Technische Universität München

TUM School of Computation, Information and Technology

Channel Coding based on Skew Polynomials and Multivariate Polynomials

Hedongliang Liu

Vollständiger Abdruck der von der TUM School of Computation, Information and Technology der Technischen Universität München zur Erlangung einer

Doktorin der Ingenieurwissenschaften

genehmigten Dissertation.

Vorsitz:		Prof. Dr.-Ing. Wolfgang Kellerer
Prüfende der Dissertation:	1.	Prof. Dr.-Ing. Antonia Wachter-Zeh
	2.	Prof. Felice Manganiello, Ph.D.

Die Dissertation wurde am 14.02.2024 bei der Technischen Universität München eingereicht und durch die TUM School of Computation, Information and Technology am 25.06.2024 angenommen.

Abstract

The exponential growth of data generated nowadays has created a high demand for novel solutions to increase efficiency in communication networks and the reliability of large-scale storage systems. Error-correcting codes with related properties have been studied intensively in recent years. Error correction is also essential for the development of quantum computers that can run useful algorithms with negligible miscalculation rate.

This dissertation considers new constructions and decoding approaches for error-correcting codes based on non-conventional polynomials, with the objective of providing new coding solutions to the applications mentioned above.

With skew polynomials, we construct codes that are dual-containing, which is a desired property of quantum error-correcting codes. By considering evaluation codes based on skew polynomials, a condition on the existence of optimal support-constrained codes is derived and an application of such codes in the distributed multi-source networks is proposed. For a class of multicast networks, the advantage of vector network coding compared to scalar network coding is investigated.

Multivariate polynomials have been attracting increasing interest in constructing codes with repair capabilities by accessing only a small amount of available symbols, which is required to build failure-resistant distributed storage systems. A new class of bivariate evaluation codes and their local recovery capability are studied. Interestingly, the well-known Reed-Solomon codes are used in a class of locally recoverable codes with *availability* (multiple disjoint recovery sets) via subspace design.

Aside from new constructions, decoding approaches are considered in order to increase the error correction capability in the case where the code is fixed. In particular, new lower and upper bounds on the success probability of joint decoding interleaved *alternant* codes by a syndrome-based decoder are derived, where alternant codes are an important class of algebraic codes containing Goppa codes, BCH codes and Reed-Muller codes as sub-classes.

Acknowledgments

This dissertation is based on the research I conducted during my doctoral studies at the Institute for Communication Engineering (ICE) at TUM, within the Coding and Cryptography group led by Antonia Wachter-Zeh. I am deeply grateful to have had the opportunity to be part of such an inspiring and motivating research environment, and I wish to express my heartfelt thanks to everyone who contributed to this incredible journey.

First and foremost, I would like to express my deepest gratitude to my advisor, Antonia, for her exceptional supervision throughout my doctoral training. Antonia has provided me with the freedom to explore a wide range of research topics, fostering my scientific curiosity. Her passion for initiating new collaborations and her insightful guidance have been instrumental in my growth, both academically and personally. I am especially thankful for her continued support from my Master's studies through to my PhD, her encouragement during challenging moments, and her exemplary role as a researcher, teacher, and leader.

I would also like to extend my sincere appreciation to Moshe Schwartz and Hengjia Wei for their invaluable guidance and collaboration throughout our collaborative projects. Working with them has been an enriching experience, and their knowledge and insights have been vital to my research. I am particular grateful for the warm hospitality from and the fruitful discussions with Moshe, Hengjia and Han Cai during my visit to Ben-Gurion University of Negev, as well as Felix Ulmer, Pierre Loidreau and Delphine Boucher during my stay in Université de Rennes 1.

I want to thank Sven Puchinger for his mentorship and all his insightful inputs through our collaborations in the early stage of my PhD. I would also like to thank Felice Manganiello for serving as my second examiner, and Wolfgang Kellerer for chairing the examination committee.

Over the past four years, I have had the privilege of collaborating with many talented researchers, and these joint efforts have been both joyful and intellectually rewarding. In particular, I would like to thank my co-authors: Alessandro Neri, Alexander Zeh, Anmoal Porwal, Antonia Wachter-Zeh, Chih-Chang Huang, Cornelia Ott, Frank Kschischang, Felix Ulmer, Georg Maringer, Hannes Bartz, Hengjia Wei, Hugo Sauerbier-Couvée, Ilya Vorobyev, Johan Rosenkilde, Julian Renner, Lukas Holzbaur, Marvin Xhemrishi, Moshe Schwartz, Nikita Polianskii, Rawad Bitar, Sabine Pircher, Sven Puchinger, Tim Janz, Thomas Jerkovits, Violetta Weger, and Volodya Sidorenko. Collaborating with all of you has expanded my knowledge in many meaningful ways.

I am also grateful to Violetta Weger and Stefan Ritterhoff for their help in proofreading parts of this dissertation.

To my office mates, Lorenz Welter, Marvin Xhemrishi, Sven Puchinger, Sabine Pircher, and Stefan Ritterhoff, thank you for the camaraderie and the positive atmosphere we shared. I would also like to acknowledge all members of the institute for creating such a welcoming and vibrate environment. Many thanks to the professors and organizers of events like JWCC, doctoral seminars, and the Instituteausflug – these events have enriched my experience and

fostered a spirit of collaboration. Outside of work, I will always cherish the memories of after-work gatherings, board games and weekend brunches, which added balance to my PhD life.

To my friends Hongdou, Shelton, Johnathan, Tobit, Keyue, Yanqin, and many others in Munich, thank you for pulling me out of the office and home-office from time to time, and for the unforgettable trips we shared. Han and Lisa, your constant support and generous hospitality during my visits to France and Austria have meant so much to me. Chen, Shiying, and Xuanxuan, thank you for our long-distance calls, deep conversations, and the warm welcomes whenever I returned to China—your friendship has been a source of strength throughout this journey.

Lastly, I am deeply thankful to my family, especially my mom, for her unwavering love, support, and care, and my dad, for always being my role model. Their dedication to my education and constant encouragement have been the foundation for everything I have accomplished.

Hedongliang Liu
Munich, September 2024

Contents

Nomenclature

Sets, Rings and Fields

$[a, b]$	Set of integers $\{i \mid a \leqslant i \leqslant b\}$
$[b]$	Set of integers $\{i \mid 1 \leqslant i \leqslant b\}$
$\mathcal{A}$ or $\mathcal{R}$	Ring
$\mathbb{Z}_p$	Integer ring of order p
F	Field (may be infinite)
$\mathbb{F}$ (or $\mathbb{F}_q$)	Finite field (with q elements)
$\mathbb{F}_{q^m}$	Extension field with q^m elements

Vectors, Matrices and Vector Spaces

$\boldsymbol{a}$	Vector
$\mathrm{supp}(\boldsymbol{a})$	Set of indices of nonzero positions of $\boldsymbol{a}$
$\boldsymbol{a}\vert_{\mathcal{I}}$	Vector $\boldsymbol{a}$ restricted to the positions indexed by $\mathcal{I}$
$\boldsymbol{a} \star \boldsymbol{b}$	Entry-wise multiplication of vectors $\boldsymbol{a}$ and $\boldsymbol{b}$
$\mathrm{diag}(\boldsymbol{a})$	Diagonal matrix with the entries of $\boldsymbol{a}$ on its diagonal
$\boldsymbol{A}$	Matrix
$\boldsymbol{A}^\top$	Transpose of the matrix $\boldsymbol{A}$
$\mathrm{supp}(\boldsymbol{A})$	Set of indices of nonzero columns of $\boldsymbol{A}$
$\boldsymbol{A}\vert_{\mathcal{I}}$	Matrix $\boldsymbol{A}$ restricted to the columns indexed by $\mathcal{I}$
$\boldsymbol{a}_i$	The i-th row of $\boldsymbol{A}$
$\langle \boldsymbol{A} \rangle$	Row span of $\boldsymbol{A}$
$\mathcal{G}_q(n, k)$	Grassmannian of dimension k of $\mathbb{F}_q^n$ (i.e., the set of all k-dimensional subspaces of $\mathbb{F}_q^n$)

Codes

$\mathcal{C}$, $\mathcal{C}^\perp$	Code (a set of vectors/matrices), Dual code
$[n, k]_q$	A $\mathbb{F}_q$-linear code of length n and dimension k
$[n, k, d]_q$	A $\mathbb{F}_q$-linear code of length n, dimension k and minimum distance d
d_{H}	Minimum Hamming distance
d_{R}	Minimum rank distance
$\mathrm{d}_{\mathsf{SR},\boldsymbol{n}_\ell}$	Minimum sum-rank distance w.r.t. an ordered partition $\boldsymbol{n}_\ell$ of n
$\boldsymbol{G}$	Generator matrix of a code
$\boldsymbol{H}$	Parity-check matrix of a code

Polynomials

$\mathcal{A}[x]$ or $\mathbb{F}[x]$	Univariate polynomial ring in x over a ring $\mathcal{A}$ or a finite field $\mathbb{F}$
$\mathcal{R}_n$	Multivariate polynomial ring in n variables
$I = \langle f_1, \ldots, f_s \rangle$	Ideal generated by $f_1, \ldots, f_s$
θ	Endomorphism
δ	θ-derivation
$\mathcal{A}[X; \theta, \delta]$	Skew polynomial ring in variable X over a ring $\mathcal{A}$
gcrd	Greatest common right divisor
lclm	Least common left multiple

Abbreviations

AAD	Almost affinely disjoint
AS	Almost sparse
BCH	Bose–Chaudhuri–Hocquenghem
GRS	Generalized Reed–Solomon
i.i.d.	Independently and identically distributed
LRS	Linearized Reed-Solomon
MDS	Maximum distance separable
MRD	Maximum rank distance
MSRD	Maximum sum-rank distance
QEC	Quantum error-correcting
QLRS	Quadratic lifted Reed-Solomon codes
RM	Reed–Muller
RS	Reed–Solomon

List of Figures

List of Tables

1
Motivation and Overview

Channel coding originates from the seminal work by Shannon [Sha48], which laid the mathematical foundation of *reliable communication in the presence of noise*. The channel coding theorem by Shannon shows that reliable communication is achievable as long as the code rate is below the capacity of the (noisy) channel. However, the proof of this result is non-constructive and focuses on the asymptotic behavior of error-correcting codes in a probabilistic setting. On the contrary, the work by Hamming [Ham50] around the similar time extracted the combinatorial basis for the theory of error-correcting codes. Their works are deeply intertwined and perfectly complementary. Due to the clear difference between the probabilistic, asymptotic viewpoint of Shannon and the combinatorial, constructive perspective of Hamming, prosperous studies following their footprints are growing into two main respective areas[1]: *information theory* and *coding theory*. The former focuses on characterizing the capacity (i.e., asymptotically achievable code rate) for various channel models with different statistical behaviors, while the cornerstone of the latter is the finite behavior of codes for scenarios where error correction is needed.

Many well-known and widely used codes are based on polynomials. For instance, Reed-Muller codes, used in deep-space communication [Mas92], wireless communications [Ari08; MHU14] and probabilistic checkable proofs in computational complexity theory [AS97; STV99], can be described as low-degree multivariate polynomials. BCH codes, used in satellite communication [CP88] and solid-state drives [MME$^+$13], are suitable for implementations on small and low-power hardwares because of their underlying polynomial structure.

Nowadays, error correction is not only used for communications, but also in abundant scenarios ranging from digital data storage in the daily life to the frontier research on quantum computing.

As the exponential growth of data generated and exchanged nowadays, large-scale distributed storage systems are needed to store vast amounts of data. The main goal of such systems is to guarantee the integrity of the stored data, i.e., to protect the data from loss even if some storage disks are defective. Instead of simple replication, several distributed storage systems have utilized error-correcting codes to provide reliable services, e.g., Facebook's f4 storage system [MLR$^+$14], Baidu's Atlas Cloud Storage [LJY$^+$15], Hadoop [Fou17] and Backblaze Vaults cloud storage [Bea19] use Reed-Solomon codes.

In quantum computing, a *qubit* is the basic unit of quantum information that can carry richer states beyond just 0 and 1. The challenge is that the qubits are so sensitive that

[1] We refer the interested reader to [Sle74] and [Ber74] for the influential papers in the development of the respective areas.

even stray light or slight temperature change can cause errors [CLSZ95]. The state-of-the-art quantum processors typically have error rates around 10^{-3} per interaction between *physical* qubits [FND+20; WBC+21], which is far beyond the error rate required to run useful algorithms. Quantum error-correcting (QEC) codes are proposed to suppress error rates of calculation by constructing *logical* qubits, where each logical qubit is composed of multiple physical qubits (i.e., by adding redundancy to reduce the error rate per logic operation). *Surface codes* [Kit03] have been thoroughly studied for QEC architectures and have been demonstrated in small examples by teams at IBM [CYK+22; SYK+23] and Google Quantum AI [Goo23]. However, surface codes have the drawback that they require too many physical qubits, possibly 200 million qubits for problems of interest, which makes them impractical due to the cost and complexity. Recently, low-density-parity-check (LDPC) codes were proposed as a promising candidate for QEC codes as they feature a more than ten-fold reduction in the number of physical qubits compared to surface codes under similar error rate level [BCG+23]. Advancements in finding codes with better code rate and fast decoding algorithms suitable for quantum circuits are still highly demanded.

This dissertation intends to provide new constructions from non-conventional polynomials and decoding approaches for error-correcting codes with the desired properties in the aforementioned applications. The structure of this dissertation is as follows.

Chapter 2 provides the basics of the polynomials and the metrics used in the remaining chapters. We first give a brief introduction of multivariate polynomials and present a powerful tool, Gröbner basis, for solving polynomial equation systems. We then introduce skew polynomials and their properties. Finally, we cover the Hamming, the rank and the sum-rank metrics.

In **Chapter 3** we construct *dual-containing* codes over rings based on skew polynomials. We first define (θ, δ)-*polycyclic codes* (in short, (θ, δ)-codes) and derive a parity-check matrix of this class of codes within the framework of skew polynomials. Based on the properties of skew polynomials and dual-containing codes, we develop an algorithm to compute all Euclidean-/Hermitian-dual-containing (θ, δ)-codes constructed from skew polynomials and apply this algorithm to several rings $\mathcal{A}$ of order 4. Moreover, we give an algorithm to test whether the dual code is also a (θ, δ)-code and apply it to the resulting dual-containing codes found by the previous algorithm.

Chapter 4 is devoted to a class of evaluation codes of skew polynomials, *linearized Reed-Solomon (LRS)* codes, and network coding. LRS codes are *maximum sum-rank distance (MSRD)* codes. Motivated by the practical and theoretical interest in support-constrained codes, we derive a necessary and sufficient condition on the existence of an MSRD code fulfilling certain support constraints and give an upper bound on the field size to construct such a code. With the help of the condition, we develop a scheme to design *distributed LRS codes* for multi-source networks. The second focus of this chapter is the advantage of vector network coding versus scalar network coding for a family of multicast networks, *generalized combination networks*. The task of this multicast network coding problem is to find the coding coefficients of the relay nodes at the middle layer of the network, so that all the receivers that connect to a fixed number of middle layer nodes can decode all the messages. The *solution* of such a network is the set of coding coefficients at each relay nodes. We investigate the advantage by bounding the gap between the minimum required alphabet size of the scalar solutions and the vector solutions.

Chapter 5 deals with codes with local properties constructed from multivariate polynomials. We first propose a class of bivariate evaluation codes, so called *quadratic lifted Reed-*

Solomon (QLRS) codes, where the codeword symbols whose coordinates lie on a quadratic curve form a local recovery set, so that any missing symbol in the set can be recovered within the set. We study the dimension and minimum Hamming distance of the QLRS codes and compare them to other multivariate evaluation codes, *linearized Reed-Solomon codes*, in terms of the performance in local recovery. As the second part of this chapter, we investigate an *almost affinely disjoint* (AAD) family of subspaces which is motivated by *batch* codes, a class of locally recoverable codes with availability. The subspaces in an AAD family form a partial spread where any affine transformation of any subspace in the family intersects with only a few other subspaces in the family. We give a construction for the AAD family using the best-known evaluation codes – Reed-Solomon codes. Aside from the explicit construction, we also provide upper and lower bounds on the cardinality of this family.

Chapter 6 concerns joint decoding of *interleaved* evaluation codes. A codeword of an interleaved code can be seen as s parallel codewords from linear codes. When s additive errors have a common support with restricted size, we can decode beyond half the minimum Hamming distance of the code with high probability. *Alternant codes* are subfield subcodes of Reed-Solomon. They contain Goppa codes and BCH codes as sub-classes. We apply the Schmidt-Sidorenko-Bossert joint decoding algorithm, which is known for decoding interleaved Reed-Solomon codes, to interleaved alternant codes, and derive a necessary and sufficient condition such that this algorithm succeeds. Based on this condition, we derive lower and upper bounds on the success probability of decoding interleaved alternant codes by the Schmidt-Sidorenko-Bossert decoder. Moreover, we briefly summarize the results on joint decoding of interleaved *generalized Goppa codes* and on improvements in decoding radius by utilizing list decoding for interleaved alternant codes.

2

Introduction to Codes based on Polynomials

Polynomials were firstly considered for error control by David E. Muller [Mul54] for simplifying switching circuits with multiple outputs via *polynomial representations* in *Boolean algebra* and by Irving S. Reed [Ree54], who exhibited the ability of Muller's polynomial codes to correct multiple errors and proposed the first efficient decoding algorithm. Reed-Muller codes can be described as evaluations of low-degree multivariate polynomials. Their works not only constructed one of the oldest classes of codes that have been extensively studied, but also brought some preliminary indications about a large body of knowledge about finite algebraic structures (rings, fields, vector spaces) that could be used for error correction. Since then, various well-known code constructions based on polynomials have appeared. For instance, Reed-Solomon codes [RS60] can be seen as a set of low-degree univariate polynomials and BCH codes [Hoc59; BR60] can be seen as principle ideals in a quotient polynomial ring.

