathbb{F}_{q^m}$ which are summarized in the following technical statement.

**Lemma 26.** — Let $A \in \mathcal{M}_m(\mathbb{F}_q)$ be a matrix whose characteristic polynomial $\chi_A$ is irreducible (for instance the companion matrix of any monic irreducible polynomial of degree $m$). Then $\mathbb{F}_q[A] \cong \mathbb{F}_{q^m}$ and

(i) a matrix $C \in \mathcal{M}_m(\mathbb{F}_q)$ commutes with any element of $\mathbb{F}_q[A]$ if and only if it is an element of $\mathbb{F}_q[A]$;

(ii) for any $x, y \in \mathbb{F}_{q^m} \setminus \{0\}$ there exists $P(A) \in \mathbb{F}_q[A]$ such that $P(A)x^\top = y^\top$;

(iii) There exists $\Theta \in \text{GL}_m(\mathbb{F}_q)$ such that

$$\tilde{\Theta} = \Theta^m = I_m$$

and $\forall P(A) \in \mathbb{F}_q[A], \ 	ilde{\Theta}P(A)\tilde{\Theta}^{-1} = P(A)^q$.

(iv) Any matrix $\Gamma \in \text{GL}_m(\mathbb{F}_q)$ satisfying $\Gamma F_q[A] \Gamma^{-1} = F_q[A]$ is of the form $\Gamma = \Theta^j P(A)$ for some $P(A) \in \mathbb{F}_q[A]^\times$ and some $j \in \{0, \ldots, m - 1\}$.

(v) for any matrix $B \in \mathcal{M}_m(\mathbb{F}_q)$ with the same characteristic polynomial as $A$, then $A$ and $B$ are conjugated, i.e. there exists $P \in \text{GL}_m(\mathbb{F}_q)$ such that $A = P^{-1}BP$ and $P$ can be computed in polynomial time.

**Proof.** — Clearly $\mathbb{F}_{q^m}[A]$ is commutative. Conversely, if $C$ commutes with any element of $\mathbb{F}_{q^m}[A]$ then it commutes with $A$. Since the characteristic polynomial $\chi_A$ of $A$ is irreducible and $\mathbb{F}_q$ is perfect, then the roots of $\chi_A$ in an algebraic closure of $\mathbb{F}_q$ are distinct. Finally, it is a well-known fact in linear algebra that if a matrix $U$ has simple eigenvalues, then any matrix commuting with $U$ is a polynomial in $U$. This proves (i).

Consider the space $F_{q^m} \overset{\text{def}}{=} \{P(A)x^\top \mid P \in \mathbb{F}_q[X]\} \subseteq \mathbb{F}_{q^m}$. This space is isomorphic to $\mathbb{F}_q[X]/(\pi_{A, x})$ where $\pi_{A, x}$ is the minimal polynomial of $A$ at $x$, that is to say the monic polynomial $P$ of smallest degree satisfying $P(A)x^\top = 0$. This polynomial divides the minimal polynomial of $A$ which itself is known to divide $\chi_A$. Since, by hypothesis $\chi_A$ is irreducible, then $\pi_{A, x} = \chi_A$ and $	ext{dim}_{\mathbb{F}_q} F = \text{dim}_{\mathbb{F}_q} \mathbb{F}_q[X]/(\chi_A) = m$. Therefore, $F = \mathbb{F}_{q^m}$ and hence contains $y$, which proves (ii).

Using the fact that $\mathbb{F}_{q^m}$ is isomorphic as a vector space with $\mathbb{F}_{q^m}$, the matrices in $\mathbb{F}_q[A]$ represent the multiplications by elements of $\mathbb{F}_{q^m}$ in some given basis. Choosing $\Theta$ as a matrix representation of the Frobenius map

$$\begin{cases}
\mathbb{F}_{q^m} & \longrightarrow \mathbb{F}_{q^m} \\
x & \longmapsto x^q
\end{cases}$$

in this basis yields (iii).

Let $\Gamma$ be a matrix such that $\mathbb{F}_q[A]$ is globally invariant by conjugation under $\Gamma$. This conjugation map induces a non-trivial automorphism of the field $\mathbb{F}_q[A] = \mathbb{F}_{q^m}$, which is nothing but an iterate of the Frobenius. Therefore,

$$\forall P(A) \in \mathbb{F}_q[A], \quad \Gamma P(A)\Gamma^{-1} = P(A)^{q^j}.$$
for some \( j \in \{0, \ldots, m-1\} \). Now, \( \Gamma^{-1} \Theta^j \) commutes with any element of \( \mathbb{F}_q[A] \) and then (ii) can be deduced from (i).

Let \( x \in \mathbb{F}_q^n \setminus \{0\} \), reasoning as in the proof of (i), we see that \( (x^\top, Ax^\top, \ldots, A^{m-1}x^\top) \) is a basis of \( \mathbb{F}_q^n \). Let \( P_0 \) be the transition matrix from the canonical basis to this basis. This matrix can be computed in polynomial time and then, \( P_0^{-1}AP_0 \) is nothing but the companion matrix of \( \chi_A \). Similarly, one can compute \( P_1 \) such that \( P_1^{-1}BP_1 \) equals this companion matrix of \( \chi_A \). The matrix \( P \overset{def}{=} P_1P_0^{-1} \) yields (ii).

This technical lemma has the following consequence on the matrix representations of \( \mathbb{F}_q^m \).

**Corollary 27.** — Let \( S_1, S_2 \subseteq \mathcal{M}_m(\mathbb{F}_q) \) be two matrix representations of \( \mathbb{F}_q^m \). Then there exists a matrix \( P \in \text{GL}_m(\mathbb{F}_q) \) such that \( S_1 = P^{-1}S_2P \). In addition, \( P \) can be computed in deterministic polynomial time if \( q = m^{O(1)} \) otherwise with a polynomial time Las Vegas algorithm.

