Contents lists available at ScienceDirect

Finite Fields and Their Applications

journal homepage: www.elsevier.com/locate/ffa

The syndromes decoding algorithm in group codes

Consuelo Martínez [*], Fabián Molina [*]

Departamento de Matemáticas, Universidad de Oviedo, Spain

ARTICLE INFO

Article history:
Received 29 November 2022
Received in revised form 28 January 2023
Accepted 9 March 2023
Available online 14 April 2023
Communicated by Gary L. Mullen

MSC:
11T71

Keywords:
Linear code
Decoding algorithm
Group algebra
Semisimple algebra
Group code

ABSTRACT

Our aim is the design of an efficient decoding algorithm in group codes. The algorithm is inspired by the well known syndrome decoding algorithm for linear codes and uses the decomposition of a semisimple group algebra $\mathbb{K}G$ as a direct sum of two-sided ideals, each of them generated by a central idempotent of $\mathbb{K}G$. When G is an abelian group the algorithm can be modified to make it very simple and efficient. Some illustrative examples are presented.

1. Introduction

Error correcting codes play a key role in guaranteeing the reliability of the information sent through a noisy channel. The algorithms for detection and correction of errors aim to recover the original sent message. This process, called decoding, is efficiently performed thanks to the existing error correcting codes.

On the other hand, it is possible to design cryptosystems based on error correcting codes, which in principle are resistant to quantum computing. In fact, cryptography

[*] Corresponding authors.
E-mail addresses: cmartinez@uniovi.es (C. Martínez), molinafabian@uniovi.es (F. Molina).

https://doi.org/10.1016/j.ffa.2023.102206

based on error correcting codes is postulated as one of the most promising post-quantum cryptographic methods. In it, a "good" error correcting code is the key in a public key cryptographic scheme.

The first system of this type was proposed in [23] by R. J. McEliece in 1978. This proposal uses Goppa codes that are hidden under a linear code. This linear code, that looks arbitrary is the public key. In order to decrypt the message, one should know how to decode using a linear code, what is known to be a key hard problem. McEliece's proposal remains unbroken until now. Although the encryption and decryption processes are simple, the main problem with this scheme is the large size of the public keys.

Successive proposals to solve this problem suggested the replacement of Goppa codes by more structured codes such as Generalized Reed-Solomon Codes ([1,3,25,30]), Binary Reed-Muller Codes ([28]), Algebraic-Geometric Codes ([21]) or LDPC Codes ([2,29]). However, all of them have been broken (see [8,9,12,24,27,31]). Another alternative, is the use of codes having a non-trivial permutation automorphism group ([4,22,26]). However, the knowledge of the permutation automorphism group of a code allows some attacks by reducing the degrees and the number of variables of the algebraic system to solve. So, these new proposals have also been totally or partially broken in [13].

In this paper, we explore the use of group codes in cryptography. These codes are linked to (two-sided) ideals of the group algebra $\mathbb{K}G$, where G is a finite group of order n and n is the length of the code. So, sometimes, we will write the elements of $\mathbb{K}G$ as $n-$tuples of elements of the field $\mathbb{K}$, assuming a fixed order for the elements in G. Let us remember that a $(n, k, d)-$linear code $\mathfrak{C}$ is a $G-$code, if there is an isomorphism of $\mathbb{K}-$vector spaces $\phi : \mathbb{K}^n \to \mathbb{K}G$, satisfying that $\phi(\mathfrak{C})$ is a two-sided ideal of $\mathbb{K}G$.

A linear code $\mathfrak{C}$ is called (abelian) group code, if there exists a (abelian) group G such that $\mathfrak{C}$ is a $G-$code. Linear codes that are (abelian) group codes were characterized by Bernal et al. ([5]) in terms of their permutation automorphism group. Notice that a group code can be realized as $G-$code for different groups, and some of them may be abelian and some non-abelian. In [14–17] the question of the existence of non-abelian group codes, that is, group codes that can't be realized as abelian group codes, was addressed. Authors proved that given a group G, if there exists a non-abelian $G-$code, then $|G| \geq 24$. An example of a $G-$code, with $G = \mathcal{S}_4$, that is not abelian group code was first constructed over the field $\mathbb{K} = \mathbb{F}_5$. Later, the existence of such codes in the non semisimple case was also addressed, constructing $G-$codes over $\mathbb{K} = \mathbb{F}_2$ and $\mathbb{K} = \mathbb{F}_3$ that are not abelian group codes. The arguments used by the authors in the cases $\mathbb{F}_3$ and $\mathbb{F}_5$ are similar and they use the fact that the weight distribution of the constructed code (for each characteristic) does not coincide with the weight distribution of any abelian group code of length 24.

The problem for $\mathbb{F}_2$ was more complicated because every code over $\mathbb{F}_2$ of length 24 has the same weight distribution than some abelian group code of the same length. In any case, authors in [18] could find a code that was not permutation equivalent to any abelian group code, as it was explicitly proved. However, codes obtained in this way (using the group $\mathcal{S}_4$) turned out to have worse parameters than abelian group codes of the same length. Then, using the group $G = \mathrm{SL}(2, \mathbb{F}_3)$, the authors shown the existence of a non-abelian group code that is optimal in the sense that it has dimension 6 and minimal distance 10, being 10 the maximal distance of a binary linear code of length 24 and dimension 6. Furthermore, this distance can't be reached using an abelian group code of length 24 and dimension 6. In [19], using $G = \mathcal{S}_4$ and $G = \mathrm{SL}(2, \mathbb{F}_3)$ finally the existence of $G-$codes over $\mathbb{F}_p$, for every prime $p \geq 3$, that are non-abelian group codes was proved. Using other previous results of the authors, they can conclude that there are non-abelian group codes of length 24 for every finite field.

The fact that we can use abelian and non-abelian groups to construct group codes and the difficulty to distinguish group codes among linear codes seem good properties for these codes to be used in the design of a McEliece type cryptosystem. But it is also essential to have an efficient decoding algorithm. In [6], a permutation decoding algorithm for abelian group codes in the semisimple case was proposed. In [11], authors suggest an algorithm that can be seen as a variant of the classical syndrome decoding algorithm. But many problems are still open. In this work, a general and efficient decoding algorithm for group codes, in the semisimple case, is presented. For abelian group codes the algorithm can be modified to make it even more simple and efficient.

The decoding algorithm explained here does not work for left (or right) group codes nor for group codes in the non semisimple case. The reason is that the main property that we use is that for every ideal I of $\mathbb{K}G$ (identified with a $G-$ code over $\mathbb{K}$), there exists an ideal I^+ such that $\mathbb{K}G = I \oplus I^+$. Thus, $x \in I$ if and only if $xy = 0$ for all $y \in I^+$. This fact is not fulfilled for left (nor right) ideals nor for two-sided ideals of non semisimple algebras.

2. Preliminaries

From now on, G will be a finite group of order n, $\mathbb{K}$ is a finite field and n is not divisible by the characteristic of $\mathbb{K}$. This is equivalent to say that the group algebra $\mathbb{K}G$ is semisimple (Maschke's Theorem, see [7]). Thus, $\mathbb{K}G$ can be written as the sum of s minimal two-sided ideals (see p.166-170 of [10]). That is,

$$\mathbb{K}G = \langle e_1 \rangle \oplus \cdots \oplus \langle e_m \rangle \oplus \langle e_{m+1} \rangle \oplus \cdots \oplus \langle e_s \rangle.$$

Each minimal two-sided ideal $\langle e_i \rangle$ is generated by a primitive central idempotent e_i. These ideals are called simple components of $\mathbb{K}G$. Furthermore, any two-sided ideal of $\mathbb{K}G$ is generated by a central idempotent and is a direct sum of some simple components of $\mathbb{K}G$ (see p.437 of [20]).