This chapter gives an introduction to codes constructed from polynomials. Section 2.1 provides basic notations used in this thesis. Sections 2.2 and 2.3 contain the basics on multivariate polynomials and skew polynomials that concern most of this thesis. Section 2.4 provides two methods of constructing linear block codes from polynomials. In Section 2.5, we present three metrics for measuring the error-correction capability of a code.

2.1 Basic Notations

Denote by $[a, b]$ the set of integers $\{a, a+1, \ldots, b-1, b\}$, and $[b] := [1, b]$. Let $\mathbb{N}$ be the set of nonnegative integers. For any set $\mathcal{A}$, denote by $\mathcal{A}^* := \mathcal{A} \backslash \{0\}$ the set of all the nonzero elements in $\mathcal{A}$. A ring $\mathcal{A}$ is *unitary* if there exists $1 \in \mathcal{A}^*$, such that $1 \cdot a = a \cdot 1 = a, \forall a \in \mathcal{A}$. A ring $\mathcal{A}$ is *commutative* if $ab = ba, \forall a, b \in \mathcal{A}$. Denote by $\mathbb{F}$ a finite field, by $\mathbb{F}_q$ a finite field of size q, and by $\mathbb{F}_{q^m}$ the extension field over $\mathbb{F}_q$ of extension degree m. The integer ring of size q is denoted by $\mathbb{Z}_q$. Note that for a prime p, $\mathbb{Z}_p = \mathbb{F}_p$. Denote by $[n, k]_q$ a linear block code of length n and dimension k over an alphabet of size q. If the minimum distance d of the code is also of importance, we denote the code by $[n, k, d]_q$.

Given two vectors $\boldsymbol{a} = (a_1, \ldots, a_n), \boldsymbol{b} = (b_1, \ldots, b_n)$, we denote the entry-wise multiplication of $\boldsymbol{a}$ and $\boldsymbol{b}$ by $\boldsymbol{a} \star \boldsymbol{b} := (a_1 b_1, a_2 b_2, \ldots, a_n b_n)$. For a vector $\boldsymbol{a}$ of length n, we denote by $\mathrm{supp}(\boldsymbol{a})$ the set of indices of the nonzero entries of $\boldsymbol{a}$ and by $\mathrm{diag}(\boldsymbol{a})$ the $n \times n$ diagonal matrix with the entries of $\boldsymbol{a}$ on its diagonal. Given a set $\mathcal{E} \subseteq [n]$, we denote by $\boldsymbol{a}|_{\mathcal{E}}$ the restriction of $\boldsymbol{a}$ to the entries indexed by the set $\mathcal{E}$. For an $m \times n$ matrix $\boldsymbol{E}$, we denote by $\mathrm{supp}(\boldsymbol{E})$ the set of indices of the nonzero columns of $\boldsymbol{E}$ and by $\boldsymbol{e}_i$ the i-the row of $\boldsymbol{E}$. Given a set $\mathcal{E} \subseteq [n]$, we denote by $\boldsymbol{E}|_{\mathcal{E}}$ the restriction of $\boldsymbol{E}$ to the columns indexed by $\mathcal{E}$.

Denote by $\mathcal{G}_q(n,k)$ the *Grassmannian* of dimension k, which is a set of all k-dimensional subspaces of $\mathbb{F}_q^n$. The cardinality of $\mathcal{G}_q(n,k)$ is the well-known *q-binomial coefficient*:

$$|\mathcal{G}_q(n,k)| = \begin{bmatrix} n \\ k \end{bmatrix}_q := \prod_{i=0}^{k-1} \frac{q^n - q^i}{q^k - q^i} = \prod_{i=0}^{k-1} \frac{q^{n-i} - 1}{q^{k-i} - 1} \ .$$

Given a ring $\mathcal{A}$, we denote by $\mathcal{A}[x]$ the univariate commutative polynomial ring in variable x with coefficients from $\mathcal{A}$. The *degree* of a nonzero polynomial $f = \sum_{i \in \mathbb{N}} a_i x^i \in \mathcal{A}[x]$ is $\deg(f) := \max\{i \in \mathbb{N} \mid a_i \neq 0\}$. We use the convention that the degree of a zero polynomial is defined as $-\infty$. The leading coefficient of f is denoted by $\mathrm{lc}(f)$. A polynomial $f \in \mathcal{A}[x]$ is *monic* if $\mathrm{lc}(f) = 1$. For $g, f \in \mathcal{A}[x]$, we denote by $g \mid f$ if g divides f, by $\mathrm{lcm}(g,f)$ the least common multiplier of g and f, and by $\gcd(g,f)$ the greatest common divisor of g and f.

Throughout the thesis, the indices start from 1. For convenience, the indices of the coefficients of polynomials start from 0. Hence, for vectors associated with polynomials, the indices of their entries are corresponding to the polynomials, i.e., the coefficient vector $(a_0, a_1, \ldots, a_d)$ corresponds to the polynomial $f = \sum_{i=0}^{d} a_i x^i$.

2.2 Multivariate Polynomials

In this thesis, we use multivariate polynomials in a wide range. Hence, we introduce the basics of multivariate polynomials and a powerful tool – *Gröbner bases* – in solving polynomial equations in this section.

Given a field F (may be infinite), we denote by $\mathcal{R}_n = F[x_1, \ldots, x_n]$ the commutative polynomial ring in n variables over F. We associate a vector $\boldsymbol{d} = (d_1, \ldots, d_n) \in \mathbb{N}^n$ to the exponents of a monomial by

$$\boldsymbol{x}^{\boldsymbol{d}} = x_1^{d_1} x_2^{d_2} \cdots x_n^{d_n} \in \mathcal{R}_n \ .$$

Definition 2.1 (Monomial order)**.** *A* monomial order *in $\mathcal{R}_n = F[x_1, \ldots, x_n]$ is a relation $\prec$ on $\mathbb{N}^n$ such that*

- *for all $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{N}^n$, either $\boldsymbol{a} = \boldsymbol{b}$ or $\boldsymbol{a} \prec \boldsymbol{b}$ or $\boldsymbol{b} \prec \boldsymbol{a}$,*
- *for all $\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c} \in \mathbb{N}^n$, $\boldsymbol{a} \prec \boldsymbol{b} \implies \boldsymbol{a} + \boldsymbol{c} \prec \boldsymbol{b} + \boldsymbol{c}$,*
- *for all $\boldsymbol{a} \in \mathbb{N}^n$, $\boldsymbol{0} \prec \boldsymbol{a}$ or $\boldsymbol{0} = \boldsymbol{a}$.*

The following orders are monomial orders [VG13, Theorem 21.6]:

- *Lexicographic order:*

$$\boldsymbol{a} \prec_{\mathrm{lex}} \boldsymbol{b} \iff \text{the leftmost nonzero entry in } \boldsymbol{a} - \boldsymbol{b} \text{ is negative.}$$

- *Graded lexicographic order:*

$$\boldsymbol{a} \prec_{\mathrm{grlex}} \boldsymbol{b} \iff \sum_{i=1}^{n} a_i < \sum_{i=1}^{n} b_i \text{ or } \left(\sum_{i=1}^{n} a_i = \sum_{i=1}^{n} b_i \text{ and } \boldsymbol{a} \prec_{\mathrm{lex}} \boldsymbol{b} \right) .$$

- *Graded reverse lexicographic order:*

$$\boldsymbol{a} \prec_{\text{grevlex}} \boldsymbol{b} \iff \sum_{i=1}^{n} a_i < \sum_{i=1}^{n} b_i \text{ or } \Big(\sum_{i=1}^{n} a_i = \sum_{i=1}^{n} b_i \text{ and the rightmost nonzero entry in } \boldsymbol{a} - \boldsymbol{b} \in \mathbb{Z}^n \text{ is positive} \Big).$$

Since there are multiple variables in a multivariate polynomial, the properties of a polynomial (such as "degree") are different from those of a univariate polynomial. We give the formal definitions of these properties in the following.

Definition 2.2. *Let F be a field, $\mathcal{R}_n = F[x_1, \ldots, x_n]$ be a polynomial ring, $f = \sum_{\boldsymbol{d} \in \mathbb{N}^n} c_{\boldsymbol{d}} \boldsymbol{x}^{\boldsymbol{d}} \in \mathcal{R}_n$ be a nonzero polynomial with all $c_{\boldsymbol{d}} \in F$, $\boldsymbol{d} \in \mathbb{N}^n$, and $\prec$ be a monomial order.*

- *Each $c_{\boldsymbol{d}} \boldsymbol{x}^{\boldsymbol{d}}$ with $c_{\boldsymbol{d}} \neq 0$ is a* term *of f.*
- *The* (total) degree *of f is* $\deg(f) := \max_{\boldsymbol{d} \in \mathbb{N}^n} \{ \sum_{i=1}^{n} d_i \mid c_{\boldsymbol{d}} \neq 0 \} \in \mathbb{N}$.
- *The* multidegree *of f is* $\operatorname{mdeg}(f) := \max_{\prec} \{ \boldsymbol{d} \mid c_{\boldsymbol{d}} \neq 0 \} \in \mathbb{N}^n$, *where* $\max_{\prec}$ *is the maximum with respect to* $\prec$.
- *The* x_i-degree *of f is* $\deg_{x_i}(f) := \max_{\boldsymbol{d} \in \mathbb{N}^n} \{ d_i \mid c_{\boldsymbol{d}} \neq 0 \} \in \mathbb{N}$.
- *The* leading coefficient *of f is* $\operatorname{lc}(f) := c_{\operatorname{mdeg}(f)} \in F^*$.
- *The* leading monomial *of f is* $\operatorname{lm}(f) := \boldsymbol{x}^{\operatorname{mdeg}(f)} \in \mathcal{R}$.
- *The* leading term *of f is* $\operatorname{lt}(f) := \operatorname{lc}(f) \cdot \operatorname{lm}(f) \in \mathcal{R}$.

Addition, subtraction, multiplication and division between two polynomials in $\mathcal{R}_n$ follow naturally from univariate polynomials. We present in Algorithm 2.1 a division algorithm with multiple divisors, which is used in Buchberger's algorithm (Algorithm 2.2) presented later to compute a Gröbner basis.

Algorithm 2.1: Multivariate division algorithm (cf. [VG13, Algorithm 21.11])

Input: Nonzero polynomials $f, f_1, \ldots, f_s \in \mathcal{R}_n = F[x_1, \ldots, x_n]$ where F is a field; a monomial order $\prec$ on $\mathcal{R}_n$.

Output: Quotients $q_1, \ldots, q_s \in \mathcal{R}_n$ and remainder $r \in \mathcal{R}_n$ such that $f = q_1 f_1 + \cdots + q_s f_s + r$ and $\forall i \in [s]$, $\operatorname{lt}(f_i) \nmid r$.

```
1 r ← 0, p ← f, q_i ← 0, ∀i ∈ [s];
2 while p ≠ 0 do
3     if for some i ∈ [s], lt(f_i) | lt(p) then
4         q_i ← q_i + lt(p)/lt(f_i), p ← p − (lt(p)/lt(f_i)) f_i
5     else
6         r ← r + lt(p), p ← p − lt(p)
7 return q_1, ..., q_s, r
```

The output of this kind of division may not be unique, since there may be more than one $i \in [s]$ such that $\operatorname{lt}(f_i)$ divides $\operatorname{lt}(p)$ at Line 3. In Algorithm 2.1, if we always chooses the

smallest possible i at Line 3, then the quotients $q_1, \ldots, q_s$ and the remainder r, denoted by

$$(q_1, \ldots, q_s) = f \text{ quo } (f_1, \ldots, f_s) ,$$
$$r = f \text{ rem } (f_1, \ldots, f_s) ,$$

are uniquely determined.

Theorem 2.1 (Combinatorial Nullstellensatz [Alo99, Theorem 1.2])**.** *Let F be an arbitrary field and f be a nonzero polynomial in $F[x_1, \ldots, x_n]$ of total degree $\deg(f) = \sum_{i=1}^{n} t_i$, where $t_i \in \mathbb{N}$, $\forall i$. Then, if $\mathcal{X}_1, \ldots, \mathcal{X}_n$ are subsets of F with $|\mathcal{X}_i| > t_i$, then there are $\hat{x}_1 \in \mathcal{X}_1, \ldots, \hat{x}_n \in \mathcal{X}_n$ so that*

$$f(\hat{x}_1, \ldots, \hat{x}_n) \neq 0 .$$

2.2.1 Ideals and Variety

Let $f_1, \ldots, f_s$ be polynomials in $\mathcal{R}_n$. The polynomials generate an *ideal* in $\mathcal{R}_n$

$$I = \langle f_1, \ldots, f_s \rangle := \left\{ \sum_{i=1}^{s} p_i f_i \;\middle|\; p_i \in \mathcal{R}_n \right\} .$$

An ideal is *principle* if it is generated by a single element of the ring. For example, $I = \langle f_1 \rangle \subseteq \mathcal{R}_n$ is a principle ideal.

The *variety* of I is

$$V(I) := \{\boldsymbol{u} \in F^n \mid f(\boldsymbol{u}) = 0, \ \forall f \in I\} = \{\boldsymbol{u} \in F^n \mid f_1(\boldsymbol{u}) = \cdots = f_s(\boldsymbol{u}) = 0\} .$$

We also write $V(f_1, \ldots, f_s)$ instead of $V(\langle f_1, \ldots, f_s \rangle)$ for short. It can be readily seen that the variety $V(I)$ is the set of all solutions to the *system of polynomial equations* $\{f_1 = 0, \ldots, f_s = 0\}$.

Interesting questions about I and $V(I)$ that also concern solving the system of polynomial equations include:

- How "big" is $V(I)$? Is $V(I) \neq \varnothing$?
- Ideal membership problem: given $f \in \mathcal{R}_n$, is $f \in I$?

The famous Hilbert's *Nullstellensatz* [Hil93] says the following: if F is algebraically closed, then, for $f \in I = \langle f_1, \ldots, f_s \rangle$, there is an integer $e \in \mathbb{N}$ such that $f^e \in I$. This implies that for any ideal I over an algebraically closed field, the variety $V(I) = \varnothing$ if and only if $1 \in I$.

For $n = 1$, $\mathcal{R}_1 = F[x]$, the ideal membership problem is easy to check. Let $g = \gcd(f_1, \ldots, f_s)$ be the greatest common divisor of $f_1, \ldots, f_s$. Then, the ideal $I = \langle f_1, \ldots, f_s \rangle = \langle g \rangle$ [AL22, Proposition 1.3.8]. Hence, for any $f \in \mathcal{R}_n$, $f \in I$ if and only if $g \mid f$. For $n \geqslant 2$ and $s = 1$, the ideal membership problem can be solved by Algorithm 2.1: $f \in \langle f_1 \rangle$ if and only if $f \text{ rem } f_1 = 0$. However, this method fails in general for $s \geqslant 2$. The next subsection introduces a special type of bases of an ideal where the ideal membership can be easily (only conceptually, not computationally) determined (see Theorem 2.2). These special bases are the analogue to the greatest common divisor for multivariate polynomials.

2.2.2 Gröbner Bases

A *Gröbner basis* of an ideal I is a special "basis" for I, in which the questions about $V(I)$ mentioned in Section 2.2.1 are easy to answer. Heisuke Hironaka introduced in [Hir64] a special type of basis for polynomial ideals, called "standard basis". Bruno Buchberger invented them independently in his dissertation [Buc65] and named them as Gröbner bases after his advisor Wolfgang Gröbner.

Definition 2.3 (Gröbner basis)**.** *Let $\prec$ be a monomial order and $I \subseteq \mathcal{R}_n$ be an ideal. A finite set $G \subseteq I$ is a* Gröbner basis *for I with respect to $\prec$ if $\langle \mathrm{lt}(G) \rangle = \langle \mathrm{lt}(I) \rangle$, where $\mathrm{lt}(G) := \{\mathrm{lt}(g) \mid g \in G\}$ and $\mathrm{lt}(I) := \{\mathrm{lt}(g) \mid g \in I\}$.*

For a polynomial ring $\mathcal{R}_n = F[x_1, \ldots, x_n]$, every ideal $I \subseteq \mathcal{R}$ has a Gröbner basis [VG13, Corollary 21.26]. The following theorem shows that given a Gröbner basis of an ideal, we can solve the ideal membership problem.

Theorem 2.2 ([VG13, Theorem 21.28])**.** *Let G be a Gröbner basis for the ideal $I \subseteq \mathcal{R}_n$ with respect to a monomial order $\prec$, and $f \in \mathcal{R}_n$. Then*

$$f \in I \iff f \text{ rem } G = 0 \ .$$

To present the Buchberger's algorithm that computes a Gröbner basis of an ideal, we need the following definition of an S-polynomial.

Definition 2.4 (S-polynomial)**.** *Let $g, h \in \mathcal{R}_n$, $\boldsymbol{a} = \mathrm{mdeg}(g)$, $\boldsymbol{b} = \mathrm{mdeg}(h)$, and $\boldsymbol{c} = (\max\{a_1, b_1\}, \ldots, \max\{a_n, b_n\})$. The S-polynomial of g and h is*

$$S(g, h) = \frac{\boldsymbol{x}^{\boldsymbol{c}}}{\mathrm{lt}(g)} g - \frac{\boldsymbol{x}^{\boldsymbol{c}}}{\mathrm{lt}(h)} h \ \in \mathcal{R}_n \ .$$

The following theorem shows the importance of the S-polynomials for computing a Gröbner basis.