**Proof.** — First, compute matrices \( A_1, A_2 \in \mathcal{M}_m(\mathbb{F}_q) \) such that \( S_1 = \mathbb{F}_q[A_1] \) and \( S_2 = \mathbb{F}_q[A_2] \). Such matrices can be computed in polynomial time. Denoting by \( \chi_A \), \( \chi_B \) their respective characteristic polynomials we have

\[
S_1 = \mathbb{F}_q[A_1] \simeq \mathbb{F}_q[X]/(\chi_A) \quad \text{and} \quad S_2 = \mathbb{F}_q[A_2] \simeq \mathbb{F}_q[X]/(\chi_B).
\]

Let us compute (in deterministic polynomial time if \( q = m^{O(1)} \) otherwise with a polynomial time Las Vegas method) a root of \( \chi_A \) in \( \mathbb{F}_q[A_1] \simeq \mathbb{F}_q[X]/(\chi_A) \), noted \( P(A_1) \), then \( P(A_1) \) has the same characteristic polynomial as \( A_2 \) and, from Lemma 26(ii), there is a matrix \( P \) which can be computed in polynomial time such that, \( A_2 = P^{-1}P(A_1)P \). Moreover, for dimensional reasons, \( \mathbb{F}_q[A_1] = \mathbb{F}_q[P(A_1)] \) and hence

\[
S_2 = \mathbb{F}_q[A_2] = P^{-1}\mathbb{F}_q[A_1]P = P^{-1}S_1P.
\]

So, the first part of our algorithm will consist in computing the left stabilizer algebras of \( \mathcal{C}_{\text{Mat}} \) and \( \mathcal{D}_{\text{Mat}} \) (which are assumed to be representations of \( \mathbb{F}_q^m \)). Next, according to Corollary 27, the stabilizer algebras are conjugated and one can compute (in deterministic polynomial time if \( q = m^{O(1)} \) otherwise with a polynomial time Las Vegas method) \( P \in \text{GL}_m(\mathbb{F}_q) \) such that:

\[
\text{Stab}_{\text{left}}(\mathcal{D}_{\text{Mat}}) = P^{-1}\text{Stab}_{\text{left}}(\mathcal{C}_{\text{Mat}})P.
\]

Replacing \( \mathcal{D}_{\text{Mat}} \) by the left equivalent code \( P\mathcal{D}_{\text{Mat}} \), then the codes turn out to have the same left stabilizer algebra. Once we are reduced to the case where the two codes have the same left stabilizer algebra, the following statement asserts that we are almost reduced to solving and instance of MCRE.

**Proposition 28.** — Let \( \mathcal{C}_{\text{Mat}}, \mathcal{D}_{\text{Mat}} \subseteq \mathcal{M}_{m,n}(\mathbb{F}_q) \) be two left equivalent matrix codes whose left stabilizer algebras are equal and are representations of \( \mathbb{F}_q^m \), then there exists \( j \in \{0, \ldots, m-1\} \) such that

\[
\mathcal{C}_{\text{Mat}} = \Theta^j \mathcal{D}_{\text{Mat}}
\]

where \( \Theta \) is a matrix defined in 20(ii).

**Proof.** — The codes are supposed to be left equivalent, then \( \mathcal{C}_{\text{Mat}} = S\mathcal{D}_{\text{Mat}} \) for some \( S \in \text{GL}_m(\mathbb{F}_q) \). In addition, if they have the same left stabilizer algebra \( S \), which is supposed to be of the form \( \mathbb{F}_q[A] \simeq \mathbb{F}_q^m \) for some \( A \in \mathcal{M}_m(\mathbb{F}_q) \) with an irreducible characteristic polynomial. Then, from a “left version” of Proposition 14 \( S \) is stable by conjugation by \( S \). Thus, from Lemma 20(iii), we have

\[
S = \Theta^j P(A)
\]

for some \( j \in \{0, \ldots, m-1\} \) and \( P(A) \in \mathbb{F}_q[A]^* \). Therefore, since \( P(A) \in \mathcal{S} = \text{Stab}_{\text{left}}(\mathcal{D}_{\text{Mat}}) \), then \( S\mathcal{D}_{\text{Mat}} = \Theta^j \mathcal{D}_{\text{Mat}} \), which concludes the proof.
Corollary 29. — Let \( \mathcal{C}_{\text{Mat}}, \mathcal{D}_{\text{Mat}} \subseteq \mathcal{M}_{m,n}(\mathbb{F}_q) \) which are equivalent, i.e. there exists \( S \in \text{GL}_m(\mathbb{F}_q) \) and \( T \in \text{GL}_n(\mathbb{F}_q) \) such that \( \mathcal{C}_{\text{Mat}} = S \mathcal{D}_{\text{Mat}} T \). Suppose that the two codes have the same left stabilizer algebra which is a representation of \( \mathbb{F}_q^m \). Then, denoting by \( \Theta \) the matrix of \( \mathcal{D}_{\text{Mat}} \), \( \mathcal{C}_{\text{Mat}} \) is right equivalent to \( \Theta^j \mathcal{D}_{\text{Mat}} \) for some \( j \in \{0, \ldots, m - 1\} \).

Proof. — Apply Proposition 28 to the pair \( (\mathcal{C}_{\text{Mat}}, \mathcal{D}_{\text{Mat}} T) \).

These results yield to Algorithm 2 which decides HV-MCE in polynomial efficiently when left stabilizer algebras are equal to \( \mathbb{F}_q^m \).

Algorithm 2: An algorithm to decide the equivalence problem for \( \mathbb{F}_q^m \)-linear code with left stabilizer algebras equal to \( \mathbb{F}_q^m \).

