GENERALIZED SKEW REED-SOLOMON CODES AND OTHER APPLICATIONS OF SKEW POLYNOMIAL EVALUATION

by

Siyu Liu

A thesis submitted in conformity with the requirements for the degree of Doctor of Philosophy
Graduate Department of Electrical and Computer Engineering
University of Toronto

© Copyright 2016 by Siyu Liu
Abstract

Generalized Skew Reed-Solomon Codes and Other Applications of Skew Polynomial Evaluation

Siyu Liu

Doctor of Philosophy

Graduate Department of Electrical and Computer Engineering

University of Toronto

2016

This thesis studies the theory of error-correcting codes based on evaluation of skew polynomials. Skew polynomials are a noncommutative generalization of ordinary polynomials that, in recent years, have found applications in coding theory and cryptography. The evaluation of skew polynomials is significantly different from that of ordinary polynomials. Using skew polynomial evaluation, we define a class of codes called Generalized Skew-Evaluation codes. This class contains a special subclass that we call Generalized Skew Reed-Solomon codes. Generalized Skew Reed-Solomon codes are intimately connected to both regular Reed-Solomon codes and Gabidulin codes. These connections allow us to design interesting codes for both the Hamming metric and the rank metric. In addition, we prove a duality theory that exists within a subclass of Generalized Skew Reed-Solomon codes. This duality theory can be viewed as an extension to the well-known duality theory of Generalized Reed-Solomon codes. The decoding of Generalized Skew Reed-Solomon codes can be done using a Berlekamp-Welch style decoder. We design such a decoder by studying the general polynomial interpolation problem for skew polynomials. In particular, we generalize Kötter interpolation to work over skew polynomial rings. In our generalization, we provide a mathematically rigorous framework, as well as two important applications: a Newton interpolation algorithm for skew polynomial evaluation and a Berlekamp-Welch style decoder. In analyzing the connection
between Generalized Skew Reed-Solomon codes and Gabidulin codes, we demonstrate the interplay between skew polynomials and linearized polynomials. This is summarized by a Structure Theorem. Using this theorem, we construct a matroid called the $\mathbb{F}_{q^m}[x; \sigma]$-matroid. The structure of this matroid is induced by skew polynomial evaluation. We examine numerous properties of this matroid, and show an application of this matroid in the context of network coding.
To 梁岩
Acknowledgements

I would like to express my sincere gratitude to my supervisor, Prof. Frank R. Kschischang. It has been almost 10 years since I first worked with Frank as an undergraduate student. Through these years, Frank has guided me in all aspects of my life. As an academic, Frank is truly a brilliant researcher. His insightfulness in asking the most fundamental questions is awe-inspiring, and has helped me immensely in my research. As a teacher, Frank is exemplary. I have benefited greatly from his extraordinary patience and devotion to his students. His passion for research and dedication to teaching have influenced me deeply. He taught me not only to be a better researcher, but also a better teacher, and a better person. Working with Frank over the years has made my graduate experience truly wonderful.

I wish to thank Prof. Stark Draper, Prof. Raymond Kwong and Prof. Wei Yu for taking their time to help evaluate my thesis. Their comments, suggestions and advice are very valuable. I am very thankful to have Prof. Heide Gluesing-Luerssen as my external examiner. Her detailed appraisal provided many useful suggestions that allowed me to improve this thesis.

I have been fortunate to collaborate with Prof. Felice Manganiello over the years. Our frequent discussions and his insightful comments led to many interesting research results and contributed largely to this thesis.

I am very grateful to Prof. Bernardo Galvão-Sousa, Prof. Adrian Nachman and Prof. Mary Pugh who helped shape my academic career from a student to becoming a teacher.

I had the great pleasure to share my journey with many wonderful friends and colleagues. In particular, I learned a great deal from research discussions with Chen Feng and Gokul Sridharan. I also enjoyed many wonderful hours discussing philosophy and solving math puzzles with Pratik Patil. I am thankful to Binbin Dai and Soroush Tabatabei for
their help in the preparation of my defense.

For their friendship and advice, I would like to thank the members of FRK group: Chris Blake, Siddarth Hari, Chunpo Pan, Chu Pang, Christian Senger, Mansoor Yousefi, and Lei Zhang.

BA4162 is the office where I enjoyed many years of hardwork, joy, and laughter. I would like to thank my officemates who shared these memories: Arvin Ayoughi, Hayssam Dahrouj, Yicheng Lin, Kaveh Mahdaviani, Rafid Mahmood, Louis Tan, Lei Zhou and Yuhan Zhou.

I would also like to thank many wonderful friends with whom I shared many memorable Friday evenings on many recreational activities: Wei Bao, Alice Gao, David Han, Weiwei Li, Yinan Liu, Huiyuan Xiong, Yue Yin, Wanyao Zhao, and Caiyi Zhu.

Finally, I am, as always, grateful to my family for their unwavering support, especially to Yan for her love, care and patience.
# Contents

1 Introduction ............................................. 1
   1.1 Previous Work ........................................ 3
   1.2 Our Contributions ..................................... 4
      1.2.1 New Simplified Notation ............................. 5
      1.2.2 Generalized Skew Reed-Solomon Codes ............... 5
      1.2.3 Kötter Interpolation in Skew Polynomial Rings ... 6
      1.2.4 Matroidal Structure of Skew Polynomials ............ 6
   1.3 Organization of Thesis ................................. 7

2 Skew Polynomials ........................................ 8
   2.1 Basic Definitions ..................................... 9
   2.2 Skew Polynomial Evaluation ............................. 14
   2.3 Minimal Polynomial and $P$-Independent Sets ........... 17
      2.3.1 Minimal Polynomials ................................ 18
      2.3.2 $P$-independent Sets ................................ 20
      2.3.3 Vandermonde Matrices ............................... 22
   2.4 Summary ................................................ 23

3 Skew Polynomials Rings over Finite Fields ................. 24
   3.1 Skew Polynomial Rings without Derivation .............. 25
      3.1.1 Notation ............................................. 25
5.1.4 Interpolation Algorithm ........................................ 64
5.2 Newton Interpolation for Skew Polynomials .................. 70
  5.2.1 Newton Interpolation Problem ............................... 70
  5.2.2 Newton Interpolation Algorithm ......................... 72
  5.2.3 Computational Complexity ................................... 76
5.3 Berlekamp-Welch Decoder for GSRS-Codes .................. 77
  5.3.1 Berlekamp-Welch Setup ..................................... 77
  5.3.2 Berlekamp-Welch via Kötter Interpolation ............... 78
5.4 Summary .......................................................... 81

6 Matroidal Structure & Application to Network Coding 83
  6.1 Matroidal Structure ............................................. 84
    6.1.1 Matroid Basics ............................................. 84
    6.1.2 The $\mathbb{F}_{q^m}[x;\sigma]$-matroid ....................... 87
    6.1.3 $\mathcal{F}(\mathbb{F}_{q^m})$ Metric Space ..................... 89
    6.1.4 $C(1)$-submatroid and Projective Geometry .............. 90
  6.2 Application to Matroidal Network Coding ................... 92
    6.2.1 Communication using $\mathbb{F}_{q^m}[x;\sigma]$-matroid ..... 93
    6.2.2 Computational Complexity .................................. 94
  6.3 Summary .......................................................... 96

7 Conclusion .......................................................... 97
  7.1 Future Work ..................................................... 98

Bibliography .......................................................... 101
Chapter 1

Introduction

In the history of algebraic coding theory, the design of codes using polynomial evaluation has always played a major role. The most well-known example is the celebrated Reed-Solomon code invented in 1960 [1]. Since then, Reed-Solomon codes have been used ubiquitously, including consumer technologies such as CDs, DVDs, data transmission technologies such as DSL, broadcast systems such as DVB, and storage systems such as RAID 6. The essential theory of Reed-Solomon codes boils down to the evaluations of polynomials over a finite field. Since polynomials over finite fields are a well-understood subject in abstract algebra, Reed-Solomon codes naturally inherit a wealth of algebraic structures. For example, the fundamental theorem of algebra dictates the number of roots a polynomial of degree $k$ can have. Thus, one can immediately determine that Reed-Solomon codes are optimal in the sense that they satisfy the Singleton bound with equality. Furthermore, the algebraic structures also allow Reed-Solomon codes to have efficient decoding procedures. The well-known decoding algorithms such as Berlekamp-Massey, Berlekamp-Welch [2], and the more modern Sudan’s list-decoding [3] are some examples.

With the tremendous success of Reed-Solomon codes, it is natural to ask if there are other types of interesting codes that are based on polynomial evaluation. One surprising
answer came from the seminal work of Gabidulin in 1985 [4]. In this work, Gabidulin defined a new distance measure called rank distance. He first showed that the codes optimal for the rank distance are also optimal for the traditional Hamming distance. Next, he defined a class of codes, now called Gabidulin codes, that are optimal for the rank distance. Gabidulin codes are intimately related to Reed-Solomon codes. Just as Reed-Solomon codes stem from the evaluations of regular polynomials over a finite field, Gabidulin codes are constructed from the evaluations of linearized polynomials over a finite field. Just like Reed-Solomon codes, Gabidulin codes also inherit significant algebraic structures. It can be shown that the roots of a linearized polynomial form a vector space (over a subfield of the given finite field). This vector space structure is of prime importance in network coding. Many important works in network coding have utilized Gabidulin codes and linearized polynomials [5, 6].

The existence of linearized polynomials begs the question of other possible types of polynomials that can be defined over a finite field. The pioneering study on the most general polynomials (not necessarily defined over fields) was conducted in the mathematics literature by Ore in 1934 [7]. Ore argued that the defining property of polynomials is that the degree of the product of two polynomials is the sum of their individual degrees. Having this property in mind, Ore defined the most general polynomial rings that satisfy such property. The modern name for such a ring is skew polynomial ring.

Although the basic mathematical structure of skew polynomial rings is well investigated by Ore, the evaluation theory of skew polynomials was only developed in 1986 by Lam [8]. This evaluation theory gave skew polynomial rings many additional algebraic structures that are useful in the context of coding theory. Equipped with this evaluation theory and specializing to the case of finite fields, this thesis examines the codes that arise from the evaluations of skew polynomials. We employ an approach that parallels both the development of Reed-Solomon codes and Gabidulin codes. The goal of the thesis is to fill the box below.
More specifically, following the development of Reed-Solomon and Gabidulin codes, we will define a class of codes that arise from skew polynomial evaluation called Generalized Skew Reed-Solomon (GSRS) codes. Just like the case of Reed-Solomon and Gabidulin codes, our new code construction is optimal under particular distance measures. GSRS codes inherit many algebraic structures from the rich theory of skew polynomials. These algebraic structures enable us to develop an important duality theory for these codes. They also allow us to establish a generalized interpolation framework for skew polynomials that contains a Berlekamp-Welch style decoder for GSRS codes as a special case.

### 1.1 Previous Work

The foundation theory of skew polynomials was first developed by Ore in 1934 [7] and followed by subsequent works of Jacobson [9] and Cohn [10]. The evaluation theory of skew polynomials was first proposed by Lam in 1985 [8], and quickly followed by subsequent works of Lam and Leroy [11, 12]. The application of skew polynomials in the coding theory only appeared in the last several years. In a series of papers [13, 14, 15, 16], Boucher and Ulmer investigated skew polynomials in the context of constructing skew cyclic codes. Just as cyclic codes are defined as ideals in an ordinary polynomial ring,
skew cyclic codes are defined as *left ideals* in a skew polynomial ring. This generalization is important due to the lack of unique factorization in a skew polynomial ring. Thus, in general, a skew polynomial can be factored (into irreducible polynomials) in many different ways. This property allows a significant increase in the number of different skew cyclic codes as compared to traditional cyclic codes. A brute force code search allowed Boucher and Ulmer to find several codes with the best known distance property to date.

While there are many subsequent papers on extensions of skew cyclic codes [17, 18, 19], there is only one notable investigation into codes that arise from evaluation of skew polynomials [20]. In this work, the authors used the skew polynomial evaluation theory in [11] to define *remainder evaluation skew codes*. They studied some basic properties of such codes and drew some connections to skew module codes (generalization of skew cyclic codes) from previous work [16]. This work serves as the basis of our investigation into evaluation codes based on skew polynomials. However, our approach differs significantly from this work.

It is worthwhile to note that skew polynomials have also been studied in cryptographic applications in works such as [21, 22]. In [23], skew polynomials are used in analyzing shift-register synthesis as a generalization of the case of ordinary polynomials. In [24], the author gave some basic ideas of skew polynomial interpolation. In [22], a version of Newton interpolation for skew polynomial was presented.

## 1.2 Our Contributions

In this thesis, we contributed to the study of coding theory using skew polynomials in the followings areas.
1.2.1 New Simplified Notation

In part of Chapter 2 and most of Chapter 3, we developed a series of new notation for the special case of skew polynomial ring without derivation. This set of notation allows us to give many original and simple proofs to important properties of skew polynomial rings. Furthermore, our notation allows us to present the evaluation theory of skew polynomials in parallel with the evaluation theory of linearized polynomials. The interplay between these two theories is developed through a series of theorems and is summarized by the Structure Theorem at the end of Chapter 3. The Structure Theorem serves as the foundation for a large part of both Chapter 4 and Chapter 6.

1.2.2 Generalized Skew Reed-Solomon Codes

To investigate the codes that arise from skew polynomial evaluation, we defined Generalized Skew Evaluation code and the special case of Generalized Skew Reed-Solomon codes in Chapter 4. Our definition here is a generalization of the definition given in [20]. Our more general definition is important for several reasons. First, it allows us to show that the well-known Gabidulin codes are a special case of Generalized Skew Reed-Solomon codes. With this observation, we can design a particular pasting construction of codes. This construction has an interesting property that individual component codes are optimal for the rank distance, while the overall code is optimal for the Hamming distance. The general definition also allows us to prove a fundamental duality theory within a subclass of Generalized Skew Reed-Solomon codes. This duality theory can be viewed as an extension to the duality theory of the classical Generalized Reed-Solomon codes [25]. Our duality theory is a cumulative result that encompasses key ideas from the theories of both linearized polynomials and skew polynomials.
1.2.3 Kötter Interpolation in Skew Polynomial Rings

In decoding polynomial evaluation codes, polynomial interpolation is a key step. One important general interpolation algorithm is the Kötter interpolation, which has been shown to be adaptable to many practical applications [26]. In previous work, Kötter interpolation has been defined over regular polynomial rings as well as linearized polynomial rings [26, 27]. In our work, we generalized the setup to skew polynomial rings. The interpolation framework in previous work was presented in vague terms and lacked mathematical rigour. In our generalization, we put forth a systematic mathematical framework that is rigorous and also encompasses previous work as special cases.

Our Kötter interpolation algorithm can be adapted to two important applications. The first is a generalization of the existing Newton interpolation algorithm for skew polynomials [22]. Compared to previous work [22], our new Newton interpolation algorithm (as a special case of Kötter interpolation in skew polynomial rings) relaxes the constraints on interpolation points and reduces the overall computational complexity. The second application is a Berlekamp-Welch style decoder for the Generalized Skew Reed-Solomon codes that we previously defined. This decoder has the same complexity as the traditional Berlekamp-Welch decoder for Reed-Solomon codes. This algorithm affirms the fact that our Generalized Skew Reed-Solomon codes have an efficient practical decoder.

1.2.4 Matroidal Structure of Skew Polynomials

A skew polynomial ring over a finite field induces an independence structure (called $P$-independence) on the underlying field. Using the Structure Theorem, we show that this independence structure is intimately related to the linear independence structure of vector spaces. Motivated by the general definition of independence for a matroid, we constructed a new matroid called the $\mathbb{F}_{q^m}[x;\sigma]$-matroid. This matroid has many interesting properties. The collection of its flats form a metric space. The collection of flats on a submatroid is bijectively isometric to the projective geometry of $\mathbb{F}_{q^m}$ over $\mathbb{F}_q$. 
equipped with the \textit{subspace metric}. Using these properties, we present an application of the \(\mathbb{F}_{q^m}[x;\sigma]\)-matroid in the context of network coding.

1.3 Organization of Thesis

The rest of the thesis is organized as follows. In Chapter 2, we introduce the basic theory of skew polynomial rings over fields, and outline a few key concepts that will be used throughout the thesis. In Chapter 3, we specialize the theory developed in Chapter 2 to the case of skew polynomial rings without \textit{derivation}. In this special case, we develop a set of notation that allows us to efficiently describe the coding theory aspect of skew polynomial evaluation. In Chapter 4, we define Generalized Skew Evaluation codes and Generalized Skew Reed-Solomon codes. Using our definitions, we present a novel pasting construction code design that results in a code with good properties in both the rank and Hamming distance. We also develop an important duality theory for a class of Generalized Reed-Solomon Codes. In Chapter 5, we set up the Kötter interpolation framework for skew polynomial rings. Using this framework, we discuss two applications: Newton interpolation for skew polynomials and Berlekamp-Welch decoder for Generalized Skew Reed-Solomon codes. In Chapter 6, we introduce the basic theory of matroids and construct a new matroid called the \(\mathbb{F}_{q^m}[x;\sigma]\)-matroid. We also present an application of the \(\mathbb{F}_{q^m}[x;\sigma]\)-matroid in the context of network coding. In Chapter 7, we give some concluding remarks and point to potential directions for future work.
Chapter 2

Skew Polynomials

In this chapter, we examine some basic properties of skew polynomial rings. In general, skew polynomial rings can be defined over any division ring. We shall restrict to the case of skew polynomial rings defined over fields, keeping in mind that most coding-theoretic and cryptographic applications use skew polynomial rings defined over finite fields. The basic theory stated here mostly originated from the pioneering work on the subject by Ore [7]. The evaluation theory of skew polynomial rings comes from the work of by Lam and Leroy [11]. The general theory developed here will serve as the foundation for the special case that we will consider in Chapter 3, and will also be used extensively in the setup of Kötter interpolation algorithm in Chapter 5.

The rest of this chapter is organized as follows. In Section 2.1, we introduce some basic definitions and define skew polynomial rings. In Section 2.2, we introduce the evaluation map for skew polynomial rings. In Section 2.3, we discuss the minimal polynomials associated with skew polynomial evaluation as well as the important concept of $P$-independent sets. A generalization of Vandermonde matrices is also described. Section 2.4 offers some concluding statements.
2.1 Basic Definitions

Let $F$ be a field. An automorphism of $F$ is a bijective map $\sigma : F \rightarrow F$ satisfying the following properties for $a, b \in F$.

\[ \sigma(1) = 1 \]
\[ \sigma(a + b) = \sigma(a) + \sigma(b) \]
\[ \sigma(ab) = \sigma(a)\sigma(b). \]

**Example 2.1.1.** For any field $F$, the identity map from $F \rightarrow F$ is an automorphism. It is called the trivial automorphism.

**Example 2.1.2.** Let $\mathbb{C}$ be the field of complex numbers. We can easily verify that the conjugation map $\sigma(a + bi) = a - bi$, for all $a + bi \in \mathbb{C}$, is an automorphism of $\mathbb{C}$.

**Example 2.1.3.** Consider the finite field $F_{16}$. Since $F_{16}$ is a field of characteristic 2. Then for any $a, b \in F_{16}$, we have $(a + b)^2 = a^2 + b^2$. Thus, the map $\sigma_1(a) = a^2$, for all $a \in F_{16}$ is an automorphism of $F_{16}$. Similarly, the maps $\sigma_2(a) = a^4$ and $\sigma_3(a) = a^8$ are also automorphisms of $F_{16}$.

Let $F$ be a field and $\sigma$ be an automorphism of $F$. A $\sigma$-derivation is a map $\delta_\sigma : F \rightarrow F$ such that, for all $a$ and $b$ in $F$,

\[ \delta_\sigma(a + b) = \delta_\sigma(a) + \delta_\sigma(b), \]  
\[ \delta_\sigma(ab) = \delta_\sigma(a)b + \sigma(a)\delta_\sigma(b). \]  

We shall ignore the subscript when there is no ambiguity about the automorphism $\sigma$. Note that from the condition on $\delta(ab)$, we see that $\delta(1) = 0$. When restricted to finite fields, we can easily characterize all $\sigma$-derivations by the following proposition.

**Proposition 2.1.4.** Over a finite field $F_q$, all $\sigma$-derivations are of the form $\delta(a) =$
Chapter 2. Skew Polynomials

10

\[ \beta(\sigma(a) - a), \text{ for } \beta \in \mathbb{F}_q. \]

Proof. Write \( \mathbb{F}_q = \mathbb{F}_{p^n} \) for some prime \( p \) and denote the automorphism \( \sigma \) by \( \sigma(a) = a^{p^n} \).

Any map from \( \mathbb{F}_q \) to \( \mathbb{F}_q \) can be represented by a polynomial function with degree less than \( q = p^n \). Thus, we can write \( \delta(x) = \sum_{i=0}^{p^n-1} c_i x^i \), and it suffices to determine the form of the \( c_i \)'s. Next, (2.1) implies,

\[
\sum_i c_i (a + b)^i = \sum_i c_i a^i + \sum_i c_i b^i.
\]

For this to hold for all \( a, b \in \mathbb{F}_q \), we need \( p \) to divide \( i \) for every \( i \). Thus, \( \delta(x) \) must have the form \( \delta(x) = \sum_k c_k x^{p^k} \). Then, (2.2) implies,

\[
\sum_k c_k (ab)^{p^k} = \sum_k c_k a^{p^k} b + \sum_k c_k b^{p^k} a^{p^m}.
\]

This must hold for all \( a, b \in \mathbb{F}_q \). By comparing coefficients, the only possible nonzero coefficients are \( c_0 \) and \( c_m \). Then,

\[
c_0(ab) + c_m(ab)^{p^m} = c_0 ab + c_m a^{p^m} b + c_0 a^{p^m} b + c_m a^{p^m} b^{p^m}.
\]

Thus, \( c_m = -c_0 \), and \( \delta(x) = c_m (x^{p^m} - x) \). As \( c_m \) is arbitrary, we can write \( \delta(x) = \beta(x^{p^m} - x) \), for \( \beta \in \mathbb{F}_q \), as desired.

\[ \square \]

Remark 2.1.5. The above result also follows as a special case from a much more general theorem by Cohn [10]. The proof we presented here is original.

Definition 2.1.6. A skew polynomial ring over a field \( \mathbb{F} \) with automorphism \( \sigma \) and derivation \( \delta \), denoted \( \mathbb{F}[x; \sigma, \delta] \), is a ring which consists of polynomials \( \sum a_i x^i \) (\( a_i \in \mathbb{F} \)) with the usual addition of polynomials and a multiplication that distributes over addition and follows the commuting rule \( xa = \sigma(a)x + \delta(a) \).
Example 2.1.7. Consider $\mathbb{F}_4[x; \sigma, \delta]$ with $\mathbb{F}_4 = \{0, 1, \alpha, \alpha^2 = 1 + \alpha\}$, $\sigma(a) = a^2$, and $\delta(a) = \sigma(a) + a$. Then,

$$(x + 1)(\alpha x + 1) = x(\alpha x + 1) + (\alpha x + 1)$$

$$= (\sigma(a)x + \delta(a))x + x + \alpha x + 1$$

$$= (\alpha^2x + \alpha^2 + \alpha)x + x + \alpha x + 1$$

$$= \alpha^2x^2 + (\alpha^2 + \alpha + 1)x + 1$$

$$= \alpha^2x^2 + \alpha x + 1$$

In this thesis, we will often consider the case where $\delta = 0$. Over a finite field, there is actually a change of variable that allows one to convert from a skew polynomial ring with $\delta \neq 0$ to one with $\delta = 0$. The following is a special case of a theorem by Cohn [10].

Proposition 2.1.8. Let $\mathbb{F}$ be a finite field and $\mathbb{F}[x; \sigma, \delta_\beta]$ be a skew polynomial ring with $\delta_\beta(a) = \beta(\sigma(a) - a)$ for all $a \in \mathbb{F}$. Let $\mathbb{F}[y; \sigma]$ be a skew polynomial ring with $\delta = 0$. Then the map

$$\Phi: \mathbb{F}[x; \sigma, \delta_\beta] \to \mathbb{F}[x, \sigma]$$

$$\sum a_i x^i \mapsto \sum a_i (y - \beta)^i$$

is a ring isomorphism.

Remark 2.1.9. Note that the above proposition shows an isomorphism of rings only. In the context of coding theory, skew polynomial rings with derivation have shown to produce codes that are different from those that arises from skew polynomial rings without derivation [20].

Remark 2.1.10. When $\sigma$ is the identity automorphism and $\delta$ is the zero derivation, we recover the ordinary polynomial ring $\mathbb{F}[x]$. 
Example 2.1.11. Consider $\mathbb{F}_4[x;\sigma]$ as before with $\sigma(a) = a^2$ but $\delta = 0$. Then,

\[(x + 1)(\alpha x + 1) = x(\alpha x + 1) + (\alpha x + 1) = \sigma(\alpha)x^2 + x + \alpha x + 1 = \alpha^2 x^2 + \alpha^2 x + 1.\]

Example 2.1.12. Let $\mathbb{F}_{q^m}$ be a field extension of the finite field $\mathbb{F}_q$. Let $\sigma$ be the automorphism of $\mathbb{F}_{q^m}$ given by $\sigma(a) = a^q$ for all $a \in \mathbb{F}_{q^m}$, and let $\delta$ be the zero derivation. Then $\mathbb{F}_{q^m}[x;\sigma]$ is isomorphic to the ring of linearized polynomials over $\mathbb{F}_{q^m}$ [28]. We shall explore this connection in greater detail in future chapters.

Clearly, since $xa \neq ax$ in general, $\mathbb{F}_{q^m}[x;\sigma,\delta]$ is a noncommutative ring. As the next example shows, it is not a unique factorization domain.

Example 2.1.13. Consider $\mathbb{F}_4[x;\sigma]$ as in Example 2.1.11. Then,

\[x^4 + x^2 + 1 = (x^2 + x + 1)(x^2 + x + 1) = (x^2 + \alpha^2)(x^2 + \alpha),\]

are two possible irreducible factorizations.

Although skew polynomial rings behave quite differently from ordinary polynomial rings, they do retain the important property of having a division algorithm. We prove the following [9].

Proposition 2.1.14. $\mathbb{F}[x;\sigma,\delta]$ is a right Euclidean domain.

