An Alternative Decoding Method for Gabidulin Codes in Characteristic Zero

Sven Müelich*, Sven Puchinger*, David Mödinger†, Martin Bossert*
* Ulm University, Institute of Communications Engineering, 89081 Ulm, Germany
Email: {sven.mueelich, sven.puchinger, martin.bossert}@uni-ulm.de
† Ulm University, Institute of Distributed Systems, 89081 Ulm, Germany
Email: david.moedinger@uni-ulm.de

Abstract—**Gabidulin codes, originally defined over finite fields, are an important class of rank metric codes with various applications. Recently, their definition was generalized to certain fields of characteristic zero and a Welch–Berlekamp like algorithm with complexity $O(n^3)$ was given. We propose a new application of Gabidulin codes over infinite fields: low-rank matrix recovery. Also, an alternative decoding approach is presented based on a Gao type key equation, reducing the complexity to at least $O(n^2)$. This method immediately connects the decoding problem to well-studied problems, which have been investigated in terms of coefficient growth and numerical stability.**

Index Terms—**Gabidulin Codes, Characteristic Zero, Rank Metric, Decoding, Matrix Recovery**

I. MOTIVATION

Finding a matrix of minimal rank is a problem which occurs in different scenarios. For example in random linear network coding [1], an error can be described by a matrix of minimal rank. Therefore, codes whose metric is based on the rank of matrices can be beneficial. The most prominent example of rank metric codes are Gabidulin codes, introduced by Delsarte [2], Gabidulin [3], and Roth [4]. Given a received word $\mathbf{R} = \mathbf{C} + \mathbf{E}$, the calculation of the error matrix $\mathbf{E}$ of minimum rank can be described by the weight-minimization problem

$$\min \operatorname{rank}(\mathbf{E}') \text{ subject to } \mathbf{H}\mathbf{E}' = \mathbf{H}\mathbf{E}, \tag{1}$$

where $\mathbf{H}$ is a parity check matrix. This minimization problem is equivalent to the problem of low-rank matrix recovery (LRMR) [5], [6], which is the matrix-analogue to compressed sensing [7], [8]. This problem aims to recover an unknown matrix $\mathbf{E} \in \mathbb{C}^{n \times n}$ of low rank, and can be solved by finding a solution for the under-determined linear system of equations $\mathbf{H}\mathbf{e} = \mathbf{s}$, where $\mathbf{H} \in \mathbb{C}^{m \times n^2}$ is the sensing matrix, $\mathbf{e} \in \mathbb{C}^{n^2 \times 1}$ is the vector representation of the matrix $\mathbf{E}$, and $\mathbf{s} \in \mathbb{C}^{m \times 1}$ is the measurement when applying the sensing matrix $\mathbf{H}$ to $\mathbf{E}$ ($m < n^2$). Applications of LRMR can be found e.g., in the fields of image processing or collaborative filtering. Since decoding of rank metric codes and LRMR is the same mathematical problem (cf. Equation (1)), the application of Gabidulin codes in characteristic zero might be promising to the LRMR problem. If we replace the rank metric by the Hamming metric, Equation (1) describes both a Hamming-metric decoder and the compressed sensing problem. An

exchange of concepts between these two areas was successfully investigated in the recent years [9]. Another important application of Gabidulin codes in characteristic zero is space-time coding.

Commonly, Gabidulin codes are defined over finite fields as evaluation codes of linearized polynomials and can be considered as rank metric equivalents of Reed-Solomon codes. In [10], Reed-Solomon codes over the complex field were investigated for applications in compressed sensing. LRMR and space-time codes indicate that there is a need for Gabidulin codes defined over fields of characteristic zero, possibly dense in $\mathbb{C}$. In [11] and [12], Gabidulin codes in characteristic zero were introduced. In contrast to the finite field case, θ-polynomials are used instead of linearized polynomials. A Welch-Berlekamp-like decoding algorithm [13] was transformed from the finite field case to the characteristic zero case, which allows decoding in cubic time. In this work, we consider an alternative method for decoding characteristic zero Gabidulin codes.

The rest of the paper is structured as follows: Section II outlines Gabidulin codes and related concepts in characteristic zero. In Section III we propose a new decoding approach. We explain how the decoding problem can be solved by using shift register synthesis to find solutions of a Gao-like key equation. We also discuss issues of coefficient growth and numerical problems which emerge when using infinite fields. Finally, Section IV concludes the paper.

II. GABIDULIN CODES OVER INFINITE FIELDS

This section first summarizes properties of θ-polynomials, which are used to define Gabidulin codes in characteristic zero. Then we recall different definitions of rank metric and the definition of Gabidulin codes.

