Sur la reconstruction des polynômes linéaires : un nouvel algorithme de décodage des codes de Gabidulin

On the reconstruction of linearized polynomials: a new decoding algorithm for Gabidulin codes

Pierre Loidreau [a]

[a] *Laboratoire de Mathématiques Appliquées, ENSTA, 32 bd Victor, 75015 Paris*

Abstract

We describe a reconstruction problem for linearized polynomials. We equally describe a polynomial-time algorithm enabling to solve this problem in a simple case. From this algorithm we deduce an alternative efficient decoding algorithm for Gabidulin codes introduced in 1985 in [3].

Résumé

Nous présentons un problème de reconstruction de polynômes linéaires ainsi qu'un algorithme en temps polynomial de résolution de ce problème dans un cas simple. Nous en déduisons un algorithme alternatif performant de décodage des codes de Gabidulin introduits en 1985 dans [3].

Abridged English version

In 1985, E. M. Gabidulin introduced the rank metric and presented a family of optimal codes for rank metric. He designed a polynomial-time decoding algorithm correcting these codes up to their error-correcting capacity [3]. Another version of the decoding algorithm was presented in 1991 in [4]. Independantly, Roth presented a different approach of this family, together with a decoding algorithm, see [9]. The different algorithms solve the following decoding problem in a finite field $GF(p^m)$.

Decoding$(\mathbf{y}, C, t)$
Find, when it exists, $\mathbf{c} \in C$, *and* $\mathbf{e}$ *where* $Rg(\mathbf{e} \mid GF(p)) \leq t$ *such that* $\mathbf{y} = \mathbf{c} + \mathbf{e}$,

Email address: `Pierre.Loidreau@ensta.fr` (Pierre Loidreau).

where $\mathbf{y}$ is a vector of length n over $GF(p^m)$, C is a Gabidulin code over $GF(p^m)$, t its error-correcting capacity, and $\mathrm{Rg}(\mathbf{e} \mid GF(p))$ is the rank of $\mathbf{e} \in GF(p^m)^n$ when considered as a $m \times n$-matrix over $GF(p)$. Gabidulin codes are closely related to the theory of linearized polynomials introduced by Öre in 1933, 1934 [6,7]. Namely, given a finite field $GF(p^m)$, and $\mathbf{g} = (g_1, \ldots, g_n)$ where the g_i's are $GF(p)$-linearly independent over $GF(p^m)$, the codewords of $Gab(\mathbf{g}, k)$ are the vectors $\mathbf{c} = (q(g_1) \ldots, q(g_n))$, where $q(z) = \sum_{i=0}^{k-1} q_i z^{p^i}$. The polynomial $q(z)$ is by definition a linearized polynomial of p-degree $\leq k-1$. Gabidulin codes are thus evaluation codes of linearized polynomials over elements of $GF(p^m)$. On the algebra of linearized polynomials, we define the following search problem.

Reconstruction$(\mathbf{y} = (y_1, \ldots, y_n), \mathbf{g} = (g_1, \ldots, g_n), k, t)$
Find the set (V, q) where V is a non-zero linearized polynomial of p-degree $\leq t$ and where q is a linearized polynomial of p-degree $< k$, such that $V(y_i) = V \times q(g_i)$, for all $i = 1, \ldots, n$.

Theorem 3.1 shows that solving **Reconstruction**$(\mathbf{y}, \mathbf{g}, k, t)$ implies solving **Decoding**$(\mathbf{y}, Gab(\mathbf{g}, k), t)$.

Let $S = \left((g_i^{[j]})_{i=1,j=0}^{n,k+t-1} \mid (y_i^{[j]})_{i=1,j=0}^{i=n,j=t} \right)$, where from now on $[j]$ states for p^j. We consider the linear system designed by (3) in the French version. This system is in the unknowns $\mathcal{V} \stackrel{def}{=} (v_0, \ldots, v_t)^T$, and $\mathcal{N} \stackrel{def}{=} (n_0, \ldots, n_{k+t-1})^T$. Proposition 3.2 shows that any solution of **Reconstruction**$(\mathbf{y}, \mathbf{g}, k, t)$ gives a solution to this system. Thus, by solving (3) we obtain a set that contains the solutions of **Reconstruction**$(\mathbf{y}, \mathbf{g}, k, t)$.