Input : Two codes \( \mathcal{C}_{\text{Mat}}, \mathcal{D}_{\text{Mat}} \subseteq \text{Mat}_{m,n}(\mathbb{F}_q) \)
Output: Codes are equivalent or not

Compute the left stabilizer algebras of \( \mathcal{C}_{\text{Mat}}, \mathcal{D}_{\text{Mat}} \) and \( P \) such that they are conjugated under \( P \);
\( \mathcal{D}_{\text{Mat}} \leftarrow P \mathcal{D}_{\text{Mat}} \)
Compute a matrix \( \Theta \) as in Lemma 29[19]
for \( j \in \{1, \ldots, m\} \) do

Compute \( \Theta^j \mathcal{D}_{\text{Mat}} \)
if \( \mathcal{C}_{\text{Mat}} \) and \( \Theta^j \mathcal{D}_{\text{Mat}} \) are right equivalent then
return Codes are equivalent

return Codes are not equivalent

Theorem 30. — Let \( \mathcal{C}_{\text{Mat}}, \mathcal{D}_{\text{Mat}} \) be two spaces of matrices representing \( \mathbb{F}_q^m \)-linear codes and whose left stabilizer algebra is a representation of \( \mathbb{F}_q^m \). Algorithm 2 succeeds to decide if they are equivalent in deterministic polynomial time if \( q = (mn)^O(1) \) and otherwise with a polynomial time Las Vegas algorithm.

Proof. — The for loop is made at most \( m \) times, which is polynomial in the size of the input. Now all the computations are clearly in polynomial time except when:
- deciding if codes \( \mathcal{C}_{\text{Mat}} \) and \( \Theta^j \mathcal{D}_{\text{Mat}} \) are right-equivalent,
- computing \( P \)

The computation of \( P \) is done according to Proposition 27 thus in deterministic polynomial time if \( q = m^O(1) \) and otherwise with a polynomial time Las Vegas method. Furthermore, as we have shown in Theorem 7 we can decide if two matrix codes are right equivalent in deterministic polynomial time if \( q = (mn)^O(1) \) and otherwise with a polynomial time Las Vegas algorithm.

Let us show the correctness of the algorithm. After \( \mathcal{D}_{\text{Mat}} \leftarrow P \mathcal{D}_{\text{Mat}} \), codes \( \mathcal{C}_{\text{Mat}} \) and \( \mathcal{D}_{\text{Mat}} \) have both the same stabilizer algebra which is a representation of \( \mathbb{F}_q^m \). Therefore, according to Corollary 29 we know that if \( \mathcal{C}_{\text{Mat}} \) and \( \mathcal{D}_{\text{Mat}} \) are equivalent then \( \mathcal{C}_{\text{Mat}} \) is right equivalent to \( \Theta^j \mathcal{D}_{\text{Mat}} \) for some \( j \) which concludes the proof.

5.3. When the left stabilizer algebras strictly contain representations of \( \mathbb{F}_q^m \). — In the general case, it is possible that the left stabilizer algebras contain representations of \( \mathbb{F}_q^m \) as proper subspaces. To understand the way to proceed, we first have to classify sub-algebras of \( \mathcal{M}_m(\mathbb{F}_q) \) containing a representation of \( \mathbb{F}_q^m \), which is the point of the next statement.

Proposition 31. — Let \( S \subseteq \mathcal{M}_m(\mathbb{F}_q) \) be an algebra such that
\[
\mathbb{F}_q[A] \simeq \mathbb{F}_q^m \subseteq S \subseteq \mathcal{M}_m(\mathbb{F}_q),
\]
then, \( S \) is isomorphic to a matrix algebra of the form \( \mathcal{M}_{\ell}(\mathbb{F}_q) \) for some \( \ell \) dividing \( m \). In addition, any \( A \in \mathcal{M}_m(\mathbb{F}_q) \) which commutes with any element of the centre of \( S \), then \( A \in S \).
Proof. — First consider the centre of $S$ (i.e. the sub–algebra of elements that commute with any other one). From Lemma 23, any matrix commuting with an element of $F_q[A]$ is in $F_q[A]$. We deduce that the centre of $S$ is a subfield of $F_{q^m}$ and hence is of the form $F_{q^m}$ for some $\ell$ dividing $m$.

Second, let us prove that $S$ is semi–simple, namely that its Jacobson radical is trivial. Indeed, let $N \in S$ be a strongly nilpotent element of $S$, i.e. an element such that for any $M \in S$, $NM$ is nilpotent. Suppose that $N \neq 0$, then there exists $x \in F_{q^m}$ such that $Nx^T \neq 0$. Then, from Lemma 23, there exists $P(A) \in F_q[A]$ such that $P(A)Nx^T = x^T$. Thus, $P(A)N \in S$ and has a non-zero fixed point which contradicts its nilpotence. Therefore, $\text{Rad}(S) = \{0\}$ and hence $S$ is semi-simple. Finally, suppose that $S$ is not simple and hence splits in a direct product of simple algebras. Then its centre is not simple by ([DK94 Cor. 1.7.8]). On the other hand, we proved that the centre is a field and hence is simple, a contradiction. Therefore, $S$ is simple and, by using Wedderburn–Artin Theorem [DK94 Cor. 2.4.5], is isomorphic to a matrix algebra over its centre $F_{q^r}$, i.e. to $M_r(F_{q^r})$ for some positive $r$.

Now, denote by $Z(S) \simeq F_{q^r}$ the centre of $S$ and by $C$ the commutant of $Z(S)$ in $M_m(F_q)$, i.e.

$$C \overset{\text{def}}{=} \{M \in M_m(F_q) \mid \forall B \in Z(S), \ BM = MB\}.$$ 

Note that $S \subseteq C$ by definition of its centre $Z(S)$. We are going to show that $C \simeq \text{Mat}_m(F_{q^r})$. Then we will show that $S = C$ which will conclude the proof.