A. θ-polynomials

Gabidulin codes over finite fields are usually defined using *linearized polynomials* [14]. θ-polynomials can be seen as a natural generalization of linearized polynomials for arbitrary fields. Let $K \subseteq L$ be fields and L/K be a Galois extension. The *Galois group* of L/K is given by

$$\operatorname{Gal}(L/K) = \{\theta : L \to L \text{ automorphism} : \theta(k) = k \ \forall k \in K\}.$$

Lemma 1. *[15] Let $\theta \in \mathrm{Gal}\,(L/K)$. The set*

$$L[x;\theta] = \left\{ a = \textstyle\sum_{i=0}^{d_a} a_i x^i : a_i \in L,\ d_a \in \mathbb{N},\ a_{d_a} \neq 0 \right\}$$

with multiplication rule $x\cdot\alpha = \theta(\alpha)\cdot x$ for all $\alpha \in L$, extended to polynomials inductively, and ordinary addition is a ring.

We call the polynomial ring of Lemma 1 a θ-polynomial ring. The degree of $a \in L[x;\theta]$ is given by $\deg a = d_a$ and a is called *monic* if $a_{d_a} = 1$.

Remark 2. *We state the following properties of $L[x;\theta]$.*
- $(L[x;\theta], +, \cdot)$ *is non-commutative in general.*
- θ-*polynomials are a special case of skew polynomials [15] with derivation $\delta = 0$.*
- *For $K = \mathbb{F}_q$, $L = \mathbb{F}_{q^m}$ and the Frobenius automorphism $\theta = \cdot^q$, $L[x;\theta]$ is isomorphic to a linearized polynomial ring. Note that $\cdot^q \in \mathrm{Gal}\,(\mathbb{F}_{q^m}/\mathbb{F}_q)$.*

Is was already proven in [14] that $L[x;\theta]$ is a left- and right-Euclidean domain. E.g., the following division lemma is true.

Lemma 3. *[14] For $a \in L[x;\theta]$, $b \in L[x;\theta]^*$, $\exists$ unique $\chi, \varrho \in L[x;\theta]$: $a = \chi \cdot b + \varrho$ (right division), where $\deg \varrho < \deg b$.*

Related to division, we can define the (right) modulo congruence relation for $a, b, c \in L[x;\theta]$:

$$a \equiv b \mod c \quad :\Leftrightarrow \quad \exists d \in L[x;\theta] : a = b + d \cdot c.$$

We can define an evaluation map[2] on $L[x;\theta]$ as

$$\mathrm{ev}_a = a(\cdot) : L \to L, \quad \alpha \mapsto \textstyle\sum_{i=0}^{d_a} a_i \theta^i(\alpha), \qquad (2)$$

where $\theta^i(\cdot) = \underbrace{\theta(\theta(\dots\theta(\cdot)\dots))}_{i \text{ times}}$. From $\theta \in \mathrm{Gal}\,(L/K)$ it follows that $\theta : L \to L$ is a K-linear map. Thus, also ev_a is a linear map and the root space of a θ-polynomial a,

$$\ker(a) = \{\alpha \in L : a(\alpha) = 0\},$$

is a linear subspace of L. The evaluation map of the multiplication of two θ-polynomials a, b equals the composition of the evaluation maps of a, b respectively, i.e. $\mathrm{ev}_{a\cdot b} = \mathrm{ev}_a \circ \mathrm{ev}_b$. Since θ is a linear map, it has well-defined eigenvalues which are the roots of its characteristic polynomial

$$\mathrm{char}_\theta(x) = \det(x \cdot \mathrm{id}_L - \theta).$$

The eigenvalues and characteristic polynomial are the same as of any matrix representation of θ in a basis of L over K. We say that a characteristic polynomial is square-free if all its roots have multiplicity one. If char_θ is square-free, θ has distinct eigenvalues and any of its matrix representations is diagonalizable. Using these properties, we can state the following theorem.

Lemma 4. *[12, Theorem 6] If char_θ is square-free, then*

$$\dim_K(\ker(a)) \leq \deg(a) \quad \forall a \in L[x;\theta] \setminus \{0\}$$

---

[2]There are several definitions of evaluation maps for θ-polynomials, cf. [16] for the general skew polynomial case.