So we will assume that $\mathfrak{C} = \langle e_0 \rangle$ is the two-sided ideal generated by $e_0 = e_{m+1} + \cdots + e_s$. Hence, $\mathfrak{C}$ can be identified with a group code of length n, as soon as we fix an order in the elements of G. Let's denote k the dimension of $\mathfrak{C}$ and d its minimal distance.

A codeword $\mathfrak{c} \in \mathfrak{C}$ has the form $\mathfrak{c} = ze_0$, for some $z \in \mathbb{K}G$. Therefore, an arbitrary element $\mathfrak{c} \in \mathbb{K}G$ is a codeword if and only if $\mathfrak{c}e_h = 0$ for all $h \in \{1, \ldots, m\}$. If a word

$\mathfrak{c} \in \mathfrak{C}$ is sent and the error $\mathfrak{e} \in \mathbb{K}G$ is produced during the transmission, the received word will be $\mathfrak{r} = \mathfrak{c} + \mathfrak{e}$. *The syndromes of* $\mathfrak{r}$ are defined as the m elements

$$S_h = \mathfrak{r}e_h, \qquad h = 1, \ldots, m.$$

Consequently, $S_h = (\mathfrak{c}+\mathfrak{e})e_h = \mathfrak{e}e_h$ for all $h \in \{1, \ldots, m\}$. Decoding by minimal distance means to look the closest codeword to $\mathfrak{r}$ and here it is equivalent to find a solution of the system $Xe_h = S_h$, $h = 1, \ldots, m$, having weight $q \leq t$, where t denotes the maximal number of errors that can be corrected by the code $\mathfrak{C}$. Clearly, we want such codeword to be unique. And this is always the case if the number of produced errors is not bigger than the error correcting capacity of the code $\mathfrak{C}$.

Theorem 2.1. *Let $\mathfrak{C}$ be a group code that corrects up to t errors. If the syndromes of a received word $\mathfrak{r}$ are $S_1, \ldots, S_m$, then there is at most one element $\mathfrak{e} \in \mathbb{K}G$, having weight less than or equal to t, that is a solution of the system $Xe_h = S_h$, $h = 1, \ldots, m$.*

Proof. If $\mathfrak{e}_1, \mathfrak{e}_2 \in \mathbb{K}G$ are two distinct solutions of the above system with weights q_1, q_2, respectively, and $q_1, q_2 \leq t$, then $\mathfrak{e}_1 e_h = \mathfrak{e}_2 e_h = S_h$ for all $h \in \{1, \ldots, m\}$. This implies that $(\mathfrak{e}_1 - \mathfrak{e}_2)e_h = 0$ for all $h \in \{1, \ldots, m\}$, that is, $\mathfrak{e}_1 - \mathfrak{e}_2 \in \mathfrak{C}$. So, the weight of $\mathfrak{e}_1 - \mathfrak{e}_2$ must be greater than or equal to d, where d is the minimal distance of $\mathfrak{C}$. But the weight of $\mathfrak{e}_1 - \mathfrak{e}_2$ is less than or equal to $q_1 + q_2 \leq 2t \leq d - 1$. This contradiction proves the Theorem. □

3. General case

Since the decoding process for group codes of dimension 1 is trivial, because the codewords are the scalar multiples of the central idempotent that generates the group code, we will consider, in what follows, only group codes of dimension $k \geq 2$.

First, let us notice that if $S_h = \mathfrak{r}e_h = 0$ for all $h \in \{1, \ldots, m\}$, then $\mathfrak{r} \in \mathfrak{C}$ and conversely. So the detection of errors produced during the transmission process is very easy.

Before starting the decoding process, the products $g_i e_h$, for all $i \in \{1, \ldots, n\}$ and $h \in \{1, \ldots, m\}$ should be computed. Denote $C^h_{g_i} \in M_{n\times 1}(\mathbb{K})$ the column matrix which consists of the coefficients of $g_i e_h$ (with the fixed order in G).

Proposition 3.1. *If $b < d$ and $g_{i_1}, \ldots, g_{i_b}$ are b distinct elements of G, then the matrix*

$$\mathcal{C}(g_{i_1}, \ldots, g_{i_b}) = \begin{pmatrix} C^1_{g_{i_1}} & \cdots & C^1_{g_{i_b}} \\ \vdots & & \vdots \\ C^m_{g_{i_1}} & \cdots & C^m_{g_{i_b}} \end{pmatrix} \in M_{mn\times b}(\mathbb{K}),$$

has rank b.

Proof. Suppose $\mathcal{C}(g_{i_1}, \ldots, g_{i_b})$ has rank less than b. Then there are elements $\nu_1, \ldots, \nu_b \in \mathbb{K}$, not all of them equal to zero, such that $\nu_1 C^h_{g_{i_1}} + \cdots + \nu_b C^h_{g_{i_b}} = 0$, for all $h \in \{1, \ldots, m\}$. Hence, $(\nu_1 g_{i_1} + \cdots + \nu_b g_{i_b})e_h = 0$, for all $h \in \{1, \ldots, m\}$, that is, $x = \nu_1 g_{i_1} + \cdots + \nu_b g_{i_b} \in \mathfrak{C}$. This fact contradicts that the minimal distance of $\mathfrak{C}$ is d, since the element $x \neq 0$ belongs to $\mathfrak{C}$ and its weight is less than or equal to b and so strictly less than d. □

Now we can explain the details of the decoding algorithm. Assume that we receive a word $\mathfrak{r}$. First of all, we have to decide if this word $\mathfrak{r}$ is the one that was sent or not. In the second case, we will need to recover the sent word. To do that, all the syndromes $S_h = \mathfrak{r}e_h$ will be computed and the column vectors S^h of their respective coordinates with respect to the fixed basis $\mathcal{B}$ will be defined. The aim of the decoding algorithm is to find an element $\mathfrak{e} = \alpha_1 g_{i_1} + \cdots + \alpha_q g_{i_q}$, having weight $q \leq t$, such that $S_h = \mathfrak{e}e_h$ for all $h \in \{1, \ldots, m\}$.

Since we do not know initially q, we consider a $t-$set of (ordered) elements of G, $\{g_{i_1}, \ldots, g_{i_t}\}$, and check if there is some element $\mathfrak{e} = \alpha_1 g_{i_1} + \cdots + \alpha_t g_{i_t}$ that is the solution of the system $Xe_h = S_h$, $h = 1, \ldots, m$. This will be the case if and only if the linear system

$$X_1 C^h_{g_{i_1}} + \cdots + X_t C^h_{g_{i_t}} = S^h, \qquad h \in \{1, \ldots, m\}, \tag{1}$$

admits the solution $X_i = \alpha_i$ for $i = 1, \ldots, t$. We know that if this system has some solution, it is unique. Therefore, there exist a solution if the matrix $\mathcal{C}(g_{i_1}, \ldots, g_{i_t})$ and the extended matrix

$$\mathcal{M}(g_{i_1}, \ldots, g_{i_t}) = \left(\begin{array}{ccc|c} C^1_{g_{i_1}} & \ldots & C^1_{g_{i_t}} & S^1 \\ \vdots & & \vdots & \vdots \\ C^m_{g_{i_1}} & \ldots & C^m_{g_{i_t}} & S^m \end{array} \right),$$

have both equal rank. By Proposition 3.1 we only need to check that the rank of $\mathcal{M}(g_{i_1}, \ldots, g_{i_t})$ is equal to t.

