A new constellation for space-time coding

Gwezheneg Robert

A new constellation for space-time coding

Gwezheneg Robert[1,2]

[1] IRMAR – Université de Rennes
[2] INRIA – Saclay - Île de France
gwezheneg.robert@inria.fr

Abstract. In this paper, we propose a new encoding scheme for space-time coding. This encoding relies on the generalisation [2] of Gabidulin codes to infinite fields. This enables to make use of the properties of Gabidulin codes, namely, we can use algebraic decoding algorithms.

Keywords: Gabidulin codes, space-time coding, number fields

1 Introduction

In the field of space-time coding, data is transmitted by the modulation of a carrier signal. Each antenna emits a signal which is modelled by a complex number. Only a finite number of elements is used, thus we send a coefficient in a finite subset of $\mathbb{C}$, called *constellation*. Thus, we need to encode binary data into complex numbers.

In 2005, Lu and Kumar [6] proposed an encoding scheme, thanks to a construction based on binary Gabidulin codes and to a mapping $(\mathbb{F}_2)^u \to \mathbb{C}$. The main drawback of this construction is that there is no polynomial time decoding algorithm, therefore a complete enumeration of the codewords has to be done.

In this paper, we propose a new encoding scheme, based on the generalisation of Gabidulin codes to infinite fields, which enables a polynomial time decoding. This construction gives rise to a new family of constellations.

In a first part, we introduce the mathematical objects used in this paper. In a second part, we present the encoding scheme proposed by Lu and Kumar. Then, we present our encoding and decoding schemes. In a third part, we propose a new family of constellations, related to our encoding scheme. We provide an example in a fourth part.

2 Preliminaries

In this part, we introduce notations and mathematical objects used in this paper.

2.1 Notations

We consider a field extension $K \hookrightarrow L$ of finite degree $[L : K] = m$, provided with a K-automorphism θ. Moreover, we make the hypothesis that:

– the K-automorphisms group of the extension is cyclic,
– θ is a generator of this group.

L can be seen as a K-linear space of dimension m, and we denote by $\mathcal{B} = (b_1, \ldots, b_m)$ a K-basis. This framework is that of finite fields provided with the Frobenius, but is also that of some number fields, for example Kummer extensions. This hypothesis is essential to have the main properties of Gabidulin codes, namely Theorems 1 to 6.

2.2 θ-polynomials

θ-polynomials have been introduced by Ore in the 30's ([7], [8]) and have properties similar to classical polynomials.

Definition 1 (θ-polynomials). *A* θ-polynomial *with coefficients in* L *is an element on the form* $\sum_{i \geqslant 0} a_i X^i$, $a_i \in L$ *with a finite number of non-zero* a_i. *The upper index* i *for which* $a_i \neq 0$ *is called the degree of the* θ*-polynomial. We denote by* $L[X;\theta]$ *the set of* θ*-polynomials, provided with the following operations. Namely, let* $A = \sum a_i X^i$, $B = \sum b_i X^i \in L[X;\theta]$ *and* $c \in L$.

– *The addition is defined component-wise:* $A + B = \sum_i (a_i + b_i) X^i$
– *The (symbolic) product for monomials is defined by* $X \cdot c = \theta(c) \cdot X$
  Extended to polynomials by distributivity, it gives: $A \cdot B = \sum_{i,j} a_i \theta^i(b_j) X^{i+j}$
– *The (operator) evaluation is defined by:* $\mathcal{L}_A(c) = \sum_i a_i \theta^i(c)$

$L[X;\theta]$ is a non-commutative ring, with no zero divisor. There are a right and a left Euclidean divisions.

The root-space of a θ-polynomial P is the set $\{a \in L : \mathcal{L}_P(a) = 0\}$. Since by hypothesis θ is a generator of the cyclic group of K-automorphisms, we have the following properties.

Theorem 1. *The root-space of a* θ*-polynomials* P *is a* K*-linear subspace whose dimension is upper-bounded by the degree of* P.