Proof. The proof can be found in [12, Theorem 6]. It uses matrix representations of θ and the fact that it is diagonalizable due to char_θ being square-free. $\qquad\square$

Theorem 5. *Let $\mathcal{U} \subseteq L$ be an s-dimensional K-subspace. If char_θ is square-free, there exists a unique monic θ-polynomial $\mathcal{A}_\mathcal{U}$ with $\mathcal{U} \subseteq \ker(\mathcal{A}_\mathcal{U})$ of minimum degree. $\mathcal{A}_\mathcal{U}$ is called annihilator polynomial of $\mathcal{U}$ and if $\theta(\cdot)$ can be calculated in $O(1)$, $\mathcal{A}_\mathcal{U}$ can be computed in $O(s^2)$ operations in L. Moreover, $\deg \mathcal{A}_\mathcal{U} = \dim \mathcal{U}$ and $\mathcal{U} = \ker(\mathcal{A}_\mathcal{U})$.*

Proof. The proof is similar to [12, Theorem 8]. It can be shown by induction that the polynomial $\mathcal{A}_s$ constructed in Algorithm 1 fulfills $\mathcal{U} \subseteq \ker(\mathcal{A}_s)$. By the Euclidean algorithm, $\mathcal{A}_s = \chi \cdot \mathcal{A}_\mathcal{U}$ for some $\chi \in L[x;\theta]$ and $\deg \mathcal{A}_\mathcal{U} \leq \deg \mathcal{A}_s = \dim \mathcal{U}$ because $\deg A_i = \deg A_{i-1} + 1\ \forall i$, $\deg A_0 = 0$ and thus $\deg \mathcal{A}_s = s = \dim \mathcal{U}$. Also, $\deg \mathcal{A}_\mathcal{U} \geq \dim \mathcal{U}$ by Lemma 4 (which assumes that char_θ is square-free), implying $\deg \mathcal{A}_\mathcal{U} = \dim \mathcal{U}$. Since $\mathcal{A}_\mathcal{U}$ is defined to be monic, it is therefore unique and $\mathcal{A}_s = \mathcal{A}_\mathcal{U}$. Due $\dim(\ker(\mathcal{A}_\mathcal{U})) \leq \deg \mathcal{A}_\mathcal{U} = \dim \mathcal{U}$, together with $\mathcal{U} \subseteq \ker(\mathcal{A}_\mathcal{U})$, it follows that $\mathcal{U} = \ker(\mathcal{A}_\mathcal{U})$. Line 3 of Algorithm 1 is executed s times and each loop requires

- one evaluation $A_{i-1}(u_i)$, costing $O(s)$ operations in L by naively applying the evaluation formula 2
- one computation of $\theta(A_{i-1}(u_i)) \Rightarrow O(1)$ and
- one addition in $L[x;\theta] \Rightarrow O(s)$,

and hence the algorithm has complexity $O(s)$ in L. $\qquad\square$

Algorithm 1: Annihilator Polynomial [12]

Input: K-basis $(u_1, \dots, u_s)$ of $\mathcal{U} \subseteq L$.
Output: $\mathcal{A}_\mathcal{U}$ as in Theorem 5.

1   $\mathcal{A}_0 \leftarrow 1$
2   **for** $i = 1, \dots, s$ **do**
3     $\mathcal{A}_i \leftarrow (x - \frac{\theta(A_{i-1}(u_i))}{A_{i-1}(u_i)}) \cdot A_{i-1}$ // $O(s)$
4   **return** $\mathcal{A}_s$

Theorem 6 ([12, Theorem 8]). *Let $g_1, \dots, g_n \in L$, linearly independent over K, and $\mathbf{r} = (r_1, \dots, r_n) \in L^n$. Then there is a unique monic θ-polynomial $\hat{r}$ of degree $n - 1$ such that*

$$\hat{r}(g_i) = r_i \quad \forall i = 1, \dots, n.$$

B. Rank Metric in Characteristic Zero

Let $K \subseteq L$ be fields, L/K a Galois extension of degree m and $\mathcal{B}$ a basis of L over K. The number of k-linearly independent columns of a matrix $\mathbf{X}$ is denoted by $\mathrm{rank}_k(\mathbf{X})$ for $k \in \{L, K\}$. We define the matrices

$$\mathbf{X}_\theta = \begin{pmatrix} x_1 & \dots & x_n \\ \theta(x_1) & \dots & \theta(x_n) \\ \vdots & \ddots & \vdots \\ \theta^{m-1}(x_1) & \dots & \theta^{m-1}(x_n) \end{pmatrix}, \mathbf{X}_\mathcal{B} = \begin{pmatrix} x_{1,1} \dots x_{n,1} \\ x_{1,1} \dots x_{n,1} \\ \vdots & \ddots & \vdots \\ x_{1,m} \dots x_{n,m} \end{pmatrix},$$

where $(x_{i,1}, \dots, x_{i,m})^T \in K^m$ is the representation of $x_i \in L$ in the basis $\mathcal{B}$. In [12, Section 2.2] four definitions of rank weight in characteristic zero are given.