From [DK94 Thm. 4.4.6(1)], the algebra $C$ is simple, and, from [DK94 Thm. 4.4.6(3)],

$$\dim_{F_q}(C) \cdot \dim_{F_q}(Z(S)) = \dim_{F_q}(M_m(F_q)) \implies \dim_{F_q}(C) = \frac{m^2}{\ell}.$$ 

Next, since $S \subseteq C$ we have that $Z(C) \subseteq Z(S)$ while by definition of $C$, $Z(S) \subseteq Z(C)$. Therefore, $Z(S) = Z(C) \simeq F_{q^r}$ and hence $C$ is a simple $F_{q^r}$–algebra whose centre is isomorphic to $F_{q^r}$. Therefore, $C$ is isomorphic to $M_u(F_{q^r})$ for some $u$ which is equal, for dimensional reason (Equation (7)), to $\frac{m}{\ell}$. There remains to prove that $S = C$. Suppose $S \not\subseteq C$. Then,

$$\dim_{F_{q^r}}S < \dim_{F_{q^r}}C \implies r^2 < \left(\frac{m}{r}\right)^2 \implies r^2 < \frac{m}{\ell}.$$ 

On the other hand $S \simeq M_r(F_{q^r})$ contains a representation of $F_{q^m}$ and such a representation is isomorphic to a subalgebra of the form $F_q[A]$ for some $A \in M_r(F_{q^r})$ but such an algebra has an $F_{q^r}$–dimension bounded by the degree of the characteristic polynomial of $A$ which equals $r < \frac{m}{\ell}$, hence has $F_{q^r}$–dimension strictly less than $m$. A contradiction. This concludes the proof. □

With Proposition 31 in hand, we can deduce an algorithm for solving this general case. We describe it as follows.

1. Compute the left stabilizer algebras $\text{Stab}_{\text{left}}(\mathfrak{C}_{\text{Mat}})$, $\text{Stab}_{\text{left}}(\mathfrak{D}_{\text{Mat}})$;
2. Compute their centres $Z(\text{Stab}_{\text{left}}(\mathfrak{C}_{\text{Mat}}))$, $Z(\text{Stab}_{\text{left}}(\mathfrak{D}_{\text{Mat}}))$ which are isometric to a same $F_{q^\ell}$ for some $\ell > 0$, otherwise codes are not equivalent;
3. These algebras are respectively isomorphic to $F_{q}[A_1], F_{q}[A_2]$ for some $A_1, A_2 \in M_m(F_q)$ whose minimal polynomial is irreducible of degree $\ell$. A very similar argument as Corollary 29 permits to assert that the algebras are conjugated and a matrix $P \in \text{GL}_m(F_q)$ such that $F_{q}[A_1] = F_{q}[A_2]P^{-1}$;
4. Replacing $\mathfrak{D}_{\text{Mat}}$ by $P \mathfrak{D}_{\text{Mat}}P^{-1}$ so that the stabilizer algebras of the codes have the same centre $Z$. But, from 31 the stabilizer algebras are then the same since they both equal to the commutator of $Z$ in $M_m(F_q)$;
5. Now, since the two codes have the same left stabilizer algebra and this stabilizer algebra contains a representation of $F_{q^m}$, such a representation can be explicitly computed using the [Kön90] which yields an explicit isomorphism with this stabilizer algebra and $M_{\frac{m}{\ell}}(F_{q^r})$. Via this explicit isomorphism it suffices to compute a representation of $F_q^{m}$ in $M_{\frac{m}{\ell}}(F_{q^r})$, which can be done by considering the algebra spanned by the companion matrix of any monic irreducible polynomial of degree $\frac{m}{\ell}$ in $F_{q^r}[X]$. 


6. Now both $C_{\text{Mat}}, D_{\text{Mat}}$ have a same representation of $\mathbb{F}_q^m$ which stabilizes them on the left and an argument very similar to that of the proof of Proposition 28 and Corollary 29 shows that if $C_{\text{Mat}}, D_{\text{Mat}}$ are equivalent, then $C_{\text{Mat}}, \Theta^j D_{\text{Mat}}$ are right equivalent for some $j$, where $\Theta$ is defined in Lemma 26(iii).

7. In summary, as in the previous case, one can decide equivalence by enumerating the pairs $(C_{\text{Mat}}, \Theta^j D_{\text{Mat}})$ for $j \in \{1, \ldots, m\}$ and check if one of them is a pair of right equivalent codes using the algorithm of Section 4.

This enable to prove the following theorem.

**Theorem 32.** — HV-MCE is in $\mathcal{P}$ if $q = (mn)^{O(1)}$ and in $\mathcal{ZPP}$ in the general case.

---

6. **Reduction from the Code Equivalence Problem in Hamming Metric**

The objective of the present section is to prove that the Matrix Code Equivalence Problem MCE (Problem 4) for rank metric codes is at least as hard as the Monomial Equivalence Problem ME (Problem 2), i.e. we will prove that if we have a polynomial time algorithm solving any instance of MCE then, there is a polynomial time algorithm solving any instance of ME.

In summary, we prove in the sequel the following statement.

**Theorem 33.** — The problem ME reduces in polynomial time to MCE.

For this sake we start from an instance $C_{\text{Vec}}, D_{\text{Vec}} \subseteq \mathbb{F}_q^n$ of ME and consider generator matrices $A, B \in M_{k,n}(\mathbb{F}_q)$ of $C_{\text{Vec}}, D_{\text{Vec}}$ respectively. From the generator matrices point of view, Problem ME may be reformulated as follows:

- **Instance:** Two matrices $A, B \in M_{k,n}(\mathbb{F}_q)$.
- **Decision:** it exists $S \in \text{GL}_k(\mathbb{F}_q), P \in M_n(\mathbb{F}_q)$ be a permutation matrix and a diagonal matrix $D \in \text{GL}_n(\mathbb{F}_q)$ such that:

$$A = SBDP.\]

Let $a_1^T, \ldots, a_n^T, b_1^T, \ldots, b_n^T \in \mathbb{F}_q^k$ denote the columns of $A$ and $B$ respectively. Similarly to the previous section, we keep on considering that vectors are row matrices, which explain why we apply the transposition operator to these column vectors.

Without loss of generality we can assume the following hypothesis:

**Hypothesis 1.** —

1. Each column of $A$ and $B$ is non-zero,
2. Columns of $A$ and $B$ are pairwise linearly independent.

We easily detect if there are columns of $A$ and $B$ which are equal to 0. We can assume that both matrices $A$ and $B$ have the same number of such columns, otherwise we already know they don’t define codes which are monomially equivalent. We can remove these columns without changing the answer of Problem ME with the instance $(C_{\text{Vec}}, D_{\text{Vec}})$. Now, the same kind of reasoning shows that we can assume the second point of the hypothesis. Indeed, the set of pairs of collinear columns in a matrix can be identified in polynomial time since the overall set of pairs of columns has size $O(n^2)$.