Theorem 2. *For any* K*-linear subspace* V *of dimension* s, *there exists a* θ*-polynomial* P *of degree exactly* s *which vanishes on* V, *i.e.* $\mathcal{L}_P(v) = 0, v \in V$, *and none of lower degree.*

2.3 Error model

We consider a channel consisting of m antennas in emission. The transmission duration n is cut into time intervals during which each antenna emits a signal. This signal is modeled by a complex number. We represent sent coefficients in a matrix of size $m \times n$. Each row of this matrix contains elements sent by a same antenna, whereas each column contains elements sent at a same time.

We assume that errors doesn't occur independently on each coefficients, but occur by row or column. Such a model is inspired by interferences that affects few antennas for example in jamming, or by a perturbation during a short instant.

If we send a codeword C, we receive a noisy word $Y = C + E$, where E is an error. We use a metric adapted to errors distributed by rows and columns, called the rank metric and defined in the next paragraph. A matrix E of rank r can be decomposed as $E = AB$, where A and B have size $m \times r$ and $r \times n$ and both rank r. From the nature of errors, matrix A or B can be known from the receiver [4]. This kind of partially known errors are called erasures.

Thus, sending a codeword C, we receive

$$Y = C + A_r B_r + A_c B_c + A_f B_f,$$

where right-hand size matrices have respectively sizes $m \times n$, $m \times s_r$, $s_r \times n$, $m \times s_c$, $s_c \times n$, $m \times r$ and $r \times n$. Matrices A_r and B_c are known by the receiver. Thus, $A_r B_r$ is a row erasure of rank s_r, $A_c B_c$ is a column erasure of rank s_c, and $A_f B_f$ is a full error of rank r.

2.4 Rank metric

In this paragraph, we define the rank metric, well adapted to our error model.

Definition 2. *Let $\boldsymbol{x} = (x_1, \ldots, x_n) \in L^n$. We decompose x_i in the basis $\mathcal{B}$, to get $x_i = \sum_{j=0}^{m} x_{i,j} b_j$. Thus, the weight of $\boldsymbol{x}$ is*

$$w(\boldsymbol{x}) = \operatorname{rank} \begin{pmatrix} x_{1,1} & \cdots & x_{1,m} \\ \vdots & \ddots & \vdots \\ x_{n,1} & \cdots & x_{n,m} \end{pmatrix}.$$

Since by hypothesis θ is a generator of the cyclic group of K-automorphisms, we have the following property.

Theorem 3. *Let $\boldsymbol{x} = (x_1, \ldots, x_n) \in L^n$. We denote by $\mathcal{A}$ the lower degree non-zero polynomial which vanishes on the K-linear subspace generated by the x_i's. Then,* $\deg(\mathcal{A}) = w(\boldsymbol{x})$.

2.5 Generalised Gabidulin codes

There are several families of rank metric codes. In this article, we focuses on Gabidulin codes, since they are optimal and since we have polynomial time decoding algorithms.

Original Gabidulin codes were design over finite fields [3]. We have generalised them to infinite fields in [2], restricted to extensions that satisfies the conditions of section 2.1. Indeed, this hypothesis is essential to have the main properties of Gabidulin codes (optimality and efficient decoding algorithms).

Definition 3. *Let $k < n \leqslant m$ be integers. Let $g_1, \ldots, g_n$ be K-linearly independent elements of L. The generalised Gabidulin code of dimension k, length n and support $\boldsymbol{g} = (g_1, \ldots, g_n)$ is*

$$Gab_{\theta,k}(\boldsymbol{g}) = \{(\mathcal{L}_f(g_1), \ldots, \mathcal{L}_f(g_n)) : f \in L[X;\theta], \deg(f) < k\}.$$

The minimal distance verify the Singleton-like bound: $d \leqslant n - k + 1$. Codes that reaches this bound are called *Maximum Rank Distance (MRD) codes.*

Theorem 4. *Generalised Gabidulin codes are MRD codes.*

A *codeword* can be written as a vector of length n over L, or equivalently as a matrix of size $m \times n$ over K, by expanding each coordinate of the vector in a K-basis of L. Similarly, the *information word* can be seen as a vector of length k over L, as a matrix of size $m \times k$ over K or as the coefficients of a polynomial of degree lower than k and with coefficients in L.