Note that this algorithm searches for $t-$sets $\{g_{i_1}, \ldots, g_{i_t}\}$ in $\mathcal{B} = G$ for which the corresponding system of equations (1) is consistent. That is, the algorithm finds $t-$sets in G that contain all error positions. Suppose the error produced is $\beta_1 g_{i_1} + \cdots + \beta_q g_{i_q}$. When $q = t$, $\{g_{i_1}, \ldots, g_{i_t}\}$ is the unique $t-$set found by the algorithm. When $q < t$, the algorithm can find several $t-$sets containing $\{g_{i_1}, \ldots, g_{i_q}\}$, for which the system (1) has a unique solution, but the error is uniquely determined, because the coefficients of g_j, for any $j \neq i_1, \ldots, i_q$, are always zero in the unique solution of the system (1) for any of the possible $t-$sets that make such system consistent.

3.1. Decoding algorithm

Let's start now the description of the proposed decoding algorithm:

Step 1.
Compute, for the received word $\mathfrak{r}$, all the syndromes. If $S_h = 0$ for all $h \in \{1, \dots, m\}$, then there are no errors, that is $\mathfrak{r} = \mathfrak{c}$, and the algorithm ends. Otherwise, go to Step 2.

Step 2.
Select at random $\{g_{i_1}, \dots, g_{i_t}\}$ a $t-$set of G. Construct the matrix $\mathcal{M}(g_{i_1}, \dots, g_{i_t})$ and compute its rank.

a. If the rank is t, find the unique solution $X_1 = \alpha_1, \dots, X_t = \alpha_t$ of the linear system

$$X_1 C^h_{g_{i_1}} + \cdots + X_t C^h_{g_{i_t}} = S^h, \qquad h \in \{1, \dots, m\}. \tag{2}$$

Then, the error is $\mathfrak{e} = \alpha_{j_1} g_{i_{j_1}} + \cdots + \alpha_{j_q} g_{i_{j_q}}$, where $\alpha_{j_1}, \dots, \alpha_{j_q}$ are the non-zero elements in the above solution, and the algorithm ends.

b. Otherwise, the $t-$set is discarded, another $t-$set of G is selected at random and Step 2 is repeated with it.

The algorithm finishes as soon as a $t-$set $\{g_{i_1}, \dots, g_{i_t}\}$ satisfying

$$\mathrm{Rank}(\mathcal{M}(g_{i_1}, \dots, g_{i_t})) = t, \tag{P1}$$

is found. When all $t-$sets of G have been checked and none satisfies property (P1), we conclude that the number of errors produced during the transmission is greater than t and the word $\mathfrak{r}$ can not be corrected.

Example 3.2. Let $G = \mathcal{S}_4$ and $\mathbb{K} = \mathbb{F}_5$. Order the elements of G as

$$\begin{array}{lll}
g_1 = (1), & g_2 = (3,4), & g_3 = (2,3), \\
g_4 = (2,3,4), & g_5 = (2,4,3), & g_6 = (2,4), \\
g_7 = (1,2), & g_8 = (1,2)(3,4), & g_9 = (1,2,3), \\
g_{10} = (1,2,3,4), & g_{11} = (1,2,4,3), & g_{12} = (1,2,4), \\
g_{13} = (1,3,2), & g_{14} = (1,3,4,2), & g_{15} = (1,3), \\
g_{16} = (1,3,4), & g_{17} = (1,3)(2,4), & g_{18} = (1,3,2,4), \\
g_{19} = (1,4,3,2), & g_{20} = (1,4,2), & g_{21} = (1,4,3), \\
g_{22} = (1,4), & g_{23} = (1,4,2,3), & g_{24} = (1,4)(2,3).
\end{array}$$

It is known that $\mathbb{F}_5\mathcal{S}_4$ decomposes as the direct sum of five simple components of dimensions 1,1,4,9 and 9. In a concrete way

$$\mathbb{F}_5\mathcal{S}_4 = \langle e_1 \rangle \oplus \langle e_2 \rangle \oplus \langle e_3 \rangle \oplus \langle e_4 \rangle \oplus \langle e_5 \rangle,$$

where

$$e_1 = (4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4),$$

$$\begin{aligned}
e_2 &= (4,1,1,4,4,1,1,4,4,1,1,4,4,1,1,4,4,1,1,4,4,1,1,4),\\
e_3 &= (1,0,0,2,2,0,0,1,2,0,0,2,2,0,0,2,1,0,0,2,2,0,0,1),\\
e_4 &= (1,3,3,0,0,3,3,3,0,2,2,0,0,2,3,0,3,2,2,0,0,3,2,3),\\
e_5 &= (1,2,2,0,0,2,2,3,0,3,3,0,0,3,2,0,3,3,3,0,0,2,3,3)
\end{aligned}$$

are the primitive central idempotents. Take $\mathfrak{C} = \langle e_5 \rangle$. The group code $\mathfrak{C}$ has parameters $n = 24$, $k = 9$, $d = 8$ and $t = 3$.

If the received word is

$$\mathfrak{r} = (2,1,4,1,1,4,0,0,1,1,3,4,1,3,4,4,1,1,1,4,4,0,4,2),$$

then the decoding process is as follows:

Step 1. We compute S_h, $h = 1,2,3,4$. In this case,

$$\begin{aligned}
S^1 &= (4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4)^T,\\
S^2 &= (1,4,4,1,1,4,4,1,1,4,4,1,1,4,4,1,1,4,4,1,1,4,4,1)^T,\\
S^3 &= (0,1,2,0,0,2,1,0,0,2,2,0,0,2,2,0,0,1,2,0,0,2,1,0)^T,\\
S^4 &= (1,3,0,2,4,0,0,1,1,0,0,3,3,0,0,4,4,4,0,1,2,0,3,4)^T.
\end{aligned}$$

Since they are not equal to zero, we go to Step 2.

Step 2. Take the 3−set $\{g_4, g_7, g_{18}\}$. We have that the matrix

$$\mathcal{M}(g_4, g_7, g_{18}) = \left(\begin{array}{ccc|c}
C^1_{g_4} & C^1_{g_7} & C^1_{g_{18}} & S^1\\
C^2_{g_4} & C^2_{g_7} & C^2_{g_{18}} & S^2\\
C^3_{g_4} & C^3_{g_7} & C^3_{g_{18}} & S^3\\
C^4_{g_4} & C^4_{g_7} & C^4_{g_{18}} & S^4
\end{array}\right),$$

has rank 3. In fact, we have that $C^1_{g_4} = C^1_{g_7} = C^1_{g_{18}}$, $C^2_{g_7} = C^2_{g_{18}}$ and $C^3_{g_7} = C^3_{g_{18}}$ where

$$\begin{aligned}
C^1_{g_{18}} &= (4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4)^T,\\
C^2_{g_{18}} &= (1,4,4,1,1,4,4,1,1,4,4,1,1,4,4,1,1,4,4,1,1,4,4,1)^T,\\
C^3_{g_{18}} &= (0,1,2,0,0,2,1,0,0,2,2,0,0,2,2,0,0,1,2,0,0,2,1,0)^T.
\end{aligned}$$

Furthermore,

$$C^2_{g_4} = (4,1,1,4,4,1,1,4,4,1,1,4,4,1,1,4,4,1,1,4,4,1,1,4)^T,$$

$$\begin{aligned}
C^3_{g_4} &= (2,0,0,1,2,0,0,2,2,0,0,1,1,0,0,2,2,0,0,2,1,0,0,2)^T,\\
C^4_{g_4} &= (0,3,3,1,0,3,2,0,0,3,2,3,3,3,2,0,0,2,2,0,3,2,3,0)^T,\\
C^4_{g_7} &= (3,3,0,2,2,0,1,3,3,0,0,3,3,0,0,2,2,3,0,3,2,0,3,2)^T,\\
C^4_{g_{18}} &= (2,3,0,2,3,0,3,2,2,0,0,3,3,0,0,3,3,1,0,2,2,0,3,3)^T.
\end{aligned}$$

The solution of the system

$$\begin{aligned}
X_1C^1_{g_4} + X_2C^1_{g_7} + X_3C^1_{g_{18}} &= S^1\\
X_1C^2_{g_4} + X_2C^2_{g_7} + X_3C^2_{g_{18}} &= S^2\\
X_1C^3_{g_4} + X_2C^3_{g_7} + X_3C^3_{g_{18}} &= S^3\\
X_1C^4_{g_4} + X_2C^4_{g_7} + X_3C^4_{g_{18}} &= S^4
\end{aligned}$$

is $X_1 = 0$, $X_2 = 4$ and $X_3 = 2$.