Theorem 2.3 ([VG13, Theorem 21.31])**.** *A finite set $G = \{g_1, \ldots, g_s\} \subseteq \mathcal{R}_n$ is a Gröbner basis of the ideal $\langle G \rangle$ if and only if*

$$S(g_i, g_j) \text{ rem } (g_1, \ldots, g_s) = 0 \text{ for } 1 \leqslant i < j \leqslant s \ .$$

We now present a simplified version of Buchberger's algorithm [Buc65] in Algorithm 2.2.

The extended Euclidean algorithm for computing the greatest common divisor (gcd) of univariate polynomials in $F[x]$ is a special case of Buchberger's algorithm. A proof of the correctness of Algorithm 2.2 can be found in [VG13, Theorem 21.34]. In general, the Gröbner basis computed by Buchberger's algorithm is neither unique nor of minimal size. However, one can further process the polynomials in G to obtain a unique *reduced Gröbner basis*, which is defined as follows. Such a unique basis exists for every ideal [VG13, Theorem 21.38].

Definition 2.5. *A subset $G \subseteq \mathcal{R}_n$ is a* minimal *Gröbner basis of $I = \langle G \rangle$ if it is a Gröbner basis for I and for all $g \in G$*

(i) $\mathrm{lc}(g) = 1$,

(ii) $\mathrm{lt}(g) \notin \langle \mathrm{lt}(G \backslash \{g\}) \rangle$.

Algorithm 2.2: Buchberger's algorithm for Gröbner basis computation (cf. [VG13, Algorithm 21.33])

Input: Nonzero polynomials $f_1, \ldots, f_s \in \mathcal{R}_n$, and a monomial order $\prec$.
Output: A Gröbner basis $G \subseteq \mathcal{R}_n$ for the ideal $I = \langle f_1, \ldots, f_s \rangle$ with respect to $\prec$.

```
1  G ← {f_1, ..., f_s};
2  repeat
3      S ← ∅;
4      order the elements in G as g_1, ..., g_t according to ≺;
5      foreach i ∈ [t − 1], j ∈ [i + 1, t] do
6          r ← S(g_i, g_j) rem (g_1, ..., g_t);          /* Apply Algorithm 2.1 */
7          if r ≠ 0 then
8              S ← S ∪ {r}
9      if S = ∅ then
10         return G
11     else
12         G ← G ∪ S
```

An element g of a Gröbner basis G is reduced with respect to G *if no monomial of g is in $\langle \mathrm{lt}(G \backslash \{g\}) \rangle$. A minimal Gröbner basis G of an ideal I is* reduced *if all its elements are reduced with respect to G.*

To understand the complexity of Buchberger's algorithm, we need to know the maximal total degree of a polynomial occurring during the computation of a Gröbner bases and the number of polynomials in the Gröbner basis. The choice of the monomial ordering is critical to these values. Buchberger investigated in [Buc83] the maximal (total) degree and the number of polynomials occurring in a Gröbner basis G of the ideal $I = \langle f_1, \ldots, f_s \rangle$ for a finite set of bivariate polynomials $\mathcal{F} = \{f_1, \ldots, f_s\} \subseteq F[x_1, x_2]$. In the case of $\prec_{\mathrm{grlex}}$, the maximum degree of the polynomials in G is $2 \cdot \max_{f \in \mathcal{F}} \deg(f) - 1$. In the case of $\prec_{\mathrm{lex}}$, the maximum degree of the polynomials in G is $\max_{f \in \mathcal{F}} \deg(f)^2$. For any valid monomial ordering, the number of polynomials in G is $|G| = \min_{f \in \mathcal{F}} \deg(\mathrm{lt}(f)) + 1$. From a practical point of view, in the computation of a Gröbner basis in Magma [BCP97] for instance, $\prec_{\mathrm{grevlex}}$ is recommended for faster computation while $\prec_{\mathrm{lex}}$ is hard for computation, though it usually presents the most information about the ideal.

The worst-case cost of Buchberger's algorithm is still unknown today. Kühnle and Mayr [KM96] presented an algorithm for computing the unique reduced Gröbner basis for a given ideal, which requires exponential space. This gives a lower bound on the worst-case cost of Buchberger's algorithm and concludes that finding a reduced Gröbner basis is an $\mathcal{EXPSPACE}$-complete problem (see [VG13, Section 25.8] for the classification of computation complexities).

In this thesis, we use Gröbner bases for solving a system of equations. Let $f_1, \ldots, f_s$ be polynomials in $\mathcal{R}_n$. The set of solutions to the set of equations $\{f_1 = 0, \ldots, f_s = 0\}$ is the variety $V(f_1, \ldots, f_s)$. Let G be a Gröbner basis of the ideal $I = \langle f_1, \ldots, f_s \rangle$. It can be shown that the variety $V(G) = V(I)$. In Section 3.3, we use the implementation of Gröbner bases in Magma [BCP97] to solve a system of polynomial equations. In Section 4.4.3, we use the facilities for multivariate polynomials in SageMath [The22] to solve a systems of linear equations.

2.3 Skew Polynomials

Skew polynomials over division rings[1] are *non-commutative* polynomials that were introduced and studied by Øystein Ore in [Ore33]. The theory of skew polynomials is quite rich and widely investigated in the literature. For instance, the division and factorization properties were studied in [Ore33; Gie98; Bau16]. Evaluation of skew polynomials and sets of roots were first considered by Tsit-Yuen Lam in [Lam86] and studied in great detail thereafter by Lam and André Leroy [LL88a; LL88b; Ler95; LL04; LLO08; Ler12]. Faster algorithms for skew polynomials have been proposed for factorization and counting the number of factorizations [CB12], interpolation [LMK14; BJR22] and multiplication [CLB17; PW18]. Properties of multivariate skew polynomial have been studied in [MK19a].

The general definition of skew polynomial rings $\mathcal{A}[X;\theta,\delta]$ involves an *endomorphism* θ of the base ring $\mathcal{A}$ and a *derivation* associated with the endomorphism. We start with the basics for the general definition in Section 2.3.1, where $\mathcal{A}$ is a general ring. This definition is mainly used in Chapter 3. In Section 2.3.2 we restrict our focus to a simpler class of skew polynomial ring $\mathbb{F}_{q^m}[X;\sigma]$ with the *Frobenius automorphism* $\sigma(a) = a^q, \forall a \in \mathbb{F}_{q^m}$ and the zero derivation $\delta = 0$, on which Chapter 4 is based.

2.3.1 The General Definition: $\mathcal{A}[X;\theta,\delta]$

Consider a ring $\mathcal{A}$ with addition $+$ and multiplication $\cdot$ (we may omit the $\cdot$ between two elements for simplicity). We consider the most general definition, where $\mathcal{A}$ is not necessarily a division ring.

Definition 2.6 (Endomorphism and derivation)**.** *An* endomorphism *of a ring $\mathcal{A}$ is a map $\theta : \mathcal{A} \to \mathcal{B} \subseteq \mathcal{A}$ such that, for all $a, b \in \mathcal{A}$,*

- $\theta(a + b) = \theta(a) + \theta(b)$,
- $\theta(ab) = \theta(a)\theta(b)$.

A map θ is an automorphism *if $\mathcal{B} = \mathcal{A}$.*

A θ-derivation *of $\mathcal{A}$ is a map $\delta : \mathcal{A} \to \mathcal{B} \subseteq \mathcal{A}$ such that, for all $a, b \in \mathcal{A}$*

- $\delta(a + b) = \delta(a) + \delta(b)$,
- $\delta(ab) = \delta(a)b + \theta(a)\delta(b)$.

A θ-derivation δ is an inner *θ-derivation if there exists $\beta \in \mathcal{A}$ such that $\delta(a) = \beta a - \theta(a)\beta$ for all $a \in \mathcal{A}$.*

It follows from the definition that for any endomorphsim θ and any θ-derivation δ of $\mathcal{A}$, it holds that

- $\theta(0) = 0,\ \theta(1) = 1$,
- $\delta(0) = 0,\ \delta(1) = 0$.

We denote by id the *identity automorphism* $\theta(a) = a, \forall a \in \mathcal{A}$. For ease of notation, we also use the exponential notation $\theta(a) = a^\theta$ and $\delta(a) = a^\delta$.

[1] A ring is a division ring if all the nonzero elements have a multiplicative inverse.

Lemma 2.1. *If $a \in \mathcal{A}^*$ is invertible, then a is not a* zero-divisor[2] *and $\theta(a)$ invertible.*

Proof. We first show that any $a \in \mathcal{A}$ cannot be both invertible and a zero-divisor. Suppose an invertible $a \in \mathcal{A}$ is a zero-divisor. Let $0 \neq b \in \mathcal{A}$ with $ab = 0$. Then $b = (a^{-1}a)b = a^{-1}(ab) = 0$, which is a contradiction. According to Definition 2.6, $\theta(a \cdot a^{-1}) = \theta(a) \cdot \theta(a^{-1})$. Together with $\theta(a \cdot a^{-1}) = \theta(1) = 1$, it can be seen that $\theta(a)$ is invertible and its inverse is $\theta(a^{-1})$. □

Definition 2.7 (Skew polynomial rings)**.** *A* skew polynomial ring $\mathcal{A}[X;\theta,\delta]$ *is a set of polynomials*

$$\mathcal{A}[X;\theta,\delta] := \left\{ \sum_{i=0}^{n} a_i X^i \;\middle|\; a_i \in \mathcal{A}, n \in \mathbb{N} \right\}$$

with addition $+$ as for usual polynomials, and multiplication $\cdot$ following the basic rule

$$Xa = \theta(a)X + \delta(a), \;\; \forall \; a \in \mathcal{A} \; . \tag{2.1}$$

The multiplication extends to all elements in $\mathcal{A}[X;\theta,\delta]$ by associativity and distributivity. The degree *of a nonzero skew polynomial $f = \sum_{i\in\mathbb{N}} f_i X^i \in \mathcal{A}[X;\theta,\delta]$ is $\deg f := \max\{\, i \mid f_i \neq 0\}$. By convention, the degree of the zero polynomial is $\deg(0) = -\infty$.*

Multiplication

Given an endomorphism θ of a ring $\mathcal{A}$, the powers of θ are $a^{\theta^{i+1}} = \theta^{i+1}(a) := \theta(\theta^i(a))$, for all $i \in \mathbb{N}, a \in \mathcal{A}$. Given a θ-derivation δ of $\mathcal{A}$ and $a \in \mathcal{A}$, we denote $a^{\theta\delta} = \delta(\theta(a))$.

For any $h = \sum_{i=0}^{d} h_i X^i$ and $g = \sum_{i=0}^{e} g_i X^i$ in $\mathcal{A}[X;\theta,\delta]$, the product of the two polynomials is

$$h \cdot g = h_d g_e^{\theta^d} X^{d+e} + \left(h_{d-1} g_e^{\theta^{d-1}} + h_d \left(g_e^{\theta^{d-1}\delta} + g_e^{\theta^{d-2}\delta\theta} + \cdots + g_e^{\delta\theta^{d-1}} \right) \right) X^{d+e-1} + \cdots \; ,$$

where $g_e^{\theta^{d-2}\delta\theta} = \theta(\delta(g_e^{\theta^{d-2}}))$ and $g_e^{\delta\theta^{d-1}} = \theta^{d-1}(\delta(g_e))$ according to the notation introduced above. It can be seen that the explicit expression of the product is quite messy when a nonzero derivation is involved. For skew polynomials with a zero θ-derivation, e.g., $h = \sum_{i\in\mathbb{N}} h_i X^i$ and $g = \sum_{i\in\mathbb{N}} g_i X^i$ in $\mathcal{A}[X;\theta]$, the expression of the product is simply

$$h \cdot g = \sum_{i\in\mathbb{N}} \sum_{j\in\mathbb{N}} h_i \theta^i(g_j) X^{i+j} = \sum_{s\in\mathbb{N}} \left(\sum_{\substack{i\in\mathbb{N} \\ i\leqslant s}} h_i \theta^i(g_{s-i}) \right) X^s \; . \tag{2.2}$$

Theorem 2.4. *For any $h, g \in \mathcal{A}[X;\theta,\delta]$, if the leading coefficient* $\mathrm{lc}(g)$ *is invertible, then* $\deg(hg) = \deg(h) + \deg(g)$.

Proof. Let $d = \deg(h)$ and $e = \deg(g)$. It can be seen from the multiplication rule in (2.1) that commuting the variable X and the coefficients does not increase the degree in X. Therefore, the coefficient of $X^{d+e+i} = 0, \forall i > 0$. It follows from Lemma 2.1 that the coefficient of the monomial X^{d+e} is $\mathrm{lc}(h) \cdot \theta^d(\mathrm{lc}(g)) \neq 0$. □

[2] An elements $b \in \mathcal{A}^*$ is a zero divisor if $\exists a \in \mathcal{A}^*$ such that $ab = ba = 0$.

Division

If the base ring $\mathcal{A}$ of $\mathcal{R} = \mathcal{A}[X;\theta,\delta]$ is a division ring, then for $f, g \in \mathcal{R}$, one can always perform a right division on f by g, i.e., find the quotient polynomial $q \in \mathcal{R}$ and the remainder polynomial $r \in \mathcal{R}$ such that $f = q \cdot g + r$ with $\deg(r) < \deg(g)$ [Ore33]. The right division algorithm is presented in Algorithm 2.3, which is the analogue to the well-known Euclidean algorithm for skew polynomials.

For a non-division ring $\mathcal{A}$, the following theorem shows that the right division can be done by Algorithm 2.3 as well, as long as the leading coefficient of the divisor is invertible.

Theorem 2.5. *For any $f, g \in \mathcal{R}$, if the leading coefficient $\mathrm{lc}(g)$ is invertible, then Algorithm 2.3 outputs a unique pair of quotient and remainder $q, r \in \mathcal{R}$.*

Proof. By Lemma 2.1, it can seen that, with an invertible $\mathrm{lc}(g)$, q_i at Line 3 in Algorithm 2.3 is nonzero and Line 5 can always remove the leading term in r. Hence, $\deg(r)$ decreases in every loop and the algorithm terminates. We show that the outputs q, r are unique by contradiction. Suppose that $f = hg + r = \tilde{h}g + \tilde{r}$, which implies $(h - \tilde{h})g = r - \tilde{r}$. If $h - \tilde{h}$ is nonzero, then $\deg((h-\tilde{h})g) \geqslant \deg(g)$. However, by the termination condition of the algorithm, $\deg(r - \tilde{r}) < \deg(g)$. □

Algorithm 2.3: Right division (Euclidean) algorithm for skew polynomials

Input: $f, g \in \mathcal{A}[X;\theta,\delta]$ where $\mathrm{lc}(g)$ is invertible
Output: A unique pair $q, r \in \mathcal{A}[X;\theta,\delta]$ such that $f = q \cdot g + r$ with $\deg(r) < \deg(g)$.

1 $q \leftarrow 0, r \leftarrow f$;
2 **while** $\deg(r) \geqslant \deg(g)$ **do**
3 $\quad q_i \leftarrow \mathrm{lc}(r)\theta^{n-m}(g_m^{-1})X^{n-m}$;
4 $\quad q \leftarrow q + q_i$;
5 $\quad r \leftarrow r - q_i \cdot g$;
6 **return** q, r

Left division between any $f, g \in \mathcal{R}$ can be performed only if θ is an automorphism of $\mathcal{A}$. For left division, the repeated procedure from Line 3 to Line 5 in Algorithm 2.3 becomes

$$\begin{aligned} q_i &\leftarrow \theta^{-m}(g_m^{-1}\,\mathrm{lc}(r))X^{n-m} \\ q &\leftarrow q + q_i \\ r &\leftarrow f - g \cdot q_i \end{aligned}$$

and the outputs $q, r \in \mathcal{R}$ are such that $f = g \cdot q + r$. The inverse map of θ is required in the computation of q_i. Hence, θ needs to be an automorphism in order to perform the left division.

We say that g right (resp., left) divides f or f is right (left) divisible by g, if the remainder of the right (left) division on f by g is 0. We denote by $g \mid_r f$ (resp., $g \mid_l f$) if f is right (left) divisible by g.

Definition 2.8 (gcrd and lclm)**.** *For any $f_1, f_2 \in \mathcal{R}$, the* greatest common right divisor *(gcrd) of f_1, f_2, denoted by* $\mathrm{gcrd}(f_1, f_2)$*, is a monic polynomial $g \in \mathcal{R}$ of the largest degree such that g right divides both f_1 and f_2.*

The least common left multiplier *(lclm) of $f_1, f_2 \in \mathcal{R}$ is a monic polynomial $m \in \mathcal{R}$ of the lowest degree which is right divisible by both f_1 and f_2, i.e.,*

$$m = g_1 \cdot f_1 = g_2 \cdot f_2 \quad \text{for some } g_1, g_2 \in \mathcal{R} \ .$$

It has been shown in [Ore33, Section 2] that for any $f_1, f_2 \in \mathcal{R}$ where $\mathcal{A}$ is a division ring, there is a unique $g = \operatorname{gcrd}(f_1, f_2)$ and it can be computed from the output of Algorithm 2.4: $\operatorname{gcrd}(f_1, f_2) = r_\ell$ (up to a scalar multiple). If $\operatorname{gcrd}(f_1, f_2) = 1$, then f_1 and f_2 are *relatively prime*.

Similarly, it has been shown in [Bau16, Section 2] that for any $f_1, f_2 \in \mathcal{R}$ where $\mathcal{A}$ is a division ring, there is a unique $m = \operatorname{lclm}(f_1, f_2)$ and it can be computed up to a scalar multiple from the output of Algorithm 2.4: $\operatorname{lclm}(f_1, f_2) = s_{\ell+1} f_1 = -t_{\ell+1} f_2$.

The degree of the lclm of f_1, f_2 and the degree of the gcrd can be related via

$$\deg \operatorname{lclm}(f_1, f_2) = \deg f_1 + \deg f_2 - \deg \operatorname{gcrd}(f_1, f_2) \ .$$

Algorithm 2.4: Extended Euclidean algorithm for skew polynomials.