Proof. We shall establish a right division algorithm for $\mathbb{F}[x;\sigma,\delta]$. That is, for any $f(x), g(x) \in \mathbb{F}[x;\sigma,\delta]$ with $\deg(f) > 0, \deg(g) > 0$, there exist unique $q(x), r(x)$ with
\[
deg r(x) < \deg g(x) \text{ such that } \\
f(x) = q(x)g(x) + r(x).
\]

Suppose \( f(x) = a_0 + a_1x + \ldots + a_nx^n, g(x) = b_0 + b_1x + \ldots + b_mx^m \), with \( a_n \neq 0 \) and \( b_m \neq 0 \). If \( n < m \), then clearly \( f(x) = 0g(x) + f(x) \). If \( n \geq m \), we have:

\[
f(x) - a_n(\sigma^{n-m}(b_m))^{-1}x^{n-m}g(x) = c_{n-1}x^{n-1} + \ldots.
\]

Thus, the existence of \( q(x) \) and \( r(x) \) follow by induction on \( n \), and uniqueness follows by degree considerations.

\[\textbf{Remark 2.1.15.} \quad \text{Since } \mathbb{F}[x;\sigma,\delta] \text{ is a noncommutative ring, the order of } q(x)g(x) \text{ in (2.3) is important. The name right Euclidean domain refers to } g(x) \text{ appearing to the right of } q(x). \text{ In fact, } \mathbb{F}[x;\sigma,\delta] \text{ is also a left Euclidean domain.}\]

Since we have a well-defined division algorithm on \( \mathbb{F}[x;\sigma,\delta] \), the standard notion of greatest common divisor (gcd) and least common multiple (lcm) also have their corresponding generalizations.

\[\textbf{Definition 2.1.16.} \quad \text{For nonzero } f_1, f_2 \in \mathbb{F}[x;\sigma,\delta], \text{ the greatest right common divisor (grcd) of } f_1 \text{ and } f_2, \text{ denoted } \text{grcd}(f_1, f_2), \text{ is the unique monic polynomial } g \in \mathbb{F}[x;\sigma,\delta] \text{ of highest degree such that there exist } u_1, u_2 \in \mathbb{F}[x;\sigma,\delta] \text{ with } f_1 = u_1g \text{ and } f_2 = u_2g.\]

\[\textbf{Definition 2.1.17.} \quad \text{For nonzero } f_1, f_2 \in \mathbb{F}[x;\sigma,\delta], \text{ the least left common multiple (llcm) of } f_1 \text{ and } f_2, \text{ denoted } \text{llcm}(f_1, f_2), \text{ is the unique monic polynomial } h \in \mathbb{F}[x;\sigma,\delta] \text{ of lowest degree such that there exist } u_1, u_2 \in \mathbb{F}[x;\sigma,\delta] \text{ with } h = u_1f_1 \text{ and } h = u_2f_2.\]

Using the division algorithm, we can easily verify the following.

\[\textbf{Proposition 2.1.18.} \quad \text{For all } f_1, f_2 \in \mathbb{F}_{q^n}[x;\sigma,\delta], \]

\[
\deg(\text{llcm}(f_1, f_2)) = \deg(f_1) + \deg(f_2) - \deg(\text{grcd}(f_1, f_2)).
\]
2.2 Skew Polynomial Evaluation

When defining an evaluation map for a skew polynomial ring, it is important to take into account the actions of $\sigma$ and $\delta$. The traditional “plug in” map that simply replaces the variable $x$ by a value $a \in F$ does not work. A suitable evaluation map was found by Lam and Leroy [11], and the method can be motivated as follows.

If $f(x) \in F[x]$, then we actually have two interpretations of $f(a)$. One way is simply plugging in $a$ for the variable $x$; the other is to define $f(a)$ as the remainder upon division of $f(x)$ by $x - a$. These two methods give the same result. For skew polynomial rings, although we cannot use the simple “plug in” map, we can still define the evaluation of $f(x) \in F[x; \sigma, \delta]$ as the remainder upon right division by $x - a$. Since we have a right division algorithm for $F[x; \sigma, \delta]$, this evaluation map is well-defined. Thus, we can state the following.

**Definition 2.2.1.** For $a \in F$, $f(x) \in F[x; \sigma, \delta]$ and $f(x) = q(x)(x - a) + r$, the evaluation of $f(x)$ at $a$ is defined as $f(a) = r$.

In fact, we can compute this evaluation without using the division algorithm. Define the following set of functions $N_i : F \to F$ recursively:

$$N_0(a) = 1,$$

$$N_{i+1}(a) = \sigma(N_i(a))a + \delta(N_i(a)),$$

for all $a \in F$. Equipped with these $N_i$ functions, Lam and Leroy [11] proved the following.

**Theorem 2.2.2.** For $f(x) = \sum a_i x^i \in F[x; \sigma, \delta]$ and $a \in F$, we have $f(a) = \sum a_i N_i(a)$.

**Example 2.2.3.** Let $F_4 = \{0, 1, \alpha, \alpha^2\}$, with $\alpha^2 = 1 + \alpha$. Consider $F_4[x; \sigma, \delta]$, with $\sigma(a) = a^2$, $\delta(a) = \sigma(a) + a$, for all $a \in F$. Let $f(x) = \alpha x^2 + x$, and evaluate $f(\alpha)$ in two different
ways. First,

\[(\alpha x)(x - \alpha) = (\alpha x)(x + \alpha)\]
\[= \alpha x^2 + \alpha(\sigma(\alpha)x + \delta(\alpha))\]
\[= \alpha x^2 + \alpha^2x + \alpha(\alpha^2 + \alpha)\]
\[= \alpha x^2 + x + \alpha.\]

So,

\[f(x) = (\alpha x)(x - \alpha) + \alpha,\]

thus, \(f(\alpha) = \alpha\). On the other hand,

\[N_0(\alpha) = 1,\]
\[N_1(\alpha) = \sigma(N_0(\alpha))\alpha + \delta(N_0(\alpha))\]
\[= \alpha,\]
\[N_2(\alpha) = \sigma(N_1(\alpha))\alpha + \delta(N_1(\alpha))\]
\[= \alpha^2\alpha + \alpha^2 + \alpha\]
\[= 1 + \alpha^2 + \alpha\]
\[= 0,\]

hence, \(f(\alpha) = \sum_i a_i N_i(\alpha) = 0 \cdot N_0(\alpha) + 1 \cdot N_1(\alpha) + \alpha \cdot N_2(\alpha) = \alpha.\)

Unlike the evaluation map for ordinary polynomial rings, this evaluation map is not a ring homomorphism. In particular, \(fg(a) \neq f(a)g(a)\). In order to evaluate a product, we need the concept of \((\sigma, \delta)\)-conjugacy.
**Definition 2.2.4.** For any two elements \( a \in \mathbb{F}, c \in \mathbb{F}^* = \mathbb{F} \setminus \{0\} \), define:

\[
a^c \triangleq \sigma(c)a^{-1} + \delta(c)c^{-1}.
\]

We call two elements \( a, b \in \mathbb{F} \) \((\sigma, \delta)\)-conjugates if there exists an element \( c \in \mathbb{F}^* \) such that \( a^c = b \). Two elements that are not \((\sigma, \delta)\)-conjugates are called \((\sigma, \delta)\)-distinct.

It is straightforward to verify that conjugacy is an equivalence relation [11]. We call the set \( C(a) = \{a^c | c \in \mathbb{F}^*\} \) the conjugacy class of \( a \).

**Example 2.2.5.** Consider the finite field \( \mathbb{F}_{16} \cong \mathbb{F}_2[\gamma] \), where \( \gamma \) is a primitive root of the irreducible polynomial \( x^4 + x + 1 \). For an element \( a \in \mathbb{F}_{16} \), let \( \sigma(a) = a^4 \), \( \delta(a) = \sigma(a) + a = a^4 + a \). The \((\sigma, \delta)\)-conjugacy classes are as follows.

\[
1^c = \sigma(c)c^{-1} + \delta(c)c^{-1} \\
= \sigma(c)c^{-1} + (\sigma(c) + c)c^{-1} \\
= cc^{-1} \\
= 1.
\]

Thus, the conjugacy class \( C(1) = \{1\} \). For \( C(0) \),

\[
0^c = \delta(c)c^{-1} \\
= c^4c^{-1} + 1 \\
= c^3 + 1
\]
so,

\begin{align*}
0^7 &= \gamma^3 + 1 = \gamma^{14}, \\
0^7 &= \gamma^6 + 1 = \gamma^{13}, \\
0^7 &= \gamma^9 + 1 = \gamma^7, \\
0^7 &= \gamma^{12} + 1 = \gamma^{11}, \\
0^7 &= \gamma^{15} + 1 = 0.
\end{align*}

Thus, the conjugacy class \( C(0) = \{0, \gamma^7, \gamma^{11}, \gamma^{13}, \gamma^{14}\} \). Similar computation shows that \( C(\gamma) = \{\gamma, \gamma^4, \gamma^5, \gamma^6, \gamma^9\} \) and \( C(\gamma^2) = \{\gamma^2, \gamma^3, \gamma^8, \gamma^{10}, \gamma^{12}\} \).

The evaluation of a product is related to the \((\sigma, \delta)\)-conjugacy in the following way.

**Theorem 2.2.6** (Product Theorem). \([11]\) Let \( f(x), g(x) \in \mathbb{F}[x; \sigma, \delta] \) and \( a \in \mathbb{F} \). If \( g(a) = 0 \), then \((fg)(a) = 0\). If \( g(a) \neq 0 \), then \((fg)(a) = f(a^{g(a)})g(a)\).

**Example 2.2.7.** Consider \( \mathbb{F}_4[x; \sigma] \) as before, with \( \sigma(a) = a^2 \). Let \( f = x^4 + x^2 + 1, \ g = x^2 + x + 1 \) and \( h = x^2 + x + 1 \), so that \( f = gh \). By Theorem 2.2.2,

\begin{align*}
f(\alpha) &= N_4(\alpha) + N_2(\alpha) + N_0(\alpha) \\
&= \alpha^{15} + \alpha^3 + 1 = 1.
\end{align*}

By Theorem 2.2.6,

\begin{align*}
gh(\alpha) &= g(\alpha^{h(\alpha)})h(\alpha) = \alpha^2 \alpha = 1.
\end{align*}

### 2.3 Minimal Polynomial and \( P \)-Independent Sets

Throughout this section, \( \Omega \) is taken to be a finite subset of the field \( \mathbb{F} \).
2.3.1 Minimal Polynomials

For any polynomial \( f \), either in \( \mathbb{F}[x;\sigma,\delta] \) or in \( \mathbb{F}[x] \), let

\[
Z(f) = \{ a \in \mathbb{F} \mid f(a) = 0 \}.
\]

That is, \( Z(f) \) is the set of zeros of \( f \).

If \( f \in \mathbb{F}[x] \) is nonzero and \( \deg(f) = n \), we know that \( |Z(f)| \leq n \). However, as the next example shows, a skew polynomial can have more zeros than its degree.

**Example 2.3.1.** Let \( f = x^2 + 1 \in \mathbb{F}_4[x;\sigma] \), with \( \sigma(a) = a^2 \) for all \( a \in \mathbb{F}_4 \). Then, \( Z(f) = \{1, \alpha, \alpha^2\} \), since, for \( a \in \mathbb{F}_4^* \),

\[
f(a) = N_2(a) + N_0(a) = a^3 + 1 = 0.
\]

**Definition 2.3.2.** Let \( \Omega \subseteq \mathbb{F} \) and let \( f_\Omega \in \mathbb{F}[x;\sigma,\delta] \) be the monic polynomial of least degree such that \( f_\Omega(a) = 0 \) for all \( a \in \Omega \). We call \( f_\Omega \) the minimal polynomial of \( \Omega \). The empty set has \( f_\varnothing = 1 \).

**Proposition 2.3.3.** Let \( \Omega \subseteq \mathbb{F} \) and let \( f_\Omega \in \mathbb{F}[x;\sigma,\delta] \) be its minimal polynomial. Then for any \( \beta \notin Z(f_\Omega) \), we have \( f_{\Omega \cup \{\beta\}} = (x - \beta^{f_\Omega(\beta)}) f_\Omega \).

**Proof.** By Theorem 2.2.6, we know that \( (x - \beta^{f_\Omega(\beta)}) f_\Omega \) vanishes on \( \Omega \cup \{\beta\} \). To check minimality, we note that \( \deg((x - \beta^{f_\Omega(\beta)}) f_\Omega) = \deg(f_\Omega) + 1 \) and no polynomial of \( \deg(f_\Omega) \) can vanish on \( \Omega \cup \{\beta\} \).

**Corollary 2.3.4.** Let \( \Omega \subseteq \mathbb{F} \). Then, \( f_\Omega = (x - a_1)(x - a_2)\cdots(x - a_n) \) where each \( a_i \) is conjugate to some element of \( \Omega \).

**Proof.** For any \( \alpha \in \Omega \), \( f_{(\alpha)} = x - \alpha \). The statement follows by iteratively applying Proposition 2.3.3. 

\( \Box \)
Proposition 2.3.3 and Corollary 2.3.4 imply that the zeros of \( f_\Omega \) are well-behaved in the following sense.

**Theorem 2.3.5.** [8] Every root of \( f_\Omega \) is a \( \sigma \)-conjugate to an element in \( \Omega \).

We shall also state the following useful theorem.

**Theorem 2.3.6.** [12] Let \( \Omega \subseteq F \). If \( f_\Omega = pg \), with \( p, g \in F[x;\sigma,\delta] \), then \( g = f_{Z(g)} \), i.e., \( g \) is a minimal polynomial.

Lastly, we prove the following important decomposition theorem for minimal polynomials.

**Theorem 2.3.7 (Decomposition Theorem).** Let \( \Omega_1, \Omega_2 \subseteq F \), with corresponding minimal polynomials \( f_{\Omega_1} \) and \( f_{\Omega_2} \) such that \( \Omega_1 = Z(f_{\Omega_1}) \) and \( \Omega_2 = Z(f_{\Omega_2}) \). Then, we have

1. \( f_{\Omega_1 \cup \Omega_2} = \text{lcm}(f_{\Omega_1}, f_{\Omega_2}) \).

2. \( f_{\Omega_1 \cap \Omega_2} = \text{gcd}(f_{\Omega_1}, f_{\Omega_2}) \).

3. \( \deg(f_{\Omega_1 \cup \Omega_2}) = \deg(f_{\Omega_1}) + \deg(f_{\Omega_2}) - \deg(f_{\Omega_1 \cap \Omega_2}) \).

**Proof.**

1. Since every \( \alpha \in \Omega_1 \) is a zero of \( f_{\Omega_1 \cup \Omega_2} \), we have \( f_{\Omega_1 \cup \Omega_2} = p_1 f_{\Omega_1} \) for some \( p_1 \in F[x;\sigma,\delta] \). Similarly, every \( \beta \in \Omega_2 \) is a zero of \( f_{\Omega_1 \cup \Omega_2} \), so we have \( f_{\Omega_1 \cup \Omega_2} = p_2 f_{\Omega_2} \) for some \( p_2 \in F[x;\sigma,\delta] \). Since \( \text{lcm}(f_1, f_2) \) is the monic polynomial of lowest degree with this property, we must have \( f_{\Omega_1 \cup \Omega_2} = \text{lcm}(f_{\Omega_1}, f_{\Omega_2}) \).

2. Every \( \alpha \in \Omega_1 \cap \Omega_2 \) is a zero of both \( f_{\Omega_1} \) and \( f_{\Omega_2} \). Thus, we can write \( f_{\Omega_1} = p_1 f_{\Omega_1 \cap \Omega_2} \) and \( f_{\Omega_2} = p_2 f_{\Omega_1 \cap \Omega_2} \) for some \( p_1, p_2 \in F[x;\sigma,\delta] \). Thus, by the definition of \( \text{gcd} \), we have \( f_{\Omega_1 \cap \Omega_2} \mid \text{gcd}(f_{\Omega_1}, f_{\Omega_2}) \), where \( \mid \) denotes right divisibility. Now, clearly every \( \beta \in Z(\text{gcd}(f_{\Omega_1}, f_{\Omega_2})) \) is a zero of both \( f_{\Omega_1} \) and \( f_{\Omega_2} \); since \( \Omega_1 = Z(f_{\Omega_1}) \) and \( \Omega_2 = Z(f_{\Omega_2}) \), we have that \( \beta \) is a zero of \( f_{\Omega_1 \cap \Omega_2} \). By Theorem 2.3.6, \( \text{gcd}(f_{\Omega_1}, f_{\Omega_2}) \) is the minimal polynomial of \( Z(\text{gcd}(f_{\Omega_1}, f_{\Omega_2})) \). Thus \( f_{\Omega_1 \cap \Omega_2} = \text{gcd}(f_{\Omega_1}, f_{\Omega_2}) \).
Remark 2.3.8. Our discussion on minimal polynomials is quite different from the original discussion in the work of Lam [8]. All the proofs in this section are original.

2.3.2 $P$-independent Sets

Extending Example 2.3.1, we see that if $\Omega = \{1, \alpha\}$, then $f_\Omega = x^2 + 1$. However, $Z(f_\Omega) = \{1, \alpha, \alpha^2\}$. This shows that, in a skew polynomial ring, it is possible that $|Z(f_\Omega)| > |\Omega|$. This motivates the following definition by Lam [8].

Definition 2.3.9. An element $\alpha \in F$ is $P$-dependent on a set $\Omega$ if $f_\Omega = f_{\Omega \cup \{\alpha\}}$ and $P$-independent of $\Omega$ otherwise. A set of elements $\Omega = \{\alpha_1, \ldots, \alpha_n\} \subseteq F$ is $P$-independent if for any $i \in \{1, \ldots, n\}$ the element $\alpha_i$ is $P$-independent of the set $\Omega \setminus \{\alpha_i\}$.

Remark 2.3.10. If we are working over the regular polynomial ring $F[x]$, the analogous notion of $P$-independence is simply “distinctness”. Thus the $P$-independence structure of $F[x]$ is trivial. Every set contains distinct elements is $P$-independent.

Definition 2.3.11. The $P$-closure of the set $\Omega$ is

$$\overline{\Omega} = \{\alpha \in F \mid f_\Omega(\alpha) = 0\}.$$ 

Any maximal $P$-independent subset of $\overline{\Omega}$ is called a $P$-basis for $\overline{\Omega}$. $\Omega$ is called $P$-closed if $\Omega = \overline{\Omega}$.

Remark 2.3.12. $\Omega$ is $P$-closed precisely when $\Omega = Z(f_\Omega)$. Thus, the condition in Theorem 2.3.7 can be rephrased by restricting $\Omega_1$ and $\Omega_2$ to be $P$-closed.

An important theorem relating $P$-independent sets to $(\sigma, \delta)$-conjugacy classes is the following by Lam [8].
**Theorem 2.3.13.** [8] Let $\Omega_1, \Omega_2 \subset \mathbb{F}$ such that $\Omega_1$ and $\Omega_2$ are $P$-independent, and subsets of two distinct conjugacy classes. Then, $\Omega = \Omega_1 \cup \Omega_2$ is also $P$-independent, where $\cup$ denotes a disjoint union.

**Example 2.3.14.** Consider $\mathbb{F}_{16}$ with primitive element $\gamma$, $\sigma(a) = a^4$, and $\delta = 0$:

- The set $\{1, \gamma^3\}$ is $P$-independent. In fact, any two-element set excluding the element $0$ is $P$-independent.

- The set $\{1, \gamma^3, \gamma^6\}$ is not $P$-independent. In fact, $\{1, \gamma^3\} = C(1)$.

- The set $\{1, \gamma^3, \gamma, \gamma^4\}$ is $P$-independent, as it is the disjoint union of $\{1, \gamma^3\} \in C(1)$ and $\{\gamma, \gamma^4\} \in C(\gamma)$.

Lam [8] also showed that the $P$-independence of a set can be determined by examining the degree of its minimal polynomial.

**Theorem 2.3.15.** [8] Let $\Omega \subset \mathbb{F}$. Then $\Omega$ is $P$-independent if and only if $\deg(f_\Omega) = |\Omega|$.

In the following, we will require the following two useful corollaries.

**Corollary 2.3.16.** Let $\Omega = \{\alpha_1, \alpha_2, \ldots, \alpha_n\}$ be an arbitrary set of $n$ points in $\mathbb{F}$. Then $\deg(f_\Omega) \leq n$.

**Corollary 2.3.17.** Let $\Omega = \{\alpha_1, \alpha_2, \ldots, \alpha_n\} \subset \mathbb{F}$ be such that $\deg(f_\Omega) = n$. Then, for any $S \subset \Omega$, we have $\deg(f_S) = |S|$.

**Proof.** Consider $\Omega_i = \Omega \setminus \{\alpha_i\}$ for each $1 \leq i \leq n$. By Corollary 2.3.16, we know that $\deg(f_{\Omega_i}) < n$. Suppose that $\deg(f_{\Omega_i}) < n - 1$, then the polynomial $g(x) = (x - \alpha_i f_{\Omega_i}(\alpha_i)) f_{\Omega_i}(x)$ vanishes on all of $\Omega$. However, $\deg(g) < \deg(f_\Omega) = n$, contradicting the minimality of $f_\Omega$. Thus, $\deg(f_{\Omega_i}) = n - 1 = |\Omega_i|$ for every $i$. The result follows by recursively applying this argument for smaller subsets of $\Omega$. \qed
2.3.3 Vandermonde Matrices

Let \((\alpha_1,\alpha_2,\ldots,\alpha_n) \in \mathbb{F}^n\). The Vandermonde matrix generated by \((\alpha_1,\ldots,\alpha_n)\) is given by:

\[
V(\alpha_1,\alpha_2,\ldots,\alpha_n) = \begin{pmatrix}
1 & 1 & \cdots & 1 \\
\alpha_1 & \alpha_2 & \cdots & \alpha_n \\
\vdots & \vdots & \ddots & \vdots \\
\alpha_1^{n-1} & \alpha_2^{n-1} & \cdots & \alpha_n^{n-1}
\end{pmatrix}
\]

The Vandermonde matrix has many interesting properties that are useful in coding theory. In particular, \(V(\alpha_1,\alpha_2,\ldots,\alpha_n)\) is invertible if and only if \(\alpha_1,\alpha_2,\ldots,\alpha_n\) are distinct.

In the skew polynomial ring, we have a similar skew-Vandermonde matrix setup [11].

**Definition 2.3.18.** Let \((\alpha_1,\alpha_2,\ldots,\alpha_n) \in \mathbb{F}^n\). The \((\sigma,\delta)\)-Vandermonde matrix is defined as

\[
V_{\sigma,\delta}(\alpha_1,\alpha_2,\ldots,\alpha_n) = \begin{pmatrix}
1 & 1 & \cdots & 1 \\
N_1(\alpha_1) & N_1(\alpha_2) & \cdots & N_1(\alpha_n) \\
\vdots & \vdots & \ddots & \vdots \\
N_{n-1}(\alpha_1) & N_{n-1}(\alpha_2) & \cdots & N_{n-1}(\alpha_n)
\end{pmatrix}
\]

Unlike the regular Vandermonde matrix, the condition for \((\sigma,\delta)\)-Vandermonde matrix \(V_{\sigma,\delta}\) to be invertible is quite different. Its rank is related to the degree of the minimal polynomial of \(\{\alpha_1,\alpha_2,\ldots,\alpha_n\}\) [8]:

**Theorem 2.3.19.** Let \(\Omega = \{\alpha_1,\alpha_2,\ldots,\alpha_n\} \subset \mathbb{F}\). Then \(\text{rank}(V_{\sigma,\delta}(\alpha_1,\alpha_2,\ldots,\alpha_n)) = \deg(f_\Omega)\).

In light of our discussion on \(P\)-independence, we arrive at the following useful corollary.
Corollary 2.3.20. Let $\Omega = \{\alpha_1, \alpha_2, \ldots, \alpha_n\} \subset \mathbb{F}$. $V_{\sigma, \delta}(\alpha_1, \alpha_2, \ldots, \alpha_n)$ is invertible if and only if $\Omega$ is a $P$-independent set.

Proof. Follows from Theorem 2.3.15 and Theorem 2.3.19.

We will extensively use skew-Vandermonde matrices in Chapter 4.

2.4 Summary

In this chapter, we defined skew polynomial rings and discussed some of its basic properties. The evaluation map for skew polynomials is defined through a right division operation, and the result can be computed by applying the $N_i$ functions. Unlike regular polynomials, the evaluation map for skew polynomials behaves differently under a product operation. This motivates the definition of $(\sigma, \delta)$-conjugacy classes. We observed that a skew polynomial of degree $n$ can have more than $n$ roots. This suggests that the minimal polynomial that vanishes on a set of points is an important object to study. We discussed several key properties of minimal polynomials including the important Decomposition Theorem. The theory of minimal polynomials naturally led to the definition of $P$-independence. Finally, a generalization of the Vandermonde matrix called $(\sigma, \delta)$-Vandermonde matrix was introduced. The connections between minimal polynomial, $P$-independence, and rank of $(\sigma, \delta)$-Vandermonde matrix were summarized in Theorem 2.3.19 and Corollary 2.3.20.

In the following chapter, we will focus on the special case of skew polynomial rings without derivation. Some of the theory developed in this chapter will be reformulated using a set of simplified notation. Additional theory that connects the evaluation of skew polynomials to both regular polynomials and linearized polynomials will be explored.
Chapter 3

Skew Polynomials Rings over Finite Fields

In this chapter, we will focus on skew polynomial rings defined over finite fields without derivation \((\delta = 0)\). In this case, we have developed a series of notations that allow for concise definitions and proofs. Most of the proofs presented here are original. It is our intention to present this theory in a way that draws a parallel with the theory of linearized polynomials. The theory and notations developed here will be useful for discussions in both Chapter 4 and Chapter 6.

The rest of this chapter is organized as follows. In Section 3.1, we introduce a set of notation that is useful for the discussion of skew polynomial rings without derivation. The concepts of \((\sigma, \delta)\)-conjugacy classes, skew polynomial evaluation, and \((\sigma, \delta)\)-Vandermonde matrices described in Chapter 2 are reformulated using our new notation. Several original proofs are presented. In Section 3.2, we present a theorem on the minimal polynomial of \(\sigma\)-conjugacy classes. We also develop the important Structure Theorem that describes the relationship between the \(P\)-independence structure of \(\sigma\)-conjugacy classes and vector spaces. Section 3.3 summarizes the chapter.
Chapter 3. Skew Polynomials Rings over Finite Fields

3.1 Skew Polynomial Rings without Derivation

3.1.1 Notation

For this chapter, we will fix a finite field $\mathbb{F}_q$ and consider a finite field extension $\mathbb{F}_{q^m}$ over $\mathbb{F}_q$. Let $\text{Aut}(\mathbb{F}_{q^m})$ be the automorphism group of $\mathbb{F}_{q^m}$. We let $\sigma_s \in \text{Aut}(\mathbb{F}_{q^m})$ be such that $\sigma_s(a) = a^{q^s}$ for all $a \in \mathbb{F}_{q^m}$. Since the maximal subfield fixed by $\sigma_s$ is $\mathbb{F}_q$ if and only if $\gcd(s,m) = 1$, we will henceforth assume $\gcd(s,m) = 1$ whenever we consider $\sigma_s \in \text{Aut}(\mathbb{F}_{q^m})$. Further, we denote the nonzero elements of $\mathbb{F}_{q^m}$ by $\mathbb{F}_q^*$ and we let $\mathbb{N} = \{0,1,2\ldots\}$.

For ease of presentation, for $i \in \mathbb{N}$, define $[i]_s = q^{is} - 1$ and $\lfloor i \rfloor_s = q^{is}$. We can verify that $[i]_s$ and $\lfloor i \rfloor_s$ satisfy the following properties.

**Proposition 3.1.1.** For any $i, j \in \mathbb{N}$ and any $a \in \mathbb{F}_{q^m}$, we have

1. $a^{[0]}_s = a$.

2. $a^{[i]}_s = a^{[j]}_s$ if $i \equiv j \mod m$.