Consequently,

$$\mathfrak{e} = 4g_7 + 2g_{18} = (0,0,0,0,0,0,4,0,0,0,0,0,0,0,0,0,0,2,0,0,0,0,0,0)$$

and

$$\mathfrak{c} = (2,1,4,1,1,4,1,0,1,1,3,4,1,3,4,4,1,4,1,4,4,0,4,2).$$

4. A special case

In this section, keeping the notation in the previous ones, we will present a variation of the previous algorithm in the case that G is abelian and the simple components of $\mathbb{K}G$ have dimension 1. We know that this is the case when $\mathbb{K}$ contains a primitive root of the unit.

Before starting the algorithm, we compute the scalars $\lambda^h_{g_i} \in \mathbb{K}$ such that $g_i e_h = \lambda^h_{g_i} e_h$, for all $i \in \{1, \dots, n\}$ and $h \in \{1, \dots, m\}$ (note that the simple components $\langle e_1\rangle, \dots, \langle e_m\rangle$ have dimension 1). Let us define $A_{g_i} = (\lambda^1_{g_i}, \dots, \lambda^m_{g_i})^T$ for all $g_i \in G$ and

$$\mathcal{A}(g_{i_1}, \dots, g_{i_b}) = \begin{pmatrix} A_{g_{i_1}} & \dots & A_{g_{i_b}} \end{pmatrix}.$$

Proposition 4.1. *If $b < d$ and $g_{i_1}, \dots, g_{i_b}$ are b distinct elements of G, then the matrix*

$$\mathcal{A}(g_{i_1}, \dots, g_{i_b}) = \begin{pmatrix} \lambda^1_{g_{i_1}} & \dots & \lambda^1_{g_{i_b}} \\ \vdots & & \vdots \\ \lambda^m_{g_{i_1}} & \dots & \lambda^m_{g_{i_b}} \end{pmatrix} \in M_{m\times b}(\mathbb{K})$$

has rank b.

Proof. The proof is analogous to the proof of Proposition 3.1. □

The algorithm starts by computing all the syndromes $S_h = \mathfrak{r}e_h = \mu^h e_h$ where $\mu_h \in \mathbb{K}$. Let us define the column vector $R = (\mu^1, \dots, \mu^m)^T$.

As before, the algorithm will look for $t-$sets of G containing all error positions. So, the aim of the algorithm is to find an element $\mathfrak{e} = \alpha_1 g_{i_1} + \cdots + \alpha_t g_{i_t}$, where $g_{i_1}, \dots, g_{i_t}$ are distinct elements of G and $\alpha_1, \dots, \alpha_t \in \mathbb{K}$ such that the linear system

$$X_1 A_{g_{i_1}} + \cdots + X_t A_{g_{i_t}} = R \tag{3}$$

has a unique solution $X_i = \alpha_i$ for all $i \in \{1, \dots, t\}$. And this is equivalent to the fact that the rank of the extended matrix

$$\mathcal{B}(g_{i_1}, \dots, g_{i_t}) = \Big(A_{g_{i_1}} \quad \dots \quad A_{g_{i_t}} \;\Big|\; R \Big) = \left(\begin{array}{ccc|c} \lambda^1_{g_{i_1}} & \dots & \lambda^1_{g_{i_t}} & \mu^1 \\ \vdots & & \vdots & \vdots \\ \lambda^m_{g_{i_1}} & \dots & \lambda^m_{g_{i_t}} & \mu^m \end{array} \right)$$

is equal to the rank of the matrix $\mathcal{A}(g_{i_1}, \dots, g_{i_t})$ that is known to be t.

4.1. Decoding algorithm

Let's start the description of the second decoding algorithm.

Step 1.
Compute all the syndromes. If $S_h = 0$ for all $h \in \{1, \dots, m\}$, then there are no errors and the algorithm ends. Otherwise, compute R and go to Step 2.

Step 2.
Select at random $\{g_{i_1}, \dots, g_{i_t}\}$ a $t-$set of G. Compute the matrix $\mathcal{B}(g_{i_1}, \dots, g_{i_t})$ and its rank.

a. If the rank is t, find the unique solution $\alpha_1, \dots, \alpha_t \in \mathbb{K}$ of the linear system

$$X_1 A_{g_{i_1}} + \cdots + X_t A_{g_{i_t}} = R.$$

If $\alpha_{j_1}, \dots, \alpha_{j_q}$ are the non-zero elements in the above solution, then the error is $\mathfrak{e} = \alpha_{j_1} g_{i_{j_1}} + \cdots + \alpha_{j_q} g_{i_{j_q}}$ and the algorithm ends.

b. Otherwise, the $t-$set is discarded, another $t-$set of G is randomly selected and Step 2 is repeated with it.

The algorithm finishes when a $t-$set $\{g_{i_1}, \dots, g_{i_t}\}$ satisfying

$$\mathrm{Rank}(\mathcal{B}(g_{i_1}, \dots, g_{i_t})) = t, \tag{P2}$$

is found. If all $t-$sets of G have been checked and none satisfies (P2), we conclude that the number of errors produced during the transmission is greater than t and the word $\mathfrak{r}$ can not be corrected.

This algorithm is more efficient than the general algorithm since the size of matrices is smaller. It computes ranks of matrices with $n-k$ rows instead of matrices with $n(n-k)$ rows.

5. Abelian case

The algorithm seen in the previous section proved to be very efficient when dealing with abelian groups. But the field must be fairly good. Our aim now is to adjust this algorithm to be used on arbitrary fields. In what follows, G is an abelian group and its order n is not divisible by the characteristic of $\mathbb{K}$.

Next result will be essential in what follows. Since, we could not find a reference for it, its proof is included by completeness.