Input: $f_1, f_2 \in \mathcal{R}$ where $\mathcal{A}$ is a division ring.
Output: $\ell \in \mathbb{N}$, $r_i, s_i, t_i \in \mathcal{R}, i \in [\ell + 1]$, and $q_i \in \mathcal{R}, i \in [\ell]$

$r_1 \leftarrow f_1, s_1 \leftarrow 1, t_1 \leftarrow 0$;
$r_2 \leftarrow f_2, s_2 \leftarrow 0, t_2 \leftarrow 1$;
$i \leftarrow 2$ **while** $r_i \neq 0$ **do**
    $q_i, r_{i+1} \leftarrow$ right divide r_i by r_{i-1} ; `/* Apply Algorithm 2.3 */`
    $s_{i+1} \leftarrow s_{i-1} - q_i s_i$;
    $t_{i+1} \leftarrow t_{i-1} - q_i t_i$;
    $i \leftarrow i + 1$
$\ell \leftarrow i - 1$;
return $r_i, s_i, t_i, \forall i \in [\ell + 1]$, *and* $q_i, \forall i \in [\ell]$

Evaluation

For a commutative polynomial $f \in \mathcal{A}[x]$, the process of evaluating f at $a \in \mathcal{A}$, denoted by $f(a)$, is simply "plugging in" the value a in place of x in f and carry out the proper operation in $\mathcal{A}$. The result by this simple method coincides with the result by the *remainder evaluation*, where $f(a) \in \mathcal{A}$ is the remainder of dividing f by $(x - a)$. In other words, the evaluation $f(a)$ is such that $f = g \cdot (x - a) + f(a)$ for some $g \in \mathcal{A}[x]$. However, for skew polynomials, simple plugging-in does not always give the same result as the remainder evaluation, as shown in Example 2.1. Nevertheless, we show in Theorem 2.6 that another way of "plugging-in" is equivalent to the remainder evaluation.

Definition 2.9 (Remainder evaluation of skew polynomials)**.** *Let $\mathcal{A}$ be a ring and $\mathcal{R} = \mathcal{A}[X; \theta, \delta]$. For any $f \in \mathcal{R}$ and $a \in \mathcal{A}$, the* evaluation *of f at a, denoted by $f(a)$, is the remainder from the right division on f by $X - a$. In other words, we can write*

$$f(a) = f - g \cdot (X - a) \quad \text{for some } g \in \mathcal{R} \ .$$

Since $X - a$ is monic, it follows from Theorem 2.5 that the remainder evaluation $f(a)$ is unique.

Example 2.1. *Let $\mathbb{F}_4[X;\sigma,\delta]$ be a skew polynomial ring with the Frobenius automorphism $\sigma(a) = a^2, \forall a \in \mathbb{F}_4$ and an inner σ-derivation $\delta(a) = \sigma(a) - a, \forall a \in \mathbb{F}_4$ (recall from Definition 2.6 that $\beta = 1$). Let α be a primitive element of $\mathbb{F}_4$. To evaluate $f = X^3 + X + 1$ at $\alpha \in \mathbb{F}_4$, by the "plugging-in" method, we get*

$$f(\alpha) = \alpha^3 + \alpha + 1 = \alpha \ .$$

By the remainder evaluation from Definition 2.9, we have

$$f(\alpha) = \alpha + 1 = f - (X^2 + \alpha X)(X - \alpha) \ .$$

The following theorem shows that the remainder evaluation for skew polynomials has an equivalent form so that one can perform the evaluation by addition and multiplication, without applying any division algorithm (e.g., Algorithm 2.3). The form has been proven in [LL88b, Lemma 2.4] for division rings and applies naturally to general rings.

Theorem 2.6. *Let $\mathcal{A}$ be a ring, θ be an endomorphism of $\mathcal{A}$ and δ be a θ-derivation. For any $a \in \mathcal{A}$, define recursively the* i-th truncated norm *of a as*

$$\begin{aligned} N_0(a) &:= 1 \ , \\ N_{i+1}(a) &:= \theta(N_i(a)) \cdot a + \delta(N_i(a)) \ , \ \forall i \in \mathbb{N} \ . \end{aligned}$$

Then, for any $f = \sum_{i\in\mathbb{N}} f_i X^i \in \mathcal{R}$, the evaluation of f at $a \in \mathcal{A}$ is

$$f(a) = \sum_{i\in\mathbb{N}} f_i N_i(a) \ .$$

Proof. We first show that for any $k \in \mathbb{N}$, $(X-a) \mid_r (X^k - N_k(a))$ by induction. This is trivial for $k = 0$, since $X^0 - N_0(a) = 1 - 1 = 0$. Assume it is true for some $k \geqslant 0$. Then

$$\begin{aligned} X^{k+1} - N_{k+1}(a) &= X^{k+1} - \theta(N_k(a)) \cdot a - \delta(N_k(a)) \\ &= X^{k+1} - \theta(N_k(a)) \cdot a + \theta(N_k(a)) \cdot X - \theta(N_k(a)) \cdot X - \delta(N_k(a)) \\ &= X^{k+1} + \theta(N_k(a))(X - a) - (\theta(N_k(a)) \cdot X + \delta(N_k(a))) \\ &= \theta(N_k(a))(X - a) + X^{k+1} - X \cdot N_k(a) \qquad (2.3) \\ &= \theta(N_k(a))(X - a) + X(X^k - \cdot N_k(a)) \ , \end{aligned}$$

where (2.3) follows from the multiplication rule in (2.1) for commuting the variable and the coefficient. With the induction hypothesis, $(X - a) \mid_r (X^k - N_k(a))$, we conclude that both terms on the last line are right divisible by $X - a$.

Then we can see that

$$\begin{aligned} f - f(a) &= \sum_{i\in\mathbb{N}} f_i X^i - \sum_{i\in\mathbb{N}} f_i N_i(a) \\ &= \sum_{i\in\mathbb{N}} f_i \cdot \big(X^i - N_i(a)\big) \ . \end{aligned}$$

Applying the result for $X^{k+1} - N_{k+1}(a)$ above to each term $\left(X^i - N_i(a)\right)$ in the sum, we conclude that $(X-a) \mid_r (f - f(a))$. Therefore, we can write $f = q \cdot (X-a) + f(a)$ for some $q \in \mathcal{R}$. Since $\deg(f(a)) < 1 = \deg(X-a)$, it is indeed the remainder of the right division on f by $X - a$. □

So far, we only considered polynomials in the *left form*, $f = \sum_{i\in\mathbb{N}} f_i X^i$. If a polynomial is given in the *right form*, i.e., $h = \sum_{i\in\mathbb{N}} X^i h_i$, then for some $a \in \mathcal{A}$, the evaluation $h(a) \neq \sum_{i\in\mathbb{N}} N_i(a) h_i$. Instead, we must first convert h to the left form $\sum_{i\in\mathbb{N}} g_i X^i$ by the multiplication rule (2.1) and then compute $h(a) = \sum_{i\in\mathbb{N}} g_i N_i(a)$.

With the definition of $N_i(a)$ in Theorem 2.6, we define the (θ,δ)-*Vandermonde matrix* on a set of elements $\Omega = \{a_1, \ldots, a_n\} \subseteq \mathcal{A}$.

Definition 2.10 ((θ,δ)-Vandermonde matrix)**.** *Let $N_i(\cdot)$ be the i-th truncated norm as defined in Theorem 2.6. Given a set $\Omega = \{a_1, \ldots, a_n\} \subseteq \mathcal{A}$, the (θ,δ)-Vandermonde matrix of Ω is given by*

$$\boldsymbol{V}^{\theta,\delta}(\Omega) := \begin{pmatrix} N_0(a_1) & N_0(a_2) & \ldots & N_0(a_n) \\ N_1(a_1) & N_1(a_2) & \ldots & N_1(a_n) \\ \vdots & \vdots & \ddots & \vdots \\ N_{n-1}(a_1) & N_{n-1}(a_2) & \ldots & N_{n-1}(a_n) \end{pmatrix} .$$

Similar to the evaluation of commutative univariate polynomials, the evaluation of a polynomial $f = \sum_{i=0}^{k} f_i X^i \in \mathcal{R}, k \in \mathbb{N}$ at all the elements in $\Omega = \{a_1, \ldots, a_n\}$ can be written as

$$(f(a_1), f(a_2), \ldots, f(a_n)) = (f_0, \ldots, f_k) \cdot \underbrace{\begin{pmatrix} N_0(a_1) & N_0(a_2) & \ldots & N_0(a_n) \\ \vdots & \vdots & \ddots & \vdots \\ N_k(a_1) & N_k(a_2) & \ldots & N_k(a_n) \end{pmatrix}}_{=:\boldsymbol{V}_k^{\theta,\delta}(\Omega)} .$$

Next, we seek for a formula for evaluating a product of two polynomials at a point $a \in \mathcal{A}$. For this purpose, we first define the notion of (θ,δ)-*conjugacy*.

Definition 2.11 ((θ,δ)-conjugacy)**.** *For any two elements $a \in \mathcal{A}, c \in \mathcal{A}^*$, define*

$$a^c := \theta(c) a c^{-1} + \delta(c) c^{-1} ,$$

where the term $\delta(c)c^{-1}$ is also known as the "logarithmic derivative" of $c \in \mathcal{A}^*$. Two elements $a, b \in \mathcal{A}$ are said to be (θ,δ)-conjugate *if there exists an element $c \in \mathcal{A}^*$ such that $a^c = b$.*

It is easy to check that (θ,δ)-conjugacy fulfills the following properties of equivalence relations:

- Reflexivity: a is (θ,δ)-conjugate to a, for any $a \in \mathcal{A}$.
- Symmetry: If a is (θ,δ)-conjugate to b, then b is (θ,δ)-conjugate to a, for any $a, b \in \mathcal{A}$.
- Transitivity: If a is (θ,δ)-conjugate to b and b is (θ,δ)-conjugate to c, then a is (θ,δ)-conjugate to c, for any $a, b, c \in \mathcal{A}$.

Since (θ, δ)-conjugacy is an equivalence relation, we can speak of (θ, δ)-conjugacy classes. For instance, the (θ, δ)-conjugacy class of 0, $\left\{0^c = \delta(c)c^{-1} \mid c \in \mathcal{A}^*\right\}$, consists of the logarithmic derivatives of all the nonzero elements in $\mathcal{A}$.

Definition 2.12 ((θ, δ)-conjugacy classes)**.** *For any $a \in \mathcal{A}$, the (θ, δ)-conjugacy class of a is*

$$C_{\theta,\delta}(a) := \{a^c \mid c \in \mathcal{A}^*\} .$$

Using the notion of (θ, δ)-conjugacy, the next result provides us with a useful formula for evaluating a product $f \cdot g$ at $a \in \mathcal{A}$.

Theorem 2.7 ([LL88b, Theorem 2.7])**.** *Let $f, g \in \mathcal{R}$ and $a \in \mathcal{A}$. Then, $(f \cdot g)(a) \neq f(a) \cdot g(a)$ in general. Instead,*

$$(f \cdot g)(a) = \begin{cases} 0 & \text{if } g(a) = 0 \\ f\left(a^{g(a)}\right) \cdot g(a) & \text{else} \end{cases} ,$$

where $a^{g(a)} := \theta(g(a)) \cdot a \cdot g(a)^{-1} + \delta(g(a)) \cdot g(a)^{-1}$.

Ideals

Since the skew polynomial rings $\mathcal{R}$ are non-commutative, we need to differentiate between left and right ideals of $\mathcal{R}$.

A *left ideal* $\langle f \rangle_l \subseteq \mathcal{R}$ generated by $f \in \mathcal{R}$ is a set of skew polynomials

$$\langle f \rangle_l := \{gf \mid g \in \mathcal{R}\} .$$

Similarly, a *right ideal* $\langle f \rangle_r \subseteq \mathcal{R}$ generated by $f \in \mathcal{R}$ is a set of skew polynomials

$$\langle f \rangle_r := \{fg \mid g \in \mathcal{R}\} .$$

Lemma 2.2. *For $g, f \in \mathcal{R}$, $\langle g \rangle_l \supseteq \langle f \rangle_l$ if and only if $g \mid_r f$.*

Proof. We first show the sufficiency. Suppose $g \mid_r f$, then $\forall p \in \langle f \rangle_l$, $p = u \cdot f$ for some $u \in \mathcal{R}$ and $p = u \cdot (q \cdot g)$ for q being the quotient of the right division on f by g. Hence, $\langle g \rangle_l \supseteq \langle f \rangle_l$. For the necessity, assume $g \nmid_r f$. Then there is a nonzero $a \in \mathcal{R}$ of $\deg(a) < \deg(g)$ such that $f = q \cdot g + a$. It can be seen that for any $u \in \mathcal{R}$ of $\deg(u) = 0$, $u \cdot f \notin \langle g \rangle_l$. □

2.3.2 With Frobenius Automorphism and Zero Derivation

In this subsection, we present the properties of skew polynomials over an extension field $\mathbb{F}_{q^m}$, with the Frobenius automorphism and zero derivation.

The *Frobenius automorphism* of an extension field $\mathbb{F}_{q^m}$ is the map

$$\begin{aligned} \sigma : \mathbb{F}_{q^m} &\to \mathbb{F}_{q^m} \\ a &\mapsto a^q . \end{aligned}$$

Let $\mathbb{F}_{q^m}[X; \sigma]$ be a skew polynomial ring over $\mathbb{F}_{q^m}$ with Frobenius automorphism σ and zero derivation $\delta = 0$. We follow the notation used in Section 2.3.1 but omit the δ for simplicity.

For two skew polynomials $h = \sum_{i\in\mathbb{N}} h_i X^i$ and $g = \sum_{j\in\mathbb{N}} g_j X^j$ in $\mathbb{F}_{q^m}[X;\sigma]$, their product $h \cdot g$ can be computed according to (2.2). Since every nonzero element in $\mathbb{F}_{q^m}$ is invertible, it follows from Theorem 2.4 that $\deg(hg) = \deg(h) + \deg(g)$.

For any $\alpha \in \mathbb{F}_{q^m}$, its i-th truncated norm $N_i(\alpha)$ (defined in Theorem 2.6) becomes

$$N_i(\alpha) = \prod_{j=0}^{i-1} \sigma^j(\alpha) = \alpha^{\sum_{j=0}^{i-1} q^j} = \alpha^{(q^i-1)/(q-1)} \ .$$

The evaluation of any $f = \sum_{i\in\mathbb{N}} f_i X^i \in \mathbb{F}_{q^m}[X;\sigma]$ at $\alpha \in \mathbb{F}_{q^m}$ becomes

$$f(\alpha) = \sum_{i\in\mathbb{N}} f_i N_i(\alpha) = \sum_{i\in\mathbb{N}} f_i \alpha^{(q^i-1)/(q-1)} \ . \tag{2.4}$$

σ-Conjugacy Classes

For any $a \in \mathbb{F}_{q^m}$, its *σ-conjugacy* w.r.t. $c \in \mathbb{F}_{q^m}^*$ becomes

$$a^c := \sigma(c) a c^{-1} = a c^{q-1} \ , \tag{2.5}$$

and its σ-conjugacy class becomes

$$C_\sigma(a) := \left\{ a c^{q-1} \;\middle|\; c \in \mathbb{F}_{q^m}^* \right\} . \tag{2.6}$$

We say that two elements $a, b \in \mathbb{F}_{q^m}$ are *σ-distinct* if $b \notin C_\sigma(a)$. The following theorem shows that the σ-conjugacy classes of $\mathbb{F}_{q^m}$ form a partition of $\mathbb{F}_{q^m}$.

Theorem 2.8 (Structure of σ-conjugacy classes [MSK+22, Theorem 2.12])**.** *Let $\gamma \in \mathbb{F}_{q^m}^*$ be a primitive element of $\mathbb{F}_{q^m}$. For the Frobenius automorphism σ, the $q-1$ elements $1, \gamma, \gamma^2, \ldots, \gamma^{q-2}$ are pair-wise σ-distinct. There are q disjoint conjugacy classes in $\mathbb{F}_{q^m}$ and*

$$\mathbb{F}_{q^m} = C_\sigma(0) \cup C_\sigma(\gamma^0) \cup \cdots \cup C_\sigma(\gamma^{q-2}) \ ,$$

where $C_\sigma(0) = \{0\}$ and $\left|C_\sigma(\gamma^i)\right| = \frac{q^m-1}{q-1}, i \in [0, q-2]$.

Polynomial Independence

In the following, we discuss the set of roots of skew polynomials, i.e., given $f \in \mathbb{F}_{q^m}[X;\sigma]$, we determine $\{\alpha \in \mathbb{F}_{q^m} \mid f(\alpha) = 0\}$. Different from commutative polynomials $g \in \mathbb{F}[x]$, where the number of roots of g is at most $\deg(g)$, the following example shows that this does not hold for skew polynomials in general.

Example 2.2. *Consider $q = 2, m = 2$ and the skew polynomial ring $\mathbb{F}_4[X;\sigma]$ with the Frobenius automorphism and zero derivation. Let α be a primitive element of $\mathbb{F}_4$. For the polynomial $f = X^2 + 1 \in \mathbb{F}_4[X;\sigma]$, it can be verified that the set of root of f is*

$$\{1, \alpha, \alpha + 1\} \ .$$

In the commutative case, we count *distinct* roots. In the case of skew polynomials, we need a different notion of *distinctness*, namely, *polynomial independence*. For this, we need the following definition.

Definition 2.13 (Minimal polynomial)**.** *Given a nonempty set $\Omega \subseteq \mathbb{F}_{q^m}$, we say f_Ω is a* minimal polynomial *of Ω if it is a monic polynomial $f_\Omega \in \mathbb{F}_{q^m}[X;\sigma]$ of minimal degree such that $f_\Omega(\alpha) = 0$ for all $\alpha \in \Omega$.*

The following theorem shows that for any nonempty $\Omega \subseteq \mathbb{F}_{q^m}$, its minimal polynomial is unique. It also implies that $\mathbb{F}_{q^m}[X;\sigma]$ is a *principle left ideal ring*[3].