Example 1. Consider the following field extensions of $\mathbb{Q}$.

$$\mathbb{Q} \hookrightarrow K = \mathbb{Q}[X]/(X^2 + 1) = \mathbb{Q}[i]$$

$$K \hookrightarrow L = K[Y]/(Y^4 - 2) = K[\alpha]$$

$$\theta \text{ is defined by } \theta : \alpha \mapsto i\alpha$$

Then, we define a $[4, 2, 3]$ generalised Gabidulin code by defining its support $\boldsymbol{g} = (1, \alpha, \alpha^2, \alpha^3)$. Finally, we consider a K-basis of L, for example $\mathcal{B} = (1, \alpha, \alpha^2, \alpha^3)$. Here is a example of information word, as a vector, as a matrix and as a θ-polynomial.

$$(i, \alpha) \in L^2 \leftrightarrow \begin{pmatrix} i & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \in \mathcal{M}_{4,2}(K) \leftrightarrow i + \alpha X \in L[X; \theta]$$

The corresponding codeword, as a vector and as a matrix, is the following.

$$(i + \alpha, i\alpha + i\alpha^2, i\alpha^2 - \alpha^3, -2i + i\alpha^3) \in L^4 \leftrightarrow \begin{pmatrix} i & 0 & 0 & -2i \\ 1 & i & 0 & 0 \\ 0 & i & i & 0 \\ 0 & 0 & -1 & i \end{pmatrix} \in \mathcal{M}_{4,4}(K)$$

Theorem 5. *According to our error model, assume that the codeword C is send, and that a full error of rank r, a row erasure of rank s_r and a column erasure of rank s_c are added. Thus, we receive*

$$Y = C + A_r B_r + A_c B_c + A_f B_f.$$

Then, we can recover the information word if

$$2r + s_c + s_r \leqslant d - 1 = n - k.$$

Theorem 6. *In this case, generalised Gabidulin codes can be decoded in $O(n^2)$ operations over the field L.*

3 Gabidulin codes and constellations

In this part, we aim to design a coding scheme for space-time coding. This encoding has two objectives. It should encode binary data into complex numbers, and apply an error correcting code adapted to the channel.

Lu and Kumar have proposed such a scheme in 2005 [6], but they propose no decoding algorithm. We present another scheme which enables a polynomial time decoding algorithm.

We consider $\mathcal{F} : (\mathbb{F}_2)^u \to \mathbb{C}$. This mapping should be injective. We also consider a (binary or generalised) Gabidulin code

$$\mathcal{G} : \mathcal{M}_{k,m}(F) \to \mathcal{M}_{n,m}(F)$$

where F denotes the base field $\mathbb{F}_2$ or K for number fields.

We present Lu-Kumar's scheme and our one. In both cases, data to be encoded are distributed in u binary matrices of size $k \times m$.

3.1 Lu-Kumar's scheme

Lu-Kumar's scheme (see the first line of Figure 1) begins by applying a binary Gabidulin code

$$\mathcal{G} : \mathcal{M}_{k,m}(\mathbb{F}_2) \to \mathcal{M}_{n,m}(\mathbb{F}_2).$$

These matrices are then merged into a single one applying $\mathcal{F}$ position by position.

$$\mathcal{F}_{n,m} : (\mathcal{M}_{n,m}(\mathbb{F}_2))^u \to \mathcal{M}_{n,m}(\mathbb{C}).$$

The set of all possible coefficients of these matrices is a finite subset of $\mathbb{C}$ denoted by $\mathcal{Q}$. The set of all possible matrices make an error correcting code $\mathcal{C}$ in rank metric, whose minimal distance is those of the binary Gabidulin code used.