Proposition 5.1. *Let $\mathfrak{C}$ be a linear code over a field $\mathbb{K}$ and $\mathbb{E}$ another field extension of $\mathbb{K}$. If $\widetilde{\mathfrak{C}} = \mathfrak{C} \otimes_{\mathbb{K}} \mathbb{E}$, then $\widetilde{\mathfrak{C}}$ has the same parameters as $\mathfrak{C}$.*

Proof. It is well known in linear algebra that for any $\mathbb{K}-$vector space V, $V \otimes_{\mathbb{K}} \mathbb{E}$ is also an $\mathbb{E}-$vector space defining

$$\xi(v \otimes \gamma) = v \otimes (\xi\gamma), \qquad v \in V, \qquad \gamma, \xi \in \mathbb{E},$$

and if $\{v_i\}$ is a $\mathbb{K}-$basis of V, then $\{v_i \otimes 1\}$ is an $\mathbb{E}-$basis of $V \otimes_{\mathbb{K}} \mathbb{E}$. Consequently, $\mathfrak{C}$ and $\widetilde{\mathfrak{C}}$ have the same length and the same dimension. Also, $\mathfrak{C}$ and $\widetilde{\mathfrak{C}}$ have a common control matrix $\mathcal{H}$ (whose elements lie in $\mathbb{K}$). Denote $\widetilde{d}$ the minimal distance of $\widetilde{\mathfrak{C}}$ and d the distance of $\mathfrak{C}$. Now we conclude the proof by using that,

$$\begin{aligned} d &= \min\{b \mid \text{There are } b \text{ columns of } \mathcal{H} \text{ linearly } \mathbb{K}-\text{dependent}\} \\ &= \min\{p \mid \text{There are } p \text{ columns of } \mathcal{H} \text{ linearly } \mathbb{E}-\text{dependent}\} = \widetilde{d}. \quad \square \end{aligned}$$

From now on, let us consider $\mathbb{E}$ the smallest extension of $\mathbb{K}$ such that the simple components of $\mathbb{E}G$ have dimension 1. Then $\mathbb{E}G$ is direct sum of (one-dimensional) components generated by its primitive central idempotents, denoted $f_1, \dots, f_n$.

Remind that if $e_1, \dots, e_s$ are the primitive central idempotents of $\mathbb{K}G$, then e_i is the sum of those $f_j \in \{f_1, \dots, f_n\}$ satisfying $e_i f_j \neq 0$. Indeed, since $\{f_1, \dots, f_n\}$ is an $\mathbb{E}-$basis of $\mathbb{E}G$, there are $\gamma_1^i, \dots, \gamma_n^i \in \mathbb{E}$ such that $e_i = \gamma_1^i f_1 + \cdots + \gamma_n^i f_n$. Then, $e_i = e_i^2 = (\gamma_1^i)^2 f_1 + \cdots + (\gamma_n^i)^2 f_n$ implies $(\gamma_j^i)^2 = \gamma_j^i$. Consequently, $\gamma_j^i = 0$ or $\gamma_j^i = 1$ for all $j \in \{1, \dots, n\}$ and $\gamma_j^i = 1$ if and only if $e_i f_j \neq 0$.

If $\mathfrak{C}$ is the group code in $\mathbb{K}G$ generated by e_0, as in Sections 1 and 2, and $\widetilde{\mathfrak{C}} = \mathfrak{C} \otimes_{\mathbb{K}} \mathbb{E}$ is the group code in $\mathbb{E}G$ generated by e_0, then we know, by Proposition 5.1, that $\widetilde{\mathfrak{C}}$ corrects the same number of errors than $\mathfrak{C}$. We can rename the idempotents $f_1, \dots, f_n$ in such a way that $e_0 = f_{n-k+1} + \cdots + f_n$.

Given an element $\mathfrak{r} \in \mathbb{K}G$, denote $S_1, \dots, S_l$ its syndromes related to the group code $\mathfrak{C}$ and $\widetilde{S}_1, \dots, \widetilde{S}_{n-k}$ its syndromes related to the group code $\widetilde{\mathfrak{C}}$ when $\mathfrak{r}$ is considered an element in $\mathbb{E}G$.

To decode a received word $\mathfrak{r} \in \mathbb{K}G$, we will consider it as element of $\mathbb{E}G$ and will decode using the code $\widetilde{\mathfrak{C}}$. The process works because of the following result, in which we keep the previous notation.

Proposition 5.2. *If there exist $\mathfrak{e} \in \mathbb{K}G$ and $\widetilde{\mathfrak{e}} \in \mathbb{E}G$, both with weight less than or equal to t, such that*

$$\mathfrak{e}e_h = S_h, \qquad h = 1, \dots, m \tag{4}$$

and

$$\widetilde{\mathfrak{e}}f_j = \widetilde{S}_j, \qquad j = 1, \dots, n-k \tag{5}$$

respectively, then $\widetilde{\mathfrak{e}} = \mathfrak{e}$ and so $\widetilde{\mathfrak{e}} \in \mathbb{K}G$.

Proof. For each $e_i \in \mathbb{K}G$, let's denote $J(i)$ the set of those $j \in \{1, \dots, n\}$ such that $e_i f_j \neq 0$. So, $e_i = \sum_{j \in J(i)} f_j$ and therefore

$$S_h = \mathfrak{r}e_h = \mathfrak{r}\left(\sum_{j \in J(h)} f_j\right) = \sum_{j \in J(h)} \widetilde{S}_j$$

for all $h \in \{1, \dots, m\}$. By the hypothesis and Theorem 2.1, $\mathfrak{e}$ and $\widetilde{\mathfrak{e}}$ are the unique elements in $\mathbb{K}G$ and $\mathbb{E}G$, respectively, satisfying (4) and (5). Note that $\mathfrak{e} \in \mathbb{K}G \subseteq \mathbb{E}G$ and (4) implies that

$$\mathfrak{e}\left(\sum_{j \in J(h)} f_j\right) = \mathfrak{e}e_h = S_h = \sum_{j \in J(h)} \widetilde{S}_j,$$

for all $h \in \{1, \dots, m\}$ and thus

$$\sum_{j \in J(h)} (\mathfrak{e}f_j - \widetilde{S}_j) = 0.$$

Since, $\mathfrak{e}f_j - \widetilde{S}_j \in \langle f_j \rangle$ for all $j \in J(h)$ then $\mathfrak{e}f_j - \widetilde{S}_j = 0$ for all $j \in J(h)$. Therefore $\mathfrak{e}f_j = \widetilde{S}_j$ for all $j \in J(h)$ and consequently, (5) is fulfilled. By uniqueness in $\mathbb{E}G$, we have that $\widetilde{\mathfrak{e}} = \mathfrak{e}$ and thus $\widetilde{\mathfrak{e}} \in \mathbb{K}G$. □

This result assures that if we can decode the element $\mathfrak{r} \in \mathbb{K}G$ using $\widetilde{\mathfrak{C}}$ and we get a codeword $\mathfrak{c} \in \widetilde{\mathfrak{C}} \setminus \mathfrak{C}$, then we can conclude that the number of errors produced during the transmission is greater than the error correcting capacity of $\mathfrak{C}$ and we can't decode in $\mathfrak{C}$.

So this second algorithm can be used with any abelian group code and on arbitrary fields (such that the characteristic of $\mathbb{K}$ does not divide the order of G) and we only need to extend scalars in a fairly way.