Definition 7 ([12]). *Let $\mathbf{x} \in L^n$. We define the rank weights*

$$\omega_1(\mathbf{x}) = deg(\mathcal{A}_{\langle x_1,\ldots,x_n \rangle})$$
$$\omega_2(\mathbf{x}) = \mathrm{rank}_L(\mathbf{X}_\theta)$$
$$\omega_3(\mathbf{x}) = \mathrm{rank}_K(\mathbf{X}_\theta)$$
$$\omega_4(\mathbf{x}) = \mathrm{rank}_K(\mathbf{X}_\mathcal{B})$$

The corresponding rank metrics can be defined as

$$\mathrm{d}_{\mathrm{R},i}(\mathbf{x},\mathbf{y}) = \omega_i(\mathbf{x}-\mathbf{y}) \quad \forall i \in \{1,2,3,4\}.$$

In the finite field case, these rank weights are the same. Over characteristic zero, the following relation can be proven.

Lemma 8. *[12, Lemmata 13, 14, and 15]*

$$\omega_1(\mathbf{x}) = \omega_2(\mathbf{x}) \leq \omega_3(\mathbf{x}) = \omega_4(\mathbf{x})$$

C. Gabidulin Codes

Gabidulin codes were originally defined by [3], [2], [4] over finite fields. In [11], the definition was extended to certain fields of characteristic zero, using θ-polynomials instead of linearized polynomials.

Definition 9. *Let $g_1,\ldots,g_n \in L$ be linearly independent over K. Then a Gabidulin code of length n and dimension $k \leq n$ is defined as*

$$\mathcal{C}_{\mathrm{G}}[n,k] = \{(f(g_1),\ldots,f(g_n)) \,:\, f \in L[x;\theta] \wedge \deg f < k\}.$$

An overview of properties can be found in [12].

III. A NEW DECODING APPROACH

In the following, let L, K and $\theta \in \mathrm{Gal}\,(L/K)$ be such that char_θ is square-free. We assume that $\theta(\cdot)$ can be computed in $O(1)$ operations in L. Under these assumptions, the latter only being important for complexity statements, we show that the decoding problem is similar to the finite field case.

Suppose that a codeword $\mathbf{c} \in \mathcal{C}_{\mathrm{G}}$ is corrupted by an error $\mathbf{e} = (e_1,\ldots,e_n) \in L^n$ of rank weight $\tau := \mathrm{wt}_{\mathrm{R}}(\mathbf{e})$. The *received word* is then given by

$$\mathbf{r} = \mathbf{c} + \mathbf{e} \in L^n.$$

We say that τ errors occurred. The goal of decoding is to recover $\mathbf{c}$ from $\mathbf{r}$ if τ is not too large.

A. Key Equation

Definition 10. *We define the error span polynomial*

$$\Lambda = \mathcal{A}_{\langle e_1,\ldots,e_n \rangle}.$$

The following lemma is, in contrary to the finite field case, not obvious (cf. Theorem 5) and only holds for the case of char_θ being square-free.

Lemma 11. $\deg \Lambda = \tau$

Proof. This follows directly from Theorem 5 together with $\deg \Lambda = \dim \langle e_1,\ldots,e_n \rangle = \mathrm{wt}_{\mathrm{R}}(\mathbf{e}) = \tau$. $\square$

The following lemma is necessary to prove Theorem 13, the main statement of this section.

Lemma 12. *Let $\mathcal{U} \subseteq L$ be a K-subspace and $a,b \in L[x;\theta]$.*

$$a \equiv b \mod \mathcal{A}_\mathcal{U} \quad \Leftrightarrow \quad a(u) = b(u) \ \forall\, u \in \mathcal{U}$$

Proof. By Lemma 3, there are $\chi, \varrho \in L[x;\theta]$ with

$$a - b = \chi \cdot \mathcal{A}_\mathcal{U} + \varrho$$

and $\deg \varrho < \deg \mathcal{A}_\mathcal{U}$. Then,

$$\begin{aligned} &a(u) = b(u) \ \forall\, u \in \mathcal{U} \\ \Leftrightarrow \quad &a(u) - b(u) = (a-b)(u) = (\chi \cdot \mathcal{A}_\mathcal{U} + \varrho)(u) \\ &\qquad = \chi(\mathcal{A}_\mathcal{U}(u)) + \varrho(u) = \chi(0) + \varrho(u) \\ &\qquad = \varrho(u) = 0 \ \forall\, u \in \mathcal{U}. \end{aligned}$$