Theorem 2.9 ([MSK+22, Theorem 2.5])**.** *Given any nonempty set $\Omega \subseteq \mathbb{F}_{q^m}$, there exists a unique monic skew polynomial $f \in \mathbb{F}_{q^m}[X;\sigma]$ such that, for any g in the left ideal $\langle f \rangle_l$,*

$$g(\alpha) = 0, \ \forall \alpha \in \Omega \ .$$

The minimal polynomial can be constructed by an iterative Newton interpolation approach as follows (cf. [MSK+22, Proposition 2.6]). First, set

$$g_1 = X - \alpha_1.$$

Then for $i = 2, 3, \ldots, n$, perform

$$g_i = \begin{cases} g_{i-1} & \text{if } g_{i-1}(\alpha_i) = 0, \\ \left(X - \alpha^{g_{i-1}(\alpha_i)}\right) \cdot g_{i-1} & \text{otherwise,} \end{cases} \tag{2.7}$$

where $\alpha^{g_{i-1}(\alpha_i)} = \sigma(g_{i-1}(\alpha_i))\alpha_i g_{i-1}(\alpha_i)^{-1}$ is the σ-conjugate of α w.r.t. $g_{i-1}(\alpha_i)$ as in (2.5). Upon termination, we have $g_n = f_\Omega$.

It can be shown that the minimal polynomial of a set Ω can be also constructed by computing

$$f_\Omega = \mathrm{lclm}_{\alpha \in \Omega}\{X - \alpha\} \ , \tag{2.8}$$

where lclm is the least common left multiple as defined in Definition 2.8.

The following theorem summarizes some properties of the minimal polynomial of a set $\Omega \subseteq \mathbb{F}_{q^m}$.

Theorem 2.10 ([LMK17, Theorem 7])**.** *Let $\Omega_1, \Omega_2 \subseteq \mathbb{F}_{q^m}$ and $f_{\Omega_1}, f_{\Omega_2}$ be their minimal polynomials. Then*

- $f_{\Omega_1 \cup \Omega_2} = \mathrm{lclm}(f_{\Omega_1}, f_{\Omega_2})$.
- $f_{\Omega_1 \cap \Omega_2} = \mathrm{gcrd}(f_{\Omega_1}, f_{\Omega_2})$.
- $\deg(f_{\Omega_1 \cup \Omega_2}) = \deg(f_{\Omega_1}) + \deg(f_{\Omega_2}) - \deg(f_{\Omega_1 \cap \Omega_2})$.

With the notion of minimal polynomials, we can now introduce the "distinctness" for the roots of skew polynomials.

Definition 2.14 (P-independent set)**.** *A set $\Omega \subseteq \mathbb{F}_{q^m}$ is P-independent in $\mathbb{F}_{q^m}[X;\sigma]$ if the degree of its minimal polynomial is equal to $|\Omega|$, i.e.,* $\deg(f_\Omega) = |\Omega|$.

Recall from Definition 2.10 that a σ-Vandermonde matrix can be constructed for a set $\Omega \subseteq \mathbb{F}_{q^m}$. It has been shown in [Lam86, Theorem 8] that

$$\Omega \text{ is P-independent} \iff |\Omega| = \mathrm{rank}(\boldsymbol{V}^\sigma(\Omega)) \ . \tag{2.9}$$

[3] A principle left ideal ring is a ring where every left ideal can be generated by only one element in the ideal.

We can derive the following result from the definition of P-independent sets.

Lemma 2.3. *Given a P-independent set Ω, for any subset $\mathcal{Z} \subset \Omega$, let $f_{\mathcal{Z}}(x) \in \mathbb{F}_{q^m}[X;\sigma]$ be the minimal polynomial of $\mathcal{Z}$. Then, for any element $\alpha \in \Omega \backslash \mathcal{Z}$, $f_{\mathcal{Z}}(\alpha) \neq 0$.*

Proof. Assume $f_{\mathcal{Z}}(\alpha) = 0$, then the minimal polynomial $f_{\mathcal{Z}\cup\{\alpha\}} = f_{\mathcal{Z}}$ and $\deg(f_{\mathcal{Z}\cup\{\alpha\}}) = |\mathcal{Z}| < |\mathcal{Z} \cup \{\alpha\}|$, which contradicts to that $\mathcal{Z} \cup \{\alpha\} \subseteq \Omega$ is P-independent. □

In the early work by Lam, the property of unions and subsets of P-independent sets has been derived, as summarized in the following theorem.

Theorem 2.11 ([Lam86, Theorem 22–23])**.** *For any two sets $\Omega_1, \Omega_2 \subseteq \mathbb{F}_{q^m}$, let $\Omega = \Omega_1 \cup \Omega_2$.*

- *If Ω_1 and Ω_2 are P-independent and no element in Ω_1 is σ-conjugate to any element in Ω_2, then Ω is P-independent.*
- *If Ω is P-independent, then Ω_1 and Ω_2 are P-independent. In other words, any subset of a P-independent set is P-independent.*

The relation between P-independence and linear independence and the structure of roots of the minimal polynomial f_Ω of some special sets $\Omega \subseteq \mathbb{F}_{q^m}$ are concluded in the following.

Theorem 2.12 ([LMK17, Lemma 1, Theorem 10])**.** *Let $\Omega = \{\alpha_1, \ldots, \alpha_n\} \subseteq C_\sigma(\gamma^l)$ for some $l \in [m]$, and $a_1, \ldots, a_n \in \mathbb{F}_{q^m}$ be such that $\alpha_i = \gamma^l a_i^{q-1}$ for all $i \in [n]$. Then, Ω is P-independent if and only if $a_1, \ldots, a_n$ are linearly independent over $\mathbb{F}_q$. Moreover, let f_Ω be the minimal polynomial of Ω and $\overline{\Omega} := \{\alpha \in \mathbb{F}_{q^m} \mid f_\Omega(\alpha) = 0\}$ be the set of roots of f_Ω. Then*

$$\overline{\Omega} = \left\{ \gamma^l a^{q-1} \;\middle|\; a \in \langle a_1, \ldots, a_n \rangle_q \right\} \subseteq C_\sigma(\gamma^l) \ ,$$

where $\langle a_1, \ldots, a_n \rangle_q$ denotes the $\mathbb{F}_q$-subspace of $\mathbb{F}_{q^m}$ generated by $a_1, \ldots, a_n$.

2.3.3 Applications of Skew Polynomials in Coding Theory

Codes constructed from skew polynomials were first introduced and studied by Boucher and Ulmer [BGU07; BU09a; BU11; BU14b]. Various classes of codes based on skew polynomials were constructed thereafter. For instance, codes over rings [AS10], evaluation codes [BU14a; LMK15], optimal codes in skew and sum-rank metric [Mar18] (well-known as linearized Reed-Solomon codes), skew Reed-Muller codes [GU19], new optimal codes in the rank metric [She20], skew convolutional codes [SLGK20], skew-cyclic codes [Glu21], sum-rank BCH and cyclic-skew-cyclic codes [Mar21], quantum codes [AA16; DBU+21], and twisted linearized Reed-Solomon codes [Ner22]. Skew polynomials have been considered to construct cryptographic schemes [CHH01; BGG+10], Shamir's secret sharing scheme [Zha10], DNA codes [GOS17] and maximal-recoverable locally repairable codes (MR-LRCs) [GG22].

2.4 Linear Block Codes Constructed from Polynomials

Let $\mathcal{R}$ be a polynomial ring, a polynomial code can be simply defined as a set of polynomials fulfilling some conditions,

$$\mathcal{C} := \{ f \in \mathcal{R} \mid f \text{ fulfills certain conditions } \} \ .$$

For instance, a Reed-Solomon (RS) code of dimension k is a set of univariate polynomials of degree less than k and a Reed-Muller (RM) code of dimension $\binom{m+d}{m}$ is a set of m-variate polynomials of total degree at most d. We denote these two set of polynomials respectively by

$$\begin{aligned} \mathsf{RS}_q(k) &:= \{ f \in \mathbb{F}_q[x] \mid \deg(f) < k \} \ , \\ \mathsf{RM}_q(d,m) &:= \{ f \in \mathbb{F}_q[x_1, \ldots, x_m] \mid \deg(f) \leqslant d \} \ . \end{aligned} \tag{2.10}$$

In most of the coding theory literature, as shown in the transmission model in Fig. 2.1, the elements in a code, called *codewords*, are considered to be a vector $\boldsymbol{c}$ of length n over a certain alphabet $\mathcal{A}$ (e.g., $\mathbb{F}_q$ for the $\mathsf{RS}_q(k)$ code) and are mapped from an *information* vector $\boldsymbol{u}$ of length k.

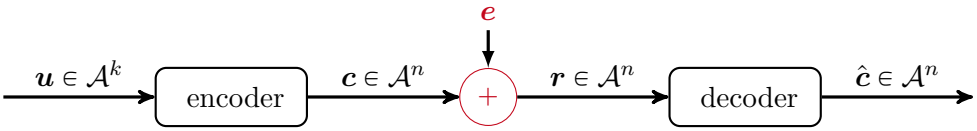


Figure 2.1: Transmission model via a linear $[n, k]$ block code with an alphabet $\mathcal{A}$.

Depending on how the symbols in $\boldsymbol{u}$ and $\boldsymbol{c}$ are related to a polynomial f in a code $\mathcal{C}$, we characterize polynomial codes into two categories: *evaluation codes* and *polycyclic codes*.

Throughout the thesis, when the code is seen as a set of polynomials as in (2.10), we denote within the parenthesis "( )" the parameters regarding the polynomials, e.g., k in $\mathsf{RS}_q(k)$ is the degree restriction of the polynomials. When the code is seen as a linear block code with fixed length n and dimension k, we denote within the rectangular brackets "[ ]" the parameters, e.g., n and k in $\mathsf{RS}_q[n, k]$ are the length and the dimension.

2.4.1 Evaluation Codes

Evaluation codes are a class of codes obtained by evaluating polynomials at some evaluation points. The number of evaluation points determines the length of the codes and the degree restriction on the polynomials determines the dimension of the codes.

The RS codes $\mathsf{RS}_q(k)$ can be also defined as a block code of length n via evaluating the polynomials at n distinct points. An $[n, k]_q$ RS code is defined as

$$\mathsf{RS}_q[n,k] := \{ (f(\alpha_1), \ldots, f(\alpha_n)) \mid f \in \mathbb{F}_q[x], \ \deg(f) < k \} \ ,$$

where $\alpha_1, \ldots, \alpha_n$ are distinct elements in $\mathbb{F}_q$ and called *code locators* of the RS codes.

Similarly, an RM code $\mathsf{RM}_q(d, m)$ is often seen as a set of $\mathbb{F}_q$-vectors of length $n = q^m$ via evaluating the polynomials at all the points in $\mathbb{F}_q^m$. Hence, $\mathsf{RM}_q(d, m)$ can be defined as an $[n, k]_q$ block code with $n = q^m$ and $k = \binom{m+d}{m}$,

$$\mathsf{RM}_q[n,k] := \{ (f(\boldsymbol{v}))_{\boldsymbol{v} \in \mathbb{F}_q^m} \mid f \in \mathbb{F}_q[x_1, \ldots, x_m], \ \deg(f) \leqslant d \} \ .$$

For evaluation codes, with the alphabet $\mathcal{A}$, each information vector $\boldsymbol{u} \in \mathcal{A}^k$ is associated to the coefficients of a polynomial of degree k in $\mathcal{A}[x]$, i.e.,

$$(u_0, u_1, \ldots, u_{k-1}) \mapsto f = u_0 + u_1 x + \ldots, u_{k-1} x^{k-1} \ .$$

Evaluation codes are an important method to construct block codes from polynomials. One of the reasons is that the minimum Hamming distance can be derived by studying the roots of the underlying polynomials. The most studied maximum distance achievable codes in the Hamming, rank and sum-rank metric are all evaluation codes (see Section 2.5). The *interpolation-based* decoders for evaluation codes are essentially developed from interpolating a polynomial from some of its evaluations.

2.4.2 Polycyclic Codes

Let $\mathcal{A}$ be a commutative ring and $\mathcal{R} = \mathcal{A}[x]$ be a polynomial ring. Denote by $\langle f \rangle \subseteq \mathcal{R}$ a principle ideal of $\mathcal{R}$ generated by $f \in \mathcal{R}$. The *quotient ring* (or *factor ring*) of $\mathcal{R}$ modulo the ideal $\langle f \rangle$ is defined as

$$\mathcal{R}/\langle f \rangle := \{u \bmod f \mid u \in \mathcal{R}\} .$$

Informally speaking, a *polycyclic code*, if seen as a set of polynomial, is an ideal in the quotient ring $\mathcal{R}/\langle f \rangle$. If seen as a set of vectors, it is the vector representation of the ideal.

Definition 2.15 (Polycyclic codes)**.** *Let $f \in \mathcal{R}$ with $\deg(f) = n$, $g \in \mathcal{R}$ with $\deg(g) = n - k$, and $g \mid f$. A* polycyclic code *w.r.t. g, f is defined as the principle ideal $\langle g \rangle / \langle f \rangle$ in $\mathcal{R}/\langle f \rangle$, i.e.,*

$$\mathcal{C}(g, f) := \{u \cdot g \bmod f \mid u \in \mathcal{R}\} .$$

A linear $[n, k]$ code is a polycyclic code *if*

$$\mathcal{C}[n, k] := \left\{ \boldsymbol{c} = (c_0, c_1, \ldots, c_{n-1}) \mid \sum_{i=0}^{n-1} c_i x^i \in \mathcal{C}(g, f) \right\} .$$

The polynomial g is a generator polynomial *of $\mathcal{C}$.*

An information vector $\boldsymbol{u} = (u_0, u_1, \ldots, u_{k-1})$ corresponds uniquely to a codeword $\boldsymbol{c} \in \mathcal{C}[n, k]$ such that

$$\left(u_0 + u_1 x + \cdots + u_{k-1} x^{k-1}\right) \cdot g = c_0 + c_1 x + \cdots + c_{n-1} x^{n-1} .$$

Remark 2.1. *The name "polycyclic" is a generalization of the well-known cyclic codes. Polycyclic codes have the following special cases that have been defined and studied in the literature.*

- *If $f = x^n - 1$, then $\mathcal{C}$ is a* cyclic code *such that*

$$(c_{n-1}, c_0, \ldots, c_{n-2}) \in \mathcal{C} \quad \text{for all } (c_0, c_1, \ldots, c_{n-1}) \in \mathcal{C} .$$

- *If $f = x^n + 1$, then $\mathcal{C}$ is a* negacyclic code *such that*

$$(-c_{n-1}, c_0, \ldots, c_{n-2}) \in \mathcal{C} \quad \text{for all } (c_0, c_1, \ldots, c_{n-1}) \in \mathcal{C} .$$

- *If $f = x^n - a$ for some $a \in \mathcal{A}^*$, then $\mathcal{C}$ is a* constacyclic code *such that*

$$(ac_{n-1}, c_0, \ldots, c_{n-2}) \in \mathcal{C} \quad \text{for all } (c_0, c_1, \ldots, c_{n-1}) \in \mathcal{C} .$$

Cyclic codes are an important class of codes in both theory and practice. Theoretically, there is rich mathematical theory in their properties, while practically, they are efficient to be encoded and decoded. The study of cyclic codes over finite fields started from the reports by Prange [Pra57; Pra85]. BCH codes are a class of cyclic codes with special generator polynomials which result in good distance property. Cyclic codes over finite rings are extensively studied since the work by Shankar [Sha79], where the Chinese Remainder Theorem was used to investigate the BCH codes over integer rings. In Chapter 3, we introduce a class of polycyclic codes where $\mathcal{R}$ is a skew polynomial ring.

2.5 Metrics

It can be seen from the last section that a *code* is simply a set of vectors or polynomials. In order to design error-correcting codes, we need to measure the distinction between the elements in the set. A distance measure $\mathrm{d}_{\mathsf{M}}(\cdot,\cdot)$ on a set $\mathcal{A}$ is called a *metric* if it fulfills the following conditions for all $a, b, c \in \mathcal{A}$:

- Positive definiteness: $\mathrm{d}_{\mathsf{M}}(a,b) \geqslant 0$ and the equality holds if and only if $a = b$.
- Symmetry: $\mathrm{d}_{\mathsf{M}}(a,b) = \mathrm{d}_{\mathsf{M}}(b,a)$.
- Triangle inequality: $\mathrm{d}_{\mathsf{M}}(a,b) + \mathrm{d}_{\mathsf{M}}(b,c) \geqslant \mathrm{d}_{\mathsf{M}}(a,c)$.

In this thesis, several metrics are used to study the error-correction capability of the codes constructed from polynomials.

2.5.1 Hamming Metric

The Hamming metric is the most used distance measure for error-correcting codes. It can be used for any linear block codes.