Example 5.3. Let $\mathbb{K} = \mathbb{F}_2$ and $G = \mathcal{C}_3 \times \mathcal{C}_3 \times \mathcal{C}_3 = \langle a, b, c \rangle$. Here

$$\mathbb{F}_2(\mathcal{C}_3 \times \mathcal{C}_3 \times \mathcal{C}_3) = \langle e_1 \rangle \oplus \cdots \oplus \langle e_{14} \rangle.$$

The dimension of the first simple component is 1 and the other simple components have dimension 2. We fix the basis

$$\{1_G, a, b, c, a^2, ab, ac, b^2, bc, c^2, a^2b, a^2c, ab^2, abc, ac^2, b^2c, bc^2, \\ a^2b^2, a^2bc, a^2c^2, ab^2c, abc^2, b^2c^2, a^2b^2c, a^2bc^2, ab^2c^2, a^2b^2c^2\}.$$

Then the simple components are generated respectively by the following primitive central idempotents:

$$\begin{aligned}
e_1 &= (1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1),\\
e_2 &= (0,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,0),\\
e_3 &= (0,1,1,1,1,1,0,1,0,1,0,1,0,1,1,1,1,1,1,0,1,0,0,0,1,1,1),\\
e_4 &= (0,1,1,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,0,1,0,1,1),\\
e_5 &= (0,1,1,1,1,0,0,1,1,1,1,1,1,1,1,0,0,0,0,0,1,1,1,1,1,0,1),\\
e_6 &= (0,1,1,0,1,1,1,1,1,0,0,1,0,1,1,1,1,1,0,1,0,1,1,1,0,0,1),\\
e_7 &= (0,1,1,0,1,0,1,1,1,0,1,1,1,0,1,1,1,0,1,1,1,0,1,0,1,1,0),\\
e_8 &= (0,1,0,1,1,1,1,0,1,1,1,0,1,1,0,1,1,1,0,1,1,0,1,0,1,0,1),\\
e_9 &= (0,1,0,1,1,1,0,0,1,1,1,1,1,0,1,1,1,1,1,0,0,1,1,1,0,1,0),\\
e_{10} &= (0,1,0,0,1,1,1,0,0,0,1,1,1,1,1,0,0,1,1,1,1,1,0,1,1,1,1),\\
e_{11} &= (0,0,1,1,0,1,1,1,1,1,1,1,1,1,1,0,0,1,1,1,0,0,1,0,0,1,1),\\
e_{12} &= (0,0,1,1,0,1,1,1,0,1,1,1,1,0,1,1,1,1,0,1,1,1,0,1,1,0,0),\\
e_{13} &= (0,0,1,0,0,1,0,1,1,0,1,0,1,1,0,1,1,1,1,0,1,1,1,1,1,1,1),\\
e_{14} &= (0,0,0,1,0,0,1,0,1,1,0,1,0,1,1,1,1,0,1,1,1,1,1,1,1,1,1).
\end{aligned}$$

The smallest extension $\mathbb{E}$ of $\mathbb{F}_2$ such that the dimensions of the simple components of $\mathbb{E}G$ are equal to 1, is $\mathbb{E} = \mathbb{F}_4 = \mathbb{F}_2(w)$ where $w^2 = w + 1$. The primitive central idempotents of $\mathbb{F}_4(\mathcal{C}_3 \times \mathcal{C}_3 \times \mathcal{C}_3)$ are:

$$\begin{aligned}
f_1 &= (1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1),\\
f_2 &= (1,w,w,w,w^2,w^2,w^2,w^2,w^2,w^2,1,1,1,1,1,1,1,w,w,w,w,w,w,w^2,w^2,w^2,1),\\
f_3 &= (1,w^2,w^2,w^2,w,w,w,w,w,w,1,1,1,1,1,1,1,w^2,w^2,w^2,w^2,w^2,w^2,w,w,w,1),\\
f_4 &= (1,w,w,w^2,w^2,w^2,1,w^2,1,w,1,w,1,w,w^2,w,w^2,w,w^2,1,w^2,1,1,1,w,w,w^2),\\
f_5 &= (1,w^2,w^2,w,w,w,1,w,1,w^2,1,w^2,1,w^2,w,w^2,w,w^2,w,1,w,1,1,1,w^2,w^2,w),\\
f_6 &= (1,w,w^2,w,w^2,1,w^2,w,1,w^2,w,1,w^2,w,1,w^2,w,1,w^2,w,1,w^2,1,w,1,w,w^2),\\
f_7 &= (1,w^2,w,w^2,w,1,w,w^2,1,w,w^2,1,w,w^2,1,w,w^2,1,w,w^2,1,w,1,w^2,1,w^2,w),\\
f_8 &= (1,w,w^2,w^2,w^2,1,1,w,w,w,w,w,w^2,w^2,w^2,1,1,1,1,1,w,w,w^2,w^2,w^2,1,w),\\
f_9 &= (1,w^2,w,w,w,1,1,w^2,w^2,w^2,w^2,w^2,w,w,w,1,1,1,1,1,w^2,w^2,w,w,w,1,w^2),\\
f_{10} &= (1,w,w,1,w^2,w^2,w,w^2,w,1,1,w^2,1,w^2,w,w^2,w,w,1,w^2,1,w^2,w^2,w,1,1,w),\\
f_{11} &= (1,w^2,w^2,1,w,w,w^2,w,w^2,1,1,w,1,w,w^2,w,w^2,w^2,1,w,1,w,w,w^2,1,1,w^2),\\
f_{12} &= (1,w,w^2,1,w^2,1,w,w,w^2,1,w,w^2,w^2,1,w,w,w^2,1,w,w^2,w^2,1,w,1,w,w^2,1),\\
f_{13} &= (1,w^2,w,1,w,1,w^2,w^2,w,1,w^2,w,w,1,w^2,w^2,w,1,w^2,w,w,1,w^2,1,w^2,w,1),\\
f_{14} &= (1,w,1,w,w^2,w,w^2,1,w,w^2,w^2,1,w,w^2,1,w,w^2,w^2,1,w,w^2,1,w^2,1,w,1,w),\\
f_{15} &= (1,w^2,1,w^2,w,w^2,w,1,w^2,w,w,1,w^2,w,1,w^2,w,w,1,w^2,w,1,w,1,w^2,1,w^2),\\
f_{16} &= (1,w,1,w^2,w^2,w,1,1,w^2,w,w^2,w,w,1,w^2,w^2,w,w^2,w,1,1,w^2,w,w,1,w^2,1),\\
f_{17} &= (1,w^2,1,w,w,w^2,1,1,w,w^2,w,w^2,w^2,1,w,w,w^2,w,w^2,1,1,w,w^2,w^2,1,w,1),\\
f_{18} &= (1,w,1,1,w^2,w,w,1,1,1,w^2,w^2,w,w,w,1,1,w^2,w^2,w^2,w,w,1,w^2,w^2,w,w^2),\\
f_{19} &= (1,w^2,1,1,w,w^2,w^2,1,1,1,w,w,w^2,w^2,w^2,1,1,w,w,w,w^2,w^2,1,w,w,w^2,w),\\
f_{20} &= (1,1,w,w,1,w,w,w^2,w^2,w^2,w,w,w^2,w^2,w^2,1,1,w^2,w^2,w^2,1,1,w,1,1,w,w),\\
f_{21} &= (1,1,w^2,w^2,1,w^2,w^2,w,w,w,w^2,w^2,w,w,w,1,1,w,w,w,1,1,w^2,1,1,w^2,w^2),\\
f_{22} &= (1,1,w,w^2,1,w,w^2,w^2,1,w,w,w^2,w^2,1,w,w,w^2,w^2,1,w,w,w^2,1,w,w^2,1,1),\\
f_{23} &= (1,1,w^2,w,1,w^2,w,w,1,w^2,w^2,w,w,1,w^2,w^2,w,w,1,w^2,w^2,w,1,w^2,w,1,1),\\
f_{24} &= (1,1,w,1,1,w,1,w^2,w,1,w,1,w^2,w,1,w^2,w,w^2,w,1,w^2,w,w^2,w^2,w,w^2,w^2),\\
f_{25} &= (1,1,w^2,1,1,w^2,1,w,w^2,1,w^2,1,w,w^2,1,w,w^2,w,w^2,1,w,w^2,w,w,w^2,w,w),\\
f_{26} &= (1,1,1,w,1,1,w,1,w,w^2,1,w,1,w,w^2,w,w^2,1,w,w^2,w,w^2,w^2,w,w^2,w^2,w^2),\\
f_{27} &= (1,1,1,w^2,1,1,w^2,1,w^2,w,1,w^2,1,w^2,w,w^2,w,1,w^2,w,w^2,w,w,w^2,w,w,w).
\end{aligned}$$

Here $e_1 = f_1$, $e_2 = f_2 + f_3$, $e_3 = f_4 + f_5$, $e_4 = f_6 + f_7$, $e_5 = f_8 + f_9$, $e_6 = f_{10} + f_{11}$, $e_7 = f_{12} + f_{13}$, $e_8 = f_{14} + f_{15}$, $e_9 = f_{16} + f_{17}$, $e_{10} = f_{18} + f_{19}$, $e_{11} = f_{20} + f_{21}$, $e_{12} = f_{22} + f_{23}$, $e_{13} = f_{24} + f_{25}$, $e_{14} = f_{26} + f_{27}$.