Also, $\varrho(u) = 0$ for all $u \in \mathcal{U}$ if and only if $\varrho = 0$, since otherwise it would contradict the minimality of $\mathcal{A}_\mathcal{U}$. $\square$

Let $\hat{r}$ be the known interpolation polynomial of degree $\deg \hat{r} < n$ corresponding to the received word $\mathbf{r}$ as in Theorem 6. Recall that f is the unknown information polynomial of degree $\deg f < k$ and Λ is the unkown error span polynomial. Also, $\mathcal{A}_{\langle g_1,\ldots,g_n \rangle}$ is known and has degree $\deg \mathcal{A}_{\langle g_1,\ldots,g_n \rangle} = n$, since the g_i's are linearly independent. The following statement is an analogue to Gao's key equation for Reed–Solomon codes and a generalization of [17, Theorem 3.6], where it was proven for finite field Gabidulin codes.

Theorem 13 (Key Equation).

$$\Lambda \cdot \hat{r} \equiv \Lambda \cdot f \mod \mathcal{A}_{\langle g_1,\ldots,g_n \rangle} \tag{3}$$

Proof. Let $u \in \langle g_1,\ldots,g_n \rangle$. Then, we can write u as a K-linear combination of the g_i's, $u = \sum_{i=1}^n \alpha_i g_i$, and

$$\begin{aligned} (\Lambda \cdot \hat{r})(u) &- (\Lambda \cdot f)(u) = \Lambda(\hat{r}(u) - f(u)) \\ &= \Lambda(\hat{r}(\sum_{i=1}^n \alpha_i g_i) - f(\sum_{i=1}^n \alpha_i g_i)) \\ &= \sum_{i=1}^n \alpha_i \Lambda(\hat{r}(g_i) - f(g_i)) = \sum_{i=1}^n \alpha_i \Lambda(r_i - c_i) \\ &= \sum_{i=1}^n \alpha_i \Lambda(e_i) = 0. \end{aligned}$$

The statement follows by Lemma 12. $\square$

B. Decoding using Shift Register Synthesis Problems

Since it is hard to directly find a solution to the key equation, which is non-linear, we try to find a solution to the following shift register synthesis problem, which is formulated in a similar way as the problem which is solved in [18] over ordinary polynomial rings.

Definition 14. *Let k, $\hat{r}$ and $\mathcal{A}_{\langle g_1,\ldots,g_n \rangle}$ be given as above. A shift register problem (SRP) is the problem of finding $(\lambda, \omega) \in (L[x;\theta]^*)^2$ such that*

$$\lambda \hat{r} \equiv \omega \mod \mathcal{A}_{\langle g_1,\ldots,g_n \rangle} \tag{4}$$
$$\deg \lambda > \deg \omega + k \tag{5}$$
$$\deg \lambda \text{ minimal} \tag{6}$$

The following theorem is, besides the key equation, the main statement of this paper. It proves that the decoding problem and the SRP are equivalent if the number of errors is less than half the minimum distance.

Theorem 15. *If $\tau < \frac{d}{2}$, the SRP has a solution (λ, ω) and any such solution satisfies*

$$(\Lambda, \Lambda f) = \alpha(\lambda, \omega)$$

for some $\alpha \in L^*$, minimum distance d and information polynomial $f \in L[x; \theta]$.

Proof. We first prove that the SRP has a solution and all solutions satisfy $\omega = \lambda f$, by applying similar arguments as in the proof of [12, Theorem 25]. Then we show that the solution is unique up to a scalar multiplication. By Theorem 13, $(\Lambda, \Lambda f)$ fulfills the congruence relation (4) and due to

$$\deg \Lambda f = \deg \Lambda + \deg f < \deg \Lambda + k,$$

it also satisfies the degree condition (5). Thus, the SRP has a solution[3] (λ, ω), and by Lemma 11, any such solution satisfies

$$\deg \lambda \leq \deg \Lambda = \tau, \tag{7}$$
$$\deg \omega < \deg \lambda + k = \tau + k. \tag{8}$$

We also know that $\dim \langle e_1, \dots, e_n \rangle = \tau$, implying

$$\deg \mathcal{A}_{\langle \lambda(e_1), \dots, \lambda(e_n) \rangle} = \dim \langle \lambda(e_1), \dots, \lambda(e_n) \rangle \leq \tau$$

and thus,

$$\deg \mathcal{A}_{\langle \lambda(e_1), \dots, \lambda(e_n) \rangle}(\omega - \lambda f) < \deg \mathcal{A}_{\langle \lambda(e_1), \dots, \lambda(e_n) \rangle} + \tau + k$$
$$\leq 2\tau + k \leq 2\frac{d-1}{2} + k = n - k + k = n,$$