After transmission, the received matrix is approximated to get a matrix with coefficients in $\mathcal{Q}$ (see the third line of Figure 1). Then, we compute the matrix of $\mathcal{C}$ closest to the received one (Closest Codeword).

$$CC : \mathcal{M}_{n,m}(\mathcal{Q}) \to \mathcal{C} \subset \mathcal{M}_{n,m}(\mathcal{Q})$$

This matrix is then decomposed in u matrices by reversing $\mathcal{F}$

$$\mathcal{F}_{n,m}^{-1} : \mathcal{M}_{n,m}(\mathbb{C}) \to (\mathcal{M}_{n,m}(\mathbb{F}_2))^u.$$

Each of these matrices, which is a codeword for the binary Gabidulin code, is decoded. Since the error has already been corrected, this step is immediate for a systematic code.

3.2 Our scheme

Permuting operations $\mathcal{G}$ and $\mathcal{F}$, we can take advantage of the structure of the Gabidulin code, which enable us to use a polynomial-time decoding algorithm.

In our scheme (see the second line of Figure 1), matrices are first merged into a single complex matrix by

$$\mathcal{F}_{n,m} : (\mathcal{M}_{k,m}(\mathbb{F}_2))^u \to \mathcal{M}_{k,m}(\mathbb{C}).$$

We denote by $\mathcal{Q}_0 \subset \mathbb{C}$ the set of all possible coefficients in this matrix. Then we apply a generalised Gabidulin code

$$\mathcal{G} : \mathcal{M}_{k,m}(\mathcal{Q}_0 \subset K) \to \mathcal{M}_{n,m}(\mathcal{Q} \subset K).$$

The set of all possible coefficients after this encoding makes another constellation $\mathcal{Q}$.

After transmission and estimation (see the fourth line of Figure 1), we can apply any decoding algorithm

$$\mathcal{G}^{-1} : \mathcal{M}_{n,m}(\mathcal{Q} \subset K) \to \mathcal{M}_{k,m}(\mathcal{Q}_0 \subset K),$$

then we reverse $\mathcal{F}$

$$\mathcal{F}_{n,m}^{-1} : \mathcal{M}_{k,m}(\mathcal{Q}_0) \to (\mathcal{M}_{k,m}(\mathbb{F}_2))^u.$$

3.3 Differences between the two encoding schemes

Using a decoding algorithm specific to Gabidulin codes, we can recover the initial data in a polynomial time. Such an algorithm is described in [5] for finite fields and generalised in [1] (in preparation). Thus, the computation of the closest codeword (CC), in $O(|\mathcal{Q}|^{n^2})$ operations, is replaced by an algebraic decoding, in $O(n^2)$.
The mapping $\mathcal{F}_{n,m}$ has no requisites, thus any bijection $\mathcal{F}$ easy to reverse can be used. Remark that the constellation used for the transmission is not directly the image of $\mathcal{F}$. It depends on the kind of field extension $K \hookrightarrow L$, on the Gabidulin code (parameters k and n, support $\boldsymbol{g} = (g_1, \ldots, g_n)$), and on the mapping $\mathcal{F}$ (image $\mathcal{Q}_0$, parameter u). We can choose these parameters to get convenient constellations.