Let us consider $\mathfrak{C} = \langle e_0 \rangle$, where $e_0 = e_9 + e_{10} + e_{11} + e_{12} + e_{13} + e_{14}$. Then $n = 27$, $k = 12$, $d = 6$, $t = 2$. Hence,

$$\widetilde{\mathfrak{C}} = \langle f_{16} + f_{17} + f_{18} + f_{19} + f_{20} + f_{21} + f_{22} + f_{23} + f_{24} + f_{25} + f_{26} + f_{27} \rangle$$

has the same parameters as $\mathfrak{C}$. We compute, working in $\widetilde{\mathfrak{C}}$,

$$\begin{aligned}
\widetilde{A}_{1_G} &= (1,1,1,1,1,1,1,1,1,1,1,1,1,1,1)^T, \\
\widetilde{A}_{a} &= (1,w^2,w,w^2,w,w^2,w,w^2,w,w^2,w,w^2,w,w^2,w)^T, \\
\widetilde{A}_{b} &= (1,w^2,w,w^2,w,w,w^2,w,w^2,w^2,w,w,w^2,1,1)^T, \\
\widetilde{A}_{c} &= (1,w^2,w,w,w^2,w^2,w,w,w^2,1,1,1,1,w^2,w)^T, \\
\widetilde{A}_{a^2} &= (1,w,w^2,w,w^2,w,w^2,w,w^2,w,w^2,w,w^2,w,w^2)^T, \\
\widetilde{A}_{ab} &= (1,w,w^2,w,w^2,1,1,1,1,w,w^2,1,1,w^2,w)^T, \\
\widetilde{A}_{ac} &= (1,w,w^2,1,1,w,w^2,1,1,w^2,w,w^2,w,w,w^2)^T, \\
\widetilde{A}_{b^2} &= (1,w,w^2,w,w^2,w^2,w,w^2,w,w,w^2,w^2,w,1,1)^T, \\
\widetilde{A}_{bc} &= (1,w,w^2,1,1,1,1,w^2,w,w^2,w,w,w^2,w^2,w)^T, \\
\widetilde{A}_{c^2} &= (1,w,w^2,w^2,w,w,w^2,w^2,w,1,1,1,1,w,w^2)^T, \\
\widetilde{A}_{a^2b} &= (1,1,1,1,1,w^2,w,w^2,w,1,1,w^2,w,w,w^2)^T, \\
\widetilde{A}_{a^2c} &= (1,1,1,w^2,w,1,1,w^2,w,w,w^2,w,w^2,1,1)^T, \\
\widetilde{A}_{ab^2} &= (1,1,1,1,1,w,w^2,w,w^2,1,1,w,w^2,w^2,w)^T, \\
\widetilde{A}_{abc} &= (1,1,1,w^2,w,w^2,w,w,w^2,w,w^2,1,1,w,w^2)^T, \\
\widetilde{A}_{ac^2} &= (1,1,1,w,w^2,1,1,w,w^2,w^2,w,w^2,w,1,1)^T, \\
\widetilde{A}_{b^2c} &= (1,1,1,w^2,w,w,w^2,1,1,w,w^2,w^2,w,w^2,w)^T, \\
\widetilde{A}_{bc^2} &= (1,1,1,w,w^2,w^2,w,1,1,w^2,w,w,w^2,w,w^2)^T, \\
\widetilde{A}_{a^2b^2} &= (1,w^2,w,w^2,w,1,1,1,1,w^2,w,1,1,w,w^2)^T, \\
\widetilde{A}_{a^2bc} &= (1,w^2,w,w,w^2,w,w^2,1,1,1,1,w^2,w,1,1)^T, \\
\widetilde{A}_{a^2c^c} &= (1,w^2,w,1,1,w^2,w,1,1,w,w^2,w,w^2,w^2,w)^T, \\
\widetilde{A}_{ab^2c} &= (1,w^2,w,w,w^2,1,1,w^2,w,1,1,w,w^2,w,w^2)^T, \\
\widetilde{A}_{abc^2} &= (1,w^2,w,1,1,w,w^2,w^2,w,w,w^2,1,1,1,1)^T, \\
\widetilde{A}_{b^2c^2} &= (1,w^2,w,1,1,1,1,w,w^2,w,w^2,w^2,w,w,w^2)^T, \\
\widetilde{A}_{a^2b^2c} &= (1,w,w^2,1,1,w^2,w,w,w^2,w^2,w,1,1,1,1)^T, \\
\widetilde{A}_{a^2bc^2} &= (1,w,w^2,w^2,w,1,1,w,w^2,1,1,w^2,w,w^2,w)^T, \\
\widetilde{A}_{ab^2c^2} &= (1,w,w^2,w^2,w,w^2,w,1,1,1,1,w,w^2,1,1)^T, \\
\widetilde{A}_{a^2b^2c^2} &= (1,1,1,w,w^2,w,w^2,w^2,w,w^2,w,1,1,w^2,w)^T.
\end{aligned}$$

Suppose

$$\mathfrak{r} = (1,0,0,1,0,1,1,0,1,0,0,1,1,1,1,0,1,1,0,0,0,0,1,0,1,0,1),$$

then:

Step 1. We have $\widetilde{S}_j \neq 0$ for all $j \in \{1, \ldots, 15\} \setminus \{1, 10, 11, 12, 13\}$ and

$$\widetilde{R} = (0, w, w^2, w^2, w, 1, 1, w, w^2, 0, 0, 0, 0, w^2, w)^T.$$

Step 2. Taking the 2−set $\{a^2b, a^2bc\}$, we have that

$$\widetilde{\mathscr{B}}(a^2b, a^2bc) = \left(\widetilde{A}_{a^2b} \quad \widetilde{A}_{a^2bc} \;\middle|\; \widetilde{R} \right)$$

has rank 2. Then, the solution of the system $X_1\widetilde{A}_{a^2b} + X_2\widetilde{A}_{a^2bc} = \widetilde{R}$ is $X_1 = X_2 = 1$.

Hence,

$$\mathfrak{e} = a^2b + a^2bc = (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0)$$

and

$$\mathfrak{c} = (1, 0, 0, 1, 0, 1, 1, 0, 1, 0, 1, 1, 1, 1, 1, 0, 1, 1, 1, 0, 0, 0, 1, 0, 1, 0, 1).$$

Remark. If we execute the general algorithm, looking for 2−sets such that the corresponding extended matrices have rank 2, these matrices have $27 \times 8 = 216$ rows. However, if we apply the last algorithm, we compute the ranks of extended matrices with 15 rows, which clearly makes the last algorithm more efficient.

6. Conclusions

1. A decoding algorithm has been found that is efficient for group codes in the semisimple case.
2. In the abelian case the algorithm can be improved to get a faster and simpler version.
3. A careful comparison of the complexities of both algorithms has to be done next.