If the dimension of the space of solutions is 0 or 1 there is a relation between the solutions of (3) and the solution of **Reconstruction**$(\mathbf{y}, \mathbf{g}, k, t)$. If its dimension is larger than 2, then there is no obvious link between them.

In the case we are interested in, that is to know decoding up to the error-correcting capability t of the code, the dimension of the space of solution of (3) is either equal to 0 or to 1 From any solution of (3) we obtain the unique solution, if this solution exists, of **Decoding**$(\mathbf{y}, Gab(\mathbf{g}, k), t)$.

To design our decoding algorithm, we need to solve (3). To do it efficiently in time complexity, we set $S = \left(\begin{array}{c|c} G_1 & Y_1 \\ \hline G_2 & Y_2 \end{array} \right)$, where $G_1 = \left(g_j^{[i]} \right)_{i=0,j=1}^{k+t-1,k+t}$ is the upper-left square matrix of S of size $k+t$. Solving (3) is equivalent to solving the system $\begin{cases} G_1 \mathcal{N} + Y_1 \mathcal{V} = 0, \\ G_2 \mathcal{N} + Y_2 \mathcal{V} = 0. \end{cases}$

Since the g_i's are linearly independent over $GF(p)$, the matrix G_1 is non singular. Solving (3) is therefore equivalent to solving system (4) where $U = -G_1^{-1}$ and $T = -G_2 G_1^{-1}$ are of respective size $(k+t) \times (k+t)$ and $(n-k-t) \times (k+t)$. These matrices do not depend on the vector $\mathbf{y}$ and can thus be precomputed.

Input : The code $Gab(\mathbf{g}, k)$ with error-correcting capability t, a vector $\mathbf{y}$ whose distance to $Gab(\mathbf{g}, k)$ is $\leq t$, and the matrices U and T.
Output : The unique 2-uple $(\mathbf{c} \in Gab(\mathbf{g}, k), \mathbf{e})$ where $\mathbf{y} = \mathbf{c} + \mathbf{e}$ and $\mathrm{Rg}(\mathbf{e} \mid GF(p)) \leq t$.

Decoding procedure :
(i) Solve system (4) and get a solution $(\mathcal{N}_0^T = (n_i)_{i=0}^{k+t-1}, \mathcal{V}_0^T = (v_i)_{i=0}^{t})$.
(ii) Let $N_0(z) = \sum_{j=0}^{k+t-1} n_j z^{[j]}$ et $V_0(z) = \sum_{i=0}^{t} v_i z^{[i]}$. Determine q_0 such that $N_0 = V_0 \times q_0$, [6].
(iii) Return $(\mathbf{c} = (q_0(g_1), \ldots, q_0(g_n)), \mathbf{e} = \mathbf{y} - \mathbf{c})$.

Since in the case of Gabidulin codes $n-k-t = t$ or $t-1$, depending on the parity of the minimum distance of the code, the time complexity of the algorithm is $\approx (k+t)(k+t^2+2t) + t^3/2$ products in $GF(p^m)$.

1. Introduction

En 1985, E. M. Gabidulin [3] introduisit la notion de métrique rang sur les espaces de codes correcteurs. Dans ce même article, il étudia une famille de codes optimaux pour cette métrique, et conçu dans le même temps un algorithme polynomial de décodage jusqu'à leur capacité de correction. Indépendamment, R. Roth redécouvrit ces codes en 1991 [9]. Ces codes, disposant d'algorithmes de décodage en temps polynomial sont utilisés dans la conception de cryptosystèmes à clé publique résistants [4,5,8].

Dans cette note, nous montrons que le problème de décodage de codes de Gabidulin jusqu'à leur capacité de correction est un cas particulier d'un problème plus général que nous nommerons problème de reconstruction des polynômes linéaires, par analogie au célèbre problème de reconstruction de polynômes [10]. Les polynômes linéaires furent étudiés par Öre, [6,7]. Nous décrivons également une méthode générale de résolution de ce problème de reconstruction et, dans le cas des codes de Gabidulin, nous montrons que cette méthode permet de dériver un algorithme performant, polynomial en temps, de décodage de ces codes jusqu'à leur capacité de correction.

2. Préliminaires

Dans la suite, $GF(p^m)$ désigne le corps fini à p^m éléments où p est une puissance d'un nombre premier. Par commodité on notera parfois $[i] \stackrel{def}{=} p^i$.