Next, if the two matrices do not have the same number of such pairs, then the corresponding codes are not monomially equivalent. If not, one can arbitrarily remove one element from each such pair without changing the answer of the problem.

Let us now define for any vector $x \in \mathbb{F}_q^n$ and any $1 \leq i \leq n$ the $n \times k$ matrix $\text{Row}_i(x)$ over $\mathbb{F}_q$ whose only non-zero row is the $i$–th one which equals $x$:

$$\text{Row}_i(x) \overset{\text{def}}{=} \begin{pmatrix} 0 & \cdots & 0 & x & 0 & \cdots & 0 \end{pmatrix}^T_i.$$
We now build in polynomial time in the size of $A$ and $B$ the following matrix codes of $\mathcal{M}_{k+n,k}(F_q)$:

\[
\mathcal{G}_{\text{Mat}} \overset{\text{def}}{=} \left\{ \sum_{i=1}^{n} \lambda_i \begin{pmatrix} a_i^\top a_i \\ \text{Row}_i(a_i) \end{pmatrix} : \lambda_i \in F_q \right\},
\]

\[
\mathcal{P}_{\text{Mat}} \overset{\text{def}}{=} \left\{ \sum_{i=1}^{n} \lambda_i \begin{pmatrix} b_i^\top b_i \\ \text{Row}_i(b_i) \end{pmatrix} : \lambda_i \in F_q \right\}.
\]

The following lemma is crucial for our reduction, it justifies our construction of $\mathcal{G}_{\text{Mat}}$ and $\mathcal{P}_{\text{Mat}}$.

**Lemma 34.** — Let $U \in \text{GL}_{k+n}(F_q)$ and $V \in \text{GL}_k(F_q)$ which verify:

\[
\mathcal{G}_{\text{Mat}} = U \mathcal{P}_{\text{Mat}} V.
\]

Then, for any $1 \leq i \leq n$, it exists $1 \leq \sigma(i) \leq n$ and $\alpha_{\sigma(i)} \in F_q^\times$ such that:

\[
\begin{pmatrix} a_i^\top a_i \\ \text{Row}_i(a_i) \end{pmatrix} = \alpha_{\sigma(i)} U \begin{pmatrix} b_{\sigma(i)}^\top b_{\sigma(i)} \\ \text{Row}_{\sigma(i)}(b_{\sigma(i)}) \end{pmatrix} V.
\]

Furthermore, the map $i \mapsto \sigma(i)$ is a permutation of $\{1, \ldots, n\}$.

**Proof.** — By definition, for any $1 \leq i \leq n$ there exist $\alpha_1, \ldots, \alpha_n \in F_q$ such that:

\[
\begin{pmatrix} a_i^\top a_i \\ \text{Row}_i(a_i) \end{pmatrix} = \sum_{j=1}^{n} \alpha_j U \begin{pmatrix} b_j^\top b_j \\ \text{Row}_j(b_j) \end{pmatrix} V = U \left( \sum_{j=1}^{n} \alpha_j \begin{pmatrix} b_j^\top b_j \\ \text{Row}_j(b_j) \end{pmatrix} \right) V.
\]

But now, since $U$ and $V$ are non-singular,

\[
\text{Rk} \left( U \left( \sum_{j=1}^{n} \alpha_j \begin{pmatrix} b_j^\top b_j \\ \text{Row}_j(b_j) \end{pmatrix} \right) V \right) = \text{Rk} \left( \sum_{j=1}^{n} \alpha_j \begin{pmatrix} b_j^\top b_j \\ \text{Row}_j(b_j) \end{pmatrix} \right) V.
\]

Recall that the rows of $\text{Row}_j(b_j)$ are all zero but the $j$–th one which equals $b_j$. Here, by Hypothesis $[\text{H}]$, the $b_j$'s are non-zero and pairwise linearly independent. Therefore, if there were at least two non zero $\alpha_j$’s, then we would have:

\[
\text{Rk} \left( \sum_{j=1}^{n} \alpha_j \begin{pmatrix} b_j^\top b_j \\ \text{Row}_j(b_j) \end{pmatrix} \right) \geq 2.
\]

A contradiction since $\begin{pmatrix} a_i^\top a_i \\ \text{Row}_i(a_i) \end{pmatrix}$ has rank 1, which proves the first part of the lemma.

There remains to prove that the map $i \mapsto \sigma(i)$ is a permutation of $\{1, \cdots, n\}$. Suppose that

\[
\begin{pmatrix} a_i^\top a_i \\ \text{Row}_i(a_i) \end{pmatrix} = \alpha_u U \begin{pmatrix} b_u^\top b_u \\ \text{Row}_u(b_u) \end{pmatrix} V = \alpha_v U \begin{pmatrix} b_v^\top b_v \\ \text{Row}_v(b_v) \end{pmatrix} V
\]

for two distinct non-zero $\alpha_u$ and $\alpha_v$. Since $U$ and $V$ are non-singular, the previous equality implies that:

\[
\alpha_u \begin{pmatrix} b_u^\top b_u \\ \text{Row}_u(b_u) \end{pmatrix} = \alpha_v \begin{pmatrix} b_v^\top b_v \\ \text{Row}_v(b_v) \end{pmatrix}.
\]

By definition, the rows of $\text{Row}_u(b_u)$ (resp. $\text{Row}_v(b_v)$) are all zero but the $u$–th (resp. $v$–th) one since $b_u$ (resp. $b_v$) is non-zero by hypothesis. Therefore $u = v$ and $i \mapsto \sigma(i)$ is one to one over $\{1, \cdots, n\}$, i.e: is a permutation. This concludes the proof. 