3. $[i]_s[j]_s = [i + j]_s$.

4. $[i]_s + [i]_s = [i + 1]_s$.

5. $[i]_s + [i]_s[j]_s = [i + j]_s$.

When $s = 1$ and there is no ambiguity, we will use the notation $\sigma, [i], [i]$, suppressing the subscript $s$.

3.1.2 Definition and Basic Properties

**Definition 3.1.2.** The skew polynomial ring over $\mathbb{F}_{q^m}$ with automorphism $\sigma_s$, denoted $\mathbb{F}_{q^m}[x; \sigma_s]$, is the ring which consists of polynomials $\sum c_i x^i$, $c_i \in \mathbb{F}_{q^m}$, with the usual addition of polynomials and a multiplication that follows the commuting rule $xa = \sigma_s(a)x$. 
As a special case of the general $F[x;\sigma,\delta]$ skew polynomial ring, $F_q^m[x;\sigma]$ inherits all the properties we discussed in Chapter 2.1. The next proposition shows that for $\sigma_{r_1}, \sigma_{r_2} \in \text{Aut}(F_q^m)$, the skew polynomial rings $F_q^m[x,\sigma_{r_1}]$ and $F_q^m[x,\sigma_{r_2}]$ are not isomorphic. This justifies our notation to specify the particular $\sigma_s$.

**Proposition 3.1.3.** Let $F_q^m$ be a finite field and let $\sigma_{r_1}, \sigma_{r_2} \in \text{Aut}(F_q^m)$. Then, the skew polynomial rings $F_q^m[x;\sigma_{r_1}]$ and $F_q^m[x;\sigma_{r_2}]$ are isomorphic as rings if and only if $\sigma_{r_1} = \sigma_{r_2}$.

**Proof.** Suppose $\Psi : F_q^m[x;\sigma_{r_1}] \rightarrow F_q^m[x;\sigma_{r_2}]$ is a ring isomorphism. Clearly, $\Psi$ restricted to $F_q^m$ is an automorphism of the field, and $\Psi(x) = \alpha x + \beta$, for $\alpha, \beta \in F_q^m$. Thus we have, on one hand, for any $a \in F_q^m$,

$$\Psi(xa) = \Psi(\sigma_{r_1}(a)x) = \Psi(\sigma_{r_1}(a))(\alpha x + \beta) = \Psi(\sigma_{r_1}(a))(\alpha x) + \Psi(\sigma_{r_1}(a))\beta,$$

and on the other,

$$\Psi(xa) = \Psi(x)\Psi(a) = (\alpha x + \beta)\Psi(a) = \sigma_{r_2}(\Psi(a))(\alpha x) + \Psi(a)\beta.$$

Thus we need $\beta = 0$ and

$$\Psi(\sigma_{r_1}(a)) = \sigma_{r_2}(\Psi(a)) \quad \text{for all } a \in F_q^m. \quad (3.1)$$

Since $\Psi$ is an automorphism when restricted to $F_q^m$ and commutes with $\sigma_{r_1}$ and $\sigma_{r_2}$, (3.1) holds if and only if $\sigma_{r_1} = \sigma_{r_2}$. 

3.1.3 $\sigma_s$-Conjugacy Classes

In the special case where $\delta = 0$, the $\sigma_s$-conjugacy relation can be described using the following map.
Definition 3.1.4. For $\sigma_s \in \text{Aut}(\mathbb{F}_{q^m})$, the $\sigma_s$-warping map $\varphi_{\sigma_s}$, is the map

$$\varphi_{\sigma_s}: \mathbb{F}_{q^m}^* \rightarrow \mathbb{F}_{q^m}^*$$

$$a \mapsto \sigma_s(a)a^{-1}.$$ 

When $s = 1$, we write $\varphi$ for $\varphi_{\sigma}$. 

Proposition 3.1.5. For $a, b \in \mathbb{F}_{q^m}^*$, $\varphi_{\sigma_s}(a) = \varphi_{\sigma_s}(b)$ if and only if $a = bc$ for some $c \in \mathbb{F}_q^*$, i.e., if and only if $a$ and $b$ are in the same multiplicative coset of $\mathbb{F}_q^*$ in $\mathbb{F}_{q^m}^*$. 

Proof. First note that the map $\varphi_{\sigma_s}$ is multiplicative. If $c \in \mathbb{F}_q^*$, then $\varphi_{\sigma_s}(c) = 1$. Thus $\varphi_{\sigma_s}(a) = \varphi_{\sigma_s}(bc)$. Conversely, if $\varphi_{\sigma_s}(a) = \varphi_{\sigma_s}(b)$, then $\varphi_{\sigma_s}(\frac{a}{b}) = 1$, showing $\frac{a}{b} \in \mathbb{F}_q^*$. □

Definition 3.1.6. For any two elements $a \in \mathbb{F}_{q^m}, c \in \mathbb{F}_{q^m}^*$, define the $\sigma_s$-conjugation of $a$ by $c$ as:

$$a^c \triangleq a \varphi_{\sigma_s}(c).$$

Definition 3.1.7. We call two elements $a, b \in \mathbb{F}_{q^m}$ $\sigma_s$-conjugates if there exists an element $c \in \mathbb{F}_{q^m}^*$ such that $a^c = b$.

Since $\sigma_s$-conjugacy is an equivalence relation, we denote the set $C_{\sigma_s}(a) = \{a^c | c \in \mathbb{F}_{q^m}^*\}$ the $\sigma_s$-conjugacy class of $a$. When $s = 1$, we write $C(a)$ for $C_{\sigma_s}(a)$.

Corollary 3.1.8. For any $a \in \mathbb{F}_{q^m}^*$, $|C_{\sigma_s}(a)| = [m]$.

Proof. It follows from Proposition 3.1.5 that there are exactly $[m]$ different values of $\varphi_{\sigma_s}(c)$ for $c \in \mathbb{F}_{q^m}^*$. □

Proposition 3.1.9. For any $a \in \mathbb{F}_{q^m}$, we have that $C_{\sigma_s}(a) = C(a)$.

Proof. Every element in $C_{\sigma_s}(a)$ has the form $a\varphi_{\sigma_s}(c)$ for some $c \in \mathbb{F}_{q^m}^*$. Then,

$$a\varphi_{\sigma_s}(c) = ac^s = a(c^{[s]})^{q-1},$$
which is in $C(a)$. Since by Corollary 3.1.8, $C_{\sigma_s}(a)$ and $C(a)$ have the same size, $C_{\sigma_s}(a) = C(a)$.

**Example 3.1.10.** Consider $\mathbb{F}_{16}$, with a primitive element $\gamma$, and $\sigma(a) = a^4$ and fixed field $\mathbb{F}_4$. Then, $C(0) = \{0\}$ is a singleton set, and

$$C(1) = \{\varphi(c) \mid c \in \mathbb{F}_{16}^*\} = \{1, \gamma^3, \gamma^6, \gamma^9, \gamma^{12}\}.$$

Note that $C(1)$ is a subgroup of $\mathbb{F}_{16}^*$, while the other nontrivial classes are cosets of $C(1)$:

$$C(\gamma) = \{\gamma, \gamma^4, \gamma^7, \gamma^{10}, \gamma^{13}\},$$

$$C(\gamma^2) = \{\gamma^2, \gamma^5, \gamma^8, \gamma^{11}, \gamma^{14}\}.$$

In the previous example, we used $1, \gamma, \gamma^2$ as class representatives. In general, as summarized in the next proposition, there are $q-1$ nontrivial (excluding $C(0)$) $\sigma$-conjugacy classes for $\mathbb{F}_{q^m}$; and we can use $\gamma^\ell$ with $0 \leq \ell < q-1$ as the class representatives.

**Proposition 3.1.11.** Let $\gamma$ be a primitive element of $\mathbb{F}_{q^m}$ and $\sigma \in \text{Aut}(\mathbb{F}_{q^m})$. Then, there are $q-1$ nontrivial $\sigma$-conjugacy classes of $\mathbb{F}_{q^m}$, each of size $[m]$. The classes can be enumerated by $C(\gamma^\ell)$ for $0 \leq \ell < q-1$. In particular, $C(1)$ is a subgroup of $\mathbb{F}_{q^m}^*$, and all other $\sigma$-conjugacy classes are cosets of $C(1)$.

**Proof.** The size of each $\sigma$-conjugacy class is determined in Corollary 3.1.8. The fact that $C(1)$ is a subgroup follows from Proposition 3.1.5. Any element in $C(\gamma^\ell)$ has the form $\gamma^\ell \varphi(a)$, where $\varphi(a) \in C(1)$. Since $C(\gamma^\ell)$ and $C(1)$ have the same size, they must be cosets.

### 3.1.4 Skew Polynomial Evaluation

To simplify the discussion of skew polynomials, we will often associate a skew polynomial in $\mathbb{F}_{q^m}[x; \sigma_s]$ with two polynomials in $\mathbb{F}_{q^m}[x]$ as follows.
Definition 3.1.12. Let \( f_s = \sum_i c_i x^i \in \mathbb{F}_{q^m}[x; \sigma_s] \). Define \( f^R_s, f^L_s \in \mathbb{F}_{q^m}[x] \) as

\[
  f^R_s = \sum_i c_i x^{[i]_s}, \\
  f^L_s = \sum_i c_i x^{[i]_s}.
\]

we call \( f^R_s \) and \( f^L_s \) the regular associate and linearized associate of \( f_s \), respectively. Moreover, we call any polynomial of the form \( \sum_i c_i x^{[i]_s} \) an \( s \)-linearized polynomial.

Using the regular associate, we can restate Lam and Leroy’s result on skew polynomial evaluation as follows.

Theorem 3.1.13. [11] For \( f_s = \sum_i c_i x^i \in \mathbb{F}_{q^m}[x; \sigma_s] \) and \( a \in \mathbb{F}_{q^m} \), \( f_s(a) = \sum_i c_i a^{[i]_s} = f^R_s(a) \).

Thus, the evaluation of a skew polynomial is equal to the evaluation of its regular associate.

Corollary 3.1.14. Zeros of \( f_s \in \mathbb{F}_{q^m}[x; \sigma_s] \) are in one-to-one correspondence with zeros of \( f^R_s \in \mathbb{F}_{q^m}[x] \).

As the next theorem shows, the evaluation of skew polynomials is intimately related to the evaluation of linearized polynomials.

Theorem 3.1.15. Let \( f_s = \sum_{i=0}^n c_i x^i \in \mathbb{F}_{q^m}[x; \sigma_s] \) and \( f^L_s = \sum_{i=0}^n c_i x^{[i]_s} \in \mathbb{F}_{q^m}[x] \) be the corresponding linearized associate. Then for any \( a \in \mathbb{F}_{q^m} \),

\[
  af_s(\varphi_{\sigma_s}(a)) = f^L_s(a).
\]
Proof.

\[ a f_s(\varphi_s(a)) = a \left( \sum_{i=0}^{n} c_i (a^{q^s-1})^{[i]_s} \right) \]

\[ = a \left( \sum_{i=0}^{n} c_i (a^{q^s-1})(q^s)^{i-1} \right) \]

\[ = \sum_{i=0}^{n} c_i a^{[i]_s} = f_s^L(a). \]

When \( s = 1 \), the linearized polynomial \( f^L = \sum_{i=0}^{n} c_i x^{[i]} \in \mathbb{F}_{q^m}[x] \) has at most \( q^n \) roots, since as a regular polynomial, it has a degree of at most \( q^n \). The next theorem shows that the \( s \)-linearized polynomial \( f_s^L = \sum_{i=0}^{n} c_i x^{[i]} \in \mathbb{F}_{q^m}[x] \) has the same bound on the number of roots, even though it has a much higher degree when viewed as a regular polynomial.

**Theorem 3.1.16.** An \( s \)-linearized polynomial of degree \( \lceil n \rceil_s \) in \( \mathbb{F}_{q^m}[x] \) has at most \( q^n \) roots.

**Proof.** We proceed by induction on \( n \). For \( n = 0 \), the polynomial \( g_0 = a_0 x \) with \( a_0 \neq 0 \) clearly has only one root at \( x = 0 \). For \( n \geq 1 \), suppose \( g_n \) is an \( s \)-linearized polynomial of degree \( \lceil n \rceil_s \) and \( \alpha \neq 0 \) is a root of \( g_n \). Since \( g_n \) is linearized, for any \( c \in \mathbb{F}_q \), \( ca \) is also a root of \( g_n \). Thus, \( g_n \) is divisible by the \( s \)-linearized polynomial \( h = x^{q^s} - \alpha^{q^s-1}x \). By the theory of symbolic product on linearized polynomials \([28]\), \( g_n \) is also symbolically divisible by the \( s \)-linearized polynomial \( h = x^{q^s} - \alpha^{q^s-1}x \). Thus, we can express \( g_n \) as \( g_n = g_{n-1}(h(x)) \), where \( g_{n-1} \) is an \( s \)-linearized polynomial of degree \( \lceil n - 1 \rceil_s \). By the induction hypothesis, \( g_{n-1} \) has at most \( q^{n-1} \) roots. Now for each root \( \beta \) of \( g_{n-1} \), since \( \gcd(s, m) = 1 \), \( h(x) = \beta \) has at most \( q \) solutions. Thus, \( g_n \) has at most \( q^{n-1}q = q^n \) roots.

### 3.1.5 Moore and \( \sigma_s \)-Vandermonde Matrices

After examining skew polynomial evaluation in relation to both regular polynomial and linearized polynomial evaluation, we now describe three Vandermonde-like matrices using
our notations.

As before, the $n$ by $n$ Vandermonde matrix generated by $(a_1, a_2, \ldots, a_n) \in \mathbb{F}_{q^m}$ is defined as

$$V(a_1, a_2, \ldots, a_n) = \begin{pmatrix} a_0^0 & a_1^0 & \cdots & a_n^0 \\ a_0^1 & a_1^1 & \cdots & a_n^1 \\ \vdots & \vdots & \ddots & \vdots \\ a_0^{n-1} & a_1^{n-1} & \cdots & a_n^{n-1} \end{pmatrix}.$$ 

We know that $V(a_1, a_2, \ldots, a_n)$ is invertible if and only if the $a_i$’s are distinct.

Next we can define the analogue in the linearized polynomial evaluation case.

**Definition 3.1.17.** Let $(a_1, \ldots, a_n) \in \mathbb{F}_{q^m}^n$. The $n$ by $n$ matrix

$$M(a_1, \ldots, a_n) = \begin{pmatrix} a_1^{[0]} & a_2^{[0]} & \cdots & a_n^{[0]} \\ a_1^{[1]} & a_2^{[1]} & \cdots & a_n^{[1]} \\ \vdots & \vdots & \ddots & \vdots \\ a_1^{[n-1]} & a_2^{[n-1]} & \cdots & a_n^{[n-1]} \end{pmatrix}$$

is called a Moore matrix.

With our set of notations, we can generalize this definition to the following.

**Definition 3.1.18.** Let $(a_1, \ldots, a_n) \in \mathbb{F}_{q^m}^n$. Define the $n$ by $n$ matrix

$$M_s(a_1, \ldots, a_n) = \begin{pmatrix} a_1^{[0],s} & a_2^{[0],s} & \cdots & a_n^{[0],s} \\ a_1^{[1],s} & a_2^{[1],s} & \cdots & a_n^{[1],s} \\ \vdots & \vdots & \ddots & \vdots \\ a_1^{[n-1],s} & a_2^{[n-1],s} & \cdots & a_n^{[n-1],s} \end{pmatrix}$$
as the generalized Moore matrix.

We will show in Chapter 4 that $M_s(a_1, a_2, \ldots, a_n)$ is invertible if and only if all $a_i$’s are linearly independent over $\mathbb{F}_q$.

Finally, translating the skew-Vandemonde matrix defined in Chapter 2, we can define the following.

**Definition 3.1.19.** The $n$ by $n$ $\sigma_s$-Vandermonde matrix generated by $(a_1, a_2, \ldots, a_n) \in \mathbb{F}_{q^m}^n$ is defined as

$$V_{\sigma_s}(\alpha_1, \alpha_2, \ldots, \alpha_n) = \begin{pmatrix} a_1[0] & a_2[0] & \cdots & a_n[0] \\ a_1[1] & a_2[1] & \cdots & a_n[1] \\ \vdots & \vdots & \ddots & \vdots \\ a_1[n-1] & a_2[n-1] & \cdots & a_n[n-1] \end{pmatrix}.$$

We know from Chapter 2 that $V_{\sigma_s}(\alpha_1, \alpha_2, \ldots, \alpha_n)$ is invertible if and only if $\{\alpha_1, \alpha_2, \ldots, \alpha_n\}$ is a $P$-independent set.

### 3.2 Structure of $\sigma$-Conjugacy Classes

#### 3.2.1 Minimal Polynomials

As shown in Proposition 3.1.11, the $\sigma_s$-conjugacy classes of $\mathbb{F}_{q^m}$ can be enumerated by $C(1), C(\gamma), \ldots, C(\gamma^{q-2})$, where $C(1)$ forms a subgroup of $\mathbb{F}_{q^m}^*$ and $C(\gamma), \ldots, C(\gamma^{q-2})$ are cosets. We now show that the minimal polynomials of the conjugacy classes follow a very regular structure.

**Theorem 3.2.1 (Minimal Polynomial).** For $0 \leq \ell < q-1$, $f_{C_{\sigma_s}(\gamma^\ell)} = x^m - a_\ell$ is the minimal polynomial of the conjugacy class $C_{\sigma_s}(\gamma^\ell)$, where $a_\ell = (\gamma^\ell)^{[m]}$, and $a_\ell \in \mathbb{F}_q$. 

Proof. An element in $C(\gamma^\ell)$ has the form $\gamma^\ell \varphi_{\sigma_s}(\gamma^k)$. Thus,

$$f_{C_{\sigma_s}(\gamma^\ell)}(\gamma^\ell \varphi_{\sigma_s}(\gamma^k)) = (\gamma^\ell \gamma^k(q^s-1))^{[m]}s - a_\ell$$

$$= (\gamma^\ell)^{[m]}s (\gamma^k)q^{sm-1} - a_\ell$$

$$= (\gamma^\ell)^{[m]}s - a_\ell. $$

Thus, every element of $C(\gamma^\ell)$ is a root of $f_{C(\gamma^\ell)}$ if $a_\ell = (\gamma^\ell)^{[m]}s$. Conversely, by the Independence Lemma (to be proved in next subsection), the largest $P$-independent subset of $C_{\sigma_s}(\gamma^\ell)$ has size $m$. Thus, $f_{C_{\sigma_s}(\gamma^\ell)}$ is indeed minimal.

To check that $a_\ell \in \mathbb{F}_q$, since $\gcd(s, m) = 1$, it suffices to check

$$a_\ell^q - 1 = ((\gamma^\ell)^{q^{sm-1}}q^{s-1})$$

$$= (\gamma^\ell)^{q^{sm}-1}$$

$$= 1. $$

Example 3.2.2. Consider $\mathbb{F}_{16}$, with a primitive element $\gamma$, and $\sigma(a) = a^4$, with fixed field $\mathbb{F}_4$. We have,

$$f_{C_{\sigma}(1)} = x^2 - 1,$$

$$f_{C_{\sigma}(\gamma)} = x^2 - \gamma^5,$$

$$f_{C_{\sigma}(\gamma^2)} = x^2 - \gamma^{10}. $$

Note that $1, \gamma^5, \gamma^{10} \in \mathbb{F}_4$, the field fixed by $\sigma$. 
3.2.2 Structure Theorem

When restricted to a single $\sigma$-conjugacy class, the $P$-independence structure of a set is related to linear independence. We now examine this connection carefully. This connection will be important for our discussions in both Chapter 4 and Chapter 6.

Lemma 3.2.3 (Independence Lemma). Let $\Omega = \{\alpha_1, \ldots, \alpha_n\} \subseteq C_{\sigma_s}(\gamma^\ell) \subseteq \mathbb{F}_{q^m}$, for $0 \leq \ell < q - 1$, and $a_1, \ldots, a_n \in \mathbb{F}_{q^m}$ be such that $\alpha_i = \gamma^\ell \varphi_{\sigma_s}(a_i)$ for $i = 1, \ldots, n$. Then, $\Omega$ is $P$-independent if and only if $a_1, \ldots, a_n$ are linearly independent over $\mathbb{F}_{q^m}$.

Proof. Without loss of generality, we assume $\ell = 0$. Let $\Omega = \{\alpha_1, \ldots, \alpha_n\} \subseteq C_{\sigma_s}(1)$ and $a_1, \ldots, a_n \in \mathbb{F}_{q^m}$ such that $\alpha_i = \varphi_{\sigma_s}(a_i)$ for all $i = 1, \ldots, n$. Let $f_\Omega \in \mathbb{F}_{q^m}[x; \sigma_s]$ be the minimal polynomial of $\Omega$ and $f^L_\Omega \in \mathbb{F}_{q^m}[x]$ be the corresponding $s$-linearized associate. Let $\lambda_1, \ldots, \lambda_n \in \mathbb{F}_q$ and $a = \sum_{i=0}^n \lambda_i a_i$ such that $a \neq 0$. By Theorem 3.1.15,

$$af_\Omega(\varphi_{\sigma_s}(a)) = f^L_\Omega(a) = \sum_{i=0}^n c_i a_i^{[i]} = \sum_{i=0}^n c_i \left( \sum_{j=0}^n \lambda_j a_j \right)^{[i]} = \sum_{j=0}^n \lambda_j \sum_{i=0}^n c_i (a_j)^{[i]} = \sum_{j=0}^n \lambda_j f^L_\Omega(a_j) = 0,$$

where the last equality follows from the fact that for each $j$, $a_j f_\Omega(\alpha_j) = f^L_\Omega(a_j)$ and $f_\Omega(\alpha_j) = 0$. This shows that for every $a \in Z(f^L_\Omega)$, $\varphi_{\sigma_s}(a) \in \overline{\Omega}$.

If $a_1, \ldots, a_n$ are linearly independent, then by Theorem 3.1.16 $\deg(f^L_\Omega) \geq [n]_s$. Thus, $\deg(f_\Omega) \geq n$. By Corollary 2.3.16, $\deg(f_\Omega)$ is at most $n$. Therefore, $\deg(f_\Omega) = n$ and $\Omega$ is $P$-independent by Theorem 2.3.15.

Conversely, assume $\Omega$ is $P$-independent. Without loss of generality, suppose $a_n$ is linearly dependent on $\{a_1, \ldots, a_{n-1}\}$. The above calculation shows that $\alpha_n$ is a root of $f_{\Omega \setminus \{a_n\}}$. This contradicts the $P$-independence assumption. \qed

Corollary 3.2.4. Let $\Omega \subseteq C_{\sigma_s}(1) \subseteq \mathbb{F}_{q^m}$, for $0 \leq \ell < q - 1$, be a $P$-independent set. Then, $\alpha$ is a root of $f_\Omega$ if and only if $\alpha = \varphi_{\sigma_s}(a)$, where $a$ is a root of $f^L_\Omega$. 

Proof. The proof of the Independence Lemma shows that if $a$ is a root of $f_{\Omega}^L$, then $\varphi_{\sigma_s}(a)$ is a root of $f_{\Omega}$. The converse follows from Theorem 2.3.5 and Theorem 3.1.15.

**Corollary 3.2.5.** Let $\Omega = \{\alpha_1, \ldots, \alpha_n\} \subseteq C_{\sigma_s}(\gamma^\ell) \subseteq \mathbb{F}_{q^m}$ be a $P$-independent set, for some $0 \leq \ell < q - 1$. Then $|\Omega| = [n]$.

*Proof.* Without loss of generality, we assume $\ell = 1$. Using the proof of the Independence Lemma, we see that the restriction of the warping map, $\varphi_{\sigma_s} : Z(f_{\Omega}^L) \setminus \{0\} \to Z(f_{\Omega})$ is a $q - 1$ to 1 map. The independence assumption implies that $|Z(f_{\Omega}^L)| = q^n$. Corollary 3.2.4 shows that the restriction of the warping map is onto. Thus $|Z(f_{\Omega})| = \frac{q^n - 1}{q - 1} = [n]$. □

**Remark 3.2.6.** In the case $s = 1$, $f_{\Omega}$ has degree $n$ and its regular associate $f_{\Omega}^R$ has degree $[n]$. This shows that $f_{\Omega}^R$ splits in $\mathbb{F}_{q^m}$. However, when $s \neq 1$, the corresponding $f_{\Omega}^R$ has degree $[n]_s$, but only has $[n]$ roots over $\mathbb{F}_{q^m}$.

**Theorem 3.2.7** (Structure Theorem). Let $\Omega = \{\alpha_1, \ldots, \alpha_n\} \subseteq C_{\sigma_s}(\gamma^\ell) \subseteq \mathbb{F}_{q^m}$, for $0 \leq \ell < q - 1$, and $a_1, \ldots, a_n \in \mathbb{F}_{q^m}$ be such that $\alpha_i = \gamma^\ell \varphi_{\sigma_s}(a_i)$ for $i = 1, \ldots, n$. Then

$$\overline{\Omega} = \{\gamma^\ell \varphi_{\sigma_s}(a) \mid a \in \{a_1, \ldots, a_n\}\} \subseteq C_{\sigma_s}(\gamma^\ell)$$

(3.2)

where $\{a_1, \ldots, a_n\}$ denotes the $\mathbb{F}_q$ subspace of $\mathbb{F}_{q^m}$ generated by $\{a_1, \ldots, a_n\}$, excluding the element $\{0\}$.

*Proof.* In light of the Independence Lemma and Corollary 3.2.4, it suffices to show that, without loss of generality, if $\alpha_1, \ldots, \alpha_k$ is a $P$-basis for $\Omega$, then $\langle a_1, \ldots, a_k \rangle = (a_1, \ldots, a_n)$. Now for any $a \in \langle a_1, \ldots, a_n \rangle$, the calculation in the proof of Independence Lemma shows that $a \in Z(f_{\Omega}^L)$. Since $\alpha_1, \ldots, \alpha_k$ are $P$-independent, we know that $\deg(f_{\Omega}) = k$ and thus $\deg(f_{\Omega}^L) = [k]_s$. Since $a_1, \ldots, a_k$ is linearly independent, we see that $Z(f_{\Omega}^L) = \langle a_1, \ldots, a_k \rangle$. Thus, $\langle a_1, \ldots, a_k \rangle = (a_1, \ldots, a_n)$.

The Structure Theorem can also be derived from the work of Lam and Leroy [11]. Here we presented a direct approach and drew the important connection to linearized
polynomials. The connection to linearized polynomials will be further explored in Chapter 4.

3.3 Summary

In this chapter, we introduced a set of notation that is useful for the discussion of skew polynomial rings without derivation. Using our new notation, we described $\sigma_s$-conjugacy classes, Moore and $\sigma_s$-Vandermonde matrices, as well as the connection between skew polynomial evaluation and evaluation of both regular and linearized polynomials. Focusing on $\sigma_s$-conjugacy classes, we characterized their minimal polynomials. The most important result of the chapter is the Structure Theorem that relates the $P$-independence structure of $\sigma$-conjugacy classes and vector spaces.