Définition 2.1 (Métrique rang, [3]) *Étant donné un corps fini $GF(p^m)$, et un vecteur $\mathbf{y} = (y_1, \ldots, y_n) \in GF(p^m)^n$, on définit le rang sur $GF(p)$ de $\mathbf{y}$ comme étant le rang de la matrice formée en prenant les coordonnées des y_i relativement à une base de $GF(p^m)$ sur $GF(p)$. On le note $Rg(\mathbf{y} \mid GF(p))$.*

Définition 2.2 (Codes de Gabidulin, [3]) *Soit k un entier positif, $\mathbf{g} = (g_1, g_2, \ldots, g_n) \in GF(p^m)^n$, où les g_i sont linéairement indépendants sur $GF(p)$. Le code de Gabidulin, noté $Gab(\mathbf{g}, k)$, est défini par*

$$Gab(\mathbf{g}, k) = \left\{(q(g_1), \ldots, q(g_n)) \mid q(x) = \sum_{i=0}^{k-1} q_i x^{[i]},\ q_i \in GF(p^m)\right\}.$$

Un polynôme linéaire ou p-polynôme de p-degré k tel que défini par Öre s'écrit sous la forme $q(x) = \sum_{i=0}^{k} q_i x^{[i]},\ q_i \in GF(p^m)$, [6,7]. Le code $Gab(\mathbf{g}, k)$ est donc un code d'évaluation de polynômes linéaires de p-degré $< k$ sur des éléments linéairement indépendants d'un corps fini. Il est de dimension k. Les codes de Gabidulin sont des codes optimaux pour la métrique rang [3].

En métrique rang le problème de décodage d'un code jusqu'à une distance t s'énonce de la façon suivante, où C désigne un code de longueur n sur $GF(p^m)$, $\mathbf{y}$ est un vecteur de longueur n, et t est un entier positif.

Décodage$(\mathbf{y}, C, t)$
Trouver, s'il en existe $\mathbf{c} \in C$ et $\mathbf{e} \in GF(p^m)^n$ avec $Rg(\mathbf{e} \mid GF(p)) \leq t$ tels que $\mathbf{y} = \mathbf{c} + \mathbf{e}$.

Il existe des algorithmes en temps polynomial résolvant le problème **Décodage**$(\mathbf{y}, Gab(\mathbf{g}, k), t)$ où t désigne la capacité de correction du code de Gabidulin $Gab(\mathbf{g}, k)$ [3,4].

3. Le problème de reconstruction des polynômes linéaires

Les polynômes linéaires sur $GF(p^m)$ sont les endomorphismes linéaires de $GF(p^m)$ considéré comme espace vectoriel sur $GF(p)$. Munis des lois d'addition, notée $+$, et de composition, notée $\times$, des applications linéaires ils forment une algèbre sur $GF(p)$. On définit le problème de reconstruction de polynômes linéaires, où $\mathbf{y} = (y_1, \ldots, y_n)$ et $\mathbf{g} = (g_1, \ldots, g_n)$ sont des vecteurs quelconques de $GF(p^m)^n$ et, où k et t sont des entiers positifs.

Reconstruction$(\mathbf{y}, \mathbf{g}, k, t)$
Trouver tous les couples (V, q) où V est un polynôme linéaire non nul de p-degré $\leq t$ et q est un polynôme linéaire de p-degré $< k$, vérifiant $V(y_i) = V \times q(g_i),\ \forall i = 1, \ldots, n$.

Si $g_1, \ldots, g_n$ sont linéairement indépendants sur $GF(p)$, alors le théorème suivant donne une relation entre **Reconstruction**$(\mathbf{y}, \mathbf{g}, k, t)$ et **Décodage**$(\mathbf{y}, Gab(\mathbf{g}, k), t)$.
Théorème 3.1
Une solution à **Reconstruction**$(\mathbf{y}, \mathbf{g}, k, t)$, *donne une solution à* **Décodage**$(\mathbf{y}, Gab(\mathbf{g}, k), t)$.