Definition 2.16 (Hamming metric)**.** *The* Hamming weight *on* $\mathbb{F}_q^n$ *is defined as*

$$\begin{aligned} \mathrm{wt}_{\mathsf{H}}(\cdot) \ : \ \mathbb{F}_q^n &\to \mathbb{N} \\ \boldsymbol{a} &\mapsto |\{i \in [n] \mid a_i \neq 0\}| \ . \end{aligned}$$

The Hamming distance *between two vectors is defined as*

$$\begin{aligned} \mathrm{d}_{\mathsf{H}}(\cdot,\cdot) \ : \ \mathbb{F}_q^n \times \mathbb{F}_q^n &\to \mathbb{N} \\ \boldsymbol{a}, \boldsymbol{b} &\mapsto |\{i \in [n] \mid a_i - b_i \neq 0\}| = \mathrm{wt}_{\mathsf{H}}(\boldsymbol{a} - \boldsymbol{b}) \ . \end{aligned}$$

For a code $\mathcal{C} \subseteq \mathbb{F}_q^n$*, its* minimum Hamming distance *is*

$$\begin{aligned} \mathrm{d}_{\mathsf{H}}(\mathcal{C}) &:= \min_{\substack{\boldsymbol{c}_1, \boldsymbol{c}_2 \in \mathcal{C} \\ \boldsymbol{c}_1 \neq \boldsymbol{c}_2}} \mathrm{d}_{\mathsf{H}}(\boldsymbol{c}_1, \boldsymbol{c}_2) \\ &= \min_{\boldsymbol{0} \neq \boldsymbol{c} \in \mathcal{C}} \mathrm{wt}_{\mathsf{H}}(\boldsymbol{c}) \ (\textit{If } \mathcal{C} \textit{ is linear}). \end{aligned}$$

Theorem 2.13 (Singleton bound [Sin64])**.** *For any block code* $\mathcal{C} \subseteq \mathbb{F}_q^n$ *(linear or non-linear) with minimum Hamming distance* $\mathrm{d}_{\mathsf{H}}(\mathcal{C}) = d$*,*

$$|\mathcal{C}| \leqslant q^{n - \mathrm{d}_{\mathsf{H}}(\mathcal{C}) + 1} \ .$$

If $\mathcal{C}$ is linear, then its dimension k fulfills

$$k \leqslant n - d + 1 \ .$$

A code whose length, cardinality and minimum Hamming distance fulfill the Singleton bound is called *maximum distance separable* (MDS). (Generalized) Reed-Solomon (GRS) codes are the most studied class of MDS codes. They require a field $\mathbb{F}_q$ of size at least the code length n. There are not many other non-trivial (i.e., not $[n, 1]$ or $[n, n]$) linear codes with this property and all of them are relatively short compared to their field size. In fact the famous MDS conjecture ([Seg55][Bal20, Conjecture 6.13]) asserts that within the range $4 \leqslant k \leqslant q - 2$, a k-dimensional MDS codes of length n satisfies $n \leqslant q + 1$.

Theorem 2.14 (Sphere-packing [Ham50] and Gilbert-Varshamov bounds [Gil52; Var57])**.** *Let $A_q^{\mathsf{H}}(n, d)$ be the maximum cardinality of a code $\mathcal{C} \subseteq \mathbb{F}_q^n$ with minimum Hamming distance $\mathrm{d}_{\mathsf{H}}(\mathcal{C}) = d$. Then,*

$$\frac{q^n}{\sum_{i=0}^{d-1} \binom{n}{i}(q-1)^i} \leqslant A_q^{\mathsf{H}}(n, d) \leqslant \frac{q^n}{\sum_{i=0}^{t} \binom{n}{i}(q-1)^i} \ ,$$

where $t = \left\lfloor \frac{d-1}{2} \right\rfloor$. The upper bound is the sphere-packing bound and the lower bound is the Gilbert-Varshamov (GV) bound.

A code whose parameters fulfill the sphere-packing bound is called a *perfect* code. The GV bound is often referred as the *random coding* bound. Namely, constructing a code by randomly taking $\frac{q^n}{\sum_{i=0}^{d-1} \binom{n}{i}(q-1)^i}$ distinct vectors from $\mathbb{F}_q^n$, with high probability, the code has minimal Hamming distance at least d.

2.5.2 Rank Metric

Fix a basis $\boldsymbol{\beta} = (\beta_1, \beta_2, \ldots, \beta_m)$ of $\mathbb{F}_{q^m}$ over $\mathbb{F}_q$. We define a mapping from $\mathbb{F}_{q^m}^n$ to $\mathbb{F}_q^{m \times n}$ by

$$\begin{aligned} \mathrm{ext}_{\boldsymbol{\beta}} : \mathbb{F}_{q^m}^n &\to \mathbb{F}_q^{m \times n} \\ \boldsymbol{c} = (c_1, c_2, \ldots, c_n) &\mapsto \boldsymbol{C} = \begin{pmatrix} c_{1,1} & c_{1,2} & \ldots & c_{1,n} \\ \vdots & \vdots & \ddots & \vdots \\ c_{m,1} & c_{m,2} & \ldots & c_{m,n} \end{pmatrix} , \end{aligned} \tag{2.11}$$

where $\boldsymbol{C}$ is unique such that $c_j = \sum_{i=1}^{m} c_{i,j}\beta_i$, for all $j = 1, \ldots, n$. The $\mathbb{F}_q$-rank of $\boldsymbol{c}$ is defined as $\mathrm{rank}_q(\boldsymbol{c}) := \mathrm{rank}(\boldsymbol{C})$.

Definition 2.17 (Rank metric)**.** *The* rank weight *on $\mathbb{F}_{q^m}^n$ is defined as*

$$\begin{aligned} \mathrm{wt}_{\mathsf{R}}(\cdot) \ : \ \mathbb{F}_{q^m}^n &\to \mathbb{N} \\ \boldsymbol{a} &\mapsto \mathrm{rank}_q(\boldsymbol{a}) \ . \end{aligned}$$

The Rank distance *between two vectors is defined as*

$$\begin{aligned} \mathrm{d}_{\mathsf{R}}(\cdot, \cdot) \ : \ \mathbb{F}_{q^m}^n \times \mathbb{F}_{q^m}^n &\to \mathbb{N} \\ \boldsymbol{a}, \boldsymbol{b} &\mapsto \mathrm{rank}_q(\boldsymbol{a} - \boldsymbol{b}). \end{aligned}$$

For a code $\mathcal{C} \subseteq \mathbb{F}_{q^m}^n$*, its* minimum rank distance *is*

$$\begin{aligned} \mathrm{d}_{\mathsf{R}}(\mathcal{C}) &:= \min_{\substack{\boldsymbol{c}_1, \boldsymbol{c}_2 \in \mathcal{C} \\ \boldsymbol{c}_1 \neq \boldsymbol{c}_2}} \mathrm{d}_{\mathsf{R}}(\boldsymbol{c}_1, \boldsymbol{c}_2) \\ &= \min_{\boldsymbol{0} \neq \boldsymbol{c} \in \mathcal{C}} \mathrm{wt}_{\mathsf{R}}(\boldsymbol{c}) \ (\textit{If } \mathcal{C} \textit{ is linear}). \end{aligned}$$

Theorem 2.15 (Singleton bound in rank metric [Del78, Theorem 5.4])**.** *For a code* $\mathcal{C} \subseteq \mathbb{F}_{q^m}^n$ *with minimum rank distance* $\mathrm{d}_{\mathsf{R}}(\mathcal{C}) = d$*,*

$$|\mathcal{C}| \leqslant q^{\min\{n(m-d+1), m(n-d+1)\}} = q^{\max\{n,m\}(\min\{n,m\}-d+1)} \ .$$

If $\mathcal{C}$ *is* $\mathbb{F}_{q^m}$*-linear, then its dimension* k *over* $\mathbb{F}_{q^m}$ *fulfills*

$$k \leqslant n - d + 1 \ .$$

A code whose length, cardinality and minimum rank distance fulfilling the Singleton bound is a *maximum rank distance* (MRD) code. *Gabidulin codes* [Del78; Gab85; Rot91] are the most well-known MRD codes. They are a class of evaluation codes based on *linearlized polynomials* (a special class of skew polynomials).

Theorem 2.16 (Sphere-packing and Gilbert-Varshamov bounds in rank metric [GY06])**.** *Let* $A_{q^m}^{\mathsf{R}}(n, d)$ *be the maximum cardinality of a code* $\mathcal{C} \subseteq \mathbb{F}_{q^m}^n$ *with minimum rank distance* $\mathrm{d}_{\mathsf{R}}(\mathcal{C}) = d$*. Then,*

$$\frac{q^{mn}}{\left|\mathcal{B}_{\mathsf{R}}^{(d-1)}\right|} \leqslant A_{q^m}^{\mathsf{R}}(n, d) \leqslant \frac{q^{mn}}{\left|\mathcal{B}_{\mathsf{R}}^{(t)}\right|} \ ,$$

where $t = \left\lfloor \frac{d-1}{2} \right\rfloor$ *and* $\left|\mathcal{B}_{\mathsf{R}}^{(\tau)}\right|$ *is a set (often called* ball*) of all the vectors of rank distance at most* τ *to a fixed vector* $\boldsymbol{b} \in \mathbb{F}_{q^m}^n$ *(e.g.,* $\boldsymbol{b} = \boldsymbol{0}$*), i.e.,*

$$\begin{aligned} &\mathcal{B}_{\mathsf{R}}^{(\tau)} := \left\{ \boldsymbol{a} \in \mathbb{F}_{q^m}^n \mid \mathrm{wt}_{\mathsf{R}}(\boldsymbol{a}) \leqslant \tau \right\} \textit{ and} \\ &\left|\mathcal{B}_{\mathsf{R}}^{(\tau)}\right| = \sum_{i=0}^{\tau} \begin{bmatrix} m \\ i \end{bmatrix}_q \prod_{j=0}^{i-1} (q^n - q^j) \textit{ with } \begin{bmatrix} m \\ i \end{bmatrix}_q = \prod_{j=0}^{i-1} \frac{q^m - q^j}{q^i - q^j} \ . \end{aligned}$$

2.5.3 Sum-Rank Metric

The sum-rank metric was first considered in coding for MIMO (multiple-input multiple-output) block-fading channels [EGH03; LK05] and the design of PSK-AM (phase-shift keying with amplitude modulation) constellations [Lu06]. It was then explicitly introduced in multi-shot network coding [NU09]. An explicit construction of optimal space-time codes in terms of *rate-diversity trade-off* from sum-rank metric codes over a finite field was first given in [SK21]. Additionally, sum-rank metric codes have been considered in applications such as network streaming [MBK16], distributed storage systems [MK19c; MN20; CMST21] and post-quantum secure code-based cryptosystems [D'A22; HBH22]. Extensive research has been done in recent years in fundamental coding-theoretical properties of sum-rank metric codes, e.g., [Mar19; BGR21; OPB21; CGL$^+$22; OLW22; OLW23], constructions of perfect/optimal/systematic sum-rank metric codes [Mar20; AMN20; Mar22; ALNW22; CD22].

Given an ordered partition $\boldsymbol{n}_\ell = (n_1, \ldots, n_\ell)$ of $n \in \mathbb{N}$, we write a vector $\boldsymbol{a}$ of length n with respect to $\boldsymbol{n}_\ell$ as

$$\boldsymbol{a} = (\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots, \boldsymbol{a}_\ell) \ ,$$

where $\boldsymbol{a}_i$ is of length n_i, for all $i \in [\ell]$.

Definition 2.18 (Sum-rank metric)**.** *The* sum-rank weight *on* $\mathbb{F}_{q^m}^n$*, w.r.t. an ordered partition* $\boldsymbol{n}_\ell = (n_1, \ldots, n_\ell)$ *of* n*, is defined as*

$$\begin{aligned} \mathrm{wt}_{\mathsf{SR},\boldsymbol{n}_\ell}(\cdot) \ : \ \mathbb{F}_{q^m}^n &\to \mathbb{N} \\ \boldsymbol{a} &\mapsto \sum_{i=1}^{\ell} \mathrm{rank}_q(\boldsymbol{a}_i) \ . \end{aligned}$$

The sum-rank distance *between two vectors is defined as*

$$\begin{aligned} \mathrm{d}_{\mathsf{SR},\boldsymbol{n}_\ell}(\cdot,\cdot) \ : \ \mathbb{F}_{q^m}^n \times \mathbb{F}_{q^m}^n &\to \mathbb{N} \\ \boldsymbol{a}, \boldsymbol{b} &\mapsto \mathrm{wt}_{\mathsf{SR},\boldsymbol{n}_\ell}(\boldsymbol{a} - \boldsymbol{b}) \ . \end{aligned}$$

For a code $\mathcal{C} \subseteq \mathbb{F}_{q^m}^n$*, its* minimum sum-rank distance *is*

$$\begin{aligned} \mathrm{d}_{\mathsf{SR},\boldsymbol{n}_\ell}(\mathcal{C}) &:= \min_{\substack{\boldsymbol{c}_1,\boldsymbol{c}_2 \in \mathcal{C} \\ \boldsymbol{c}_1 \neq \boldsymbol{c}_2}} \mathrm{d}_{\mathsf{SR},\boldsymbol{n}_\ell}(\boldsymbol{c}_1, \boldsymbol{c}_2) \\ &= \min_{\boldsymbol{0} \neq \boldsymbol{c} \in \mathcal{C}} \mathrm{wt}_{\mathsf{SR},\boldsymbol{n}_\ell}(\boldsymbol{c}) \ (\textit{If } \mathcal{C} \textit{ is linear}) \ . \end{aligned}$$

It is known that for $\ell = 1$, the sum-rank metric coincides with the rank metric, and for $\ell = n$, the sum-rank metric is the Hamming metric [MSK+22, Proposition 1.4, 1.5]. The following lemma gives a relation among the Hamming, the sum-rank and the rank weights of a fixed vector $\boldsymbol{x} \in \mathbb{F}_{q^m}^n$.

Lemma 2.4. *For a vector* $\boldsymbol{x} \in \mathbb{F}_{q^m}^n$ *and any ordered partition* $\boldsymbol{n}_\ell = (n_1, \ldots, n_\ell)$ *of* n*,* $\mathrm{wt}_{\mathsf{R}}(\boldsymbol{x}) \leqslant \mathrm{wt}_{\mathsf{SR},\boldsymbol{n}_\ell}(\boldsymbol{x}) \leqslant \mathrm{wt}_{\mathsf{H}}(\boldsymbol{x})$.

Proof. We first show that $\mathrm{wt}_{\mathsf{SR},\boldsymbol{n}_\ell}(\boldsymbol{x}) \leqslant \mathrm{wt}_{\mathsf{H}}(\boldsymbol{x})$. Consider $\boldsymbol{x} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_\ell) \in \mathbb{F}_{q^m}^n$ with $\mathrm{wt}_{\mathsf{H}}(\boldsymbol{x}) = n - t = \sum_{i=1}^{\ell} n_i - t_i$ where t_i is the number of zero entries in $\boldsymbol{x}_i$ and $\sum_{i=1}^{\ell} t_i = t$. For any $i \in [\ell]$, $\mathrm{rank}_q(\boldsymbol{x}_i) \leqslant \mathrm{wt}_{\mathsf{H}}(\boldsymbol{x}_i)$ since the rank of a matrix is at most the number of its nonzero columns. By the definition of $\mathrm{wt}_{\mathsf{SR},\boldsymbol{n}_\ell}$, we have $\mathrm{wt}_{\mathsf{SR},\boldsymbol{n}_\ell}(\boldsymbol{x}) = \sum_{i=1}^{\ell} \mathrm{rank}_q(\boldsymbol{x}_i) \leqslant \sum_{i=1}^{\ell} n_i - t_i = n - t = \mathrm{wt}_{\mathsf{H}}(\boldsymbol{x})$.

Now we show $\mathrm{wt}_{\mathsf{R}}(\boldsymbol{x}) \leqslant \mathrm{wt}_{\mathsf{SR},\boldsymbol{n}_\ell}(\boldsymbol{x})$. Suppose $\mathrm{wt}_{\mathsf{SR},\boldsymbol{n}_\ell}(\boldsymbol{x}) = t = \sum_{i=1}^{\ell} t_i$, which means each $\boldsymbol{x}_i$ has t_i $\mathbb{F}_q$-linearly independent entries for $i \in [\ell]$. Then $\boldsymbol{x}$ has at most t $\mathbb{F}_q$-linearly independent entries, which corresponds to the rank weight of $\boldsymbol{x}$. □

It can be seen that $\mathbb{F}_{q^m}^n$ is isometric to $\mathbb{F}_q^{m \times n}$ or $\mathbb{F}_q^{n \times m}$ as $\mathbb{F}_q$-vector spaces. The definition of the sum-rank metric on $\mathbb{F}_q^{m \times n}$ and $\mathbb{F}_q^{n \times m}$ follows naturally from Definition 2.18. For later usage, we give the definition of sum-rank metric on these matrix spaces in the following. For simplicity, we abuse the notation $\mathrm{wt}_{\mathsf{SR},\boldsymbol{n}_\ell}$ for the matrix space.

Let $\ell \in \mathbb{N}$ and $\boldsymbol{n}_\ell = (n_1, \ldots, n_\ell) \in \mathbb{N}^\ell$ be an ordered partition of $n = \sum_{l=1}^{\ell} n_l$. For matrices

$$\boldsymbol{A} = \begin{pmatrix} \boldsymbol{A}_1 & \boldsymbol{A}_2 & \ldots & \boldsymbol{A}_\ell \end{pmatrix} \in \mathbb{F}_q^{m \times n} \qquad \text{or} \qquad \boldsymbol{B} = \begin{pmatrix} \boldsymbol{B}_1 \\ \boldsymbol{B}_2 \\ \vdots \\ \boldsymbol{B}_\ell \end{pmatrix} \in \mathbb{F}_q^{n \times m} ,$$

where $\boldsymbol{A}_i \in \mathbb{F}_q^{m \times n_i}$ and $\boldsymbol{B}_i \in \mathbb{F}_q^{n_i \times m}$, $i \in [\ell]$, we say $\boldsymbol{A}$ has a *column-wise* partition with respect to $\boldsymbol{n}_\ell$ and $\boldsymbol{B}$ has a *row-wise* partition w.r.t. $\boldsymbol{n}_\ell$. The sum-rank weights of $\boldsymbol{A}$ and $\boldsymbol{B}$ w.r.t. $\boldsymbol{n}_\ell$ are, respectively,

$$\mathrm{wt}_{\mathsf{SR},\boldsymbol{n}_\ell}(\boldsymbol{A}) := \sum_{l=1}^{\ell} \mathrm{rank}(\boldsymbol{A}_l) \qquad \text{and} \qquad \mathrm{wt}_{\mathsf{SR},\boldsymbol{n}_\ell}(\boldsymbol{B}) := \sum_{l=1}^{\ell} \mathrm{rank}(\boldsymbol{B}_l) .$$

We can find the following relation between the sum-rank weight and the rank of a matrix.