In the following chapter, we will use most of the theory developed in this chapter to design a new class of error-correcting codes called Generalized Skew-Evaluation codes.
Chapter 4

Generalized Skew-Evaluation Codes

In this chapter, we present a class of error-correcting codes that are based on the evaluation of skew polynomials called Generalized Skew-Evaluation (GSE) codes [29]. The idea of designing codes using evaluation of skew polynomials first appeared in the work of Boucher and Ulmer [20]. The codes we present here generalize the definition in [20]. Such generalization allows us to develop a duality theory for a subclass of GSE codes that is akin to the duality theory of Generalized Reed-Solomon codes [25]. Using the Structure Theorem developed in Chapter 3, we draw connections between a subclass of GSE codes and the well-known Gabidulin codes. Recognizing that Gabidulin codes are in fact a special case of GSE codes, we construct a new class of codes called pasting MDS construction. This class of codes has interesting distance properties in both the Hamming metric and the rank metric.

The rest of this chapter is organized as follows. Section 4.1 introduces the Generalized Skew-Evaluation (GSE) codes, establishes their connection to Gabidulin codes, and discusses a particular pasting construction that combines both the MDS and MRD properties. Section 4.2 develops a duality theory on a subclass of GSE codes. Section 4.3 summarizes our contributions in the chapter.
4.1 Generalized Skew-Evaluation Codes

In this chapter, we will continue with the notation developed in Chapter 3. We consider an automorphism $\sigma_s$ of $\mathbb{F}_{q^m}$ and the skew polynomial ring $\mathbb{F}_{q^m}[x;\sigma_s]$.

4.1.1 Basic Definition

Definition 4.1.1. Let $\{\alpha_1, \ldots, \alpha_n\} \subset \mathbb{F}_{q^m}$ with $\text{rank}(V_{\sigma_s}(\alpha_1, \ldots, \alpha_n)) \geq k$ and $\{v_1, \ldots, v_n\} \subset \mathbb{F}_{q^m}^*$. The Generalized Skew-Evaluation (GSE) code of length $n$, dimension $k$, with code locators $\{\alpha_1, \ldots, \alpha_n\}$ and column multipliers $\{v_1, \ldots, v_n\}$ is given by

$$C = \{(v_1f(\alpha_1), \ldots, v_nf(\alpha_n)) \mid f \in \mathbb{F}_{q^m}[x;\sigma_s], \deg(f) < k\}.$$ 

If $\text{rank}(V_{\sigma_s}(\alpha_1, \ldots, \alpha_n)) = n$, we call $C$ a Generalized Skew Reed-Solomon (GSRS) code.

Remark 4.1.2. This definition generalizes the remainder evaluation skew codes defined in [20]. In particular, we shall see that the inclusion of column multipliers is very important for our duality theory.

Remark 4.1.3. As shown in [20], as a consequence of Proposition 2.1.8, when considering codes that arise from skew polynomial evaluation, we do not lose any generality to restrict to the skew polynomial rings without derivation. This justifies our restriction to $\mathbb{F}_{q^m}[x;\sigma_s]$ in this chapter, and allows us to employ the notation developed in Chapter 3.

Recall that the generator matrix of an $(n,k)$ Generalized Reed-Solomon code with code locators $\{\alpha_1, \ldots, \alpha_n\}$ and column multipliers $\{v_1, \ldots, v_n\}$ has the form [25]:

$$G = \begin{pmatrix}
\alpha_0^0 & \alpha_0^1 & \cdots & \alpha_0^n \\
\alpha_1^0 & \alpha_1^1 & \cdots & \alpha_1^n \\
\vdots & \vdots & \ddots & \vdots \\
\alpha_1^{k-1} & \alpha_1^{k-1} & \cdots & \alpha_1^{k-1}
\end{pmatrix}
\begin{pmatrix}
v_1 & 0 & \cdots & 0 \\
0 & v_2 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & v_n
\end{pmatrix}.$$
Similarly, the generator matrix of a \((n, k)\) GSE code with code locators \(\{\alpha_1, \ldots, \alpha_n\}\) and column multipliers \(\{v_1, \ldots, v_n\}\) has the form

\[
G = \begin{pmatrix}
\alpha_1^{[0]} & \alpha_2^{[0]} & \cdots & \alpha_n^{[0]} \\
\alpha_1^{[1]} & \alpha_2^{[1]} & \cdots & \alpha_n^{[1]} \\
\vdots & \vdots & \ddots & \vdots \\
\alpha_1^{[k-1]} & \alpha_2^{[k-1]} & \cdots & \alpha_n^{[k-1]}
\end{pmatrix}
\begin{pmatrix}
v_1 & 0 & \cdots & 0 \\
0 & v_2 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & v_n
\end{pmatrix}.
\]

Recall the Singleton bound states that for a linear \((n, k)\) code \(C\), the Hamming distance \(d\) is bounded by \(d \leq n - k + 1\). A code that achieves the Singleton bound is called Maximum Distance Separable (MDS). It is well-known that the GRS codes are MDS. We show below that GSRS codes are also MDS, hence justifying the analogous name to GRS codes.

**Theorem 4.1.4.** GSRS codes are MDS.

*Proof.* Let \(\Omega = \{\alpha_1, \ldots, \alpha_n\}\) be the code locators of a GSRS code. Since \(\text{rank}(V_{\alpha_1^{[0]}, \ldots, \alpha_n^{[0]}}) = n\), we know that \(\Omega\) is a \(P\)-independent set. By Corollary 2.3.17, a degree \(k-1\) polynomial can only vanish on at most a subset of size \(k - 1\) of \(\Omega\). Thus, any codeword must have at least weight \(n - k + 1\).

\[\square\]

### 4.1.2 Rank Metric

The *rank metric* is a distance measure that has been found useful in many engineering applications including network coding [5]. We shall briefly define the rank distance here.

Let \(x = (x_1, \ldots, x_n) \in \mathbb{F}_q^n\). Each \(x_i\) is a length \(m\) vector of \(\mathbb{F}_q\). Then, \(V_x = \text{span}(x_1, \ldots, x_n)\) is a vector space over \(\mathbb{F}_q\). Define \(\text{rank}(x) = \dim(V_x)\).

**Definition 4.1.5.** A metric on \(\mathbb{F}_q^n\) called the rank distance, denoted by \(d_{\text{rank}}\), is given by \(d_{\text{rank}}(x, y) = \text{rank}(x - y)\), for all \(x, y \in \mathbb{F}_q^n\).
Example 4.1.6. Consider \( \mathbb{F}_8 \) over \( \mathbb{F}_2 \) with a primitive element \( \gamma \) such that \( \gamma \) is a root of \( x^3 + x + 1 \), and a basis \( \{1, \gamma, \gamma^2\} \). Let \( x = (1, \gamma, \gamma^3) \), \( y = (\gamma, \gamma^2, \gamma^5) \) \( \in \mathbb{F}_8^3 \). Then we can expand \( x \) and \( y \) as matrices \( M(x) \) and \( M(y) \) over \( \mathbb{F}_2 \).

\[ M(x) = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \end{pmatrix}, \quad M(y) = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix}. \]

Thus, \( \text{rank}(x) = 2 \) and \( \text{rank}(y) = 3 \). Furthermore, we have \( x - y = \gamma + \gamma^2 + \gamma^3 + \gamma^5 \), and

\[ M(x - y) = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}. \]

Thus, \( d_{\text{rank}}(x, y) = 3 \).

Definition 4.1.7. Let \( C \) be a \( (n,k) \) linear code over \( \mathbb{F}_{q^m} \). The minimum rank distance \( d_r \) of \( C \) is the minimum nonzero rank of its codewords. \( C \) is said to be Maximum Rank Distance (MRD) if \( d_r = n - k + 1 \).

4.1.3 Gabidulin Codes

A well-known class of MRD codes are the Gabidulin codes [4]. Just as GSE codes are defined as evaluations of skew polynomials, a Gabidulin code is a similar construction that involves evaluation of linearized polynomials.

Definition 4.1.8. Let \( \{a_1, \ldots, a_n\} \subset \mathbb{F}_{q^m} \) be a linearly independent set over \( \mathbb{F}_q \). Then,
an \((n,k)\) Gabidulin code is a code with a generator matrix of the form:

\[
G_{Gab} = \begin{pmatrix}
    a_1^0 & a_2^0 & \cdots & a_n^0 \\
    a_1^1 & a_2^1 & \cdots & a_n^1 \\
    \vdots & \vdots & \ddots & \vdots \\
    a_1^{k-1} & a_2^{k-1} & \cdots & a_n^{k-1}
\end{pmatrix}.
\]

A generalization of Gabidulin code was later proposed in [30], by considering different automorphisms. Using our notation, and for easy reference, we shall call it the Generalized Gabidulin code, defined as follows.

**Definition 4.1.9.** Let \(\{a_1, \ldots, a_n\} \subset \mathbb{F}_{q^m}\) be a linearly independent set over \(\mathbb{F}_q\). Then, an \((n,k)\) Generalized Gabidulin code is a code with a generator matrix of the form:

\[
G_{Gab} = \begin{pmatrix}
    a_1^{[0]} & a_2^{[0]} & \cdots & a_n^{[0]} \\
    a_1^{[1]} & a_2^{[1]} & \cdots & a_n^{[1]} \\
    \vdots & \vdots & \ddots & \vdots \\
    a_1^{[k-1]} & a_2^{[k-1]} & \cdots & a_n^{[k-1]}
\end{pmatrix}.
\]

It is known that Generalized Gabidulin codes are MRD [30].

### 4.1.4 Gabidulin Codes are GSE Codes

The next theorem shows that with appropriate choice of column multipliers, Gabidulin codes can be constructed as a special case of GSE codes.

**Theorem 4.1.10.** Let \(\{a_1, \ldots, a_n\} \subset \mathbb{F}_{q^m}\) be a linearly independent set over \(\mathbb{F}_q\) and \(G_{Gab}\) be the corresponding generator matrix as defined in Definition 4.1.9. Let \(\alpha_1, \ldots, \alpha_n \in \mathbb{F}_{q^m}\).
be such that $\alpha_i = \varphi_s(a_i)$ for $i = 1, \ldots, n$. Then

$$G_{Gab} = \begin{pmatrix} \alpha_1^{[0]} & \alpha_2^{[0]} & \cdots & \alpha_n^{[0]} \\ \alpha_1^{[1]} & \alpha_2^{[1]} & \cdots & \alpha_n^{[1]} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_1^{[k-1]} & \alpha_2^{[k-1]} & \cdots & \alpha_n^{[k-1]} \end{pmatrix} \begin{pmatrix} a_1 & 0 & \cdots & 0 \\ 0 & a_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a_n \end{pmatrix}.$$}

i.e., every Gabidulin code is a GSE (in fact GSRS) code with an appropriate choice of column multipliers.

**Proof.** We have that $\alpha_i = \varphi_s(a_i) = a_i^{q^s-1}$. Thus for each $i, 1 \leq i \leq n$ and each $j, 0 \leq j \leq k-1$

$$\alpha_i^{[j]}a_i = (a_i^{q^s-1})^{q^s_j-1}a_i = a_i^{q^s_j-1}a_i = a_i^{[j]}.$$ 

Note that in Theorem 4.1.10, $\alpha_i \in C(1)$ for all $i$. If we are in class $\gamma^\ell \neq 1$, we can consider the following generator matrix $G_\ell$ for a GSRS.
\[ G_\ell = \begin{pmatrix} \alpha_1^{[0]} & \alpha_2^{[0]} & \cdots & \alpha_n^{[0]} \\ \alpha_1^{[1]} & \alpha_2^{[1]} & \cdots & \alpha_n^{[1]} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_1^{[k-1]} & \alpha_2^{[k-1]} & \cdots & \alpha_n^{[k-1]} \end{pmatrix} \begin{pmatrix} a_1 & 0 & \cdots & 0 \\ 0 & a_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a_n \end{pmatrix} = \begin{pmatrix} \left( \gamma^\ell \alpha_1 \right)^[0] & 0 & \cdots & 0 \\ 0 & \left( \gamma^\ell \alpha_2 \right)^[1] & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \left( \gamma^\ell \alpha_n \right)^[k-1] \end{pmatrix} \begin{pmatrix} \alpha_1^{[0]} & \alpha_2^{[0]} & \cdots & \alpha_n^{[0]} \\ \alpha_1^{[1]} & \alpha_2^{[1]} & \cdots & \alpha_n^{[1]} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_1^{[k-1]} & \alpha_2^{[k-1]} & \cdots & \alpha_n^{[k-1]} \end{pmatrix} \begin{pmatrix} a_1 & 0 & \cdots & 0 \\ 0 & a_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a_n \end{pmatrix} = G_{G\text{ab}}. \]

Thus, a GSRS code with code locators in \( C(\gamma^\ell) \) with the appropriate column multipliers is a Gabidulin code with a “row-multiplier” matrix. This “row-multiplier” matrix is invertible and does not change the row space, thus the code is the same and the MRD property of \( G_{G\text{ab}} \) is preserved. Alternatively, we can view the “row-multiplier” matrix as just a scrambling of the input symbols. This allows us to conclude the following.

**Theorem 4.1.11.** Let \( \{a_1, a_2, \ldots, a_n\} \in \mathbb{F}_{q^m} \) and let \( \alpha_i = \varphi_{\sigma_s}(a_i) \). Then the GSRS code with code locators \( \gamma^\ell \alpha_1, \gamma^\ell \alpha_2, \ldots, \gamma^\ell \alpha_n \in C(\gamma^\ell) \) and column multipliers \( \{a_1, a_2, \ldots, a_n\} \) is an MRD code.

### 4.1.5 Pasting MDS construction

The above construction of MRD codes using different \( \sigma_s \)-conjugacy classes suggest the following *pasting construction*. 
For each $\ell \in \{0, \ldots, m-2\}$, we can construct a $(n, k)$ GSRS code $C_{\ell}$ by taking $n_{\ell}$ $P$-independent code locators from the class $C(\gamma^\ell)$ and the appropriate column multipliers as stated in Theorem 4.1.10. Each $C_{\ell}$ is both MDS and MRD. Now, let $G_{\ell}$ be a generator matrix for $C_{\ell}$ and define a code $C$ with generator matrix $G$ as follows:

$$G = (G_0 | G_\ell | \ldots | G_{m-2}).$$

The code $C$ is an $(n, k)$ code with $n = \sum_{\ell=0}^{m-2} n_{\ell}$. The component code given by $G_\ell$ is MRD. By Theorem 2.3.13, the set of code locators for $G$ is formed as a disjoint union of $P$-independent code locators from distinct classes and hence is a $P$-independent set. Thus, $G$ is in fact a GSRS code. Therefore, we have constructed a code $G$ in which each component code is MRD while the overall code still remains MDS.

**Example 4.1.12.** Consider $\mathbb{F}_{64}$, with a primitive element $\gamma$, and fix $\sigma(a) = a^4$. Pick $P$-independent sets $\{\alpha_1, \alpha_2, \alpha_3\}, \{\beta_1, \beta_2, \beta_3\} \subset C(1) = C(\gamma^0)$ and let $a_i, b_i$ be such that $\alpha_i = a_i^{q-1}, \beta_i = b_i^{q-1}$ for $1 \leq i \leq 3$. Note that $\{\gamma \beta_1, \gamma \beta_2, \gamma \beta_3\}$ is a $P$-independent set in $C(\gamma)$. Let $C_0$ and $C_1$ be two $(3, 2)$ GSRS codes with generator matrices as follows:

$$G_0 = \begin{pmatrix} a_1^{[0]} & a_2^{[0]} & a_3^{[0]} \\ a_1^{[1]} & a_2^{[1]} & a_3^{[1]} \end{pmatrix} \begin{pmatrix} a_1 & 0 & 0 \\ 0 & a_2 & 0 \\ 0 & 0 & a_3 \end{pmatrix},$$

$$G_1 = \begin{pmatrix} (\gamma \beta_1)^{[0]} & (\gamma \beta_2)^{[0]} & (\gamma \beta_3)^{[0]} \\ (\gamma \beta_1)^{[1]} & (\gamma \beta_2)^{[1]} & (\gamma \beta_3)^{[1]} \end{pmatrix} \begin{pmatrix} b_1 & 0 & 0 \\ 0 & b_2 & 0 \\ 0 & 0 & b_3 \end{pmatrix}.$$
Then, the pasting construction gives a $(6, 2)$ GSRS code with generator matrix

\[
G = (G_\gamma^0 \mid G_\gamma)
\]

\[
= \begin{pmatrix}
\alpha_1^{[0]} & \alpha_2^{[0]} & \alpha_3^{[0]} & (\gamma \beta_1)^{[0]} & (\gamma \beta_2)^{[0]} & (\gamma \beta_3)^{[0]} \\
\alpha_1^{[1]} & \alpha_2^{[1]} & \alpha_3^{[1]} & (\gamma \beta_1)^{[1]} & (\gamma \beta_2)^{[1]} & (\gamma \beta_3)^{[1]}
\end{pmatrix}
\begin{pmatrix}
a_1 & 0 & 0 & 0 & 0 & 0 \\
0 & a_2 & 0 & 0 & 0 & 0 \\
0 & 0 & a_3 & 0 & 0 & 0 \\
0 & 0 & 0 & b_1 & 0 & 0 \\
0 & 0 & 0 & 0 & b_2 & 0 \\
0 & 0 & 0 & 0 & 0 & b_3
\end{pmatrix}
\]

### Chapter 4. Generalized Skew-Evaluation Codes

#### 4.2 Duality Theory

In this section, we shall outline a duality theory that exists for a certain subclass of GSE codes. Our duality theory is akin to the duality theory of Generalized Reed-Solomon codes [25], however, the proof technique is much more involved. The central idea is to find a structured inverse of $\sigma_s$-Vandermonde matrices. We will show, using the relationship between Moore and $\sigma_s$-Vandermonde matrices, we have a nicely structured inverse for a certain class of $\sigma_s$-Vandermonde matrices. This inverse will allow us to construct the duality theory.

##### 4.2.1 $\sigma_s$-Vandermonde Form

We first show that special submatrices of a $\sigma_s$-Vandermonde matrix retains a Vandermonde form if we allow column multipliers.

**Lemma 4.2.1** (Vandermonde Form Lemma). Let $a_1, \ldots, a_n \in \mathbb{F}_{q^m}$, and $V_{\sigma_s}(a_1, \ldots, a_n)$ be the $n$ by $n$ Vandermonde matrix generated by $a_1, \ldots, a_n$. Then, the $k$ by $n$ submatrix
A formed by \( k \) consecutive rows of \( V_\sigma(a_1, \ldots, a_n) \) starting from row \( i + 1 \) has the form:

\[
A = \begin{pmatrix}
\frac{a_i[0]}{a_1} & \frac{a_i[0]}{a_2} & \cdots & \frac{a_i[0]}{a_n} \\
\frac{a_i[1]}{a_1} & \frac{a_i[1]}{a_2} & \cdots & \frac{a_i[1]}{a_n} \\
\vdots & \vdots & \ddots & \vdots \\
\frac{a_i[k-1]}{a_1} & \frac{a_i[k-1]}{a_2} & \cdots & \frac{a_i[k-1]}{a_n}
\end{pmatrix}
\begin{pmatrix}
v_1 & 0 & \cdots & 0 \\
0 & v_2 & \cdots & 0 \\
0 & 0 & \ddots & \vdots \\
0 & 0 & \cdots & v_n
\end{pmatrix},
\]

where \( \overline{a}_j = a_j^{[i]}, v_j = a_j^{[i]} \) for \( 1 \leq j \leq n \).

**Proof.** It suffices to consider any one column of \( A \). Any column of \( A \) has the form \([a_i[0], a_i[i+1], \ldots, a_i[i+k-1]]^\top\) for some \( a \). Note that by Proposition 3.1.1:

\[
a_i[i+j] = a_i[i]_{s+i+j}\]

\[
= a_i[i]_{s}(a_i[j]_{s}).
\]

Thus \([a_i[0], a_i[i+1], \ldots, a_i[i+k-1]]^\top = a_i[i]_{s}[(a_i[i]_{s})[0], (a_i[i]_{s})[1], \ldots, (a_i[i]_{s})[k-1]]^\top\), and \( \overline{a} = a_i[0], v = a_i[i]_{s}. \)

**Example 4.2.2.** Consider \( \mathbb{F}_{16} \) with primitive element \( \gamma \) and \( \sigma(a) = a^4 \). Then,

\[
V_\sigma(\gamma, \gamma^2, \gamma^3, \gamma^4) = \begin{pmatrix}
\gamma^{[0]} & (\gamma^2)^{[0]} & (\gamma^3)^{[0]} & (\gamma^4)^{[0]} \\
\gamma^{[1]} & (\gamma^2)^{[1]} & (\gamma^3)^{[1]} & (\gamma^4)^{[1]} \\
\gamma^{[2]} & (\gamma^2)^{[2]} & (\gamma^3)^{[2]} & (\gamma^4)^{[2]} \\
\gamma^{[3]} & (\gamma^2)^{[3]} & (\gamma^3)^{[3]} & (\gamma^4)^{[3]}
\end{pmatrix} = \begin{pmatrix}
1 & 1 & 1 & 1 \\
\gamma & \gamma^2 & \gamma^3 & \gamma^4 \\
\gamma^5 & \gamma^{10} & \gamma^{15} & \gamma^5 \\
\gamma^6 & \gamma^{12} & \gamma^3 & \gamma^9
\end{pmatrix}
\]
The 3 by 2 submatrix formed by the last 3 rows and first 2 columns has the form:

\[
\begin{pmatrix}
\gamma & \gamma^2 \\
\gamma^5 & \gamma^{10} \\
\gamma^6 & \gamma^{12}
\end{pmatrix}
= \begin{pmatrix} 1 & 1 \\ \gamma^4 & \gamma^8 \\ (\gamma^4)^2 & (\gamma^8)^2 \end{pmatrix}
\begin{pmatrix} \gamma & 0 \\ 0 & \gamma^2 \end{pmatrix}
= \begin{pmatrix} 1 & 1 \\ \gamma^4 & \gamma^8 \\ (\gamma^4)^2 & (\gamma^8)^2 \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ \gamma & \gamma^4 \\ \gamma^2 & \gamma^8 \end{pmatrix}
\]

4.2.2 Generalized Moore Matrices

Recall, for \(a_1, \ldots, a_k \in \mathbb{F}_{q^m}\), we defined the Generalized Moore Matrix as the following.

\[
M_s(a_1, \ldots, a_k) = \begin{pmatrix}
a_1^{[0]} & a_2^{[0]} & \cdots & a_k^{[0]} \\
a_1^{[1]} & a_2^{[1]} & \cdots & a_k^{[1]} \\
a_1^{[2]} & a_2^{[2]} & \cdots & a_k^{[2]} \\
\vdots & \vdots & \ddots & \vdots \\
a_1^{[k-1]} & a_2^{[k-1]} & \cdots & a_k^{[k-1]}
\end{pmatrix}
\]

When \(s = 1\), the regular Moore matrix has a well-known determinant formula.

**Proposition 4.2.3.** [31] Let

\[
M = M(a_1, \ldots, a_k) = \begin{pmatrix}
a_1^{[0]} & a_2^{[0]} & \cdots & a_k^{[0]} \\
a_1^{[1]} & a_2^{[1]} & \cdots & a_k^{[1]} \\
a_1^{[2]} & a_2^{[2]} & \cdots & a_k^{[2]} \\
\vdots & \vdots & \ddots & \vdots \\
a_1^{[k-1]} & a_2^{[k-1]} & \cdots & a_k^{[k-1]}
\end{pmatrix}
\]
be the $k \times k$ Moore matrix over $\mathbb{F}_{q^m}$. Then

$$\det(M) = \prod_{c} (c_1a_1 + \ldots + c_ka_k),$$

where $c = (c_1, \ldots, c_k)$ runs over all possible direction vectors in $\mathbb{F}_{q^m}$. Equivalently,

$$\det(M) = \prod_{1 \leq i \leq n} \prod_{c_1, \ldots, c_{i-1}} (c_1a_1 + \ldots + c_{i-1}a_{i-1} + a_i).$$

From this, it is easy to see that a Moore matrix is invertible if and only if $a_1, a_2, \ldots, a_k$ are linearly independent over $\mathbb{F}_{q^m}$.

For the generalized Moore matrix, the same conclusion about independence can be drawn by directly using Theorem 3.1.16. Furthermore, note that for $a, b \in \mathbb{F}_{q^m}$, we have $(a + b)^q = a^q + b^q$. This allows us to state the following useful Lemma.

**Lemma 4.2.4.** Let $a_1, \ldots, a_k \in \mathbb{F}_{q^m}$. Let $M = M_s(a_1, \ldots, a_k)$ and $N = M_s(a_1^q, \ldots, a_k^q)$ be two $k \times k$ generalized Moore matrices. Then $\det(N) = \det(M)^q$.

Finally, we arrive the main theorem of the section. If we choose the maximal number of points $\mathbb{F}_{q^m}$ that are linearly independent over $\mathbb{F}_q$, then the generalized Moore matrix has a well-structured inverse.

**Theorem 4.2.5** (Moore matrix inverse). Let $c_1, \ldots, c_m \in \mathbb{F}_{q^m}^*$ and linearly independent over $\mathbb{F}_q$. Let

$$M = \begin{pmatrix}
   c_1^{[0]} & c_2^{[0]} & \ldots & c_m^{[0]} \\
   c_1^{[1]} & c_2^{[1]} & \ldots & c_m^{[1]} \\
   c_1^{[2]} & c_2^{[2]} & \ldots & c_m^{[2]} \\
   \vdots & \vdots & \ddots & \vdots \\
   c_1^{[m-1]} & c_2^{[m-1]} & \ldots & c_m^{[m-1]}
\end{pmatrix}$$

be the corresponding $m \times m$ generalized Moore matrix. Then the inverse matrix $M^{-1}$
has the form

\[(M^{-1})^\top = \frac{1}{\det M} \begin{pmatrix}
1 & 0 & 0 & \ldots & 0 \\
0 & (-1)^{m-1} & 0 & \ldots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & 0 & \ldots & (-1)^{(m-1)^2}
\end{pmatrix}
\begin{pmatrix}
\bar{c}_1^0 & \bar{c}_2^0 & \ldots & \bar{c}_m^0 \\
\bar{c}_1^1 & \bar{c}_2^1 & \ldots & \bar{c}_m^1 \\
\bar{c}_1^2 & \bar{c}_2^2 & \ldots & \bar{c}_m^2 \\
\ddots & \ddots & \ddots & \ddots \\
\bar{c}_1^{m-1} & \bar{c}_2^{m-1} & \ldots & \bar{c}_m^{m-1}
\end{pmatrix}\]

for some \(\bar{c}_1, \bar{c}_2, \ldots, \bar{c}_m \in \mathbb{F}_{q^m}\).