Let us now prove that $(\mathcal{G}_{\text{Vec}}, \mathcal{P}_{\text{Vec}})$ is a positive instance of ME if and only if $(\mathcal{G}_{\text{Mat}}, \mathcal{P}_{\text{Mat}})$ is a positive instance of MCE.
Lemma 35

By Lemma 34, for any code $\mathcal{C}$ of dimension $n$, it exists by definition $S \in \text{GL}_n(F_q)$, $P \in \text{GL}_n(F_q)$ a permutation matrix and $D \in \text{GL}_n(F_q)$ a diagonal matrix such that:

$$A = SBDP.$$  

Denote by $a_1, \ldots, a_n \in F_q^*$ the diagonal entries of $D$. Then, for any $1 \leq i \leq n$, it exists $1 \leq j \leq n$ (image of $i$ by the permutation given by $P$) such that:

$$a_i^\top = a_j S b_j^\top \quad \text{and} \quad a_i^\top a_i = a_j^2 S b_j^\top b_j S^\top$$

This gives:

$$\mathcal{C}_{\text{Mat}} = U D_{\text{Mat}} V,$$

where

$$U \triangleq \begin{pmatrix} S & 0 \\ 0 & P \end{pmatrix} \in \mathcal{M}_{k+n}(F_q) \quad \text{and} \quad V \triangleq S^\top \in \mathcal{M}_k(F_q),$$

which are nonsingular matrices. Therefore, $(\mathcal{C}_{\text{Mat}}, D_{\text{Mat}})$ is a positive instance of MCE.

6.2. If $(\mathcal{C}_{\text{Vec}}, \mathcal{D}_{\text{Vec}})$ is a positive instance of MCE then $(\mathcal{C}_{\text{Mat}}, D_{\text{Mat}})$ is a positive instance of ME. — By definition, there exist two non-singular matrices $U$ and $V$ such that:

$$\mathcal{C}_{\text{Mat}} = U D_{\text{Mat}} V.$$  

By Lemma 34 for any $1 \leq i \leq n$, it exists $1 \leq \sigma(i) \leq n$ and $a_{\sigma(i)} \in F_q^*$ such that:

$$\begin{pmatrix} a_i^\top a_i \\ \text{Row}_i(a_i) \end{pmatrix} = a_{\sigma(i)} U \begin{pmatrix} b_{\sigma(i)}^\top b_{\sigma(i)} \\ \text{Row}_{\sigma(i)}(b_{\sigma(i)}) \end{pmatrix} V = a_{\sigma(i)} U \begin{pmatrix} b_{\sigma(i)}^\top b_{\sigma(i)} V \\ \text{Row}_{\sigma(i)}(b_{\sigma(i)} V) \end{pmatrix}.$$  

(8)

The rows $b_{\sigma(i)} V$ and $a_i$ define non-zero linear forms as, by Hypothesis $b_{\sigma(i)}$ and $a_i$ are non-zero and $V$ is non-singular. We are going to prove that they are collinear. To this end we use the following classical lemma.

Lemma 35. — Let $\varphi, \psi : F_q^n \to F_q$ be two non-zero linear forms, then

$$\text{Ker}(\varphi) = \text{Ker}(\psi) \iff \exists \lambda \in F_q^* \quad \varphi = \lambda \psi.$$  

Let us prove that $\text{Ker}(b_{\sigma(i)} V) = \text{Ker}(a_i)$ for any $i$. These spaces have the same dimension as they are defined by non-zero linear forms. In this way it is enough to prove the inclusion $\text{Ker}(a_i) \subseteq \text{Ker}(b_{\sigma(i)} V)$. Let $x^\top \in \text{Ker}(a_i)$. Let us denote by $e_1, \ldots, e_n$ the canonical basis of $F_q^n$. By right multiplying $x^\top$ by $x^\top$, we get:

$$\begin{pmatrix} a_i^\top a_i x^\top \\ (a_i x^\top) e_i^\top \end{pmatrix} = a_{\sigma(i)} U \begin{pmatrix} b_{\sigma(i)}^\top b_{\sigma(i)} V x^\top \\ \text{Row}_{\sigma(i)}(b_{\sigma(i)} V x^\top) e_{\sigma(i)}^\top \end{pmatrix}.$$  

Here by hypothesis $a_i x^\top = 0$. Thus, as $U$ is non-singular and $a_{\sigma(i)} \neq 0$ we have that $b_{\sigma(i)}^\top V x = 0$ i.e. $x \in \text{Ker}(b_{\sigma(i)} V)$. Finally, by applying Lemma 35 to $a_i$ and $b_{\sigma(i)} V$, there exists for any $i$ some $\lambda_{\sigma(i)} \in F_q^*$ such that

$$a_i = \lambda_{\sigma(i)} b_{\sigma(i)} V.$$  

Then, if $P$ denotes the permutation matrix associated to $i \mapsto \sigma(i)$ (it is a permutation as shown in Lemma 34) one can verify that we have:

$$A = V^\top BDP$$

where $D$ is the diagonal matrix whose diagonal entries are $\lambda_1, \ldots, \lambda_n$. Therefore, as $V$ is non-singular, the pair of codes $(\mathcal{C}_{\text{Vec}}, \mathcal{D}_{\text{Vec}})$ with respective generator matrices $(A, B)$ is a positive instance of ME. This concludes the proof of Theorem 33.
Conclusion

This work has presented the equivalence problem of matrix codes endowed with the rank metric as well as some of its natural variants. Our contribution is threefold. First, as summarized in Figure 2, we have shown that the code equivalence problem for matrix codes (MCE) is harder than the monomial equivalence problem (ME). This last problem has been pursued for many years and works on it tend to show its hardness. Thus, the equivalence problem for matrix codes is not likely to be easy and could for instance be considered for cryptographic applications. On the other hand, if one restricts instances in the equivalence problem to $\mathbb{F}_q$-linear codes (particular matrix codes that are appealing for cryptographic applications) or only considers the right-equivalence then it leads to an easy problem. Using algorithms from the mathematical field of algebras we have shown that these problems fall down in $\mathcal{P}$ or $\mathcal{ZPP}$ depending of the ground field size $q$. To our opinion, this work may have extensions that are of interest.
Classification of matrix codes. Our algorithm to decide if two matrix codes are right equivalent takes advantage of the fact that right-equivalence only shifts (multiplicatively) right stabilizer algebras of codes by a non-singular matrix. In this way, stabilizer algebras share, roughly speaking, the same algebraic structure and as we have shown, by decomposing them we can decide if two codes are right equivalent. Interestingly, our algorithm decomposes right stabilizer algebras as “minimal atoms”. Then it punctures each code according to these atoms and tries to put them in correspondence.