Lemma 2.5. *For a matrix $\boldsymbol{A} \in \mathbb{F}_q^{m \times n}$ and an ordered partition $\boldsymbol{n}_\ell = (n_1, \ldots, n_\ell)$ of n, $\mathrm{rank}(\boldsymbol{A}) \leqslant \mathrm{wt}_{\mathsf{SR},\boldsymbol{n}_\ell}(\boldsymbol{A}) \leqslant \ell \cdot \mathrm{rank}(\boldsymbol{A})$. Similarly, for a matrix $\boldsymbol{B} \in \mathbb{F}_q^{n \times m}$, $\mathrm{rank}(\boldsymbol{B}) \leqslant \mathrm{wt}_{\mathsf{SR},\boldsymbol{n}_\ell}(\boldsymbol{B}) \leqslant \ell \cdot \mathrm{rank}(\boldsymbol{B})$.*

Proof. Denote by $\langle \boldsymbol{A} \rangle_{\mathsf{c}}$ the column space of a matrix $\boldsymbol{A}$. For the first inequality,

$$\begin{aligned} \mathrm{rank}(\boldsymbol{A}) = \dim(\langle \boldsymbol{A} \rangle_{\mathsf{c}}) &= \dim(\langle \boldsymbol{A}_1 \rangle_{\mathsf{c}} + \cdots + \langle \boldsymbol{A}_\ell \rangle_{\mathsf{c}}) \\ &\leqslant \dim(\langle \boldsymbol{A}_1 \rangle_{\mathsf{c}}) + \cdots \dim(\langle \boldsymbol{A}_\ell \rangle_{\mathsf{c}}) \\ &= \mathrm{rank}(\boldsymbol{A}_1) + \cdots + \mathrm{rank}(\boldsymbol{A}_\ell) = \mathrm{wt}_{\mathsf{SR},\boldsymbol{n}_\ell}(\boldsymbol{A}) . \end{aligned}$$

For the second inequality,

$$\mathrm{wt}_{\mathsf{SR},\boldsymbol{n}_\ell}(\boldsymbol{A}) = \sum_{i=1}^{\ell} \mathrm{rank}(\boldsymbol{A}_i) \leqslant \sum_{i=1}^{\ell} \mathrm{rank}(\boldsymbol{A}) = \ell \cdot \mathrm{rank}(\boldsymbol{A}) .$$

For the matrix $\boldsymbol{B} \in \mathbb{F}_q^{n \times m}$ with a row-wise partition, the proof is similar, by considering the row space of $\boldsymbol{B}$ and $\boldsymbol{B}_i$'s. □

With the relation between the sum-rank metric and the Hamming metric in Lemma 2.4, the following Singleton bound for sum-rank-metric codes can be easily derived from the Singleton bound for Hamming-metric codes in Theorem 2.13.

Theorem 2.17 (Singleton bound in the sum-rank metric [MSK+22, Theorem 1.4])**.** *Let $\boldsymbol{n}_\ell$ be an ordered partition of $n \in \mathbb{N}$. For a code $\mathcal{C} \subseteq \mathbb{F}_{q^m}^n$ (linear or non-linear) with minimum sum-rank distance $\mathrm{d}_{\mathsf{SR},\boldsymbol{n}_\ell}(\mathcal{C}) = d$,*

$$|\mathcal{C}| \leqslant q^{m(n-d+1)} .$$

If $\mathcal{C}$ is $\mathbb{F}_{q^m}$-linear, then its dimension k over $\mathbb{F}_{q^m}$ fulfills

$$k \leqslant n - d + 1 .$$

The equality holds in both equations if and only if $\mathcal{C}\boldsymbol{A} := \{\boldsymbol{c}\boldsymbol{A} \mid \boldsymbol{c} \in \mathcal{C}\}$ is an MDS code,

i.e., $\mathrm{d}_{\mathsf{H}}(\mathcal{C}\boldsymbol{A}) = d$*, for all* $\boldsymbol{A} = \mathrm{diag}(\boldsymbol{A}_1, \ldots, \boldsymbol{A}_\ell) \in \mathbb{F}_q^{n\times n}$ *where every* $\boldsymbol{A}_i \in \mathbb{F}_q^{n_i\times n_i}, i \in [\ell]$ *is invertible.*

A code whose length, cardinality and minimum sum-rank distance fulfilling the Singleton bound is a *maximum sum-rank distance* (MSRD) code. *Linearized Reed-Solomon codes* [Mar18; CD18] are a class of MSRD codes which gained a lot of research interest. They are evaluation codes based on skew polynomials.

Theorem 2.18 (Sphere-packing and Gilbert-Varshamov bounds in sum-rank metric [BGR21])**.** *Let* $\boldsymbol{n}_\ell$ *be an ordered partition of* $n \in \mathbb{N}$ *and* $A_{q^m}^{\mathsf{SR}}(n, d)$ *be the maximum cardinality of a code* $\mathcal{C} \subseteq \mathbb{F}_{q^m}^n$ *with minimum sum-rank distance* $\mathrm{d}_{\mathsf{SR},\boldsymbol{n}_\ell}(\mathcal{C}) = d$*. Then,*

$$\frac{q^{mn}}{\left|\mathcal{B}_{\mathsf{SR}}^{(d-1)}\right|} \leqslant A_{q^m}^{\mathsf{SR}}(n, d) \leqslant \frac{q^{mn}}{\left|\mathcal{B}_{\mathsf{SR}}^{(t)}\right|} \ ,$$

where $t = \left\lfloor \frac{d-1}{2} \right\rfloor$ *and* $\left|\mathcal{B}_{\mathsf{SR}}^{(\tau)}\right|$ *is a set (often called* ball*) of all the vectors of sum-rank distance at most* τ *to a fixed vector in* $\mathbb{F}_{q^m}$ *(e.g.,* $\mathbf{0}$*), i.e.,*

$$\mathcal{B}_{\mathsf{SR}}^{(\tau)} := \left\{ \boldsymbol{a} \in \mathbb{F}_{q^m}^n \mid \mathrm{wt}_{\mathsf{SR},\boldsymbol{n}_\ell}(\boldsymbol{a}) \leqslant \tau \right\} \text{ and}$$
$$\left|\mathcal{B}_{\mathsf{SR}}^{(\tau)}\right| = \sum_{s=0}^{\tau} \sum_{\substack{(s_1,\ldots,s_\ell)\in\mathbb{N}^\ell \\ s_1+\cdots+s_\ell=s}} \prod_{i=1}^{\ell} \begin{bmatrix} n_i \\ s_i \end{bmatrix}_q \prod_{j=0}^{s_i-1} (q^m - q^j) \ . \tag{2.12}$$

Due to the second sum over all the ordered partitions of s in (2.12), computing the ball size is expensive. An efficient algorithm is given in [PRR22, Algorithm 1] for computing the $\left|\mathcal{B}_{\mathsf{SR}}^{(\tau)}(\mathbf{0})\right|$ with complexity $O\big(\tau\big(\ell d^3 + d^4(\ell m + n)\log(q)\big)\big)$. Simplified forms and asymptotic behaviors of the sphere-packing bound and the Gilbert-Varshamov bound in the sum-rank metric are given in [OPB21].

3

Dual-Containing Polycyclic Codes over Rings based on Skew Polynomials

Finite rings are considered to be possible alphabets for linear codes first by Assmus Jr. and Mattson [AM63]. Blake investigated in [Bla72] the structure of cyclic codes over $\mathbb{Z}_m$ and studied in [Bla75] the analogues to Hamming, Reed-Solonmon and BCH codes over $\mathbb{Z}_{p^r}$. Spiegel [Spi77; Spi78] generalized Blake's results to any integer ring $\mathbb{Z}_m$ by using the Chinese Remainder Theorem. The interest in codes over finite rings has been evoked since the works by Carlderbank et al. [CHK$^+$93; HKC$^+$94], which use linear codes over $\mathbb{Z}_4$ to explain the duality between the nonlinear binary Kerkock and Preparata codes. The works by Wood [Woo99; Woo08; Woo09] laid a foundation of algebraic coding theory over finite rings by extending the two classical theorems by MacWilliams [Mac61; Mac62] to codes over finite rings. The extension theorem is also known as the equivalence theorem and the MacWilliams identities deal with the relation between the Hamming weight enumerators of a linear code and its dual.

Self-dual codes (or in general, dual-containing codes) have attracted a lot of research interest since the work by Calderbank et al. [CRSS98], which transformed the problem of constructing quantum error-correcting (QEC) codes into the problem of finding classical additive codes which are dual-containing. Several QEC codes have been constructed from classical codes, such as BCH codes [AKS07], Reed-Solomon codes [LXW08], Reed-Muller codes [Ste99] and algebraic geometric codes [CLX05]. Many good QEC codes have been constructed from cyclic codes over finite rings [QMG09; KZ11; GG14; TZKD16; BDUY19]. Constacyclic codes and negacyclic codes, as generalizations of cyclic codes (cf. Remark 2.1), have also been used to construct quantum codes [CLZ15; GW18; WLLG20; AIP$^+$21].

Skew-cyclic codes (also called θ-cyclic code) are a class of polycyclic codes $\mathcal{C}(g, f)$ where g, f are polynomials in a skew polynomial ring $\mathcal{A}[X;\theta]$ and $f = X^n - 1$, so that the *θ-cyclic shift* of any codeword in a skew-cyclic code $\mathcal{C}$ is also a codeword, i.e.,

$$(\theta(c_{n-1}), \theta(c_0), \ldots, \theta(c_{n-2})) \in \mathcal{C} \text{ for all } (c_0, c_1, \ldots, c_{n-1}) \in \mathcal{C} \ .$$

The concept of skew-cyclic codes was introduced over finite fields by Boucher, Geiselmann and Ulmer [BGU07]. Analogue to Definition 2.15 of the polycyclic codes, each skew-cyclic code corresponds to a right divisor g of $f = X^n - 1$. Since skew polynomials do not necessarily have unique irreducible factorizations, a polynomial $X^n - 1$ may have a considerable number of distinct right divisors of the same degree, which leads to many skew cyclic codes. Therefore, there is better chance to obtain codes with good parameters. This was one motivation of [BGU07] to introduce the notion of skew-cyclic codes. The works [BSU08; BU09b; BU09a]

further investigate skew-cyclic codes over finite rings. The notions of *skew-constacyclic* and *skew-negacyclic* are the generalizations with $f = X^n - a, a \in \mathcal{A}^*$ and $f = X^n + 1$, respectively. The skew polynomials ring $\mathcal{A}[X;\theta,\delta]$ with nonzero derivation was first considered by Boucher and Ulmer [BU14a] to construct (θ,δ)-*cyclic codes*, where the following (θ,δ)-*cyclic shift* of a codeword is also a codeword:

$$(\theta(c_{n-1}) + \delta(c_0), \theta(c_0) + \delta(c_1), \ldots, \theta(c_{n-2}) + \delta(c_{n-1})) \in \mathcal{C} \text{ for all } (c_0, c_1, \ldots, c_{n-1}) \in \mathcal{C}.$$

Boulagouaz and Leroy [BL13] generalized the notion to (θ,δ)-*polycyclic codes*. These works both considered codes over finite fields. Since the work [SB18] by Sharma and Bhaintwal, several works also investigated the (θ,δ)-polycyclic codes over finite rings [MGLF21; ST21; MGF22; PP22; Sup23]. Recently, θ-cyclic codes over various rings have been used to construct quantum codes [DBU+21; VPIS22; PVS23], as well as (θ,δ)-polycyclic codes [PIP22].

This chapter considers constructions of dual-containing codes over finite commutative rings from skew polynomials with derivations. We first introduce in Section 3.1 the rings over which we search for dual-containing codes. In Section 3.2, we define the (θ,δ)-polycyclic codes and introduce some properties regarding the codes and their dual codes. We then present in Section 3.3 an algorithm using Gröbner bases to compute all the dual-containing (θ,δ)-polycyclic codes. The resulting codes found by the algorithm are presented in Section 3.4.

This chapter is based on the work [LOU23], submitted to Advances in Mathematics of Communications.

3.1 Base Rings, Endomorphisms and Derivations

Throughout the chapter, we denote by $\mathcal{A}$ a finite commutative ring, θ an endomorphism of $\mathcal{A}$, δ a θ-derivation of $\mathcal{A}$, and $\mathcal{R} = \mathcal{A}[X;\theta,\delta]$ a skew polynomial ring.

For the base ring $\mathcal{A}$ over which dual-containing codes are constructed, we consider the finite commutative ring $\mathcal{A}$ that is a *free $\mathcal{B}$-algebra*, denoted by $\mathcal{A} = \mathcal{B}[\beta_1, \beta_2, \ldots, \beta_s]$, where $\beta_1, \beta_2, \ldots, \beta_s$ form a *basis* of $\mathcal{A}$ over $\mathcal{B}$. This means that any element $a \in \mathcal{A}$ can be written as a $\mathcal{B}$-linear combinations of $\beta_1, \beta_2, \ldots, \beta_s$, i.e., $a = b_1\beta_1 + \cdots + b_s\beta_s$ for some $b_1, \ldots, b_s \in \mathcal{B}$.

In this chapter we consider the following rings $\mathcal{A}$ of order 4, which are all free $\mathbb{F}_2$-algebras, to construct self-dual codes $\mathcal{C} \subseteq \mathcal{A}^n$:

- $\mathcal{A} = \mathbb{F}_2[v]$ where $v^2 = v$,
- $\mathcal{A} = \mathbb{F}_2[u]$ where $u^2 = 0$,
- $\mathcal{A} = \mathbb{F}_4 = \mathbb{F}_2[\alpha]$ where $\alpha^2 = \alpha + 1$.

Here we add some insights on the notations. For the finite field $\mathbb{F}_4$, it is well-known that it is also a quotient ring $\mathbb{F}_2[x]/\langle x^2 - x - 1\rangle$ where $x^2 - x - 1$ is an irreducible polynomial in $\mathbb{F}_2[x]$, and $\{1, \alpha\}$ constitutes a basis of $\mathbb{F}_4$ over $\mathbb{F}_2$ (α is often called the primitive element in the context of finite fields). Analogously, the ring $\mathbb{F}_2[v]$ (resp. $\mathbb{F}_2[u]$) is the quotient ring $\mathbb{F}_2[x]/\langle x^2 - x\rangle$ ($\mathbb{F}_2[x]/\langle x^2\rangle$), and v (u) constitutes a basis of the ring over $\mathbb{F}_2$. Since $x^2 - x$ (resp. x^2) is not irreducible in $\mathbb{F}_2[x]$, there are zero divisors $\{v, v+1\}$ ($\{u\}$) in the ring.

For the endomorphisms θ and θ-derivations δ of $\mathcal{A}$ that we use to define the skew polynomial ring $\mathcal{A}[X;\theta,\delta]$, we consider those that can be written as *polynomial maps* in the subring $\mathcal{B} \subseteq \mathcal{A}$.

Definition 3.1 (Polynomial maps)**.** *A* polynomial map *on a ring $\mathcal{B}$ is a map*

$$f \ : \ \mathcal{B} \to \mathcal{B}$$
$$x \mapsto \sum_{i=0}^{s} b_i x^i,$$

where $s \in \mathbb{N}$ and $b_i \in \mathcal{B}$.

For the rings $\mathbb{F}_2[v], \mathbb{F}_2[u]$ and $\mathbb{F}_2[\alpha]$, the endomorphisms θ and derivations δ which are polynomial maps in the subring $\mathcal{B} = \mathbb{F}_2$ are listed in Table 3.1. Note that since $\theta(0) = 0, \theta(1) = 1, \delta(0) = 0$ and $\delta(1) = 0$, it is sufficient to give the map on the basis (v, u or α) of $\mathcal{A}$ over $\mathcal{B}$ to determine the value $\theta(a)$ and $\delta(a)$ for all $a \in \mathcal{A}$.

Table 3.1: Endomorphisms and derivations of the rings. The gray cells indicate the inner derivations.

(a) The endomorphisms θ and derivations δ of the ring $\mathbb{F}_2[v]$.

$\mathbb{F}_2[v]$	Automorphism		Endomorphism	
	$\theta_1 = \mathrm{id}$	$\theta_2 : v \mapsto v+1$	$\theta_3 : v \mapsto 0$	$\theta_4 : v \mapsto 1$
$\delta_1 = 0$	$v \mapsto 0$	$v \mapsto 0$	$v \mapsto 0$	$v \mapsto 0$
δ_2		$v \mapsto 1$		
δ_3		$v \mapsto v$	$v \mapsto v$	
δ_4		$v \mapsto v+1$		$v \mapsto v+1$

(b) The endomorphisms θ and derivations δ of the ring $\mathbb{F}_2[u]$.

$\mathbb{F}_2[u]$	Automorphism	Endomorphism
	$\theta_1 = \mathrm{id}$	$\theta_2 : u \mapsto 0$
$\delta_1 = 0$	$u \mapsto 0$	$u \mapsto 0$
δ_2	$u \mapsto 1$	
δ_3	$u \mapsto u$	$u \mapsto u$
δ_4	$u \mapsto u+1$	

(c) The endomorphisms θ and derivations δ of the ring $\mathbb{F}_2[\alpha]$.