**Proof.** Let \(D\) be the matrix of cofactors of \(M\), i.e., \(D = (d_{ij})\) with \(d_{ij} = (-1)^{i+j}M_{ij}\), where \(M_{ij}\) denote the \(ij\)th minor of \(M\). Then, \((M^{-1})^\top = \frac{1}{\det(M)} D\). Consider the \(j\)th column of \(D\) given by \([d_{1j}, d_{2j}, \ldots, d_{mj}]^\top\). Note that

\[d_{ij} = (-1)^{i+j}M_{ij} = (-1)^{i+j}c_j^\top \begin{vmatrix}
\bar{c}_1^0 & \bar{c}_2^0 & \ldots & \bar{c}_m^0 \\
\bar{c}_1^1 & \bar{c}_2^1 & \ldots & \bar{c}_m^1 \\
\bar{c}_1^2 & \bar{c}_2^2 & \ldots & \bar{c}_m^2 \\
\ddots & \ddots & \ddots & \ddots \\
\bar{c}_1^{m-1} & \bar{c}_2^{m-1} & \ldots & \bar{c}_m^{m-1}
\end{vmatrix},\]

where the quantities of the form \(\bar{c}_j\) denote that \(c_j\) is removed.

Note that for any element \(c \in \mathbb{F}_{q^m}\), we have \(c^{[m]} = c\). Let \(\overline{M}_{ij}\) denote the submatrix of \(M\) corresponding to the minor \(M_{ij}\), then \(\overline{M}_{ij}\) has a “cyclic” structure. In particular,
by interchanging rows of \( \overline{M}_{ij} \), we can obtain a matrix:

\[
\begin{pmatrix}
c_1^{[i]s} & c_2^{[i]s} & \cdots & c_j^{[i]s} & \cdots & c_m^{[i]s} \\
\vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\
c_1^{[m-1]s} & c_2^{[m-1]s} & \cdots & c_j^{[m-1]s} & \cdots & c_m^{[m-1]s} \\
c_1^{[0]s} & c_2^{[0]s} & \cdots & c_j^{[0]s} & \cdots & c_m^{[0]s} \\
\vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\
c_1^{[i-2]s} & c_2^{[i-2]s} & \cdots & c_j^{[i-2]s} & \cdots & c_m^{[i-2]s}
\end{pmatrix} = (4.2)
\]

\[
\begin{pmatrix}
(c_1^{[1]s})^{[0]s} & (c_2^{[1]s})^{[0]s} & \cdots & (c_j^{[1]s})^{[0]s} & \cdots & (c_m^{[1]s})^{[0]s} \\
(c_1^{[1]s})^{[1]s} & (c_2^{[1]s})^{[1]s} & \cdots & (c_j^{[1]s})^{[1]s} & \cdots & (c_m^{[1]s})^{[1]s} \\
\vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\
(c_1^{[1]s})^{[m-2]s} & (c_2^{[1]s})^{[m-2]s} & \cdots & (c_j^{[1]s})^{[m-2]s} & \cdots & (c_m^{[1]s})^{[m-2]s}
\end{pmatrix}, (4.3)
\]

where we used the properties of \([i]s\) from Proposition 3.1.1. A total of \((i-1)(m-2)\) row exchanges are needed to obtain the matrix in (4.2) from \( \overline{M}_{ij} \). Thus, by Lemma 4.2.4

\[ M_{ij} = (-1)^{(i-1)(m-2)} M_{1j}^{[i-1]s}. \]

Since \( d_{1j} = (-1)^{i+j} M_{1j} \), we get

\[ d_{1j}^{[i-1]s} = (-1)^{(1+j)[i-1]} M_{1j}^{[i-1]s}. \]

Thus,

\[ d_{ij} = (-1)^{i+j} M_{ij} \]

\[ = (-1)^{i+j} (-1)^{(i-1)(m-2)} M_{1j}^{[i-1]s} \]

\[ = (-1)^{i+j} (-1)^{(i-1)(m-2)} (-1)^{-(1+j)[i-1]} d_{1j}^{[i-1]s}. \]
If we are in characteristic 2, then clearly \( d_{ij} = d_{ij}^{[i-1]s} \). Otherwise, \( q \) is odd, so \((-1)^{(1+j)[i-1]s} = (-1)^{(1+j)}\). So,

\[
d_{ij} = (-1)^{i+j}(-1)^{(i-1)(m-2)}(-1)^{(1+j)}d_{ij}^{[i-1]s} \\
= (-1)^{i-1}(-1)^{(i-1)(m-2)}d_{ij}^{[i-1]s} \\
= (-1)^{(i-1)(m-1)}d_{ij}^{[i-1]s}.
\]

Thus, the theorem follows by setting \( \bar{c}_j = d_{1j} \). \( \Box \)

In appropriate field characteristics, the Moore inverse has the following simplification.

**Corollary 4.2.6.** If the characteristic of \( \mathbb{F}_{q^m} \) is 2 or \( m \) is odd, then

\[
(M^{-1})^\top = \frac{1}{\det M} \begin{pmatrix}
\begin{array}{cccc}
\bar{c}_1^{[0]s} & \bar{c}_2^{[0]s} & \cdots & \bar{c}_m^{[0]s} \\
\bar{c}_1^{[1]s} & \bar{c}_2^{[1]s} & \cdots & \bar{c}_m^{[1]s} \\
\vdots & \vdots & \ddots & \vdots \\
\bar{c}_1^{[m-1]s} & \bar{c}_2^{[m-1]s} & \cdots & \bar{c}_m^{[m-1]s}
\end{array}
\end{pmatrix}
\]

For simplicity of notation, we shall denote

\[
D_M = \frac{1}{\det M} \begin{pmatrix}
1 & 0 & 0 & \cdots & 0 \\
0 & (-1)^{m-1} & 0 & \cdots & 0 \\
0 & 0 & (-1)^{2(m-1)} & \cdots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & 0 & \cdots & (-1)^{(m-1)^2}
\end{pmatrix},
\]
so that

\[
(M^{-1})^T = D_M \begin{pmatrix}
\tilde{c}_1^{[0]} & \tilde{c}_2^{[0]} & \ldots & \tilde{c}_m^{[0]} \\
\tilde{c}_1^{[1]} & \tilde{c}_2^{[1]} & \ldots & \tilde{c}_m^{[1]} \\
\tilde{c}_1^{[2]} & \tilde{c}_2^{[2]} & \ldots & \tilde{c}_m^{[2]} \\
\vdots & \vdots & \ddots & \vdots \\
\tilde{c}_1^{[m-1]} & \tilde{c}_2^{[m-1]} & \ldots & \tilde{c}_m^{[m-1]}
\end{pmatrix}
\]

### 4.2.3 Inverse $\sigma_s$-Vandermonde Matrix

We shall use the special inverse we found for the Generalized Moore matrix to describe the inverse of a class of $\sigma_s$-Vandermonde matrices. Note that the special inverse we found in the previous subsection required $c_1, \ldots, c_m \in \mathbb{F}_{q^m}^*$ to be linearly independent over $\mathbb{F}_q$. This is equivalent to requiring $c_1, \ldots, c_m \in \mathbb{F}_{q^m}^*$ to form a basis for $\mathbb{F}_{q^m}$ over $\mathbb{F}_q$. Translating this into the framework of $P$-independence using the Structure Theorem, we have the following theorem.

**Theorem 4.2.7** (Inverse $\sigma_s$-Vandermonde Matrix). Let $\gamma$ be a primitive element of $\mathbb{F}_{q^m}$, $a_1, \ldots, a_m \in \mathbb{F}_{q^m}$ be a $P$-basis for $C(\gamma^\ell)$, and let $V = V_{\sigma_s}(a_1, \ldots, a_m)$ be the corresponding $\sigma_s$-Vandermonde matrix. Then

\[
(V^{-1})^T = D_M V_{\sigma_s}(b_1, \ldots, b_m) diag(v_1, \ldots, v_m)
\]

\[
= D_M \begin{pmatrix}
b_1^{[0]} & b_2^{[0]} & \ldots & b_m^{[0]} \\
b_1^{[1]} & b_2^{[1]} & \ldots & b_m^{[1]} \\
\vdots & \vdots & \ddots & \vdots \\
b_1^{[m-1]} & b_2^{[m-1]} & \ldots & b_m^{[m-1]}
\end{pmatrix} \begin{pmatrix}v_1 & 0 & \ldots & 0 \\
0 & v_2 & \ldots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \ldots & v_m
\end{pmatrix},
\]

where $b_1, \ldots, b_m$ forms a $P$-basis for $C(\gamma^{-\ell})$, and $v_1, \ldots, v_m \in \mathbb{F}_{q^m}^*$.

**Proof.** We can write each $a_i = \gamma^{\ell \varphi_{\sigma_s}(c_i)}$ for some $c_i \in \mathbb{F}_{q^m}^*$. Since $\varphi_{\sigma_s}(c_i)[k] = c_i^{[k]} = 1$,
Now, since \( A \) is a Moore matrix, by Theorem 4.2.5, we have

\[
V = \begin{pmatrix}
  a_1^{[0]} & a_2^{[0]} & \cdots & a_m^{[0]} \\
  a_1^{[1]} & a_2^{[1]} & \cdots & a_m^{[1]} \\
  \vdots & \vdots & \ddots & \vdots \\
  a_1^{[m-1]} & a_2^{[m-1]} & \cdots & a_m^{[m-1]}
\end{pmatrix}
\]

\[
= \begin{pmatrix}
  (\gamma^\ell)^{[0]} & 0 & \cdots & 0 \\
  0 & (\gamma^\ell)^{[1]} & \cdots & 0 \\
  \vdots & \vdots & \ddots & \vdots \\
  0 & 0 & \cdots & (\gamma^\ell)^{[m-1]}
\end{pmatrix}
\begin{pmatrix}
  c_1^{[0]} & c_2^{[0]} & \cdots & c_m^{[0]} \\
  c_1^{[1]} & c_2^{[1]} & \cdots & c_m^{[1]} \\
  \vdots & \vdots & \ddots & \vdots \\
  c_1^{[m-1]} & c_2^{[m-1]} & \cdots & c_m^{[m-1]}
\end{pmatrix}
\begin{pmatrix}
  c_1^{-1} & 0 & \cdots & 0
\end{pmatrix}
\]

= AMC.

Now, since \( A \) and \( C \) are diagonal matrices, \(((AMC)^{-1})^\top = A^{-1}(M^{-1})^\top C^{-1}\). As \( M \) is a Moore matrix, by Theorem 4.2.5, we have

\[
((AMC)^{-1})^\top = A^{-1}(M^{-1})^\top C^{-1}
\]

\[
= \begin{pmatrix}
  (\gamma^{-\ell})^{[0]} & 0 & \cdots & 0 \\
  0 & (\gamma^{-\ell})^{[1]} & \cdots & 0 \\
  \vdots & \vdots & \ddots & \vdots \\
  0 & 0 & \cdots & (\gamma^{-\ell})^{[m-1]}
\end{pmatrix}
\begin{pmatrix}
  \bar{c}_1^{[0]} & \bar{c}_2^{[0]} & \cdots & \bar{c}_m^{[0]} \\
  \bar{c}_1^{[1]} & \bar{c}_2^{[1]} & \cdots & \bar{c}_m^{[1]} \\
  \vdots & \vdots & \ddots & \vdots \\
  \bar{c}_1^{[m-1]} & \bar{c}_2^{[m-1]} & \cdots & \bar{c}_m^{[m-1]}
\end{pmatrix}
\begin{pmatrix}
  c_1 & 0 & \cdots & 0
\end{pmatrix}
\]

\[
= \begin{pmatrix}
  b_1^{[0]} & b_2^{[0]} & \cdots & b_m^{[0]} \\
  b_1^{[1]} & b_2^{[1]} & \cdots & b_m^{[1]} \\
  \vdots & \vdots & \ddots & \vdots \\
  b_1^{[m-1]} & b_2^{[m-1]} & \cdots & b_m^{[m-1]}
\end{pmatrix}
\begin{pmatrix}
  v_1 & 0 & \cdots & 0
\end{pmatrix}
\]

\[
= D_M V_\sigma(b_1, \ldots, b_n) \text{diag}(v_1, \ldots, v_n),
\]

where \( b_i = \gamma^{-\ell} \varphi_{\sigma_i}(\bar{c}_i) \) and \( u_i = c_i / \bar{c}_i^{-1} \). Lastly, \( b_1, \ldots, b_m \) form a \( P \)-basis for \( C(\gamma^{-\ell}) \) since
Example 4.2.8. Consider $\mathbb{F}_{256}$ with primitive element $\gamma$ and $\sigma(a) = a^4$.

$$V_\sigma(\gamma, \gamma^4, \gamma^7, \gamma^{10}) = \begin{pmatrix} \gamma^0 & (\gamma^4)^0 & (\gamma^7)^0 & (\gamma^{10})^0 \\ \gamma^1 & (\gamma^4)^1 & (\gamma^7)^1 & (\gamma^{10})^1 \\ \gamma^2 & (\gamma^4)^2 & (\gamma^7)^2 & (\gamma^{10})^2 \\ \gamma^3 & (\gamma^4)^3 & (\gamma^7)^3 & (\gamma^{10})^3 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ \gamma & \gamma^4 & \gamma^7 & \gamma^{10} \\ \gamma^5 & \gamma^{20} & \gamma^{35} & \gamma^{50} \\ \gamma^{21} & \gamma^{84} & \gamma^{147} & \gamma^{210} \end{pmatrix}$$

$$(V_\sigma^{-1})^\top = \begin{pmatrix} \gamma^{25} & \gamma^{50} & \gamma^{223} & \gamma^{199} \\ \gamma^{20} & \gamma^{196} & \gamma^{120} & \gamma^{21} \\ \gamma^{140} & \gamma^{15} & \gamma^{218} & \gamma^{74} \\ \gamma^{49} & \gamma^{56} & \gamma^{100} & \gamma^{31} \end{pmatrix} \begin{pmatrix} 1 & 1 & 1 & 1 \\ \gamma^{74} & \gamma^{146} & \gamma^{152} & \gamma^{77} \\ (\gamma^{74})^2 & (\gamma^{146})^2 & (\gamma^{152})^2 & (\gamma^{77})^2 \\ (\gamma^{74})^3 & (\gamma^{146})^3 & (\gamma^{152})^3 & (\gamma^{77})^3 \end{pmatrix} = \begin{pmatrix} \gamma^{25} & 0 & 0 & 0 \\ 0 & \gamma^{50} & 0 & 0 \\ 0 & 0 & \gamma^{223} & 0 \\ 0 & 0 & 0 & \gamma^{199} \end{pmatrix}$$

Note that $\gamma, \gamma^4, \gamma^7, \gamma^{10} \in C(\gamma)$ and $\gamma^{74}, \gamma^{146}, \gamma^{152}, \gamma^{77} \in C(\gamma^2) = C(\gamma^{-1})$.

4.2.4 Duality Theory of GSRS Codes

Lemma 4.2.9. Let $G$ be a full-rank $k$ by $n$ matrix with $k < n$. Let $\overline{G}$ be an invertible $n$ by $n$ matrix formed by appending rows to $G$ from below. Let $H^\top$ denote the last $n - k$ columns of $\overline{G}^{-1}$. Then $GH^\top = 0$, where $0$ denotes a zero matrix of appropriate size.

Proof. We have that $\overline{G}G^{-1} = I_{n,n}$, where $I_{n,n}$ is the $n \times n$ identities matrix. This shows that the inner product of any of the first $k$ rows of $\overline{G}$ with any of the last $n - k$ columns of $\overline{G}^{-1}$ must be zero. Consider all such inner products, we have that $GH^\top = 0$. \qed
Theorem 4.2.10 (GSRS Duality). Let $\gamma$ be a primitive element of $\mathbb{F}_{q^m}$, and $a_1, \ldots, a_m \in \mathbb{F}_{q^m}$ be a $P$-basis for $C(\gamma^f)$. Let $C$ be a $(m, k)$ GSRS code with code locators $a_1, \ldots, a_m$, with column multipliers $w_1, \ldots, w_m$, $w_i \in \mathbb{F}_{q^m}$ for $1 \leq i \leq m$. Then the dual code $C^\perp$ is a $(m, m - k)$ GSRS code with code locators $\bar{b}_1, \ldots, \bar{b}_m$, where $\bar{b}_1, \ldots, \bar{b}_m$ forms a $P$-basis for $C(\gamma^{-f})$.

Proof. It suffices to prove the case where $w_i = 1$ for $1 \leq i \leq m$. Let

$$G = \begin{pmatrix}
\begin{array}{cccc}
a_1^{[0]} & a_2^{[0]} & \cdots & a_m^{[0]} \\
a_1^{[1]} & a_2^{[1]} & \cdots & a_m^{[1]} \\
\vdots & \vdots & \ddots & \vdots \\
a_1^{[k-1]} & a_2^{[k-1]} & \cdots & a_m^{[k-1]} \\
\end{array}
\end{pmatrix}$$

be a generator matrix for $C$. We can complete $G$ to the $m$ by $m$ Vandermonde matrix $V_m(a_1, \ldots, a_m)$. From Theorem 4.2.7,

$$(V^{-1})^\top = D_M \begin{pmatrix}
\begin{array}{cccc}
b_1^{[0]} & b_2^{[0]} & \cdots & b_m^{[0]} \\
b_1^{[1]} & b_2^{[1]} & \cdots & b_m^{[1]} \\
\vdots & \vdots & \ddots & \vdots \\
b_1^{[m-1]} & b_2^{[m-1]} & \cdots & b_m^{[m-1]} \\
\end{array}
\end{pmatrix} \begin{pmatrix}
v_1 \\
v_2 \\
\vdots \\
v_m \\
\end{pmatrix} = \begin{pmatrix}
v_1 \\
v_2 \\
\vdots \\
v_m \\
\end{pmatrix}.$$

Thus, by Lemma 4.2.9, $H$ can be take the last $m - k$ rows of $(V^{-1})^\top$.

$$H = \frac{1}{\det M} \begin{pmatrix}
(-1)^{k(m-1)} & 0 & \cdots & 0 \\
0 & (-1)^{(k+1)(m-1)} & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & (-1)^{(m-1)^2} \\
\end{pmatrix} \begin{pmatrix}
b_1^{[k]} & b_2^{[k]} & \cdots & b_m^{[k]} \\
b_1^{[k+1]} & b_2^{[k+1]} & \cdots & b_m^{[k+1]} \\
\vdots & \vdots & \ddots & \vdots \\
b_1^{[m-1]} & b_2^{[m-1]} & \cdots & b_m^{[m-1]} \\
\end{pmatrix} \begin{pmatrix}
v_1 \\
v_2 \\
\vdots \\
v_m \\
\end{pmatrix}.$$
Now, by Lemma 4.2.1,

\[
\begin{pmatrix}
    b_1^{[k]} & b_2^{[k]} & \cdots & b_m^{[k]} \\
    b_1^{[k+1]} & b_2^{[k+1]} & \cdots & b_m^{[k+1]} \\
    \vdots & \vdots & \ddots & \vdots \\
    b_1^{[m-1]} & b_2^{[m-1]} & \cdots & b_m^{[m-1]}
\end{pmatrix}
= \begin{pmatrix}
    -b_1^{[0]} & -b_2^{[0]} & \cdots & -b_m^{[0]} \\
    -b_1^{[1]} & -b_2^{[1]} & \cdots & -b_m^{[1]} \\
    \vdots & \vdots & \ddots & \vdots \\
    -b_1^{[m-k+1]} & -b_2^{[m-k+1]} & \cdots & -b_m^{[m-k+1]}
\end{pmatrix}
\begin{pmatrix}
    b_1^{[k]} & 0 & \cdots & 0 \\
    0 & b_2^{[k]} & \cdots & 0 \\
    \vdots & \vdots & \ddots & \vdots \\
    0 & 0 & \cdots & b_m^{[k]}
\end{pmatrix}
\]

where \( \overline{b}_j = b_j^{[k]} \). Thus,

\[ H = \frac{1}{\det M} \begin{pmatrix}
    0 & (-1)^{k(m-1)} & 0 & \cdots & 0 \\
    0 & 0 & (-1)^{(k+1)(m-1)} & \cdots & 0 \\
    \vdots & \vdots & \vdots & \ddots & \vdots \\
    0 & 0 & 0 & \cdots & (-1)^{(m-1)^2}
\end{pmatrix}
\times
\begin{pmatrix}
    -b_1^{[0]} & -b_2^{[0]} & \cdots & -b_m^{[0]} \\
    -b_1^{[1]} & -b_2^{[1]} & \cdots & -b_m^{[1]} \\
    \vdots & \vdots & \ddots & \vdots \\
    -b_1^{[m-k+1]} & -b_2^{[m-k+1]} & \cdots & -b_m^{[m-k+1]}
\end{pmatrix}
\begin{pmatrix}
    v_1 b_1^{[k]} & 0 & \cdots & 0 \\
    0 & v_2 b_2^{[k]} & \cdots & 0 \\
    \vdots & \vdots & \ddots & \vdots \\
    0 & 0 & \cdots & v_m b_m^{[k]}
\end{pmatrix}
\]

The left factor of

\[
\frac{1}{\det M} \begin{pmatrix}
    0 & (-1)^{k(m-1)} & 0 & \cdots & 0 \\
    0 & 0 & (-1)^{(k+1)(m-1)} & \cdots & 0 \\
    \vdots & \vdots & \vdots & \ddots & \vdots \\
    0 & 0 & 0 & \cdots & (-1)^{(m-1)^2}
\end{pmatrix}
\]
does not change the kernel of $H$. Let

$$
\overline{H} = \begin{pmatrix}
\overline{b}_1^{[0]} & \overline{b}_2^{[0]} & \ldots & \overline{b}_m^{[0]} \\
\overline{b}_1^{[1]} & \overline{b}_2^{[1]} & \ldots & \overline{b}_m^{[1]} \\
\vdots & \vdots & \ddots & \vdots \\
\overline{b}_1^{[m-k+1]} & \overline{b}_2^{[m-k+1]} & \ldots & \overline{b}_m^{[m-k+1]} \\
\end{pmatrix}
\begin{pmatrix}
v_1 b_1^{[k]} \\
v_1 b_2^{[k]} \\
\vdots \\
v_1 b_m^{[k]} \\
0 \\
v_2 b_2^{[k]} \\
\vdots \\
v_2 b_m^{[k]} \\
0 \\
\vdots \\
v_m b_m^{[k]} \\
\end{pmatrix}
$$

Since $H$ and $\overline{H}$ has the same kernel, $\overline{H}$ is a generator matrix for $C^\perp$. Clearly, $\overline{H}$ is a generator matrix for a GSRS code with code locators $\overline{b}_1, \ldots, \overline{b}_m$. By Theorem 4.2.7, $\overline{b}_1, \ldots, \overline{b}_m$ form a basis for $C(\gamma^\ell)$.

Remark 4.2.11. Our duality theory is very similar to the duality theory of Generalized Reed-Solomon codes. In both cases, column multipliers are essential to construct the generator for the dual code. Our construction requires the code locators to be from a single $\sigma_s$-conjugacy class, and the number of code locators have to be the maximal $P$-independent subset of a $\sigma_s$-conjugacy class. This requirement allows us to take advantage of the Moore matrix inverse developed earlier. An interesting consequence of our result is that each class can be naturally associated to a “dual class”. It is ongoing research to extend this duality theory to more general cases.

Example 4.2.12. Fix $\mathbb{F}_{256}$ with primitive element $\gamma$ and $\sigma(a) = a^4$. Consider a $(4, 2)$-GSRS code with generator matrix

$$
G = \begin{pmatrix}
\gamma^{[0]} & (\gamma^4)^{[0]} & (\gamma^7)^{[0]} & (\gamma^{10})^{[0]} \\
\gamma^{[1]} & (\gamma^4)^{[1]} & (\gamma^7)^{[1]} & (\gamma^{10})^{[1]} \\
\end{pmatrix}
= \begin{pmatrix}
1 & 1 & 1 & 1 \\
\gamma & \gamma^4 & \gamma^7 & \gamma^{10} \\
\end{pmatrix}
$$
Chapter 4. Generalized Skew-Evaluation Codes

We can compute $H$ from $(V^{-1})^\top$ in the previous example.

\[
H = \begin{pmatrix}
\gamma^{140} & \gamma^{15} & \gamma^{218} & \gamma^{74} \\
\gamma^{49} & \gamma^{56} & \gamma^{100} & \gamma^{31}
\end{pmatrix}
= \begin{pmatrix}
1 & 1 & 1 & 1 \\
\gamma^{164} & \gamma^{41} & \gamma^{137} & \gamma^{212}
\end{pmatrix}
\begin{pmatrix}
\gamma^{140} & 0 & 0 & 0 \\
0 & \gamma^{15} & 0 & 0 \\
0 & 0 & \gamma^{218} & 0 \\
0 & 0 & 0 & \gamma^{74}
\end{pmatrix}.
\]

Note that $\gamma^{164}, \gamma^{41}, \gamma^{137}, \gamma^{212} \in C(\gamma^2) = C(\gamma^{-1})$.

4.3 Summary

In this chapter, we introduced Generalized Skew-Evaluation codes which contain Gabidulin codes as a special case. A particular pasting construction resulted in a MDS GSE code with MRD component codes. We also developed a duality theory for a class of GSRS codes. The construction of this duality theory required several important ideas: the idea of column multipliers from the duality theory of Generalized Reed-Solomon codes, the generalized Moore matrix inverse, as well as the structure of $\sigma_s$-conjugacy classes.

In the following chapter, we will develop a general Kötter interpolation algorithm. A special adaptation of the algorithm will be a Berlekamp-Welch-like decoder for the GSRS codes defined in this chapter.
Chapter 5

Kötter Interpolation in Skew Polynomial Rings

In this chapter, we present the Kötter interpolation algorithm [32] for free \( \mathbb{F}[x; \sigma, \delta] \)-modules [33]. This algorithm is a generalization of [26], where it was first stated for free \( \mathbb{F}[x] \)-modules. Reference [26] also provides many applications of the Kötter interpolation algorithm. The case for free modules over the ring of linearized polynomials is given in [27]. Our presentation here is not only more general, but also puts the entire setup into a mathematically rigorous framework.

We will also present two specific applications of Kötter interpolation algorithm for free \( \mathbb{F}[x; \sigma, \delta] \)-modules. The first application is a construction of Newton interpolation for skew polynomials. A method of constructing Newton interpolation for skew polynomials is given in [22]. This construction constrains the points of interpolation to come from distinct \((\sigma, \delta)\)-conjugacy classes. Our Kötter interpolation-based algorithm relaxes this constraint and provides a new algebraic condition for interpolation. Furthermore, it is simple to describe and has considerably lower computational complexity compared to the algorithm proposed in [22]. The second application is a Berlekamp-Welch-like decoder that allows us to decode the GSRS codes constructed in Chapter 4.
Chapter 5. Kötter Interpolation in Skew Polynomial Rings

The rest of this chapter is organized as follows. In Section 5.1, we briefly discuss the concept of a free module and present a rigorous formulation of Kötter interpolation algorithm over a free $\mathbb{F}[x;\sigma,\delta]$-module. In Section 5.2, we specialize our interpolation algorithm to develop a Newton interpolation algorithm for skew polynomials. In Section 5.3.2, we discuss a different specialization of the interpolation algorithm that gives a Berlekamp-Welch-like decoder for the GSRS code defined in Chapter 4. Section 5.4 summarizes our contributions in the chapter and discusses some potential extensions.

5.1 Interpolation over Free $\mathbb{F}[x;\sigma,\delta]$-modules

5.1.1 Modules and Free Modules

In this subsection, we present some basic theory on modules and free modules.