This may remind techniques that are used in Support Splitting approaches [Leo82, Sen00]. It consists for two codes endowed with the Hamming metric to puncture them and to look if they both satisfy the same “invariants”. In [Leo82] is proposed to look at the weight distribution of the punctured codes while in [Sen00] it is this distribution but for hulls. In this light, decomposition of right stabilizer algebras can be seen as a good invariant for matrix codes. By good we mean that it is an efficient tool to solve equivalence problems. On the other hand, such invariant remains weakly discriminant since for arbitrary codes, the stabilizer algebra will be trivial.

This decomposition of stabilizer algebras could for instance be used to classify matrix codes. Furthermore, stabilizer algebras may help us to identify some properties that could be useful for decoding algorithms. This would be particularly interesting as we know very few families of matrix codes that we can decode efficiently. Except simple codes [SKK10] (that have a trivial structure), all matrix codes with an efficient decoding algorithm are $F_{q^m}$-linear [Gab85, GMRZ13] which exactly corresponds to the case where codes have a rich left stabilizer algebraic structure (that we use to decide the equivalence of these codes).

Finally, it should be emphasized that in the present article, we made the choice to work only over finite fields, which permits a simpler treatment since any central simple algebra is isomorphic to a matrix algebra in this setting (trivial Brauer group). However, some literature exists on codes over infinite fields [Aug14, ALR13, ALR18, ACLN20] and the similar question of solving code equivalence over arbitrary fields makes sense and is left as an open question. Such a treatment will probably represent a slightly harder task while involving non trivial central simple algebras.

Our Reduction. We can deduce from our reduction that any algorithm solving a “particular” instance of the matrix code equivalence problem (codes are generated by matrices of rank one) provides an algorithm solving the monomial equivalence problem. Therefore, this reduction gives a new manner to solve the monomial equivalence problem. For instance we could model the matrix equivalence problem into a system of polynomial equations that we would solve with Gröbner bases techniques. This approach, thanks to our reduction, would give a new modelling of the monomial equivalence problem. It may be interesting to study the complexity of this approach as Gröbner and to compare of with the results of [ST17] where permutation equivalence of codes is treating by solving a system of polynomial equations.

References

[AAB+17] Carlos Aguilar Melchor, Nicolas Aragon, Slim Bettaieb, Loïc Bidoux, Olivier Blazy, Jean-Christophe Deneuville, Philippe Gaborit, and Gilles Zémor. Rank quasi cyclic (RQC). First round submission to the NIST post-quantum cryptography call, November 2017.

[ABD+19] Nicolas Aragon, Olivier Blazy, Jean-Christophe Deneuville, Philippe Gaborit, Adrien Hauteville, Olivier Ruatta, Jean-Pierre Tillich, Gilles Zémor, Carlos Aguilar Melchor, Slim Bettaieb, Loïc Bidoux, Bardet Magali, and Ayoub Otmami. ROLLO (merger of Rank-Ourboros, LAKE and LOCKER). Second round submission to the NIST post-quantum cryptography call, March 2019.

[ABG+19] Nicolas Aragon, Olivier Blazy, Philippe Gaborit, Adrien Hauteville, and Gilles Zémor. Durandal: a rank metric based signature scheme. In Advances in Cryptology - EUROCRYPT 2019 - 38th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Darmstadt, Germany, May 19-23, 2019, Proceedings, Part III, volume 11478 of LNCS, pages 728–758. Springer, 2019.
[ACLN20] Daniel Augot, Alain Couvreur, Julien Lavauzelle, and Alessandro Neri. Rank-metric codes over arbitrary Galois extensions and rank analogues of Reed–Muller codes. [arXiv:2006.14189] June 2020.

[ALR13] Daniel Augot, Pierre Loidreau, and Gwezheneg Robert. Rank metric and Gabidulin codes in characteristic zero. In 2013 IEEE International Symposium on Information Theory, pages 509–513. IEEE, 2013.

[ALR18] Daniel Augot, Pierre Loidreau, and Gwezheneg Robert. Generalized Gabidulin codes over fields of any characteristic. Des., Codes Cryptogr., 86(8):1807–1848, 2018.

[Aug14] Daniel Augot. Generalization of Gabidulin codes over fields of rational functions. In 21st International Symposium on Mathematical Theory of Networks and Systems (MTNS 2014), 2014.

[Ber68] Elwyn R. Berlekamp. Factoring polynomials over finite fields. In E. R. Berlekamp, editor, Algebraic Coding Theory, chapter 6. McGraw-Hill, 1968.

[Ber70] Elwyn R. Berlekamp. Factoring polynomials over large finite fields. Math. Comp., 24:713–735, 1970.

[Ber03] Thierry P. Berger. Isometries for rank distance and permutation group of Gabidulin codes. IEEE Trans. Inform. Theory, 49(11):3016–3019, 2003.

[BG13] Aleams Barra and Heide Gluesing-Luerssen. MacWilliams extension theorems and the local-global property for codes over rings. Corr, abs/1307.7159, 2013.

[BGK17] Thierry P. Berger, Cheikh Thiécoumba Gueye, and Jean Belo Klamti. A NP-complete problem in coding theory with application to code based cryptography. In Codes, Cryptology and Information Security - Second International Conference, C2SI 2017, Rabat, Morocco, April 10-12, 2017, Proceedings - In Honor of Claude Carlet, pages 230–237, 2017.

[BOS19] Magali Bardet, Ayoub Otmani, and Mohamed Saeed-Taha. Permutation code equivalence is not harder than graph isomorphism when hulls are trivial. In Proc. IEEE Int. Symposium Inf. Theory - ISIT 2019, pages 2464–2468, July 2019.