Preuve
Soit $\mathcal{L}$ l'ensemble des solutions de **Reconstruction**$(\mathbf{y}, \mathbf{g}, k, t)$. Prenons $(V_1, q_1) \in \mathcal{L}$. Alors pour tout i, $V_1(y_i) = V_1 \times q_1(g_i)$ soit, par linéarité de V_1, $V_1(y_i - q_1(g_i)) = 0$ pour tout i. Ces égalités sont strictement équivalentes aux égalités $e_i = y_i - q_1(g_i)$ appartiennent à un espace vectoriel sur $GF(p)$ de dimension au plus égale au p-degré de V_1 soit, au plus égal à t [6,7]. Ce qui est encore équivalent à écrire $y_i = q_1(g_i) + e_i$ pour tout i, où le vecteur $e = (e_1, \ldots, e_n)$ est de rang au plus égal à t. Donc (c, e) où $c = (q_1(g_1), \ldots, q_1(g_n))$ est une solution de **Décodage**$(\mathbf{y}, Gab(\mathbf{g}, k), t)$. $\diamondsuit$

On s'intéresse désormais à la résolution du problème **Reconstruction**$(\mathbf{y}, \mathbf{g}, k, t)$ qui est plus général que **Décodage**$(\mathbf{y}, Gab(\mathbf{g}, k), t)$. Résoudre **Reconstruction**$(\mathbf{y}, \mathbf{g}, k, t)$ correspond à résoudre

$$V(y_i) = V \times q(g_i), \quad \forall i = 1, \ldots, n, \tag{1}$$

dont les inconnues sont les polynômes linéaires $V(z) \stackrel{def}{=} \sum_{i=0}^{t} v_i z^{[i]}$ et $q(z) \stackrel{def}{=} \sum_{j=0}^{k-1} q_j z^{[j]}$. C'est un système quadratique de n équations à $k+t+1$ inconnues. Il n'existe pas de résultat générique concernant la complexité exacte de résolution de tels systèmes. Les meilleurs connus consistent à trouver des base de Gröbner de l'idéal engendré par ces équations. Considérons désormais le système d'équations

$$V(y_i) = N(g_i), \quad \forall i = 1, \ldots, n, \tag{2}$$

dont les inconnues sont les polynômes linéaires $V(z) \stackrel{def}{=} \sum_{i=0}^{t} v_i z^{[i]}$ et $N(z) \stackrel{def}{=} \sum_{j=0}^{k+t-1} n_j z^{[j]}$. C'est un système linéaire dont les inconnues sont les $k + 2t + 1$ coefficients de $N(z)$ et de $V(z)$.
Proposition 3.2 *L'ensemble des solutions du système* (2) *contient toutes les solutions du système* (1).

Preuve
Soit (V_0, q_0) une solution de (1), le couple $(V_0, N_0 = V_0 \times p_0)$ est une solution de (2). $\diamondsuit$

On s'attache dorénavant à la résolution du système (2).

Écrivons $\mathcal{V} \stackrel{def}{=} (v_0, \ldots, v_t)^T$, et $\mathcal{N} \stackrel{def}{=} (n_0, \ldots, n_{k+t-1})^T$. Posons $S = \left.\left(\begin{array}{ccc|ccc} g_1 & \cdots & g_1^{[k+t-1]} & y_1 & \cdots & y_1^{[t]} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ g_n & \cdots & g_n^{[k+t-1]} & y_n & \cdots & y_n^{[t]} \end{array}\right)\right\} n.$

Résoudre le système (2) est équivalent à résoudre le système matriciel aux inconnues $\mathcal{N}$ et $\mathcal{V}$

$$S \times \begin{pmatrix} \mathcal{N} \\ \mathcal{V} \end{pmatrix} = 0. \tag{3}$$

L'ensemble des solutions de ce système forme un espace vectoriel $\mathcal{S}$ dont la dimension nous donne des informations sur les solutions du système (1).

(i) Si $\mathcal{S}$ est de dimension 0, la seule solution possible de (3), donc de (1), est le vecteur nul. Comme 0 n'est pas éligible en tant que solution, il n'y a pas de solution au problème $\mathbf{Reconstruction}(\mathbf{y}, \mathbf{g}, k, t)$.

(ii) Si $\mathcal{S}$ est de dimension 1, on peut montrer que, quelle que soit la solution de (3), elle donne toujours la même solution pour (1), quand celle-ci existe. Donc $\mathbf{Reconstruction}(\mathbf{y}, \mathbf{g}, k, t)$ a au plus une solution.

(iii) si $\mathcal{S}$ est de dimension $s \geq 2$, le nombre de solutions de (3) est égal à p^{ms}, et il n'y a pas de lien évident entre les espaces de solutions de (1) et de (3). Pour déterminer les solutions de $\mathbf{Reconstruction}(\mathbf{y}, \mathbf{g}, k, t)$, on énumère les $p^{ms} - 1$ solutions non nulles du système (3), et on teste leur validité.