Definition 5.1.1. Let $R$ be a commutative ring with multiplicative identity $1_R$. An $R$-module $M$ consists an abelian group $(M,+)$ and an operation $\circ : R \times M \to M$ such that for all $r,s \in R$ and $x,y \in M$, we have:

1. $r \circ (x + y) = r \circ x + r \circ y$
2. $(r + s) \circ x = r \circ x + s \circ x$
3. $(rs) \circ x = r \circ (s \circ x)$
4. $1_R \circ x = x$.

Example 5.1.2. Consider the ring of integers $\mathbb{Z}$. Then every $\mathbb{Z}$-module is in fact an abelian group.

Example 5.1.3. Let $\mathbb{F}$ be a field, then a $\mathbb{F}$-module is simply a vector space over $\mathbb{F}$.

Thus, we can view modules as generalizations of vector spaces. We know that every vector space has a basis. The concept of basis can also be generalized to the case of
modules.

**Definition 5.1.4.** Let $M$ be a $R$-module. The set $E \subset M$ is a basis for $M$ if:

1. $E$ is a generating set for $M$. i.e., every element of $M$ is a finite sum of elements of $E$ multiplied by coefficients in $R$.

2. $E$ is linearly independent. i.e, $r_1 e_1 + r_2 e_2 + \ldots + r_n e_n = 0_M$ with distinct elements $e_1, e_2, \ldots, e_n \in E$, implies that $r_1 = r_2 = \ldots = r_n = 0_R$, where $0_M$ is the zero element in $M$ and $0_R$ is the zero element in $R$.

**Definition 5.1.5.** A free $R$-module is a module with a basis.

**Remark 5.1.6.** Note that just like the case of vector spaces, a basis is not necessarily finite. However, in the case where a free $R$-module $M$ has a finite basis of size $k$, then $M$ is isomorphic to $R^k$, where $R^k$ denotes the direct sum of $k$ copies of $R$.

**Example 5.1.7.** Let $M$ be a free $\mathbb{Z}$-module with basis size 2. Then $M$ is isomorphic to $\mathbb{Z}^2$, where $\mathbb{Z}^2 = \{(a, b) \mid a, b \in \mathbb{Z}\}$ with addition and multiplication rules such that for all $a_1, b_2, a_2, b_2, c \in \mathbb{Z}$:

1. $(a_1, b_1) + (a_2, b_2) = (a_1 + a_2, b_1 + b_2)$.

2. $c(a_1, b_1) = (ca_1, cb_1)$.

**Remark 5.1.8.** An easy way to picture free modules to think about vector spaces are multiple copies of the underlying field $\mathbb{F}$, and free modules as multiple copies of the underlying ring $R$.

### 5.1.2 Free $\mathbb{F}[x; \sigma, \delta]$-module Setup

Let $\mathbb{F}[x; \sigma, \delta]$ be a skew polynomial ring. Let $V$ be a free $\mathbb{F}[x; \sigma, \delta]$-module with a finite ordered module basis $B = \{y_0, y_1, \ldots, y_J\}$, where $J + 1$ is the rank of $V$. Any $Q \in V$ can
be uniquely represented as a finite $\mathbb{F}[x;\sigma,\delta]$-linear combination of basis elements as:

$$Q = \sum_j p_j(x) \circ y_j = \sum_{i,j} a_{i,j} x^i \circ y_j,$$

where $a_{i,j} \in \mathbb{F}$, $p_j(x) \in \mathbb{F}[x;\sigma,\delta]$ and $\circ$ denotes the multiplication between a ring element in $\mathbb{F}[x;\sigma,\delta]$ and a module element in $V$. Note that only finitely many $a_{i,j}$’s are nonzero. Accordingly, we can also view $V$ as a vector space over $\mathbb{F}$ with vector space basis

$$M = \{x^i \circ y_j, i \geq 0, 0 \leq j \leq J\}.$$

We can put a total ordering, $<_o$, on $M$ that satisfies:

1. For any $m \in M$, $m <_o x \circ m$.

2. For any $m_1, m_2 \in M$, if $m_1 <_o m_2$, then $x \circ m_1 <_o x \circ m_2$.

3. $<_o$ is a well-ordering on $M$.

Examples of such orderings can be found in [34]. We will present two instances in Example 5.1.14 and Section 5.2.2.

For a basis element $x^i \circ y_j \in M$, define:

$$\text{Ind}_x(x^i \circ y_j) = i, \quad \text{Ind}_y(x^i \circ y_j) = j.$$

Let $V^* = V \setminus \{0\}$, where 0 is the zero element in the module. For any $Q \in V^*$, define the leading monomial of $Q$, $\text{LM}(Q) \in M$, to be the basis element of the highest order (with respect to $<_o$) in the basis expansion $Q = \sum_{i,j} a_{i,j} x^i \circ y_j$ having $a_{i,j} \neq 0$. Further define:

$$\text{Ind}_x(Q) = \text{Ind}_x(\text{LM}(Q)), \text{Ind}_y(Q) = \text{Ind}_y(\text{LM}(Q))$$
Chapter 5. Kötter Interpolation in Skew Polynomial Rings

Define relations on $V^*$, symbolically denoted by $<$, $\simeq$, as follows. For $Q, Q' \in V^*$, write $Q < Q'$ if $\text{LM}(Q) <_{\sigma} \text{LM}(Q')$, and $Q \simeq Q'$ if $\text{LM}(Q) = \text{LM}(Q')$. We say $Q'$ has a higher order than $Q$ if $Q < Q'$, and we say that $Q$ and $Q'$ have the same order if $Q \simeq Q'$. Note that $\simeq$ is an equivalence relation on $V^*$, and $<$ is a total ordering on $V/\simeq$.

Lastly, we introduce the following notation:

$$S_j = \{Q \in V; \text{Ind}_y(Q) = j\} \cup \{0\}, \quad 0 \leq j \leq L.$$ 

Clearly, $S_i \cap S_j = \{0\}$ for all $0 \leq i, j \leq L$ and $V = \bigcup_j S_j$. With this we prove the following two lemmas, which will be useful in Section 5.1.4.

**Lemma 5.1.9.** If $Q \in S_j$, then $x \circ Q \in S_j$.

*Proof.* $\text{LM}(Q) = x^i \circ y_j$ for some $i$. By the ordering on $M$ (condition 2), multiplication by $x$ cannot change the $j$-index of the leading monomial; hence $\text{LM}(x \circ Q) = x^{i+1} \circ y_j$, so $x \circ Q \in S_j$. $\square$

**Lemma 5.1.10** (Sandwich Lemma). If $Q, Q' \in S_j$ are such that $Q \simeq Q' \simeq x \circ Q$, then either $Q \simeq Q'$ or $Q' \simeq x \circ Q$.

*Proof.* Let $\text{LM}(Q) = x^i \circ y_j$, then $\text{LM}(x \circ Q) = x^{i+1} \circ y_j$. By the ordering on $M$ (condition 1), for fixed $j$-index, the order increases monotonically in the exponent of $x$. Thus $\text{LM}(Q')$ is either $x^i \circ y_j$ or $x^{i+1} \circ y_j$, so we cannot have $Q < Q' < x \circ Q$, thus the result follows. $\square$

### 5.1.3 Interpolation Problem

Let $V$ be a free $\mathbb{F}[x; \sigma, \delta]$-module with basis $B = \{y_0, y_1, \ldots, y_J\}$. Viewing $V$ as a vector space over $\mathbb{F}$ with basis $M = \{x^i \circ y_j; i \geq 0, 0 \leq j \leq J\}$, we choose a set of linear functionals $D_\ell, \ell = 1, 2, \ldots, L$ on $V$. Let $K_\ell = \{Q \in V : D_\ell(Q) = 0\}$ be the kernel of $D_\ell$. Define $\overline{K}_\ell = K_1 \cap K_2 \cap \cdots \cap K_\ell$ and assume $\overline{K}_\ell$ are $\mathbb{F}[x; \sigma, \delta]$-submodules for every $\ell$ (the requirement for $\overline{K}_\ell$ to be submodules will become clear in the proof of the interpolation algorithm).
Then the generalized interpolation problem is to find a nonzero $Q \in \overline{K}_L$ of minimal order (with respect to $\prec$). We call such an element a minimum of the set $\overline{K}_L$. Note that since $\prec_o$ is a well-ordering on $M$, a minimum always exists in $\overline{K}_L$; and our algorithm will imply that a nonzero element always exists in $\overline{K}_L$.

**Example 5.1.11.** Consider the problem of finding a polynomial $f(x) \in \mathbb{F}[x]$ of minimal degree that vanishes on a set of points $\{\alpha_1, \alpha_2, \ldots, \alpha_L\}$. We can view $\mathbb{F}[x]$ as module over itself (with a module basis $y_0 = 1$). As a vector space over $\mathbb{F}$, it has a basis $\{1, x, x^2, \ldots\}$. For $g(x) \in \mathbb{F}[x]$, define $D_\ell(g) = g(\alpha_\ell)$, for $1 \leq \ell \leq L$. Clearly, the $D_\ell$'s are linear functionals from $\mathbb{F}[x]$ to $\mathbb{F}$. The kernels $K_\ell$'s are $\mathbb{F}[x]$-submodules, so $\overline{K}_\ell$'s are $\mathbb{F}[x]$-submodules as well. Hence, the desired $f(x)$ is a minimal degree polynomial in $\overline{K}_L$.

Other interesting examples of interpolation problems for ordinary polynomial rings can be put in this module framework, see [26].

**Lemma 5.1.12.** The minimum $Q \in \overline{K}_L$ is unique up to multiplication by a scalar (in $\mathbb{F}$).

**Proof.** Let $Q, Q'$ be two minimal elements in $\overline{K}_L$ such that $Q \neq cQ'$, for $c \in \mathbb{F}^\star$. Then $Q \approx Q'$, so they have the same leading monomial. Thus, there exists a nontrivial linear combination $\alpha Q + \beta Q' \neq 0$ ($\alpha, \beta \in \mathbb{F}$) such that $\alpha Q + \beta Q' < Q$. But by linearity, $\alpha Q + \beta Q' \in \overline{K}_L$, contradicting the minimality of $Q$.  

### 5.1.4 Interpolation Algorithm

Let $T_{\ell,j} = \overline{K}_\ell \cap S_j$, and $g_{\ell,j}$ be a minimum in $T_{\ell,j}$. Define $T_{0,j} = S_j$ and initialize $g_{0,j} = y_j$. Note that $y_j$ is indeed a minimum in $S_j$. The interpolation algorithm iteratively uses update equations to get $g_{\ell+1,j}$ from $g_{\ell,j}$. This type of update idea originated from Berlekamp’s iterative-discrepancy decoding algorithm [2]. We break down the iterative algorithm in the following cases:

1. (No update): If $g_{\ell,j} \in K_{\ell+1}$, then $g_{\ell+1,j} = g_{\ell,j}$.
2. If \( g_{\ell,j} \notin K_{\ell+1} \), find the \( g_{\ell,j}^* \) among all \( g_{\ell,j} \notin K_{\ell+1} \) such that \( g_{\ell,j}^* < g_{\ell,j'} \), for all \( g_{\ell,j'} \notin K_{\ell+1} \) and \( g_{\ell,j'} \neq g_{\ell,j}^* \). Note that \( g_{\ell,j}^* \) is unique.

(a) \((\text{Order-preserving update})\): For \( g_{\ell,j} \neq g_{\ell,j}^* \), construct \( g_{\ell+1,j} = D_{\ell+1}(g_{\ell,j}^*)g_{\ell,j} - D_{\ell+1}(g_{\ell,j})g_{\ell,j}^* \). Note that by this construction \( g_{\ell+1,j} \approx g_{\ell,j} \).

(b) \((\text{Order-increasing update})\): For \( g_{\ell,j}^* \), construct \( g_{\ell+1,j}^* = D_{\ell+1}(g_{\ell,j}^*)(x \circ g_{\ell,j}^*) - D_{\ell+1}(x \circ g_{\ell,j}^*)g_{\ell,j}^* \). Note that \( g_{\ell,j}^* < g_{\ell+1,j}^* \).

The different type of updates are illustrated in Figure 5.1.4. We shall see that 2.b is the case where the condition on \( \overline{K}_\ell \) being \( \mathbb{F}[x; \sigma, \delta] \)-submodules is used.

**Theorem 5.1.13.** In each of the cases above, the updated \( g_{\ell+1,j} \) is a minimum in \( T_{\ell+1,j} \).

**Proof.** We shall prove each case separately.

1. \((\text{No update})\): We have \( g_{\ell+1,j} = g_{\ell,j} \) and \( g_{\ell,j} \in T_{\ell+1,j} \). Since \( g_{\ell,j} \) is a minimum in \( T_{\ell,j} \supseteq T_{\ell+1,j} \), \( g_{\ell,j} \) must also be a minimum in \( T_{\ell+1,j} \).

2. (a) \((\text{Order-preserving update})\): By construction, we have \( g_{\ell+1,j} = D_{\ell+1}(g_{\ell,j}^*)g_{\ell,j} - \)}
\[D_{\ell+1}(g_{\ell,j})g_{\ell,j^*}.\] So, by linearity of \(D_{\ell+1}\)

\[
D_{\ell+1}(g_{\ell+1,j}) = D_{\ell+1}(g_{\ell,j^*})D_{\ell+1}(g_{\ell,j}) - D_{\ell+1}(g_{\ell,j})D_{\ell+1}(g_{\ell,j^*})
\]

\[= 0.
\]

Thus, \(g_{\ell+1,j} \in K_{\ell+1}\). Now, \(\text{Ind}_{y}(g_{\ell+1,j}) = \text{Ind}_{y}(g_{\ell,j})\) (since \(g_{\ell+1,j} \simeq g_{\ell,j}\)), so \(g_{\ell+1,j} \in S_{j}\) and thus \(g_{\ell+1,j} \in T_{\ell+1,j}\). Furthermore, for any \(k \leq \ell\), since \(g_{\ell,j}, g_{\ell,j^*} \in \overline{K}_{\ell}\), we get

\[
D_{k}(g_{\ell+1,j}) = D_{\ell+1}(g_{\ell,j^*})D_{k}(g_{\ell,j}) - D_{\ell+1}(g_{\ell,j})D_{k}(g_{\ell,j^*})
\]

\[= 0 + 0
\]

\[= 0.
\]

So, \(g_{\ell+1,j} \in T_{\ell,j}\). But as observed, \(g_{\ell+1,j} \simeq g_{\ell,j}\), and since \(g_{\ell,j}\) is a minimum in \(T_{\ell,j}\), \(g_{\ell+1,j}\) is a minimum in \(T_{\ell,j}\) as well. As \(T_{\ell+1,j} \subset T_{\ell,j}\), \(g_{\ell+1,j}\) is a minimum in \(T_{\ell+1,j}\).

(b) (Order-increasing update): By construction,

\[
D_{\ell+1}(g_{\ell+1,j^*}) = D_{\ell+1}(g_{\ell,j^*})D_{\ell+1}(x \circ g_{\ell,j^*}) - D_{\ell+1}(x \circ g_{\ell,j^*})D_{\ell+1}(g_{\ell,j^*})
\]

\[= 0,
\]

\(g_{\ell+1,j^*} \in K_{\ell+1}\). For any \(k \leq \ell\), applying \(D_{k}\), we get:

\[
D_{k}(g_{\ell+1,j^*}) = D_{\ell+1}(g_{\ell,j^*})D_{k}(x \circ g_{\ell,j^*}) - D_{\ell+1}(x \circ g_{\ell,j^*})D_{k}(g_{\ell,j^*})
\]

\[= D_{\ell+1}(g_{\ell,j^*})D_{k}(x \circ g_{\ell,j^*}) - 0
\]

\[= D_{\ell+1}(g_{\ell,j^*}) \cdot 0
\]

\[= 0.
\]
Here the second line follows from the fact $g_{\ell,j^*} \in \overline{K}_k$ for all $k \leq \ell$, and the third line follows from the fact that each $\overline{K}_k$ is a $\mathbb{F}[x;\sigma,\delta]$-submodule, so $x \circ g_{\ell,j^*} \in \overline{K}_k$. Thus $g_{\ell+1,j^*} \in \overline{K}_{\ell+1}$. Also, $\text{Ind}_y(g_{\ell+1,j^*}) = \text{Ind}_y(x \circ g_{\ell,j^*}) = j^*$, so $g_{\ell+1,j^*} \in T_{\ell+1,j^*}$.

We are now left to show that $g_{\ell+1,j^*}$ is a minimum in $T_{\ell+1,j^*}$. Suppose to the contrary that there exists $f \in T_{\ell+1,j^*}$ such that $f < g_{\ell+1,j^*}$. Now, $T_{\ell+1,j^*} \subset T_{\ell,j^*}$, so $f \in T_{\ell,j^*}$. We get the following relation:

$$g_{\ell,j^*} \prec f < x \circ g_{\ell,j^*},$$

where the first inequality follows from the minimality of $g_{\ell,j^*} \in T_{\ell,j^*}$, and the second inequality follows from our assumption on $f$ and the fact $g_{\ell+1,j^*} \succeq x \circ g_{\ell,j^*}$. Then, by the Sandwich Lemma, $f \simeq g_{\ell,j^*}$. In this case, we can find $\alpha, \beta \in \mathbb{F}$ such that $h = \alpha f + \beta g_{\ell,j^*}$ and $h < g_{\ell,j^*}$. We deduce $h \in \overline{K}_\ell$ and $h \notin \overline{K}_{\ell+1}$, as $g_{\ell,j^*} \notin T_{\ell+1,j^*}$, so $h \in \overline{K}_\ell \setminus \overline{K}_{\ell+1}$. Now, $g_{\ell,j^*}$ has the minimal order in $\overline{K}_\ell \setminus \overline{K}_{\ell+1}$, so $h < g_{\ell,j^*}$ contradicts the minimality of $g_{\ell,j^*}$. Hence $f$ cannot exist.

The above iterative method is summarized in Algorithm 1.

**Remarks**

1. The only instance where we used the fact $\overline{K}_\ell$'s are $\mathbb{F}[x;\sigma,\delta]$-submodules is in case 2.b (order-increasing update). By our choice of $j^*$, it is in fact sufficient that only $\overline{K}_\ell \cap S_{j^*}$'s are $\mathbb{F}[x;\sigma,\delta]$-submodules. Note that this means $\overline{K}_\ell \cap S_{j^*}$'s are $\mathbb{F}[x;\sigma,\delta]$-submodules for each $j^*$ from every iteration $i$.

2. The requirement that $V$ is a free module allows us to properly define the conditions for the ordering $<_o$. These conditions are ultimately used to prove the Sandwich Lemma, which in turn is critical in proving case 2.b (order-increasing update). We
Algorithm 1 Free $\mathbb{F}[x;\sigma,\delta]$-modules Interpolation Algorithm

**Input:** Linear functionals $D_\ell$, $1 \leq \ell \leq L$.  
**Output:** Minimum $Q \in \mathbb{K}_L$.

1. for $j = 0 : J$ do
2. \hspace{1em} $g_j \leftarrow y_j$;
3. end for
4. for $\ell = 1 : L$ do
5. \hspace{1em} for $j = 0 : J$ do
6. \hspace{2em} $\Delta_j \leftarrow D_\ell(g_j)$
7. \hspace{1em} end for
8. \hspace{1em} $A \leftarrow \{ j : \Delta_j \neq 0 \}$
9. \hspace{1em} if $A \neq \emptyset$ then
10. \hspace{2em} $j^* \leftarrow \text{argmin}\{g_j : \Delta_j \neq 0\}$
11. \hspace{2em} for $j \in A$ do
12. \hspace{3em} if $j \neq j^*$ then
13. \hspace{4em} $g_j \leftarrow \Delta_j g_j - \Delta_j g_{j^*}$
14. \hspace{3em} else if $j = j^*$ then
15. \hspace{4em} $g_j \leftarrow \Delta_j (x \circ g_{j^*}) - \Delta_i (x \circ g_{j^*}) g_{j^*}$
16. \hspace{3em} end if
17. \hspace{2em} end for
18. \hspace{1em} end if
19. end for
20. $Q \leftarrow \min\{g_j\}$

shall later in see in Section 5.2 that these conditions can be relaxed if we have additional information on when the order-increasing update can happen.

**Example 5.1.14.** Let $V$ be a free $\mathbb{F}[x;\sigma,\delta]$-modules with basis $\{y_0, y_1\}$, and represent an element $Q \in V$ as $Q = \sum_{i,j} a_{i,j} x^i \circ y_j$. The graded lexicographical order is used on $M = \{x^i \circ y_j, i \geq 0, j = 0, 1\}$, defined as follows for $m_1, m_2 \in M$:

1. If $\text{Ind}_x(m_1) \trianglerighteq \text{Ind}_x(m_2)$, then $m_1 \trianglerighteq m_2$.

2. If $\text{Ind}_x(m_1) = \text{Ind}_x(m_2)$ and $\text{Ind}_y(m_1) \trianglerighteq \text{Ind}_y(m_2)$, then $m_1 \trianglerighteq m_2$. 
Define the following three linear functionals from $V$ to $F$:

\[ D_1(Q) = a_{0,0}, \]
\[ D_2(Q) = a_{0,1}, \]
\[ D_3(Q) = a_{1,0} + a_{1,1}. \]

It is easy to check that $K_\ell = \ker(D_\ell)$ are $F[x; \sigma, \delta]$-submodules for $\ell = 1, 2$. While $K_3$ is not a $F[x; \sigma, \delta]$-submodule, the intersection $\overline{K}_3$ is a $F[x; \sigma, \delta]$-submodule since any $Q \in \overline{K}_3$ has the form $Q = a_{1,0}x \circ y_0 + a_{1,1}x \circ y_1 + \text{higher order terms}$, with $a_{1,0} = -a_{1,1}$. Elements of such form are clearly closed under addition. Furthermore, multiplication by the ring element $x$ preserves $a_{1,0} = -a_{1,1}$, as both will be zero.

We apply the interpolation algorithm as follows:

1. $\overline{K}_1 = K_1$:

Consider $g_{1,0}$, a minimal element of $T_{1,0} = \overline{K}_1 \cap S_0$. Thus, $g_{1,0} = b_{1,0}x \circ y_0$ for some $b_{1,0} \in F^*$. For simplicity, we can pick $b_{1,0} = 1$, so $g_{1,0} = xy_0$. Similar, by the given ordering on elements, $g_{1,1} = b_{0,1}x^0 \circ y_1 = b_{0,1}y_1$ for some $b_{0,1} \in F^*$. Again, we shall pick $b_{0,1} = 1$, so $g_{1,1} = y_1$.

2. $\overline{K}_2 = K_1 \cap K_2$:

Since $D_2(Q) = a_{0,1}$ has no effect on $y_0$, so $g_{1,0} \in K_2$ and $g_{2,0} = g_{1,0} = x \circ y_0$. This is case 1, where no update occurs. Next, clearly $D_2(g_{1,1}) = 1 \neq 0$, so $g_{1,1} \notin T_{2,1}$. As $g_{1,1} = g_{1,1}^*$, we are in the case of order-increasing update, with:

\[ g_{2,1} = D_2(g_{1,1})(x \circ g_{1,1}) - D_2(x \circ g_{1,1})g_{1,1} \]
\[ = D_2(y_1)x \circ y_1 - D_2(x \circ y_1)y_1 \]
\[ = x \circ y_1. \]

3. $\overline{K}_3 = K_1 \cap K_2 \cap K_3$:
It is clear both $g_{2,0}, g_{2,1} \notin K_3$. Since $g_{2,0} < g_{2,1}$, $g_{2,1} = g_{2,0}$. Then, using the order-increasing update,

$$g_{3,0} = D_3(g_{2,0})(x \circ g_{2,0}) - D_3(x \circ g_{2,0})g_{2,0}$$
$$= D_3(x \circ y_0)x^2y_0 - D_2(x^2 \circ y_1)x \circ y_0$$
$$= x^2 \circ y_0.$$

For $g_{3,1}$, we will use the order-preserving update:

$$g_{3,1} = D_3(g_{2,j}')(g_{2,1}) - D_3(g_{2,1})g_{2,j'}$$
$$= D_3(g_{2,0})(g_{2,1}) - D_3(g_{2,1})g_{2,0}$$
$$= D_3(x \circ y_0)x \circ y_1 - D_3(x \circ y_1)x \circ y_0$$
$$= x \circ y_1 - x \circ y_0.$$

## 5.2 Newton Interpolation for Skew Polynomials

### 5.2.1 Newton Interpolation Problem

In this section, we discuss the Newton form of the polynomial interpolation problem (Newton interpolation) for skew polynomial rings. We shall show that the interpolation problem can be solved by adapting the general algorithm in Section 5.1 to this special case. The construction technique follows the one used in [26].

In the ordinary polynomial ring $\mathbb{F}[x]$, given a set of distinct points $\{\alpha_1, \alpha_2, \ldots, \alpha_n\}, \alpha_i \in \mathbb{F}$, and a set of points $\{\beta_1, \beta_2, \ldots, \beta_n\}, \beta_i \in \mathbb{F}$, the interpolation polynomial $f(x) \in \mathbb{F}[x]$ is a polynomial of minimal degree such that $f(\alpha_i) = \beta_i$ for all $1 \leq i \leq n$. 
Expressed in the Newton form
\[ f(x) = c_0 + c_1(x - \alpha_1) + c_2(x - \alpha_1)(x - \alpha_2) + \ldots + c_{n-1}(x - \alpha_1)\cdots(x - \alpha_{n-1}), \]

we have the well-known divided-difference formula for finding all the \(c_i\)'s. An important advantage of interpolation in the Newton form using the divided-difference formula (as compared to, for example, Lagrange interpolation), is that if an additional pair of points \(\{\alpha_{n+1}, \beta_{n+1}\}\) is added, the new coefficient \(c_n\) can be computed in one step and all previous \(c_i\)'s remain unmodified.

In the skew polynomial case, the noncommutativity makes it difficult to have a straightforward generalization of the recursive divided-difference formula for \(c_i\)'s. Our approach, however, will retain the important property of having a simple update equation for an additional pair of points.

The problem setup in the skew polynomial case is as follows:

Given a set of points \(\{\alpha_1, \alpha_2, \ldots, \alpha_n\}, \alpha_i \in F\) such that \(\text{rank}(V_{\sigma,\delta}(\alpha_1, \alpha_2, \ldots, \alpha_n)) = n\), and a set of points \(\{\beta_1, \beta_2, \ldots, \beta_n\}, \beta_i \in F\), find a polynomial \(f(x) \in F[x; \sigma, \delta]\) of minimum degree such that \(f(\alpha_i) = \beta_i\) for all \(1 \leq i \leq n\).

Remarks

1. The requirement that \(\text{rank}(V_{\sigma,\delta}(\alpha_1, \alpha_2, \ldots, \alpha_n)) = n\) is necessary for the proof of Lemma 5.2.1, which in turn is crucial for the correctness of the Newton interpolation algorithm.

2. The requirement that \(\text{rank}(V_{\sigma,\delta}(\alpha_1, \alpha_2, \ldots, \alpha_n)) = n\) is equivalent to the set \(\{\alpha_1, \alpha_2, \ldots, \alpha_n\}\) having minimal polynomial of degree \(n\). However, the condition that \(\text{rank}(V_{\sigma,\delta}(\alpha_1, \alpha_2, \ldots, \alpha_n)) = n\) is usually simpler to verify than finding the minimal polynomial of the set \(\{\alpha_1, \alpha_2, \ldots, \alpha_n\}\).