[BR17] Eimear Byrne and Alberto Ravagnani. Covering radius of matrix codes endowed with the rank metric. SIAM J. Discret. Math., 31(2):927–944, 2017.

[Bre11] Murray R. Bremner. How to compute the Wedderburn decomposition of a finite-dimensional associative algebra. Groups Complex. Cryptol., 3(1):47–66, 2011.

[CZ81] David G. Cantor and Hans Zassenhaus. A new algorithm for factoring polynomials over finite fields. Math. Comp., pages 587–592, 1981.

[Del78] Philippe Delsarte. Bilinear forms over a finite field, with applications to coding theory. J. Comb. Theory, Ser. A, 25(3):226–241, 1978.

[DK94] Yuri A. Drozd and Vladimir V. Kirichenko. Finite dimensional algebras. Springer–Verlag Berlin Heidelberg, 1994. Original Russian edition published by: Publisher of Kiev State University, Kiev 1980, Translated by V. Drab.

[Feu09] Thomas Feulner. The automorphism groups of linear codes and canonical representatives of their semilinear isometry classes. Adv. Math. Commun., 3(4):363–383, 2009.

[FR85] Katalin Friedl and Lajos Rónyai. Polynomial time solutions of some problems of computational algebra. In Proceedings of the Seventeenth Annual ACM Symposium on Theory of Computing, STOC ’85, pages 153–162, New York, NY, USA, 1985. Association for Computing Machinery.

[Gab85] Ernest M. Gabidulin. Theory of codes with maximum rank distance. Problemy Peredachi Informatsii, 21(1):3–16, 1985.

[Gab08] Ernst M. Gabidulin. Attacks and counter-attacks on the GPT public key cryptosystem. Des. Codes Cryptogr., 48(2):171–177, 2008.

[GMRZ13] Philippe Gaborit, Gaëtan Murat, Olivier Ruatta, and Gilles Zémor. Low rank parity check codes and their application to cryptography. In Proceedings of the Workshop on Coding and Cryptography WCC’2013, Bergen, Norway, 2013.

[GO01] Ernst M. Gabidulin and Alexei V. Ourivski. Modified GPT PKC with right scrambler. Electron. Notes Discrete Math., 6:168–177, 2001.

[Gor19] Elisa Gorla. Rank–metric codes. [arXiv:1902.02650]. To appear in “A Concise Encyclopedia of Coding Theory”, W.C. Huffman, J.-L. Kim, and P. Sole’ Eds., CRC Press, 2019.

[GPT91] Ernst M. Gabidulin, A. V. Paramonov, and O. V. Tretjakov. Ideals over a non-commutative ring and their applications to cryptography. In Advances in Cryptology - EUROCRYPT’91, number 547 in LNCS, pages 482–489, Brighton, April 1991.
[GRSZ14] Philippe Gaborit, Olivier Ruatta, Julien Schrek, and Gilles Zémor. Ranksign: An efficient signature algorithm based on the rank metric (extended version on arxiv). In Post-Quantum Cryptography 2014, volume 8772 of LNCS, pages 88–107. Springer, 2014.

[HZ19] Luca Giuzzi and Ferdinando Zullo. Identifiers for MRD-codes. Linear Algebra Appl., 575:66–86, 2019.

[HP03] W. Cary Huffman and Vera Pless. Fundamentals of error-correcting codes. Cambridge University Press, Cambridge, 2003.

[HR14] Ishay Haviv and Oded Regev. On the lattice isomorphism problem. In Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA ’14, pages 391–404, USA, 2014. Society for Industrial and Applied Mathematics.

[HTM17] Anna-Lena Horlemann-Trautmann and Kyle Marshall. New criteria for MRD and Gabidulin codes and some rank-metric code constructions. Adv. Math. Commun., 11(3):533–548, 2017.

[Hua51] Loo-Keng Hua. A theorem on matrices over a field and its applications. J. Chinese Math. Soc., 1(2):109–163, 1951.

[Leo82] Jeffrey Leon. Computing automorphism groups of error-correcting codes. IEEE Trans. Inform. Theory, 28(3):496–511, 1982.

[Mac62] Florence J. MacWilliams. Combinatorial properties of elementary abelian groups. PhD thesis, Radcliffe College, 1962.

[Mor14] Katherine Morrison. Equivalence for rank-metric and matrix codes and automorphism groups of gabidulin codes. IEEE Trans. Inform. Theory, 60(11):7035–7046, 2014.

[NPHT20] Alessandro Neri, Sven Puchinger, and Anna-Lenna Horlemann-Trautmann. Equivalence and characterizations of linear rank-metric codes based on invariants. Linear Algebra and its Applications, pages 418–469, 2020.

[Ore33] Oystein Ore. On a special class of polynomials. Trans. Amer. Math. Soc., 35(3):559–584, 1933.

[PR97] Érez Petrank and Ron. Roth. Is code equivalence easy to decide? IEEE Trans. Inform. Theory, 43(5):1602–1604, 1997.

[Rón90] Lajos Rónyai. Computing the structure of finite algebras. J. Symbolic Comput., 9(3):355–373, 1990.

[Sen97] Nicolas Sendrier. On the dimension of the hull. SIAM J. Discrete Math., 10(2):282–293, 1997.

[Sen00] Nicolas Sendrier. Finding the permutation between equivalent linear codes: The support splitting algorithm. IEEE Trans. Inform. Theory, 46(4):1193–1203, 2000.

[She19] John Sheekey. MRD codes: constructions and connections, pages 255–286. De Gruyter, Berlin, Boston, 2019.

[SKK10] Danilo Silva, Frank R. Kschischang, and Ralf Kötter. Communication over finite-field matrix channels. IEEE Trans. Information Theory, 56(3):1296–1305, 2010.

[SS13] Nicolas Sendrier and Dimitris E. Simos. The hardness of code equivalence over and its application to code-based cryptography. In Post-Quantum Cryptography 2013, volume 7932 of LNCS, pages 203–216. Springer, 2013.

[ST17] Mohamed Ahmed Saeed-Taha. Algebraic Approach for Code Equivalence. PhD thesis, Normandy University, France, 2017.