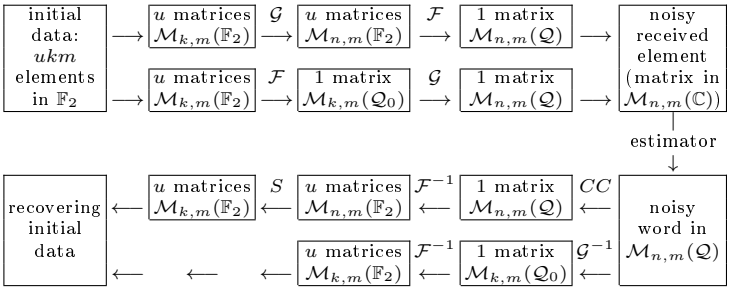

Figure 1 This figure represents the different coding and decoding schemes described in this section:
(line 1): Lu-Kumar's encoding scheme
(line 2): our encoding scheme
(line 3): decoding Lu-Kumar's scheme
(line 4): decoding our scheme

4 A new constellation

We have seen in the previous part that the constellation $\mathcal{Q}$ depends on the image of the mapping $\mathcal{F}$ and on the Gabidulin code. In this part, we present a mapping $\mathcal{F}$ and a Gabidulin code which enables to have a constellation $\mathcal{Q}$ with nice properties.

Let m be an integer and $K = \mathbb{Q}[\zeta]$ be a field containing all m-th roots of unity. It is a cyclotomic extension of degree $\varphi(m)$ where φ is the Euler phi function. Let $y \in K$ be an element such that $x^d = y$ has no solution for any divisor d of m. Then the field $L = K[\alpha] = K[Y]/(Y^m - y)$ is called a Kummer extension. The group of K-automorphisms is cyclic and a generator is $\theta : \alpha \mapsto \zeta\alpha$. We consider the $\mathbb{Q}$-basis $(\zeta^1, \ldots, \zeta^{\varphi(m)})$ of K. Finally, we consider the generalised Gabidulin code defined on this extension, and with support $(1, \alpha, \ldots, \alpha^{n-1})$. This code has length $n = m$ and dimension k.

Let $u = \lfloor \log_2(m-1) \rfloor$. The mapping from binary data to K is defined by

$$\mathcal{F} : (\mathbb{F}_2)^u \to \mathcal{Q}_0 \subset L : \overline{a_{u-1}, \ldots, a_0} \mapsto \zeta^a$$

where $\overline{a_{u-1}, \ldots, a_0}$ denotes the decomposition of a in basis 2.

Thus, the coefficient constellation $\mathcal{Q}_0$ is a subset of $u = \lfloor \log_2(m-1) \rfloor$ n-th roots of unity. Remark that the constellation is more regular if m is a power of 2, in which case all roots are present.

Then, the information symbols, *i. e.* the coefficients of the polynomials to be encoded, are selected in the coefficients constellation $\mathcal{Q}_0$. The modulation constellation $\mathcal{Q}$ is the image of all polynomials with coefficients in $\mathcal{Q}_0$ by the encoding function, and after decomposition in the K-basis of L.

Theorem 7. *With a Kummer extension and a Gabidulin code as define above, the modulation constellation $\mathcal{Q}$ is the set of all sums of k n-th roots of the unity.*

Proof. Indeed, $f = \sum_{i=0}^{k-1} f_i X^i = \sum_{i=0}^{k-1} (\sum_{j=1}^{m} f_{i,j} \alpha^j) X^j$. Thus, its evaluation in an element g_l of the support gives : $\mathcal{L}_f(g_l) = \sum_{j=1}^{m} (\sum_{i=0}^{k-1} f_{i,j} \zeta^{ij}) \alpha^{j+l}$. Thus, the decomposition of the evaluation along the basis $\mathcal{B}$ gives coefficients on the form $\sum_{i=0}^{k-1} f_{i,j} \zeta^{ij}$, which are sums of k n-th roots of the unity. (Or $y \sum_{i=0}^{k-1} f_{i,j} \zeta^{ij}$ if $j + l > m$, which corresponds to coefficients above the diagonal.)