3. This problem setup is a generalization of [22], where \(\alpha_1, \alpha_2, \ldots, \alpha_n\) are assumed to have come from \(n\ (\sigma, \delta)\)-distinct conjugacy classes.


5.2.2 Newton Interpolation Algorithm

We shall adapt the generalized interpolation algorithm in Section 5.1 for the following $\mathbb{F}[x;\sigma,\delta]$-module.

Set $V = \{cy + h(x), c \in \mathbb{F}, h(x) \in \mathbb{F}[x;\sigma,\delta]\}$, and define a binary operation $\circ : \mathbb{F}[x;\sigma,\delta] \times V \to V$ as:

$$p(x) \circ (cy + h(x)) = (a_0 c)y + p(x)h(x), \quad \forall p(x) \in \mathbb{F}[x;\sigma,\delta]$$

where $a_0$ is the constant term of $p(x)$. We can check that $V$ is a $\mathbb{F}[x;\sigma,\delta]$-module, but it is not a free $\mathbb{F}[x;\sigma,\delta]$-module. Clearly $V$ is still a finitely generated module, with a generating set $\{y_0, y_1\} = \{1, y\}$. As a vector space over $\mathbb{F}$, $V$ has a vector space basis $M = \{x^i, i \geq 0\} \cup \{y\}$. We can assign an ordering on $M$ as $x^i <_o x^{i+1}$ and $x^i <_o y$, for all $i \in \mathbb{N}$. Define linear functionals $D_\ell$ as

$$D_\ell(cy + h(x)) = c\beta_\ell + h(\alpha_\ell) \quad 1 \leq \ell \leq n.$$ 

Note that this setup if slightly different from the one in Section 5.1. Specifically, $V$ is not a free module, and the ordering $<_o$ do not satisfy the order conditions in Section 5.1.2. Furthermore from our choice of $D_\ell$’s, $\overline{K}_\ell$’s are not $\mathbb{F}[x;\sigma,\delta]$-submodules. As remarked in Section 5.1.4, all these conditions are necessary only for the order-increasing update. The following lemma and its corollary tell us that the order-increasing update can only happen in $S_0$ (i.e., $j^* = 0$ for all $\ell$).

Lemma 5.2.1. Let $g_{\ell,0} = \min\{\overline{K}_\ell \cap S_0\}$, then $D_{\ell+1}(g_{\ell,0}) \neq 0$.

Proof. Since $g_{\ell,0} \in \overline{K}_\ell \cap S_0$, we can let $g_{\ell,0} = f(x)$. So $f(\alpha_k) = 0, 1 \leq k \leq \ell$. Since $f$ is a polynomial of minimal degree, by Corollary 2.3.17, $\deg(f(x)) = \ell$. Now suppose that $D_{\ell+1}(g_{\ell,0}) = f(\alpha_{\ell+1}) = 0$, then $f(x)$ is a polynomial of degree $\ell$ that vanishes on the set $\{\alpha_1, \alpha_2, \ldots, \alpha_{\ell+1}\}$. However, by Corollary 2.3.17, the minimal polynomial of
Algorithm 2 Newton Interpolation Algorithm

Input: \( \{\alpha_1, \alpha_2, \ldots, \alpha_n\}, \{\beta_1, \beta_2, \ldots, \beta_n\} \), with \( \alpha_i, \beta_i \in \mathbb{F} \) and \( \text{rank}(V_{\sigma, \delta}(\alpha_1, \alpha_2, \ldots, \alpha_n)) = n \).

Output: \( Q = f(x) \) such that \( f(\alpha_i) = \beta_i \) for \( 1 \leq i \leq n \).

1: Initialize. \( g_0 \leftarrow 1; \) \( g_1 \leftarrow y \);
2: for \( \ell = 1 : n \) do
3: \( \Delta_0 \leftarrow D_\ell(g_0); \) \( \Delta_1 \leftarrow D_\ell(g_1) \)
4: if \( \Delta_1 \neq 0 \) then
5: \( g_1 \leftarrow \Delta_0 g_1 - \Delta_1 g_0 \)
6: end if
7: \( g_0 \leftarrow \Delta_0 (x - \alpha_\ell \alpha_\ell^{g_0(\alpha_\ell)}) g_0(\alpha_\ell) \)
8: end for
9: \( Q \leftarrow g_1 \)

\( \{\alpha_1, \alpha_2, \ldots, \alpha_{\ell+1}\} \) has degree \( \ell + 1 \). This gives us the desired contradiction.

\[ \text{Corollary 5.2.2.} \] At every iteration \( \ell \), \( j^* = 0 \).

Proof. Following Lemma 5.2.1, \( D_{\ell+1}(g_{\ell,0}) \neq 0 \). Hence, at every iteration, \( g_{\ell,j^*} = g_{\ell,0} \).

Indeed, when restricted to the generator \( \{1\} \) (i.e., in \( S_0 \)), the submodule \( \mathbb{F}[x; \sigma, \delta] \) is clearly a free module (over itself), and the ordering on \( \{x^i\} \) satisfies the ordering conditions in Section 5.1.2. Also, \( \overline{K}_\ell \cap S_0 \) is a \( \mathbb{F}[x; \sigma, \delta] \)-submodule for every \( \ell \). Thus, the interpolation algorithm in Section 5.1 can be adapted to this setup.

Let \( f(x) \in S_0 \), which means \( f(x) \in \mathbb{F}[x; \sigma, \delta] \). Then, by the Product Theorem, we have \( D_k(xf(x)) = \alpha_k^{f(\alpha_k)} f(\alpha_k) \). Hence, the order-increasing update \( D_k(f(x))xf(x) - D_k(xf(x))f(x) = D_k(f(x))(x - \alpha_k^{f(\alpha_k)})f(x) \). By Lemma 5.2.1, the order-increasing update will be used in every iteration \( i \) for \( g_0 = \min \{\overline{K}_\ell \cap S_0\} \). The adapted version of the general algorithm can be summarized in Algorithm 2.

\[ \text{Example 5.2.3.} \] Following Example 2.2.5 in Chapter 2, we use the finite field \( \mathbb{F}_{16} \cong \mathbb{F}_2[\gamma] \), where \( \gamma \) is a primitive root of the irreducible polynomial \( x^4 + x + 1 \), with \( \sigma(a) = a^4 \), \( \delta(a) = \sigma(a) + a = a^4 + a \), for all \( a \in \mathbb{F}_{16} \). We now consider the Newton interpolation problem on set of points in \( \mathbb{F}_{16} \), with \( \{\alpha_1, \alpha_2, \alpha_3\} = \{0, 1, \gamma\} \) and \( \{\beta_1, \beta_2, \beta_3\} = \{\gamma, \gamma^2, \gamma^3\} \). Note
that, as shown in Example 2.2.5, the elements of the set \{0, 1, \gamma\} all belong to different conjugacy classes.

Following Algorithm 2, denote \(g_0, g_1\) in the \(\ell\)th step as \(g_{i,0}\) and \(g_{i,1}\) respectively. Then, in the initialization step, we have \(g_{0,0} = 1\) and \(g_{0,1} = y\). In the first iteration \((\ell = 1)\), applying the algorithm, we get:

\[
g_{1,0} = D_1(g_{0,0})(x - \alpha_1^{g_{0,0}(\alpha_1)})g_{0,0} = x - \alpha_1 = x.
\]

\[
g_{1,1} = D_1(g_{0,0})g_{0,1} - D_2(g_{0,1})g_{0,0} = y - \beta_1 = y + \gamma,
\]

as we do not distinguish between addition and subtraction in characteristic 2. In iteration \(\ell = 2\),

\[
g_{2,0} = D_2(g_{1,0})(x - \alpha_2^{g_{1,0}(\alpha_2)})g_{1,0} = (\alpha_2 + \alpha_1)(x + \alpha_2^{g_{1,0}(\alpha_2)})g_{1,0} = (1 + 0)(x + 1)g_{1,0} = x^2 + x.
\]
Also,

\[ g_{2,1} = D_2(g_{1,0})g_{1,1} - D_2(g_{1,1})g_{1,0} = (\alpha_2 + \alpha_1)(y + \beta_1) + (\beta_2 + \beta_1)(x + \alpha_1) = (1 + 0)(y + \gamma) + (\gamma^2 + \gamma)(x + 0) = y + \gamma^5x + \gamma. \]

In the next iteration (\(\ell = 3\)), we have:

\[ g_{3,0} = D_3(g_{2,0})(x - \alpha_3^{g_{2,0}(a_3)})g_{2,0} = (N_2(\gamma) + N_1(\gamma))(x + \gamma^{N_2(\gamma) + N_1(\gamma)})(x^2 + x) = (\gamma^{10} + \gamma)(x + \gamma^{10} + \gamma)(x^2 + x) = \gamma^8(x + \gamma^8)(x^2 + x) = \gamma^8(x + \gamma^6)(x^2 + x) = \gamma^8x^3 + (\gamma^8 + \gamma^{14})x^2 + \gamma^{14}x = \gamma^8x^3 + \gamma^6x^2 + \gamma^{14}x. \]

\[ g_{3,1} = D_3(g_{2,0})g_{2,1} - D_3(g_{2,1})g_{2,0} = \gamma^8(y + \gamma^5x + \gamma) + (\beta_3 + \gamma^5\alpha_3 + \gamma)(x^2 + x) = \gamma^8(y + \gamma^5x + \gamma) + (\gamma^3 + \gamma^5\gamma + \gamma)(x^2 + x) = \gamma^8(y + \gamma^5x + \gamma) + (\gamma^5)(x^2 + x). \]
The desired module element is $g_{3,1}$. Setting $g_{3,1} = 0$ and isolating for $y$, we get:

\[
y = (\gamma^8)^{-1}(\gamma^5)(x^2 + x) + \gamma^5 x + \gamma
= \gamma^{12} x^2 + (\gamma^{12} + \gamma^5)x + \gamma
= \gamma^{12} x^2 + \gamma^{14} x + \gamma.
\]

With this, we can easily verify:

\[
y(0) = \gamma \\
y(1) = \gamma^{12} + \gamma^{14} + \gamma
= \gamma^2 \\
y(\gamma) = \gamma^{12} N_2(\gamma) + \gamma^{14} N_1(\gamma) + \gamma
= \gamma^{12}\gamma^{10} + \gamma^{14}\gamma + \gamma
= \gamma^3.
\]

### 5.2.3 Computational Complexity

To analyze the complexity of this algorithm, we assume that each application of the functions $\sigma, \delta$, as well as multiplication of two field elements has $O(1)$ complexity. We do not consider extra complexity for field additions. An evaluation of a skew polynomial of degree $k$ will require the computations of $N_1, N_2, \ldots, N_k$. By the recursive definition of $N_i$ functions, computing $N_{i+1}$ given that $N_i$ has been computed takes $O(1)$ complexity. Thus, an evaluation of a skew polynomial of degree $k$ will cost $O(k)$ complexity. In Algorithm 2, at the $\ell$th iteration, we need to evaluate two skew polynomials of degree at most $\ell$, as well as an update equation which is also $O(\ell)$. Summing this over $n$ iterations, we achieve an overall complexity of $O(n^2)$.

In the Newton interpolation algorithm given in [22], one writes the resulting polyno-
Polynomial as \( f(x) = \sum_{i=1}^{n} a_i g_i(x) \). Finding the \( g_i(x) \) functions take a total of \( O(n^2) \) operations. From the \( g_i \)'s, one needs to iteratively compute the \( a_i \)'s. Computing each \( a_i \) requires evaluating \( g_1 \) up to \( g_i \) and doing some extra multiplications (the number grows with \( i \)). The complexity for each \( a_i \) computation is \( O(n^2) \), giving the overall complexity of \( O(n^3) \).

By comparison, taking advantage of the framework of the generalized interpolation algorithm, our Newton interpolation algorithm is both simple to describe and computationally efficient.

5.3 Berlekamp-Welch Decoder for GSRS-Codes

5.3.1 Berlekamp-Welch Setup

A Berlekamp-Welch-type decoder for Gabidulin codes (in rank metric) was presented in [35]. The setup for skew polynomial evaluation in Hamming metric was presented in [20]. The reformulation for GSRS codes is as follows:

**Theorem 5.3.1.** Let \( C \) be an \((n, k)\) GSRS code over \( \mathbb{F}_{q^m} \) with code locators \( \{\alpha_1, \ldots, \alpha_n\} \) and column multipliers \( \{v_1, \ldots, v_n\} \). If \( c = (v_1 f(\alpha_1), \ldots, v_n f(\alpha_n)) \in C \) and \( y = (\beta_1, \ldots, \beta_n) \in \mathbb{F}_{q^m}^n \) are such that the Hamming weight of \( y - c \leq t < (n - k + 1)/2 \). Then, we can find \( Q_0, Q_1 \in \mathbb{F}_{q^m}[x; \sigma] \) such that

- \( \deg(Q_0) \leq k + t, \deg(Q_1) \leq k \).
- \( Q_0(\alpha_i) + Q_1(\alpha_i^{\frac{\beta_i}{v_i}}) \frac{\beta_i}{v_i} = 0 \) for all \( i \), where \( \beta_i \neq 0 \).
- \( Q_0(\alpha_i) = 0 \) for all \( i \), where \( \beta_i = 0 \).
- \( f \) is the quotient of the left division of \( Q_0 \) by \( -Q_1 \).

In the following section, we shall show that \( Q_0 \) and \( Q_1 \) can be constructed using Kötter interpolation in skew polynomial rings. We shall see that our algorithm allows us to replace the condition \( \deg(Q_1) \leq k \) by a weaker condition that \( \deg(Q_0) > \deg(Q_1) \).
Lastly, for simplicity of notation, we can assume, without loss of generality, that \( v_i = 1 \) for all \( i \).

### 5.3.2 Berlekamp-Welch via Kötter Interpolation

Define a free \( \mathbb{F}_{q^m}[x; \sigma] \)-module \( V \) with basis \( \{1, y\} \) and write elements in \( V \) as:

\[
Q(x, y) = Q_0(x) + Q_1(x) \circ y.
\]

Multiplication of a ring element \( f(x) \) with a module element \( Q(x, y) \) is defined as:

\[
f(x)Q(x, y) = f(x)(Q_0(x) + Q_1(x) \circ y) = f(x)Q_0(x) + (f(x)Q_1(x)) \circ y.
\]

Symbolically \( y \) is simply a “place holder” for a module basis element. However, we want to define an evaluation map on \( V \). Given a pair \( (a, b) \in \mathbb{F}_{q^m}^2 \), define \( Q(a, b) \) as follows:

\[
Q(a, b) = Q_0(a) + Q_1(a^b)b \quad \text{if } b \neq 0,
\]

\[
Q(a, b) = Q_0(a) \quad \text{if } b = 0
\]

where we use the standard skew polynomial evaluation on \( Q_0 \) and \( Q_1 \), and \( a^b \) denotes the \( \sigma \)-conjugation of \( a \) by \( b \).

As a vector space, \( V \) has a basis

\[
M = \{x^i \circ y^j, i \geq 0, 0 \leq j \leq 1\},
\]

where \( y_0 = 1 \) and \( y_1 = y \). Following Example 5.1.14, we can define the graded lexicograph-
ical order on $M$. Namely,

$$x^i <_0 x^j y <_0 x^{j+1}.$$  

As before, we define

$$S_j = \{Q \in V; \text{Ind}_y(Q) = j\} \cup \{0\}, \quad 0 \leq j \leq 1.$$  

Finally, given the list of points $\alpha_i, \beta_i, 1 \leq i \leq n$ in Theorem 5.3.1, define a set of $n$ linear functionals $D_i$ on $V$:

$$D_i(Q_0(x) + Q_1(x) \circ y) = Q_0(\alpha_i) + Q_1(\alpha_i^{\beta_i})\beta_i.$$  

It is not difficult to show, by our evaluation rule, that each $K_i = \ker(D_i)$ is a $\mathbb{F}_{q^m}[x; \sigma]$-submodule.

With the above setup in the generalized interpolation algorithm framework, we can find a $Q$ with the lowest order satisfying the linear functionals $D_i$. The only extra requirement of Berlekamp-Welch is that $\text{deg}(Q_0) > \text{deg}(Q_1)$. This amounts to restricting the minimal solution to be in $S_0$. The resulting algorithm is described in Algorithm 3, where, in the $i$th iteration, we let $g_{0,i} = V^i(x, y) = V_1^i(x) + V_2^i(x)y \in K_i \cap S_0$ and $g_{1,i} = W^i(x, y) = W_1^i(x) + W_2^i(x)y \in K_i \cap S_1$.

**Example 5.3.2.** Consider $\mathbb{F}_{16}$ with a primitive element $\gamma$ and $\sigma(a) = a^4$. Let $C$ be a $(4, 2)$ GSRS code with code locators $(\alpha_1, \alpha_2, \alpha_3, \alpha_4) = (1, \gamma, \gamma^2, \gamma^3)$ and unit column multipliers $(v_i = 1$ for all $i$). Let $f(x) = x + 1$ and $c = (f(\alpha_1), f(\alpha_2), f(\alpha_3), f(\alpha_4)) = (0, \gamma^4, \gamma^8, \gamma^{14})$.

Suppose the codeword $c$ was transmitted over a channel and we received $y = (\beta_1, \beta_2, \beta_3, \beta_4) = (0, \gamma^4, 0, \gamma^{14})$, where a single symbol error occurred in $y$. We shall use our Skew Berlekamp-Welch algorithm to reconstruct $f(x)$ as follows:

1. $i = 1$, with $(g_{0,0} = 1, g_{1,0} = y), (\alpha_1, \beta_1) = (1, 0)$. 
Algorithm 3 Skew Berlekamp-Welch Algorithm

1: Initialize. $g_0 \leftarrow 1$; $g_1 \leftarrow y$;
2: for $i = 1 : n$ do
3: \hspace{1em} $\Delta_0 \leftarrow D_i(g_0)$; $\Delta_1 \leftarrow D_i(g_1)$
4: \hspace{1em} if $\Delta_0 = 0$ and $\Delta_1 = 0$ then
5: \hspace{2em} $g_0 \leftarrow g_0$; $g_1 \leftarrow g_1$;
6: \hspace{1em} else if $\Delta_0 = 0$ then
7: \hspace{2em} $g_1 \leftarrow D_i(g_1)xg_1 - D_i(xg_1)g_1$
8: \hspace{1em} else if $\Delta_1 = 0$ then
9: \hspace{2em} $g_0 \leftarrow D_i(g_0)xg_0 - D_i(xg_0)g_0$
10: \hspace{1em} else
11: \hspace{2em} if $\deg(W_{i}^{(i)}) < \deg(V_{i}^{(i)})$ then
12: \hspace{3em} $g_0 \leftarrow \Delta_1 g_0 - \Delta_0 g_1$
13: \hspace{3em} $g_1 \leftarrow D_i(g_1)xg_1 - D_i(xg_1)g_1$
14: \hspace{2em} else
15: \hspace{3em} $g_1 \leftarrow \Delta_0 g_0 - \Delta_1 g_0$
16: \hspace{3em} $g_0 \leftarrow D_i(g_0)xg_0 - D_i(xg_0)g_0$
17: \hspace{1em} end if
18: \hspace{1em} end if
19: \hspace{1em} end if
20: $Q \leftarrow g_0$

- $D_1(g_{0,0}) = 1$ and $D_1(g_{1,0}) = 0$.
- Since $\Delta_1 = 0$, we get $g_{1,1} = g_{1,0}$ and

$$g_{0,1} = D_1(g_{0,0})g_{0,0} - D_1(xg_{0,0})g_{0,0} = x + 1$$

2. $i = 2$, with $(\alpha_2, \beta_2) = (\gamma, \gamma^4)$.

- $D_2(g_{0,1}) = \gamma^4$ and $D_2(g_{1,1}) = \gamma^4$.
- Since $y < o x + 1$, we get:

$$g_{0,2} = \Delta_1 g_{0,1} - \Delta_0 g_{1,1} = \gamma^4 x + \gamma^4 + \gamma^4 y,$$

$$g_{1,2} = D_2(g_{1,1})xg_{1,1} + D_2(xg_{1,1})g_{1,1}$$

$$= (\gamma^4 x + \gamma^2)y.$$
3. \(i = 3\), with \((\alpha_3, \beta_3) = (\gamma^2, 0)\).

- \(D_3(g_{0,2}) = \gamma^{12}\) and \(D_3(g_{1,2}) = 0\).
- Since \(D_3(g_{1,2}) = 0\), we get \(g_{1,3} = g_{1,2}\) and

\[
g_{0,3} = \Delta_1 g_{0,2} - \Delta_0 g_{1,2} = \gamma^{13} x^2 + \gamma^{10} x + \gamma^9 + (\gamma^{13} x + \gamma^9) y.
\]

4. \(i = 4\), with \((\alpha_4, \beta_4) = (\gamma^3, \gamma^{14})\).

- \(D_4(g_{0,3}) = 0\).
- Since \(D_4(g_{0,3}) = 0\), immediately we get \(g_{0,4} = g_{0,3}\).

Thus, we finally arrived at

\[
Q(x, y) = g_{0,4} = Q_0(x) + Q_1(x) y = \gamma^{13} x^2 + \gamma^{10} x + \gamma^9 + (\gamma^{13} x + \gamma^9) y.
\]

We can easily verify that substituting \(f(x) = x + 1\) for \(y\), we get \(Q_0(x) = Q_1(x) f(x)\). Thus \(f(x)\) is the left quotient of \(Q_0(x)\) by \(Q_1(x)\).

5.4 Summary

In this chapter, we presented a formulation of Kötter interpolation algorithm over a free \(\mathbb{F}[x; \sigma, \delta]\)-module. This formulation is more general than the previous formulations in [26] and [27] and is also mathematically more rigorous. We investigated two applications of our Kötter interpolation algorithm. The first is a Newton interpolation algorithm for skew polynomials. This algorithm generalizes the previously known Newton interpolation
algorithm in [22] by relaxing the condition on interpolation points, and reduces the overall complexity from $O(n^3)$ to $O(n^2)$. The second application is a Berlekamp-Welch style decoder for the Generalized Skew Reed-Solomon codes defined in Chapter 4.

A possible extension to our formulation of Kötter interpolation is to consider a graded ring. This setup is much more general than that of a free module. We have done some preliminary investigation on this extension. However, a general theory and applications are still lacking.

In the next chapter, we will switch topic to investigate skew polynomial rings from a different perspective. We shall show that the $P$-independence structure is in fact a matroidal structure. This observation will lead to many interesting properties and applications.
Chapter 6

Matroidal Structure & Application
to Network Coding

In this chapter, we explore the connection between a skew polynomial ring $\mathbb{F}_q^m[x;\sigma]$ and a representable matroid [36]. We will define a matroid called the $\mathbb{F}_q^m[x;\sigma]$-matroid, which has many interesting properties. For example, the collection of its flats form a metric space. Moreover, the collection of flats on a submatroid is bijectively isometric to the projective geometry of $\mathbb{F}_q^m$ over $\mathbb{F}_q$ equipped with the subspace metric. Using these connections, we present an application of the $\mathbb{F}_q^m[x;\sigma]$-matroid in the context of network coding.

The rest of this chapter is organized as follows. In Section 6.1, we briefly introduce the basic matroid theory. In Section 5.3.2, we discuss a different specialization of the interpolation algorithm that gives a Berlekamp-Welch-like decoder for the GSRS code defined in Chapter 4. Section 5.4 summarizes our contributions in the chapter and discusses some potential extensions.
Chapter 6. Matroidal Structure & Application to Network Coding

6.1 Matroidal Structure

6.1.1 Matroid Basics

In the following, we will only give the basics of matroid theory and follow the notation given in [37]. All the important results in this subsection can be found in [37] and are only restated here for completeness.

**Definition 6.1.1.** A matroid $M$ is an ordered pair $(E,I)$, where $E$ is a finite set and $I$ is a set of subsets of $E$ satisfying the following three conditions:

- (I1) $\emptyset \in I$.
- (I2) If $I \in I$ and $I' \subseteq I$, then $I' \in I$.
- (I3) If $I_1, I_2 \in I$ and $|I_1| < |I_2|$, then there is an element $e \in I_2 - I_1$ such that $I_1 \cup \{e\} \in I$.

If $M = (E,I)$ is a matroid, then $M$ is called a matroid on $E$. The members of $I$ are called the independent sets of $M$ and $E$ is called the ground set of $M$.

A simple class of matroids is defined as follows.

**Definition 6.1.2.** Let $E = \{1, \ldots, n\}$ and let $0 \leq m \leq n$. For any subset $X \subseteq E$, declare $x \in I$ if and only if $|X| \leq m$. Then, $M = (E,I)$ is called the $(n,m)$-uniform matroid and is denoted by $U_{n,m}$.

An important class of matroids comes from linear algebra.

**Definition 6.1.3.** Let $A$ be an $m \times n$ matrix over a field $\mathbb{F}$. Let $E = \{1, \ldots, n\}$. For any $X \subseteq E$, $X \in I$ if the columns indexed by $X$ are linearly independent over $\mathbb{F}$. The pair $(E,I)$ forms a matroid called the vector matroid of $A$. 
Example 6.1.4. Let

\[ A = \begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \]

be a 3×4 matrix over \( \mathbb{F}_2 \). Then \( E = \{1, 2, 3, 4\} \) and

\[ I = \{\varnothing, \{1\}, \{2\}, \{3\}, \{4\}, \{1, 2\}, \{1, 3\}, \{1, 4\}, \{2, 3\}, \{2, 4\}, \{3, 4\}, \{1, 2, 3\}, \{1, 3, 4\}, \{2, 3, 4\}\}. \]

Two matroids \((E_1, I_1)\) and \((E_2, I_2)\) are isomorphic if there exists a bijection \( f : E_1 \rightarrow E_2 \) such that \( I \in I_1 \) if and only if \( f(I) \in I_2 \).

Definition 6.1.5. A matroid \( M \) is representable over a field \( \mathbb{F} \) (\( \mathbb{F}\)-representable) if it is isomorphic to the vector matroid of some matrix over \( \mathbb{F} \). A matroid is representable if it is representable over some field.

Definition 6.1.6. Let \( M \) be a matroid. A maximal independent set in \( M \) is a basis of \( M \).

It is easy to see that all bases of a matroid \( M \) have the same size.

Example 6.1.7. In Example 6.1.4, the sets \( \{1, 2, 3\} \), \( \{1, 3, 4\} \), \( \{2, 3, 4\} \) are all bases of \((E, I)\).

Let \( M \) be the matroid \((E, I)\) and let \( X \subseteq E \). Let \( \mathcal{I}|X = \{I \in X : I \in \mathcal{I}\} \). Then the pair \((X, \mathcal{I}|X)\) is a matroid. We call this matroid the restriction of \( M \) to \( X \), and denote it by \( M|X \).

Definition 6.1.8. The rank \( r(X) \) of \( X \) is the size of a basis of \( M|X \).