Due to (4) and Lemma 12, $\lambda(\hat{r}(g_i)) = (\lambda \hat{r})(g_i) = \omega(g_i)$ for all i, we obtain

$$\mathcal{A}_{\langle \lambda(e_1), \dots, \lambda(e_n) \rangle}(\omega - \lambda f)(g_i)$$
$$= \mathcal{A}_{\langle \lambda(e_1), \dots, \lambda(e_n) \rangle}(\omega(g_i) - \lambda(f(g_i)))$$
$$= \mathcal{A}_{\langle \lambda(e_1), \dots, \lambda(e_n) \rangle}(\lambda(\hat{r}(g_i)) - \lambda(f(g_i)))$$
$$= \mathcal{A}_{\langle \lambda(e_1), \dots, \lambda(e_n) \rangle}(\lambda(r_i - c_i))$$
$$= \mathcal{A}_{\langle \lambda(e_1), \dots, \lambda(e_n) \rangle}(\lambda(e_i)) = 0$$

Thus, we obtain $\mathcal{A}_{\langle \lambda(e_1), \dots, \lambda(e_n) \rangle}(\omega - \lambda f) = 0$ because the polynomial has degree $< n$ but evaluates to 0 at n linearly independent positions (cf. Lemma 4). Since $L[x; \theta]$ is an integral domain, we get $\omega = \lambda f$.

Together with the congruence relation (4), it follows that

$$\lambda(\hat{r} - f) \equiv 0 \mod \mathcal{A}_{\langle g_1, \dots, g_n \rangle},$$

thus, $\lambda(e_i) = \lambda(\hat{r} - f)(g_i) = 0 \ \forall i = 1, \dots, n$. Due to $\deg \lambda \leq \deg \Lambda$, λ must be the annihilator polynomial Λ of $\langle e_1, \dots, e_n \rangle$ multiplied by a scalar $\alpha^{-1} = \lambda_{\deg \lambda} \in L^*$, the leading coefficient of the polynomial λ. Hence, also

$$\alpha \omega = \alpha \lambda f = \Lambda f \tag{9}$$

and the claim is proven. $\square$

Remark 16. *In the case $\tau < \frac{d}{2}$, a solution of the SRP is also a solution to the* linear reconstruction problem *discussed in [12]. This follows from the degree conditions (7) and (8), and the observation that $\lambda(\hat{r}(g_i)) = \omega(g_i)$ for all $i = 1, \dots, n$.*

[3]Either $(\Lambda, \Lambda f)$ or a "smaller" solution in terms of $\deg \lambda$

We can conclude that for rank errors up to half the minimum distance $\mathrm{d}_{R,i}(\mathbf{r}, \mathbf{c}) = \omega_i(\mathbf{e}) < \frac{d}{2}$, using any rank metric $\mathrm{d}_{R,i}$ with $i \in \{1, 2, 3, 4\}$ of Definition 7, we can solve the decoding problem by finding a solution of the SRP since the number of errors is $\tau = \omega_1(\mathbf{e}) \leq \omega_i(\mathbf{e}) < \frac{d}{2}$ (cf. Lemma 8). Note that certain Gabidulin codes over finite fields cannot be decoded beyond half the minimum distance in polynomial time (cf. [19]). Investigating whether this is also true over fields of characteristic zero is beyond the scope of this paper. The next section summarizes known algorithms to solve SRPs.

C. Solving Shift Register Problems

SRPs over $L[x]$ and $L[x; \theta]$ are well-studied and have been used for decoding of several algebraic codes, including Reed–Solomon and (finite field) Gabidulin codes.

Two of the most important algorithms to solve these kinds of problems are:

1) The *Extended Euclidean Algorithm*.
   Since $L[x; \theta]$ is a Euclidean domain, it admits a Euclidean algorithm. It is shown e.g. in [17] that the Euclidean algorithm over $\mathbb{F}[x; \cdot^q]$ can be performed in $O(\mathcal{D}(n))$ time, where $\mathcal{D}(n)$ is the complexity of dividing two polynomials in $\mathbb{F}[x; \cdot^q]$. These results directly translate to $L[x; \theta]$.
   Using the classical division algorithm, $\mathcal{D}(n) \in O(n^2)$. However, it is justifiable that the division method described in [20] generalizes to $L[x; \theta]$ where $\theta(\cdot)$ can be computed in $O(1)$, implying $\mathcal{D}(n) \in O(n^{1.69} \log(n))$.