4. Algorithme de décodage des codes de Gabidulin

Nous venons d'établir un lien entre le décodage des codes de Gabidulin et le problème de reconstruction de polynômes linéaires. Considérons le décodage jusqu'à la capacité de correction. Cela correspond au cas où le récepteur reçoit le vecteur $\mathbf{y}$ de longueur n tel que $\mathbf{y} = \mathbf{c} + \mathbf{e}$ avec $\mathrm{Rg}(\mathbf{e} \mid GF(p)) \leq t = \lfloor (n-k)/2 \rfloor$, et $c = (q(g_1), \ldots, q(g_n))$ où q est un polynôme linéaire de p-degré $< k$. La matrice S correspondant au système (3) est donc une matrice de taille $n \times (k + 2t + 1)$, où $k + 2t + 1 \leq n$. On peut montrer dans ce cas que l'espace vectoriel des solutions $\mathcal{S}$ est de dimension au plus égale à 1. Une solution du problème $\mathbf{Reconstruction}(\mathbf{y}, \mathbf{g}, k, t)$ permet donc de retrouver la solution que l'on sait unique du problème de décodage. Il s'ensuit l'algorithme de décodage suivant :

ENTRÉE : le code $Gab(\mathbf{g}, k)$ de capacité de correction t, un vecteur $\mathbf{y}$ à distance inférieure à t de $Gab(\mathbf{g}, k)$.
SORTIE : L'unique couple $(\mathbf{c}, \mathbf{e})$, où $\mathbf{c} \in Gab(\mathbf{g}, k)$ tel que $\mathbf{y} = \mathbf{c} + \mathbf{e}$, avec $\mathrm{Rg}(\mathbf{e} \mid GF(p)) \leq t$
ALGORITHME DE DÉCODAGE :

(i) Construction de la matrice S utilisée dans le système (3).

(ii) Résolution de (3). On choisit une solution $(\mathcal{N}_0 = (n_i)_{i=0}^{k+t-1}, \mathcal{V}_0 = (v_i)_{i=0}^{t})$.

(iii) Soit $N_0(z) = \sum_{j=0}^{k+t-1} n_j z^{[j]}$ et $V_0 = \sum_{i=0}^{t} v_i z^{[i]}$. Déterminer q_0 tel que $N_0 = V_0 \times q_0$, par division euclidienne.

(iv) Retourner $(\mathbf{c} = (q_0(g_1), \ldots, q_0(g_n)), \mathbf{e} = \mathbf{y} - \mathbf{c})$.

En négligeant la complexité de l'addition dans $GF(p^m)$ et l'élévation à la puissance $p$ième, par rapport au coût d'une multiplication dans $GF(p^m)$, la complexité totale en temps de l'algorithme est $\approx (k + 2t)^3/2 + t(k+1) + kn$ multiplications. Le terme au cube $(k+2t)^3$ prépondérant correspond à une méthode de pivot de Gauss afin de résoudre le système (3), $t(k+1)$ correspond au coût de la division euclidienne par l'algorithme de Öre, et kn correspond au calcul de $\mathbf{c}$ en évaluant le polynôme q sur les g_i.

Par rapport aux autres algorithmes de décodage, la complexité de celui-ci n'est pas bonne. En effet, le facteur cubique est en général très grand. Cependant, comme la matrice S a une partie fixe qui ne dépend que de $\mathbf{g}$ et une partie dépendant du vecteur $\mathbf{y}$ reçu, on peut diminuer notablement la complexité de l'algorithme.

Écrivons $S = \left(\begin{array}{c|c} G_1 & Y_1 \\ \hline G_2 & Y_2 \end{array} \right)$, où $G_1 = (g_i^{[j]})_{i=1,j=0}^{k+t,k+t-1}$ est la matrice carrée supérieure gauche de taille $k + t$ de S. Résoudre (3) revient à résoudre $\begin{cases} G_1 \mathcal{N} + Y_1 \mathcal{V} = 0, \\ G_2 \mathcal{N} + Y_2 \mathcal{V} = 0. \end{cases}$

Comme les g_i sont linéairement indépendants sur $GF(p)$, la matrice G_1 est inversible. Résoudre le système (3) est donc équivalent à résoudre le système suivant :

$$\begin{cases} \mathcal{N} = U \times (Y_1 \mathcal{V}), \\ ((T \times Y_1) + Y_2)\mathcal{V} = 0, \end{cases} \tag{4}$$

où $U = -G_1^{-1}$ et $T = -G_2 G_1^{-1}$ sont deux matrices de taille respectivement $(k+t) \times (k+t)$ et $(n-k-t) \times (k+t)$. Celles-ci ne dépendent pas du vecteur $\mathbf{y}$ reçu et peuvent donc être précalculées.