It can be verified the rank function \( r \) satisfies the following:

- \((R1)\) If \( X \subseteq E \), then \( 0 \leq r(X) \leq |X| \).
• (R2) If $X \subseteq Y \subseteq E$, then $r(X) \leq r(Y)$.

• (R3) If $X, Y \subseteq E$, then

$$r(X \cup Y) + r(X \cap Y) \leq r(X) + r(Y).$$

Conversely, as the following theorem shows, conditions (R1)-(R3) characterize the rank function of a matroid.

**Theorem 6.1.9.** Let $E$ be a set and $r$ be a function that maps $2^E$ into the set of non-negative integers and satisfies (R1)-(R3). Let $\mathcal{I}$ be the collection of subsets $X$ of $E$ for which $r(X) = |X|$. Then $(E, \mathcal{I})$ is a matroid having rank function $r$.

**Definition 6.1.10.** Let $M = (E, \mathcal{I})$ be a matroid, for any $X \subseteq E$, define the closure of $X$, denoted $\text{cl}(X)$ as

$$\text{cl}(X) = \{x \in E \mid r(X \cup x) = r(X)\}.$$

If $X = \text{cl}(X)$, then $X$ is called a flat.

Let $\mathcal{F}(M)$ be the set of all flats of a matroid $M = (E, \mathcal{I})$. Furthermore, for any $X \subseteq E$, let

$$\mathcal{F}(X) = \{U \subseteq X \mid U = \text{cl}(U)\},$$

i.e., $\mathcal{F}(X)$ denotes the set of all flats contained in $X$.

**Example 6.1.11.** In Example 6.1.4, $\{1, 3\}, \{2, 3\}$ are flats. However, $\{1, 2\}$ is not a flat as $\text{cl}(\{1, 2\}) = \{1, 2, 4\}$. We have $\mathcal{F}(\{1, 2, 3\}) = \varnothing, \{1\}, \{2\}, \{3\}, \{1, 2\}, \{2, 3\}$. 
6.1.2 The $\mathbb{F}_q^m[x;\sigma]$-matroid

For the rest of the chapter, we will consider the ring $\mathbb{F}_q^m[x;\sigma]$. The setup for the general ring $\mathbb{F}_q^m[x;\sigma_s]$ is similar. We shall see, in light of the Structure Theorem in Chapter 3, that we do not lose generality with this restriction.

**Theorem 6.1.12.** Let $\mathbb{F}_q^m[x;\sigma]$ be a skew polynomial ring. Then the pair $M = (\mathbb{F}_q^m,\mathcal{I})$, where

$$\mathcal{I} = \{\Omega \subseteq \mathbb{F}_q^m | |\Omega| = \deg(f_{\Omega})\}$$

is the set of all $P$-independent sets of $\mathbb{F}_q^m$, is a matroid.

**Proof.** Nonzero constant polynomials have no roots, thus $\emptyset \in \mathcal{I}$. Suppose $I \in \mathcal{I}$ and let $I' \subseteq I$. From Corollary 2.3.17, $I'$ is a $P$-independent set.

Now let $I_1, I_2 \in \mathcal{I}$ with $|I_1| < |I_2|$. We need to prove that there exists an element $e \in I_2 \setminus I_1$ such that $I_1 \cup \{e\}$ is still a $P$-independent set. Suppose to the contrary that for all $e \in I_2 \setminus I_1$ it holds that $I_1 \cup \{e\} \notin \mathcal{I}$. It follows that $I_2$ is $P$-dependent on $I_1$. This contradicts the fact that $|I_1| < |I_2|$ and $I_2 \in \mathcal{I}$. 

We can easily verify the following correspondences between notions in matroid theory and notions defined in terms of $P$-independence.

**Lemma 6.1.13.** Let $M = (\mathbb{F}_q^m,\mathcal{I})$ be the matroid constructed from $\mathbb{F}_q^m[x;\sigma]$ and let $X \subseteq M$. Then:

- $X$ is an independent set in $M$ if and only if $X$ is a $P$-independent subset of $\mathbb{F}_q^m$;
- $\text{cl}(X)$ is equal to the $P$-closure of $X$;
- $\deg(f_X)$ is a rank function on $M$.

**Theorem 6.1.14.** $M = (\mathbb{F}_q^m,\mathcal{I})$ is an $\mathbb{F}_q$-representable matroid.
Proof. Fix a basis of $\mathbb{F}_{q^m}$ over $\mathbb{F}_q$ and represent each element of $\mathbb{F}_{q^m}$ as a column vector over $\mathbb{F}_q$. Consider a class $C(\gamma^\ell) = \{\alpha_1, \ldots, \alpha_{[m]}\}$. For any $\alpha_i \in C(\gamma^\ell)$, we can find $a_i$ such that $\alpha_i = \gamma^\ell a_i^{q-1}$. Consider the $m \times [m]$ matrix over $\mathbb{F}_q$

$$A = \begin{pmatrix} a_1 & a_2 & \ldots & a_{[m]} \end{pmatrix}.$$ 

By the Structure Theorem, any subset of columns of $A$ are linearly independent over $\mathbb{F}_q$ if and only if the corresponding $\alpha_i$'s are $P$-independent. Thus, the column linear independence structure of $A$ exactly represents the $P$-independence structure of $C(\gamma^\ell)$.

Since union of $P$-independent sets from distinct classes remain $P$-independent, we can consider the following construction. Let $A$ be a $(m(q-1)+1) \times ([m](q-1)+1)$ matrix given by:

$$A = \begin{pmatrix} A_1 & 0 & \ldots & 0 & 0 \\ 0 & A_2 & \ldots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \ldots & A_{q-1} & 0 \\ 0 & 0 & \ldots & 0 & 1 \end{pmatrix},$$

where each $A_\ell = A$ for $0 \leq \ell \leq q-1$, and the last column is a column of $[m](q-1)$ zeros followed by a 1. Clearly, if we associate the columns in $A_\ell$ with the class $C(\gamma^\ell)$ and the last column with the class $C(0) = \{0\}$, then the linear independence structure of the columns of $A$ will correspond to the $P$-independence structure of $\mathbb{F}_{q^m}$. Thus $M = (\mathbb{F}_{q^m}, I)$ is an $\mathbb{F}_q$-representable matroid.

Example 6.1.15. Consider $\mathbb{F}_{16}$ with primitive element $\gamma$ as a root of the primitive polynomial $x^4 + x + 1$, and $\sigma(a) = a^4$. Let $M = (\mathbb{F}_{16}, I)$. Let $\{1, \gamma\}$ be a basis of $\mathbb{F}_{16}$ over $\mathbb{F}_4$, where $\mathbb{F}_4^* = \{1, \gamma^5, \gamma^{10}\}$. Then the vector $(1, \gamma, \gamma^2, \gamma^3, \gamma^4)$ expands into a $2 \times 5$ $A$ matrix.
over $\mathbb{F}_4$ as:

$$A = \begin{pmatrix} 1 & 0 & \gamma^5 & \gamma^5 & 1 \\ 0 & 1 & 1 & \gamma^{10} & 1 \end{pmatrix}$$

The matrix

$$A = \begin{pmatrix} 1 & 0 & \gamma^5 & \gamma^5 & 1 & 0 & 0 & 0 & 0 & 0 \gamma^5 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & \gamma^{10} & 1 & 0 & 0 & 0 & 0 \gamma^5 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & \gamma^5 & \gamma^5 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & \gamma^{10} & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & \gamma^5 & \gamma^5 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & \gamma^{10} & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

is an $\mathbb{F}_4$-representation of $M$.

**Remark 6.1.16.** The representation we gave in the proof of Theorem 6.1.14 is the most “efficient” representation of $M = (\mathbb{F}_q^m, I)$ over $\mathbb{F}_q$ in the sense that the associated $A$ matrix has the smallest dimension over $\mathbb{F}_q$. Indeed, the largest independent set in $M$ has size $m(q-1)+1$, which corresponds to the number of rows of $A$.

### 6.1.3 $\mathcal{F}(\mathbb{F}_q^m)$ Metric Space

Let $\mathcal{F}(\mathbb{F}_q^m)$ denote the set of all flats in the $\mathbb{F}_q^m[x, \sigma]$-matroid. We now show that $\mathcal{F}(\mathbb{F}_q^m)$ is a metric space.

**Theorem 6.1.17.** Define the map

$$d_\mathcal{F} : \mathcal{F}(\mathbb{F}_q^m) \times \mathcal{F}(\mathbb{F}_q^m) \rightarrow \mathbb{N}$$

$$(X, Y) \mapsto r(X \cup Y) - r(X \cap Y).$$
Chapter 6. Matroidal Structure & Application to Network Coding

Then, \( d_\mathcal{F} \) is a metric on \( \mathcal{F}(\mathbb{F}_{q^m}) \).

Proof. Since symmetry and non-negative definiteness are obvious, it suffices to show that \( d_\mathcal{F} \) satisfies the triangle equality. Let \( X, Y, Z \in \mathcal{F}(\mathbb{F}_{q^m}) \). We want to show that

\[
d_\mathcal{F}(X, Y) - d_\mathcal{F}(X, Z) - d_\mathcal{F}(Y, Z) \leq 0.
\]

By Theorem 2.3.7, we know that

\[
d_\mathcal{F}(X, Y) = \deg(f_X) + \deg(f_Y) - 2\deg(f_{X \cap Y}).
\]

Thus,

\[
d_\mathcal{F}(X, Y) - d_\mathcal{F}(X, Z) - d_\mathcal{F}(Y, Z) = \\
= 2\deg(f_{X \cap Z}) + 2\deg(f_{Y \cap Z}) - 2\deg(f_Z) - 2\deg(f_{X \cap Y}) \\
= 2\deg(f_{(X \cap Z) \cup (Y \cap Z)}) + 2\deg(f_{X \cap Y \cap Z}) - 2\deg(f_Z) - 2\deg(f_{X \cap Y}) \\
= 2(\deg(f_{(X \cap Z) \cup (Y \cap Z)}) - \deg(f_Z)) + 2(\deg(f_{X \cap Y \cap Z}) - \deg(f_{X \cap Y})) \leq 0,
\]

since both \( \deg(f_{(X \cup Z) \cap (Y \cup Z)}) - \deg(f_Z) \leq 0 \) and \( \deg(f_{X \cap Y \cap Z}) - \deg(f_{X \cap Y}) \leq 0 \). \(\square\)

Thus \( \mathcal{F}(\mathbb{F}_{q^m}) \) together with the map \( d_\mathcal{F} \) is a metric space. We shall denote it as \( (\mathcal{F}(\mathbb{F}_{q^m}), d_\mathcal{F}) \).

### 6.1.4 \( C(1) \)-submatroid and Projective Geometry

From the matroid representation in Theorem 6.1.14, it is easy to see that any single conjugacy class of \( \mathbb{F}_{q^m} \) is itself a representable matroid. Since all nontrivial classes have the same structure, we shall examine \( C(1) \). Denote \( \mathcal{F}(C(1)) \) as the set of all flats of the \( C(1) \)-submatroid. Clearly the restriction of \( d_\mathcal{F} \) to \( \mathcal{F}(C(1)) \) makes \( (\mathcal{F}(C(1)), d_\mathcal{F}) \) a metric space. We now show the correspondence between \( (\mathcal{F}(C(1)), d_\mathcal{F}) \) and projective
geometry of vector space $\mathbb{F}_{q^m}$ over $\mathbb{F}_q$.

Viewing $\mathbb{F}_{q^m}$ as a vector space over $\mathbb{F}_q$, let $\mathcal{P}(\mathbb{F}_{q^m})$ denote the set of all nontrivial subspaces of $\mathbb{F}_{q^m}$. Then, as shown in [6], the subspace metric, $d_S$, defined for all $V, W \in \mathcal{P}(\mathbb{F}_{q^m})$ as

$$d_S(V, W) = \dim(V + W) - \dim(V \cap W),$$

is a metric on $\mathcal{P}(\mathbb{F}_{q^m})$.

Let $(\mathcal{P}(\mathbb{F}_{q^m}), d_S)$ be the metric space $\mathcal{P}(\mathbb{F}_{q^m})$ with the subspace metric. We arrive at the following correspondence theorem.

**Definition 6.1.18.** Define the extended warping map, $\Phi$, between the metric spaces $(\mathcal{P}(\mathbb{F}_{q^m}), d_S)$ and $(\mathcal{F}(C(1)), d_F)$, via

$$\Phi : \mathcal{P}(\mathbb{F}_{q^m}) \longrightarrow \mathcal{F}(C(1))$$

$$V \longmapsto \{\varphi(a) \mid a \in V \setminus \{0\}\}.$$

**Theorem 6.1.19.** $\Phi$ is a bijective isometry.

**Proof.** We first show the map is injective. Let $V_1, V_2 \in \mathcal{P}(\mathbb{F}_{q^m})$ be such that $V_1 \neq V_2$. Let $a \in V_1 \setminus V_2$. By Proposition 3.1.5 in Chapter 3, it follows that $\varphi(a) \notin V_2$. Therefore $\varphi(a) \in V_1 \setminus V_2$, so $\Phi$ is injective.

For surjectivity, let $\{\alpha_1, \ldots, \alpha_n\}$ be a $P$-basis for a flat in $\mathcal{F}(C(1))$. By the Independence Lemma, there exists $a_1, \ldots, a_n \in \mathbb{F}_{q^m}$ such that $\varphi(a_i) = \alpha_i$ for all $i$, and $\{a_1, \ldots, a_n\}$ is linearly independent over $\mathbb{F}_q$. Thus $(a_1, \ldots, a_n) \in \mathcal{P}(\mathbb{F}_{q^m})$.

To show isometry, note that for $V, W \in \mathcal{P}(\mathbb{F}_{q^m})$, $\dim(V + W) = \dim(V) + \dim(W) - \dim(V \cap W)$. Clearly, $\dim(V) = r(\Phi(V))$. Thus, in light of Theorem 2.3.7, it suffices to show $\dim(V \cap W) = r(\Phi(V) \cap \Phi(W))$. Towards this end, let $a_1, \ldots, a_j$ be a basis for $V \cap W$. Clearly $\varphi(a_1), \ldots, \varphi(a_j) \in \Phi(V) \cap \Phi(W)$. By the Independence Lemma, $\varphi(a_1), \ldots, \varphi(a_j)$ are linearly independent over $\mathbb{F}_q$. Therefore $\dim((\Phi(V) \cap \Phi(W)) = r(\Phi(V) \cap \Phi(W))$. Hence $\Phi$ is a bijective isometry.
are $P$-independent. Thus $\dim(V \cap W) \leq r(\Phi(V) \cap \Phi(W))$. Conversely, if $\alpha_1, \ldots, \alpha_k$ is a $P$-basis for $\Phi(V) \cap \Phi(W)$, then there exist linearly independent $a_1, \ldots, a_k \in V \cap W$. This shows $\dim(V \cap W) \geq r(\Phi(V) \cap \Phi(W))$. Thus, $\dim(V \cap W) = r(\Phi(V) \cap \Phi(W))$. \hfill \Box

### 6.2 Application to Matroidal Network Coding

Network coding, introduced in the seminal paper [38], is based on the simple idea that, in a packet network, intermediate nodes may forward functions of the packets that they receive, rather than simply routing them. Using network coding, rather than just routing, greater transmission rates can often be achieved. In linear network coding, packets are interpreted as vectors over a finite field, and intermediate nodes forward linear combinations of the vectors that they receive. Sink nodes receive such linear combinations, and are able to recover the original message provided that they can solve the corresponding linear system. In random linear network coding (RLNC), the linear combinations are chosen at random, with solvability of the linear system assured with high probability when the underlying field is sufficiently large [39].

As a means of introducing error-control coding in RLNC, recognizing that random linear combinations of vectors are subspace-preserving, Kötter and Kschischang [6] introduced the concept of transmitting information over a network encoded in subspaces. In this framework, the packet alphabet is the set of all vectors of a vector space, and the message alphabet is the set of all subspaces of that space. The source node encodes a message in a subspace and transmits a basis of that space. Each intermediate node then forwards a random linear combination of its incoming packets. Each sink collects incoming packets and reconstructs the subspace that was selected at the transmitter.

Gadouleau and Goupil [40] generalized the subspace framework to a matroidal one. In this framework, the packet alphabet is the ground set of a matroid, and the message alphabet is the set of all flats of that matroid. The source node encodes a message in a flat...
of the matroid and transmits a basis of that flat. Each intermediate node then forwards a random element of the flat generated by its incoming packets. Each sink collects incoming packets and reconstructs the flat that was selected at the transmitter. In our work, we will use the $\mathbb{F}_{q^m}[x; \sigma]$-matroid in this matroidal network coding framework.

### 6.2.1 Communication using $\mathbb{F}_{q^m}[x; \sigma]$-matroid

We first consider using only the $C(1)$-submatroid. The setup can be summarized as the following.

- The packet alphabet is $C(1)$ and the message alphabet is $\mathcal{F}(C(1))$.
- The source node encodes a message into a flat $\Omega$ of $C(1)$ and sends a basis of $\Omega$.
- An intermediate node receives $\alpha_1, \ldots, \alpha_h \in \Omega$ and forwards a random root of the minimal polynomial $f_{\{\alpha_1, \ldots, \alpha_h\}} \in \mathbb{F}_{q^m}[x; \sigma]$.
- Each sink node collects sufficiently many packets to generate $\Omega$.

**Remark 6.2.1.** As a consequence of Theorem 6.1.19, this $C(1)$-submatroid communication model has the same message alphabet size as the subspace communication model and has the packet size of the projective network coding model in [40].

We can extend the message alphabet size in the $C(1)$-submatroid setup as follows.

- The message alphabet is

  $$\mathcal{F}(C(1)) \cup \mathcal{F}(C(\gamma)) \cup \ldots \cup \mathcal{F}(C(\gamma^{q-2}))$$

  and the packet alphabet is $\mathbb{F}_{q^m}^*$.

- The source node encodes a message into a flat $\Omega_\ell \in \mathcal{F}(C(\gamma^\ell))$. 
• An intermediate node receives $\alpha_1, \ldots, \alpha_h \in \Omega_\ell$ and forwards a random root of the minimal polynomial $f_{\{\alpha_1, \ldots, \alpha_h\}} \in \mathbb{F}_q[x; \sigma]$.

• Each sink node collects sufficiently many packets to generate $\Omega_\ell$.

This setup increases the message alphabet size by a factor of $q - 1$.

Remark 6.2.2. In both cases above, we could have included the $C(0)$-submatroid. This amounts to sending the zero packet at the source, which each intermediate node simply forwards.

6.2.2 Computational Complexity

The computation at an intermediate node in $\mathbb{F}_q^m[x; \sigma]$ matroid network coding is considerably more complex than that of subspace transmission. In the latter case, an intermediate node simply needs to compute a random linear combination of the incoming packets. In the $\mathbb{F}_q^m[x; \sigma]$-matroidal scheme, an intermediate node must forward a random root of the minimal polynomial of its incoming packets. Following the Structure Theorem, this can be accomplished as follows.

Let $\alpha_1, \ldots, \alpha_h \in C(\gamma^{\ell})$ be the incoming packets at an intermediate node. Note that all incoming packets are elements of the same class; the intermediate node must first determine this class (which we call the Class Membership problem). Next, the intermediate node can find $a_i \in \mathbb{F}_q^m$ such that $a_i^{q-1} = \alpha_i \gamma^{-\ell} \in C(1)$ for $i = 1, \ldots, h$ (which we call the Root Finding problem). Finally, the intermediate node can compute a random nonzero $\mathbb{F}_q$-linear combination $a \in \langle a_1, \ldots, a_h \rangle$, and then forward $\alpha = \gamma^{\ell}a^{q-1} \in C(\gamma^{\ell})$. Since the complexity of the last two tasks is well-known, we shall focus on the complexity of the first two.
Class Membership

Without loss of generality, we focus on the first received packet $\alpha_1 \in C(\gamma^\ell)$. It holds that $\alpha_1 = \gamma^\ell a_1^{q-1}$ for some $a_1 \in \mathbb{F}_q^m$. It is possible to isolate the parameter $\ell$ by using the following exponentiation:

$$\alpha_1^m = \gamma^\ell a_1^{(q-1)m} = (\gamma^m)^\ell \in \mathbb{F}_q^*.$$  

The class membership problem can then be solved by means of an exponentiation by $[m]$ and the use of a look-up table for a reasonably small parameter $q$.

Root Finding

We propose two different approaches. The first one is general and based on solving a multivariate linear system of equations over $\mathbb{F}_q$, while the second method is more efficient, but only works in specific field extensions.

Method 1  For $\alpha \in C(1) \subset \mathbb{F}_q^m$, we can compute a $(q - 1)$-th root of $\alpha$ by solving the equation $x^{q-1} - \alpha = 0$. This is equivalent to finding a nonzero root of the polynomial $x^q - \alpha x$. Since $x^q - \alpha x$ is a linearized polynomial, this amounts to solving a linear system with $m$ equations over $\mathbb{F}_q$; using Gaussian elimination this can be done using $O(m^3)$ operations over $\mathbb{F}_q$.

Method 2  Let $\mathbb{F}_q^m$ be an extension of $\mathbb{F}_q$ such that $\gcd([m], q-1) = 1$. Given $\alpha = a^{q-1} \in \mathbb{F}_q^m$, find $t$ such that $(q-1)t = 1 \mod [m]$, and compute $\alpha^t = a^{(q-1)t} = a$. Note that $q-1$ is invertible modulo $[m]$ if and only if $\gcd([m], q-1) = 1$. Thus, our condition on the field extension size is necessary. Furthermore, $t$ can be precomputed since the field extension is fixed. Computing $\alpha^t$ takes $O(\log t)$ multiplications in $\mathbb{F}_q^m$. Assuming each multiplication is $O(m \log m)$ complexity in $\mathbb{F}_q$, the overall algorithm takes $O((\log t)m \log m)$ complexity.
6.3 Summary

In this chapter, we revisited the Structure Theorem from Chapter 3. The Structure Theorem allows us to construct the $\mathbb{F}_q^m[x;\sigma]$-matroid. Using a decomposition theorem for minimal polynomials, we showed that the $\mathbb{F}_q^m[x;\sigma]$-matroid is a metric space. Furthermore, the $C(1)$-submatroid is bijectively isometric to projective geometry of $\mathbb{F}_q^m$ equipped with the subspace metric. Using this isometry, we showed that the $\mathbb{F}_q^m[x;\sigma]$-matroid can be used in a matroidal network coding framework.
Chapter 7

Conclusion

The evaluation of regular polynomials and linearized polynomials have both found important applications in coding theory. As skew polynomial rings are considered to be the most general polynomial rings over a field, it is natural to investigate the potential applications of skew polynomial evaluation. This thesis contributed to this study.

First, we defined a set of simplified notations that allow for concise definitions and proofs. Our notation makes it easy to draw parallels between evaluation of skew polynomials and evaluation of linearized polynomials. The connection between the two is summarized through an important Structure Theorem that is used throughout this thesis.

Next, we defined Generalized Skew Evaluation codes. An important subset of GSE codes are the Generalized Skew Reed-Solomon codes. On one hand, GSRS codes can be viewed as an analogue to the traditional Reed-Solomon codes, as both satisfy the MDS property. On the other hand, GSRS code is a generalization of Gabidulin codes as it encompasses Gabidulin codes as a special case. Understanding the structure properties of GSRS codes allow us to design a pasting construction that encapsulates both MDS and MRD properties in an interesting way. Furthermore, our general definition allows us to set up a duality theory. Our duality theory is akin to the duality theory of Generalized
Chapter 7. Conclusion

Reed-Solomon codes. Interestingly, our theory explicitly shows a duality between $\sigma_s$-conjugacy classes that arise from skew polynomial evaluation.

In order to decode GSRS codes, we examined polynomial interpolation for skew polynomials. We generalized the setup of Kötter interpolation to skew polynomial rings. In our generalization, we put forth a systematic mathematical framework that is rigorous, and also encompasses previous work as special cases. Our Kötter interpolation algorithm can be adapted to two important applications. The first is a generalization of the existing Newton interpolation algorithm that relaxes the constraints on interpolation points, and reduces the overall computational complexity. The second application is a Berlekamp-Welch style decoder for the GSRS codes, which shows GSRS codes have an efficient practical decoder.

Taking a different perspective on skew polynomials by focusing on the independence structure, we examined the connection between $P$-independence and matroid theory. We constructed a new matroid called the $\mathbb{F}_{q^m}[x;\sigma]$-matroid. Both this matroid and some special submatroids have many interesting properties. In particular, the collection of its flats forms a metric space, and induces a metric on submatroids. Using submatroids as metric spaces, we present an application of the $\mathbb{F}_{q^m}[x;\sigma]$-matroid in the context of network coding.

7.1 Future Work

There are a wealth of potential future research directions involving skew polynomial rings in coding theory. We will outline a few that are the most immediate extensions to our work.

In the context of Generalized Skew Evaluation codes, it is important to investigate the potential practical applications that can arise from our pasting construction of codes. One can also study other code designs that come from the interplay between MDS and
MRD properties of GSRS codes. An essential theoretical question is to address the possibility of extending our duality theory. So far our duality theory is only developed for a subclass of GSRS codes. It is of great theoretical interest to understand if a general duality theory for GSRS codes exists.

Another direction concerning GSRS codes is to examine the possibility of performing list-decoding of such codes. We have attempted to fit Sudan’s list-decoding approach [3] in the context of GSRS code. The main difficulty lies in defining an appropriate evaluation map on a bi-variate skew polynomial ring. We have shown that simple extension of the uni-variate skew polynomial ring evaluation is not a good choice. In particular, we could not find an evaluation map on a pair \((x, y)\) that is compatible with the substitution map \((x, f(x))\) (replacing \(y\) by \(f(x)\)). The fact that the regular polynomial evaluation map is compatible with substitution is a key step in establishing list-decoding. Another approach of list-decoding that we attempted is in the same spirit as the folded Reed-Solomon construction in [41]. However, our result here is that the list size is exponential, and hence not very useful. It will be very interesting study the list-decoding problem further and examine if any good list-decoding approach is possible for GSRS codes.

In Kötter interpolation, it is natural to ask if there are other interesting applications in the skew polynomial framework that one can establish beyond the two that we described in this thesis. Furthermore, our setup with free modules over skew polynomial rings is perhaps not the most general one. We have done some preliminary investigation into generalizing the framework further to capture the structure of graded rings. However, a general theory is still lacking. It will be interesting to understand the most general framework in which Kötter interpolation can be applied, and the potential new applications that come with such generalization.

With regard to matroid theory, there are several important questions regarding the \(\mathbb{F}_{q^n}[x; \sigma]\)-matroid that are worth investigating. While we found an application for certain submatroids of \(\mathbb{F}_{q^n}[x; \sigma]\)-matroid, we have not found a useful application for the full
\( \mathbb{F}_{q^m}[x; \sigma] \)-matroid. The potential applications might not be related to coding theory. It is also worthwhile to understand if the \( \mathbb{F}_{q^m}[x; \sigma] \)-matroid has any properties that can be interesting additions to matroid theory.
Bibliography

[1] I. S. Reed and G. Solomon, “Polynomial codes over certain finite fields,” *Journal of


in random network coding,” *Information Theory, IEEE Transactions on*, vol. 54,