2) *Module Minimization*.
   The algorithms described in [21] solve a generalized version of the SRP described in this paper. If $\theta(\cdot)$ can be computed in $O(1)$, the complexity of finding a solution of the SRP becomes $O(n^2)$. Moreover, as already mentioned in [21], there is the substantiated hope for similar speed-ups as in the $L[x]$ case, such as the divide-and-conquer variant described in [22].

Alternatively, a variant of the Berlekamp–Massey algorithm (cf. [23]) can be used, which might have advantages in practical scenarios.

D. Issues Besides Complexity

Since we are dealing with infinite fields, we have to deal with some issues that do not appear in the finite field case.

As already mentioned in [12], when computing in exact computation domains, such as number fields, we have to face the problem of coefficient growth. Fortunately, our proposed decoding method reduces the decoding problem to a problem that was already studied in terms of coefficient growth before (cf. [24]). As described in Section III-C, we can use module minimization to obtain a solution of the SRP. More precisely, in [21] a solution of the SRP is obtained by transforming a basis of a certain $L[x; \theta]$-module into a normal form, called *weak Popov form*. Instead of using the algorithms described in [21] to obtain a weak Popov form, we can use the methods from [24]. The algorithms in [24] are slower than those in [21],

but have a better control of coefficient growth in intermediate results using fraction-free methods.

On the other hand, especially in the application of LRMR, it might be advantageous in terms of complexity not to use exact but approximate computations. Thus, one has to deal with numerical issues. In the Hamming metric analogy, this problem was already investigated for complex Reed–Solomon codes (cf. [9, Chapter 7]). There, it turned out that a modification of the Berlekamp–Massey algorithm is the numerically most stable one among the classical approaches for solving an SRP. It should also be noted that the interpolation algorithm presented in [25] is a reasonable choice to compute $\hat{r}$, since it is the skew polynomial analogue of the numerically stable Newton interpolation with divided differences.

E. Summary of the Decoding Algorithm

Algorithm 2 summarizes the decoding procedure.

Algorithm 2: Decode Gabidulin Codes

Input: $\mathbf{r} = \mathbf{c} + \mathbf{e}$
Output: f such that $\mathbf{c} = (f(g_1), \dots, f(g_n))$
or "decoding failure".

1 Calculate $\hat{r}$ as in Theorem 6
2 Calculate $\mathcal{A}_{\langle g_1, \dots, g_n \rangle}$ as in Definition 10
3 $(\lambda, \omega) \leftarrow$ Solve SRP with input $\hat{r}$, $\mathcal{A}_{\langle g_1, \dots, g_n \rangle}$
4 $(\Lambda, \Omega) \leftarrow \alpha^{-1}(\lambda, \omega)$ with α as in (9)
5 $(\chi, \varrho) \leftarrow$ Right-divide Ω by Λ (cf. Lemma 3)
6 **if** $\varrho = 0$ **then**
7   **return** χ
8 **else**
9   **return** *"decoding failure"*

Theorem 17. *Alg. 2 is correct and has complexity $O(n^2)$.*

Proof. Correctness follows from Theorems 13 and 15. The lines of the algorithm have the following complexities, implying the overall statement:

- Line 1: We can use the interpolation algorithm for skew polynomials presented in [25], having complexity $O(n^2)$.
- Line 2: $O(n^2)$ by Algorithm 1.
- Line 3: $O(n^2)$ using e.g. module minimization as in [21].
- Line 4: Negligible.
- Line 5: $O(n^2)$ using the standard algorithm [15]. □

IV. Conclusion

We have proposed a new method for decoding Gabidulin codes over fields with characteristic zero, reducing the decoding complexity to $O(n^2)$ compared to $O(n^3)$ in [12]. This alternative procedure reduces decoding to a linear shift register synthesis problem, which can be efficiently solved using several known algorithms, each having advantages in terms of speed, coefficient growth or numerical stability. The presented work can be used for applying Gabidulin codes over characteristic zero to space-time coding and to the low-rank matrix recovery problem. The latter one is, to the best of our knowledge, a new application for these codes.

References

[1] R. Koetter and F. R. Kschischang, "Coding for Errors and Erasures in Random Network Coding," *IEEE Transactions on Information Theory*, vol. 54, no. 8, pp. 3579–3591, 2008.

[2] P. Delsarte, "Bilinear Forms over a Finite Field, with Applications to Coding Theory," *Journal of Combinatorial Theory, Series A*, vol. 25, no. 3, pp. 226–241, 1978.