Theorem 8. *Constellations designed on this way have the following properties:*

- *elements closer to* 0 *are more frequently used than elements of large radius.*
- *the number of element in the constellations depends on* k*.*
- *the maximal radius is* k
- 0 *is an element of the constellation if and only if we can write* $k = d_1 + \cdots + d_s$*, where the* d_i*'s are divisors of* n *other than* 1*.*

Remark 1. We aim to build a constellation with few elements. Thanks to the choice of a Kummer extension, we can take advantage of the properties of roots of unity. Thus, the image $\mathcal{Q}_0$ of $\mathcal{F}$, and the support of the code are roots of unity. This set is stable by multiplication, and by application of θ (after decomposition in the basis $\mathcal{B}$). Thus, the product of a monomial of f by a coefficient in $\mathcal{Q}_0$ doesn't create new values. Only the sum of monomials is responsible to the size of the constellation $\mathcal{Q}$.

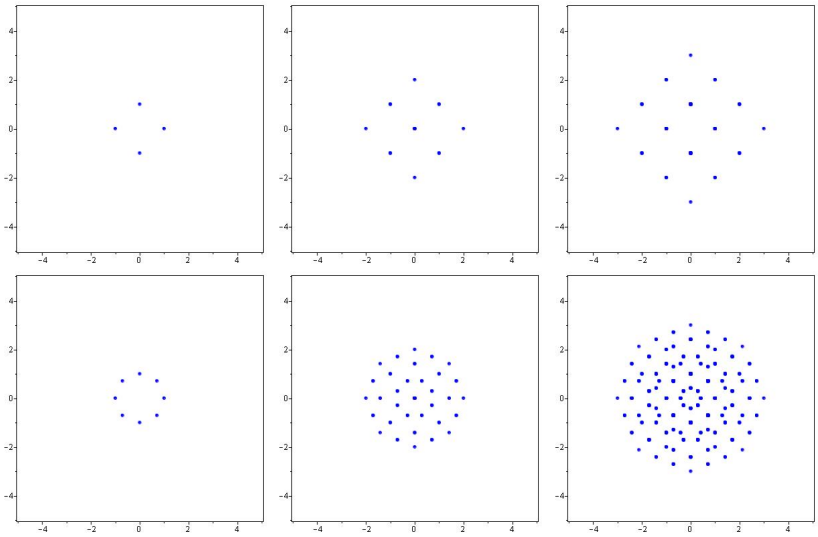

Figure 2 Some constellations:
Top: constellations for $n = m = 4$, $u = 2$ and $k = 1, 2, 3$
Bottom: constellations for $n = m = 8$, $u = 3$, and $k = 1, 2, 3$

5 An example of constellation

Example 2. In this section, we dispose of a channel with 4 antennas. We assume that to a matrix 4×4 send throughout this channel is added an error whose

weight is at most 1. Thus, we can use a $[4, 2, 3]$ generalised Gabidulin code.

We consider the following field extensions of $\mathbb{Q}$.

$$\mathbb{Q} \hookrightarrow K = \mathbb{Q}[X]/(X^2 + 1) = \mathbb{Q}[i]$$

$$K \hookrightarrow L = K[Y]/(Y^4 - 2) = K[\alpha]$$

$$\theta \text{ is defined by } = \theta : \alpha \mapsto i\alpha$$

Then, we define a $[4, 2, 3]$ generalised Gabidulin code by defining its support $\boldsymbol{g} = (1, \alpha, \alpha^2, \alpha^3)$. Finally, we consider a K-basis of L, for example $\mathcal{B} = (1, \alpha, \alpha^2, \alpha^3)$. We can combine 2 bits as follows to define an element of K:

$$\mathcal{F} : \overline{\beta_1\beta_0} \mapsto i^{(2\beta_1+\beta_0)} = (-1)^{\beta_1} i^{\beta_0} \in \mathcal{Q}_0 = \{\pm 1, \pm i\}.$$

With 16 bits, we can define a polynomial $f_0 + f_1 X$, which is encoded and send throughout the channel.