7. Acknowledgments

This work was supported by the project MCIU-22-PID2021-123461NB-C22.

Data availability

No data was used for the research described in the article.

References

[1] M. Baldi, M. Bianchi, F. Chiaraluce, J. Rosenthal, D. Schipani, Enhanced public key security for the McEliece cryptosystem, J. Cryptol. 29 (2016) 1–27, https://doi.org/10.1007/s00145-014-9187-8.

[2] P. Barreto, F. Biasi, R. Misoczki, W. Ruggiero, Scaling efficient code-based cryptosystems for embedded platforms, J. Cryptogr. Eng. 4 (2) (2014) 123–134, https://doi.org/10.48550/arXiv.1212.4317.

[3] T. Berger, P. Loidreau, How to mask the structure of codes for a cryptographic use, Des. Codes Cryptogr. 35 (2005) 63–79, https://doi.org/10.1007/s10623-003-6151-2.

[4] T. Berger, P. Cayrel, P. Gaborit, A. Otman, Reducing key length of the McEliece cryptosystem, in: B. Preneel (Ed.), Prog. Cryptology - AFRICACRYPT 2009, in: Lect. Notes Comp. Sci., vol. 5580, Springer, Berlin, Heidelberg, 2009, pp. 77–97.

[5] J.J. Bernal, A. del Río, J.J. Simón, An intrinsical description of group codes, Des. Codes Cryptogr. 51 (3) (2009) 289–300, https://doi.org/10.1007/s10623-008-9261-z.

[6] J.J. Bernal, A. del Río, J.J. Simón, Partial permutation decoding for abelian codes, IEEE Trans. Inf. Theory (2012), https://doi.org/10.1109/ISIT.2012.6284011.

[7] M. Burrow, Representation Theory of Finite Groups, Academic Press, New York, 1965.

[8] A. Couvreur, P. Gaborit, V. Gauthier, A. Otmani, J. Tillich, Distinguisher-based attacks on public-key cryptosystems using Reed-Solomon codes, Des. Codes Cryptogr. 73 (2) (2014) 641–666, https://doi.org/10.48550/arXiv.1307.6458.

[9] A. Couvreur, I. Márquez, R. Pellikaan, A polynomial time attack against algebraic geometry code based public key cryptosystem, IEEE Trans. Inf. Theory (2014) 1446–1450, https://doi.org/10.1109/ISIT.2014.6875072.

[10] R. Curtis, I. Reiner, Representation Theory of Finite Groups and Associative Algebras, John Wiley and Sons. N.Y., London, 1962.

[11] M. Elía, C. García, Ideal group codes and their syndrome decoding, in: Proc. 21st Int. Symp. Math. Theory Netw. Syst., Groningen, the Netherlands, 2014, pp. 7–11.

[12] T. Fabsic, V. Hromada, P. Stankovski, P. Zajac, Q. Guo, T. Johansson, A reaction attack on the QC-LDPC McEliece cryptosystem, in: Int. Workshop Post-Quantum Cryptogr., 2017, pp. 51–68.

[13] J. Faugère, L. Perret, F. de Portzamparc, Algebraic attack against variants of McEliece with Goppa polynomial of a special form, in: P. Sarkar, T. Twata (Eds.), Adv. Cryptology ASIACRYPT 2014. Lect. Notes Comput. Sci., in: Lect. Notes Comp. Sci., vol. 8873, 2014, pp. 21–41.

[14] C. García, Códigos grupo no abelianos, Tesis Doctoral, Universidad de Oviedo, Oviedo, 2015.

[15] C. García, S. González, V. Markov, C. Martínez, A. Nechaev, Group codes which are not abelian group codes, in: Joaquim Borges, Merce Villanueva (Eds.), Proc. 3rd Int. Castle Meeting Cod. Theory Appl., Universitat Autonoma de Barcelona, Servei de Publicacions, 2011, pp. 123–127.

[16] C. García, S. González, V. Markov, C. Martínez, A. Nechaev, When all group codes of a non commutative group are abelian (a computational approach), J. Math. Sci. 186 (2012) 575–585, https://doi.org/10.1007/s10958-012-1006-x.

[17] C. García, S. González, V. Markov, C. Martínez, A. Nechaev, Group codes over non-abelian groups, J. Algebra Appl. 12 (7) (2013), https://doi.org/10.1142/S0219498813500370.

[18] C. García, S. González, V. Markov, C. Martínez, A. Nechaev, New examples of non-abelian group codes, Adv. Math. Commun. 10 (1) (2016) 1–10, https://doi.org/10.3934/amc.2016.10.1.

[19] C. García, S. González, V. Markov, C. Martínez, Non-abelian group codes over an arbitrary finite field, J. Math. Sci. 223 (5) (2017) 504–507, https://doi.org/10.1007/s10958-017-3363-y.

[20] T. Hungerford, Algebra, Springer Verlag, Nueva York, 2003.

[21] H. Janwa, O. Moreno, McEliece public cryptosystem using algebraic-geometric codes, Des. Codes Cryptogr. 8 (1996) 293–307, https://doi.org/10.1007/BF00173300.

[22] Z. Li, C. Xing, S. Ling Yeo, Reducing the key size of McEliece cryptosystem from automorphism-induced Goppa codes via permutations, in: D. Lin, K. Sako (Eds.), Public-Key Cryptogr. - PKC 2019, in: Lect. Notes Comput. Sci., vol. 11443, 2019, pp. 599–617.

[23] J. McEliece, A public-key cryptosystem based on algebraic coding theory, DSN Progress Report, Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA, 1978, pp. 114–116.

[24] L. Minder, A. Shokrollah, Cryptanalysis of the Sidelnikov cryptosystem, in: M. Naor (Ed.), Adv. Cryptology - EUROCRYPT 2007, in: Lect. Notes Comp. Sci., vol. 4515, Springer, Berlin, Heidelberg, 2007, pp. 347–360.

[25] H. Niederreiter, Knapsack-type cryptosystems and algebraic coding theory, Probl. Control Inf. Theory 15 (2) (1986) 157–166.

[26] E. Persichetti, Compact McEliece keys based on quasi-dyadic Srivastava codes, J. Math. Cryptol. 6 (2) (2012) 149–169, https://doi.org/10.1515/jmc-2011-0099.

[27] S. Shestakov, V. Sidelnikov, On insecurity of cryptosystems based on generalized Reed-Solomon codes, Discrete Math. Appl. 2 (4) (1992) 439–444.

[28] V. Sidelnikov, A public-key cryptosystem based on binary Reed-Muller codes, Discrete Math. Appl. 4 (3) (1994), https://doi.org/10.1515/dma.1994.4.3.191.

[29] Y. Wang, Quantum resistant random linear code based public key encryption scheme RLCE, in: Proc. IEEE Int. Symp. Inf. Theory-ISIT, 2016, pp. 2519–2523.

[30] C. Wieschebrink, Cryptoanalysis of the Niederreiter public key scheme based on GRS subcodes, in: N. Sendrier (Ed.), Post-Quantum Cryptogr. PQCrypto 2010, in: Lec. Notes Comp. Sci., vol. 6061, Springer, Berlin, 2010, pp. 61–72.

[31] C. Wieschebrink, An attack on the modified Niederreiter encryption scheme, in: M. Yung, Y. Dodis, A. Kiayias, T. Malkin (Eds.), Public Key Cryptogr. - PKC 2006, PKC 2006, in: Lect. Notes Comp. Sci., vol. 3958, Springer, Berlin, 2006, pp. 14–26.