[3] E. M. Gabidulin, "Theory of Codes with Maximum Rank Distance," *Problemy Peredachi Informatsii*, vol. 21, no. 1, pp. 3–16, 1985.

[4] R. M. Roth, "Maximum-Rank Array Codes and their Application to Crisscross Error Correction," *IEEE Transactions on Information Theory*, vol. 37, no. 2, pp. 328–336, 1991.

[5] E. J. Candès and B. Recht, "Exact Matrix Completion via Convex Optimization," *Foundations of Computational mathematics*, vol. 9, no. 6, pp. 717–772, 2009.

[6] D. Gross, "Recovering Low-Rank Matrices From Few Coefficients in Any Basis," *IEEE Transactions on Information Theory*, vol. 57, no. 3, pp. 1548–1566, 2011.

[7] E. J. Candès, J. Romberg, and T. Tao, "Robust Uncertainty Principles: Exact Signal Reconstruction From Highly Incomplete Frequency Information," *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 489–509, 2006.

[8] D. L. Donoho, "Compressed Sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[9] H. A. Zörlein, "Channel Coding Inspired Contributions to Compressed Sensing," Ph.D. dissertation, Universität Ulm, 2015.

[10] M. Mohamed, S. Rizkalla, H. A. Zörlein, and M. Bossert, "Deterministic Compressed Sensing with Power Decoding for Complex Reed-Solomon Codes," in *International ITG Conference on Systems, Communications and Coding*, 2015, pp. 1–6.

[11] D. Augot, P. Loidreau, and G. Robert, "Rank Metric and Gabidulin Codes in Characteristic Zero," in *ISIT 2013 IEEE International Symposium on Information Theory*, 2013.

[12] G. Robert, "Codes de Gabidulin en Caractéristique Nulle. Application au Codage Espace-Temps," Ph.D. dissertation, Université Rennes 1, 2015.

[13] P. Loidreau, "A Welch–Berlekamp Like Algorithm for Decoding Gabidulin Codes," in *Coding and Cryptography*. Springer, 2006, pp. 36–45.

[14] O. Ore, "On a Special Class of Polynomials," *Transactions of the American Mathematical Society*, vol. 35, no. 3, pp. 559–584, 1933.

[15] ——, "Theory of Non-Commutative Polynomials," *Annals of mathematics*, pp. 480–508, 1933.

[16] D. Boucher and F. Ulmer, "Linear Codes using Skew Polynomials with Automorphisms and Derivations," *Designs, codes and cryptography*, vol. 70, no. 3, pp. 405–431, 2014.

[17] A. Wachter-Zeh, "Decoding of Block and Convolutional Codes in Rank Metric," Ph.D. dissertation, Université Rennes 1; Ulm University, 2013.

[18] P. Fitzpatrick, "On the Key Equation," *IEEE Transactions on Information Theory*, vol. 41, no. 5, pp. 1290–1302, 1995.

[19] N. Raviv and A. Wachter-Zeh, "Some Gabidulin Codes Cannot be List Decoded Efficiently at any Radius," in *IEEE International Symposium on Information Theory (ISIT)*, June 2015, pp. 6–10.

[20] S. Puchinger and A. Wachter-Zeh, "Fast Operations on Linearized Polynomials and their Applications in Coding Theory," *arXiv preprint http://arxiv.org/abs/1512.06520*, Dec. 2015.

[21] W. Li, J. S. Nielsen, S. Puchinger, and V. Sidorenko, "Solving Shift Register Problems over Skew Polynomial Rings using Module Minimisation," in *International Workshop on Coding and Cryptography, arXiv: http://arxiv.org/abs/1501.04797*, Apr. 2015.

[22] M. Alekhnovich, "Linear Diophantine Equations Over Polynomials and Soft Decoding of Reed–Solomon Codes," *IEEE Transactions on Information Theory*, vol. 51, no. 7, pp. 2257–2265, Jul. 2005.

[23] G. Richter and S. Plass, "Error and Erasure Decoding of Rank-Codes with a Modified Berlekamp-Massey Algorithm," *ITG FACHBERICHT*, pp. 203–210, 2004.

[24] B. Beckermann, H. Cheng, and G. Labahn, "Fraction-Free Row Reduction of Matrices of Ore Polynomials," *Journal of Symbolic Computation*, vol. 41, no. 5, pp. 513–543, 2006.

[25] S. Liu, F. Manganiello, and F. R. Kschischang, "Kötter interpolation in skew polynomial rings," *Designs, codes and cryptography*, vol. 72, no. 3, pp. 593–608, 2014.