The modulation constellation $\mathcal{Q}$, which is the set of all possible coefficients in the matricial codeword, is

$$\{0, 2, -2, 2i, -2i, i + 1, i - 1, -i + 1, -i - 1\}$$

for the elements on and above the diagonal, and

$$\{0, 4, -4, 4i, -4i, 2i + 2, 2i - 2, -2i + 2, -2i - 2\}$$

for the elements over it. This is due to the Kummer extension, indeed, $\alpha^4 = 2$. But we can divide these elements by 2 before transmission, and multiply them by 2 at the reception to have once constellation.

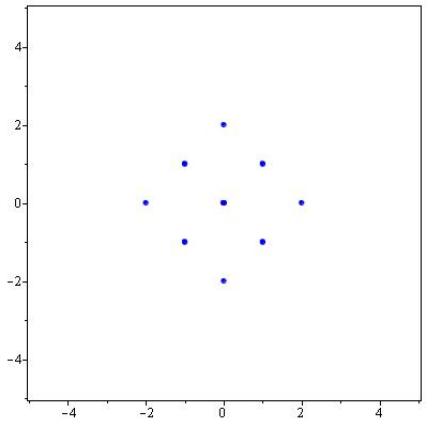

Figure 3 The constellation of the example

We now compare our constellations to classical ones. For the same rate (number of bits per channel use), Lu-Kumar's constructions deals with a constellation

of four elements.

constellation	PAM	QAM	PSK	$\mathcal{Q}$
elements	$\{\pm 3, \pm 1\}$	$\{\pm 1 \pm i\}$	$\{\pm 1, \pm i\}$	$\{0, \pm 2, \pm 2i, \pm 1 \pm i\}$
average energy $\mathcal{E}$	$\sqrt{5}$	$\sqrt{2}$	1	$\sqrt{2}$
minimal distance δ	2	2	$\sqrt{2}$	$\sqrt{2}$
$\delta/\mathcal{E}$	$\frac{2}{5}\sqrt{5}$	$\sqrt{2}$	$\sqrt{2}$	1

Our constellations have lower efficiency, but in space-time coding area, differences of codewords should have full rank. According to the Singleton bound, this case $d = n$ corresponds to codes of dimension $k = 1$. In that case, the constellation Q is a classical PSK.

6 Conclusion

In this article, we have introduced a new encoding scheme for space-time coding purpose, based on a $[n, k, d]$ generalised Gabidulin code. Assuming that the rank of the error is smaller than $\lfloor \frac{d-1}{2} \rfloor$, we can correct the received word in polynomial time. This encoding scheme involves a constellation which depends on several parameters (field extension, parameters and support of the code). A family of such constellations is described. They contain more elements than constellations of others constructions, but this effect is partly compensated by the fact that elements of small radius and more frequently used.

References

1. Augot, D., Loidreau, P., Robert, G.: Rank metric and Gabidulin codes over infinite fields, in preparation
2. Augot, D., Loidreau, P., Robert, G.: Rank metric and gabidulin codes in characteristic zero. ISIT 2013 IEEE International Symposium on Information Theory (2013)
3. Gabidulin, E.M.: Theory of codes with maximum rank distance. Problemy Peredachi Informatsii 21(1), 3–16 (1985)
4. Li, W., Sidorenko, V., Silva, D.: On transform-domain error and erasure correction by Gabidulin codes. Designs, Codes and Cryptography (2014)
5. Loidreau, P.: A welch–berlekamp like algorithm for decoding gabidulin codes. In: Coding and Cryptography, pp. 36–45. Springer (2006)
6. Lu, H.F., Kumar, P.: A unified construction of space-time codes with optimal rate-diversity tradeoff. Information Theory, IEEE Transactions on 51(5), 1709–1730 (2005)
7. Ore, O.: On a special class of polynomials. Transactions of the American Mathematical Society 35(3), 559–584 (1933)
8. Ore, O.: Theory of non-commutative polynomials. Annals of mathematics pp. 480–508 (1933)