Entrée : Le code $Gab(\mathbf{g}, k)$ de capacité de correction t, un vecteur $\mathbf{y}$ à distance inférieure à t de $Gab(\mathbf{g}, k)$, les matrices U et T.
Sortie : L'unique couple $(\mathbf{c}, \mathbf{e})$, où $\mathbf{c} \in Gab(\mathbf{g}, k)$ tel que $\mathbf{y} = \mathbf{c} + \mathbf{e}$, avec $\mathrm{Rg}(\mathbf{e} \mid GF(p)) \leq t$.
Algorithme de décodage :

(i) Résolution du système (4) par pivot de Gauss et multiplications matricielles. On choisit une solution $(\mathcal{N}_0 = (n_i)_{i=0}^{k+t-1}, \mathcal{V}_0 = (v_i)_{i=0}^{t})$.
(ii) Soit $N_0(z) = \sum_{j=0}^{k+t-1} n_j z^{[j]}$ et $V_0(z) = \sum_{i=0}^{t} v_i z^{[i]}$. Déterminer q_0 tel que $N_0 = V_0 \times q_0$, par division euclidienne.
(iii) Retourner $(\mathbf{c} = (q_0(g_1), \ldots, q_0(g_n)), \mathbf{e} = \mathbf{y} - \mathbf{c})$.

Seule la première étape de résolution du système est modifiée par rapport à l'algorithme précédent. Comme, dans notre cas, $n-k-t = t$ ou bien $t-1$ suivant la parité de $n-k$, on passe d'un coût de résolution en $(k+2t)^3/2$ pour le premier algorithme, à une complexité de $\approx t^2(k+t) + t^3/2 + t(k+t) + (k+t)^2 = (k+t)(k+t^2+2t) + t^3/2$. Le gain de complexité entre les deux algorithmes est très significatif.

A titre comparatif, la complexité des algorithmes décrits dans [4,9] est sensiblement égale à $t(2n+m)+t^3$ opérations dans le corps $GF(p^m)$.

Références

[1] Berlekamp (E. R.). – *Algebraic Coding Theory.* – Aegean Press Park, 1984, revised 1984 édition.

[2] Berlekamp (E. R.), Welch (L.) . – Error Correction of Algebraic Block Codes, *US Patent, Number 4,633,470,* 1986.

[3] Gabidulin (E. M.). – Theory of codes with maximal rank distance. *Problems of Information Transmission*, vol. 21, juillet 1985, pp. 1–12.

[4] Gabidulin (E. M.). – A fast matrix decoding algorithm for rank-error correcting codes. *In : Algebraic coding*, éd. par Cohen (G.), Litsyn (S.), Lobstein (A.) et Zémor (G.), LNCS. pp. 126–133. – Springer-Verlag, 1991.

[5] Gabidulin (E .M.), Paramonov (A. V.) et Tretjakov (O. V.). – Ideals over a non-commutative ring and their application in cryptology. *LNCS*, vol. 573, 1991, pp. 482 – 489.

[6] Öre (O.). – On a special class of polynomials. *Transactions of the American Mathematical Society*, vol. 35, 1933, pp. 559–584.

[7] Öre (O.). – Contribution to the theory of finite fields. *Transactions of the American Mathematical Society*, vol. 36, 1934, pp. 243–274.

[8] Ourivski (A. V.), Gabidulin (E. M.), Honary (B.), Ammar (B.). – Reducible rank codes and their applications to cryptography. *IEEE Transactions on Information Theory*, vol. 49, nN° 12, décembre 2003, pp 3289–3293.

[9] Roth (R. M.). – Maximum-Rank array codes and their application to crisscross error correction. *IEEE Transactions on Information Theory*, vol. 37, nN° 2, mars 1991, pp. 328–336.

[10] Sudan (M.). – Decoding Reed-Solomon codes beyond the error-correction diameter. *Proceedings of the 35th Annual Allerton Conference on Communication, Control and Computing* pp 215–224, 29 septembre – 1er octobre, 1997.