$\mathbb{F}_2[\alpha]$	Automorphism	
	$\theta_1 = \mathrm{id}$	$\theta_2 : \alpha \mapsto \alpha+1$
$\delta_1 = 0$	$\alpha \mapsto 0$	$\alpha \mapsto 0$
δ_2		$\alpha \mapsto 1$
δ_3		$\alpha \mapsto \alpha$
δ_4		$\alpha \mapsto \alpha+1$

The following examples show that a map which is a polynomial map in $\mathcal{B}$ is not necessarily a polynomial map in $\mathcal{A} \supset \mathcal{B}$. The principle to determine whether a map θ (or δ) is a polynomial map in $\mathcal{A}$ is to check whether θ can be written as $\theta(x) = \sum_{i \in \mathbb{N}} a_i x^i$ with fixed $a_i \in \mathcal{A}$ for all $x \in \mathcal{A}$.

Example 3.1. *Consider the ring $\mathcal{A} = \mathbb{F}_2[v]$ of order 4. As listed in Table 3.1a, there are two automorphisms $\theta_1 = \mathrm{id}$ and θ_2, and two non-trivial endomorphisms θ_3 and θ_4.*

The automorphism $\theta_1 = \mathrm{id}$ is trivially a polynomial map on $\mathcal{A}$. Suppose that the automorphism θ_2 is a polynomial map on $\mathcal{A}$ such that for any $x \in \mathcal{A}$,

$$\theta_2 : x \mapsto \sum_{i \in \mathbb{N}} \underbrace{(b_{i,0} + b_{i,1} v)}_{\in \mathcal{A}} x^i \ = \ \sum_{i \in \mathbb{N}} b_{i,0} x^i + \sum_{i \in \mathbb{N}} b_{i,1} v x^i \qquad (b_{i,j} \in \mathbb{F}_2) \ .$$

Then $\theta_2(0) = 0 \implies b_{0,0} = 0$. Since $b_{i,j} \in \{0, 1\}$, $\theta_2(v)$ is a sum of positive powers of v. Since $v^2 = v$, we have that $\theta_2(v)$ is a sum of v, which is either v or 0 in this ring of characteristic 2. However, since $\theta_2(v) = v + 1$, we conclude that θ_2 is not a polynomial map on $\mathcal{A}$.

Example 3.2. *Consider the ring $\mathcal{A} = \mathbb{F}_2[u]$. As listed in Table 3.1b, the only automorphism of $\mathcal{A}$ is $\theta_1 = \mathrm{id}$, which is a polynomial map in $\mathcal{A}$. Suppose that a θ_1-derivation δ of $\mathcal{A}$ is a polynomial map in $\mathcal{A}$ such that for any $x \in \mathcal{A}$,*

$$\delta : x \mapsto \sum_{i=0}^{t} a_i x^i \qquad (a_i \in \mathcal{A}) \ .$$

Since $\delta(1) = 0$, we must have $a_0 = 0$ in the polynomial map. From $u^2 = 0$, we obtain $\delta(u) = \sum_{i=1}^{t} a_i u^i = a_1 u$. Write $a_1 = b_{1,0} + b_{1,1}u \in \mathcal{A}$ with some $b_{1,0}, b_{1,1} \in \mathbb{F}_2$. Then $\delta(u) = b_{1,0}u + b_{1,1}u^2 = b_{1,0}u$, which can never be $u+1$ or 1. Hence, $\delta_2(u) = 1$ and $\delta_4(u) = u+1$ are not polynomial maps on $\mathcal{A}$.

3.2 Polycyclic Codes over Rings based on Skew Polynomials

Denote by $\langle f \rangle_l \subseteq \mathcal{R}$ a left ideal in $\mathcal{R}$ generated by $f \in \mathcal{R}$. The quotient ring (or factor ring) of $\mathcal{R}$ modulo the left ideal $\langle f \rangle_l$ is defined as

$$\mathcal{R}/\langle f \rangle_l := \{ h \bmod f \mid h \in \mathcal{R} \} \ ,$$

where $h \bmod f$ gives the remainder of right dividing h by f (applying Algorithm 2.3).

Proposition 3.1. *Let $\mathcal{I}$ be a left ideal in $\mathcal{R}/\langle f \rangle_l$. Then*

(i) There is a unique monic polynomial $g \in \mathcal{I}$ of minimal degree.

(ii) $\mathcal{I}$ is principle with a generator g.

(iii) $g \mid_r f$ in $\mathcal{R}$.

Proof. (i) Suppose that there are two monic skew polynomials with minimum degree in $\mathcal{I}$, say h and g and $g \neq h$. Since $\mathcal{I}$ forms an additive group, $r = g - h \in \mathcal{I}$ and $\deg(r) < \deg(g) = \deg(h)$, which is contrary to the minimal degree assumption on g and h.

(ii) Suppose that there exists an $h \in \mathcal{I}$ which is not a right multiple of g. Then $h = mg + r$ for some $m \in \mathcal{R}$, $r \in \mathcal{I}$ and $\deg(r) < \deg(g)$. This contradicts the degree minimality of g.

(iii) Suppose that $g \nmid_r f$. We can then write $f = mg + r$ for some $m, r \in \mathcal{R}$ and $\deg(r) < \deg(g)$. Since g is a generator of $\mathcal{I}$, $mg \in \mathcal{I}$ and $r = f - mg \equiv -mg \bmod f$, which means r is an additive inverse of mg in $\mathcal{R}/\langle f \rangle_l$ and therefore $r \in \mathcal{I} \subseteq \mathcal{R}/\langle f \rangle_l$. However, this contradicts the degree minimality of g. □

The results above shows that every left ideal in the quotient ring $\mathcal{R}/\langle f \rangle_l$ is a principle left ideal and different left ideals are generated by different right factors of f.

Analogue to the polycyclic codes (Definition 2.15) which are ideals in the quotient ring $\mathcal{A}[x]/\langle f \rangle$ for some $f \in \mathcal{A}[x]$, for a non-commutative skew polynomial ring, we define the polycyclic codes as the left ideals in the quotient ring $\mathcal{R}/\langle f \rangle_l$ for some $f \in \mathcal{R}$.

Definition 3.2 ((θ, δ)-polycyclic code)**.** *Let $f \in \mathcal{R}$ be a monic skew polynomial with a right divisor $g \in \mathcal{R}$. A* (θ, δ)-polycyclic code *(in short (θ, δ)-code) w.r.t. g, f is defined as a left ideal in $\mathcal{R}/\langle f \rangle_l$ generated by g, i.e.,*

$$\mathcal{C}(g, f) := \langle g \rangle_l / \langle f \rangle_l = \{ u \cdot g \bmod f \mid u \in \mathcal{R} \} \ .$$

The polynomial g is a generator polynomial *of $\mathcal{C}(g, f)$.*

*Let $n = \deg(f)$ and $k = n - \deg(g)$. A linear block code $\mathcal{C}[n, k]$ is a (θ, δ)-*code *if*

$$\mathcal{C}[n,k] := \left\{ \boldsymbol{c} = (c_0, c_1, \ldots, c_{n-1}) \mid c_0 + c_1 X + \cdots + c_{n-1} X^{n-1} \in \mathcal{C}(g,f) \right\} .$$

3.2.1 Module Codes

Linear codes of length n over a finite field $\mathbb{F}$ can be seen as subspaces of the vector spaces $\mathbb{F}^n$. Analogously, linear codes over a finite ring $\mathcal{A}$ can be seen as *submodules* of the *module* $\mathcal{A}^n := \{(a_0, a_1, \ldots, a_{n-1}) \mid a_i \in \mathcal{A}, \forall i \in [n]\}$. Informally speaking, the concept of *module* is a generalization of the vector space, in the sense that the set of scalars is a ring instead of a field.

Definition 3.3 (Module)**.** *Let $\mathcal{A}$ be a ring. A* left $\mathcal{A}$-module $\mathcal{M}$ *consists of an abelian group $(\mathcal{M}, +)$ and a* left scalar multiplication $\cdot : \mathcal{A} \times \mathcal{M} \to \mathcal{M}$ *such that for all $a, b \in \mathcal{A}$ and $x, y \in \mathcal{M}$,*

(i) $a \cdot (x + y) = a \cdot x$,

(ii) $(a + b) \cdot x = a \cdot x + b \cdot x$,

(iii) $(ab) \cdot x = a \cdot (b \cdot x)$,

(iv) $1 \cdot x = x$.

A right $\mathcal{A}$-module $\mathcal{M}$ *is defined similarly with a right scalar multiplication $\cdot : \mathcal{M} \times \mathcal{A} \to \mathcal{M}$.*

Vector spaces over a finite field $\mathbb{F}$ always have a *basis* (i.e., every element in the vector space is a unique $\mathbb{F}$-linear combination of the elements in the basis), and the dimension is unique. However, modules do not always have a basis. The modules that have a basis are called *free* modules.

Proposition 3.2. *Let $f \in \mathcal{R}$ be a monic skew polynomial. The quotient ring $\mathcal{R}/\langle f \rangle_l$ is a left $\mathcal{R}$-module and a free left $\mathcal{A}$-module. Moreover, $\mathcal{R}/\langle f \rangle_l \cong \mathcal{A}^n$.*

Proof. By definition, $\mathcal{R}/\langle f \rangle_l$ is a left $\mathcal{R}$-module. Since f is monic, we can perform the right division (Algorithm 2.3) on any element $u \in \mathcal{R}$ by f and obtain a unique remainder polynomial of degree $< n$. For any $r \in \mathcal{R}$ of $\deg(r) < n$, there exist $u, q \in \mathcal{R}$ (possibly infinite pairs) such that $u = qf + r$. Therefore, $\mathcal{R}/\langle f \rangle_l = \{r \in \mathcal{R} \mid \deg(r) < n\}$. It is easy to see that $\{r \in \mathcal{R} \mid \deg(r) < n\}$ is a free $\mathcal{A}$-module with a basis $(1, X, \ldots, X^{n-1})$ and

$$\left\{ (r_0, r_1, \ldots, r_{n-1}) \mid r_0 + r_1 X + \cdots + r_{n-1} X^{n-1} \in \mathcal{R}/\langle f \rangle_l \right\} = \mathcal{A}^n .$$

□

Proposition 3.3. *Let $f \in \mathcal{R}$ be a monic skew polynomial with a right divisor $g \in \mathcal{R}$. The (θ, δ)-code $\mathcal{C}(g, f)$ is a left $\mathcal{R}$-submodule of $\mathcal{R}/\langle f \rangle_l$ and a free left $\mathcal{A}$-submodule of dimension $k = \deg(f) - \deg(g)$.*

Proof. By definition, $\mathcal{C}(g, f) = \langle g \rangle_l / \langle f \rangle_l$ is a left $\mathcal{R}$-submodule of $\mathcal{R}/\langle f \rangle_l$. Since f is monic, the leading coefficient $\mathrm{lc}(g)$ is a right divisor of 1 and is therefore invertible. For any $w \in \mathcal{C}(g, f)$, $g \mid_r w$ and the quotient polynomial q is unique (Theorem 2.5) and of degree at most k. This implies that the $\mathcal{C}(g, f)$ is a free $\mathcal{A}$-submodule of dimension k with a basis $(g, Xg, \ldots, X^{k-1}g)$. □

The relations between the quotient ring $\mathcal{R}/\langle f\rangle_l$, the $\mathcal{A}$-module $\mathcal{A}^n$, the left ideal $\mathcal{C}(g,f)$ in $\mathcal{R}/\langle f\rangle_l$, and the block code $\mathcal{C}[n,k]$ are illustrated below.

$$\begin{array}{ccc} \mathcal{R}/\langle f\rangle_l & \cong & \mathcal{A}^n \\ \cap| & & \cap| \\ \mathcal{C}(g,f) & \cong & \mathcal{C}[n,k] \end{array}$$

The code $\mathcal{C}[n,k]$ has a generator matrix of the following form:

$$\boldsymbol{G} = \begin{pmatrix} g_0 & g_1 & \cdots & g_{n-k} & 0 & \cdots & \\ g_0^{\delta} & g_1^{\delta}+g_0^{\theta} & g_2^{\delta}+g_1^{\theta} & \cdots & g_{n-k}^{\theta} & 0 & \cdots \\ \vdots & \ddots & \ddots & \cdots & \ddots & \ddots & \\ g_0^{\delta^{k-1}} & & \cdots & & \cdots & & g_{n-k}^{\theta^{k-1}} \end{pmatrix}.$$

The rows are given by the coefficients of $g, Xg, \ldots, X^{k-1}g$, which can be computed using the rule $Xa = a^{\theta}X + a^{\delta}$ for $a \in \mathcal{A}$. In particular, the code is completely determined by g, θ and δ.

Example 3.3. *Let $f \in \mathcal{R}$ be a monic skew polynomial of degree 4 and $g = g_0 + g_1X$ be a right divisor of f. The (θ,δ)-code $\mathcal{C}[4,3] \cong \mathcal{C}(g,f)$ has a generator matrix*

$$\boldsymbol{G} = \begin{pmatrix} g_0 & g_1 & 0 & 0 \\ g_0^{\delta} & g_1^{\delta}+g_0^{\theta} & g_1^{\theta} & 0 \\ g_0^{\delta^2} & g_0^{\delta\theta}+g_0^{\theta\delta}+g_1^{\delta^2} & g_0^{\theta^2}+g_1^{\delta\theta}+g_1^{\theta\delta} & g_1^{\theta^2} \end{pmatrix}.$$

If θ is of the form $a \mapsto a^q$ and δ is an inner θ-derivation $a \mapsto \beta a - \theta(a)\beta$, which are the only possibilities if $\mathcal{A}$ is a finite field $\mathbb{F}_{q^m}$, then the entries of the generator matrix $\boldsymbol{G}$ become polynomial expressions in the coefficients of g, which allows sophisticated computations over $\mathcal{A}$. Hence, most studies on self dual (θ,δ)-code so far considered $\mathcal{A}$ to be a finite field.

3.2.2 Parity-Check Polynomials/Matrices of (θ,δ)-Codes

For $\mathcal{A} = \mathbb{F}_q$, a parity-check matrix of (θ,δ)-codes has been derived for $\delta = 0$ in [BU09a, Corollary 1] for general $\delta \neq 0$ in [BL13]. A later work [BD18] derived a parity-check matrix for $\mathcal{A}$ being a finite commutative rings. In [SB18], a parity-check matrix for (θ,δ)-codes $\mathcal{C}(g,f)$ over the ring $\mathcal{A} = \mathbb{Z}_4[u], u^2 = 1$ is studied when $f = hg$ is a central polynomial[1]. The works [BL13; BD18] used the framework of pseudo-linear transformations, while we only use the framework of skew polynomial rings to derive a parity-check matrix in this section.

In order to obtain a parity-check matrix for (θ,δ)-codes $\mathcal{C}(g,f)$, we make the additional assumption that there exists $\hbar \in \mathcal{R}$ such that $f = hg = g\hbar$ (i.e., g is a left and right divisor of f). This assumption is weaker than the assumption that f is central, which allows us to find more $g, f \in \mathcal{R}$ to construct dual-containing codes (see the $[6,4]$ example in Section 3.4.1).

In the rest of this chapter, we also make the assumptions in following proposition on the leading coefficients of $g, h, \hbar$.

Proposition 3.4. *For a (θ,δ)-codes $\mathcal{C}(g,f)$, where f is monic and g is a left and right divisor of f, i.e., $f = hg = g\hbar$ for some $h, \hbar \in \mathcal{R}$,*

[1] A polynomial $f \in \mathcal{R}$ is called *central* if $hf = fh$, for all $h \in \mathcal{R}$. Equivalently, a polynomial $f \in \mathcal{A}[X;\theta,\delta]$ is central if $Xf = fX$.

(i) we can assume w.l.o.g. that g and h are monic;

(ii) if θ is an automorphism, then we can also assume w.l.o.g. that $\hbar$ is monic.

Proof. Let $g = g_{n-k}X^{n-k} + \cdots + g_0$ and $h = h_kX^k + \cdots + h_0$. Since $f = hg$ and f is monic, the leading coefficient of $f = hg$ is $h_k\theta^k(g_{n-k}) = 1$, showing that h_k and $\theta^k(g_{n-k})$ are invertible. We can write

$$\begin{aligned} f &= \underbrace{(h_kX^k + \cdots + h_0)}_{h} \cdot \underbrace{(g_{n-k}X^{n-k} + \cdots + g_0)}_{g} \\ &= \underbrace{(h_kX^k + \cdots + h_0) \cdot g_{n-k}}_{\tilde{h}} \cdot \underbrace{g_{n-k}^{-1} \cdot (g_{n-k}X^{n-k} + \cdots + g_0)}_{\tilde{g}} \ . \end{aligned}$$

Note that $\tilde{g}$ is a monic polynomial. Since the endomorphism θ maps 1 to 1, the leading coefficient of $\tilde{h}\tilde{g}$ is the leading coefficient of $\tilde{h}$. Because $\tilde{h}\tilde{g} = f$ is monic, we obtain that $\tilde{h}$ is also a monic polynomial. The polynomials g differ from $\tilde{g}$ by multiplying an invertible element. It can be seen that any left multiple of g by a polynomial of degree $\leqslant k-1$ is also a left multiple of $\tilde{g}$ by another polynomial of degree $\leqslant k-1$ and vice versa. Therefore, $\mathcal{C}(g,f) = \mathcal{C}(\tilde{g},f)$ and we can assume w.l.o.g. that g is a monic polynomial. With the argument on $\tilde{h}$ above, we can assume that h is also monic.

We now show that $\hbar = \hbar_kX^k + \cdots + \hbar_0$ is monic if θ is an automorphism. The leading coefficient $\mathrm{lc}(f) = \mathrm{lc}(g\hbar) = g_{n-k}\theta^{n-k}(\hbar_k) = 1$. Since both f and g are monic, $\theta^{n-k}(\hbar_k)$ must be 1. If θ is an automorphism, we obtain that $\hbar_k = 1$. □

Proposition 3.4 (ii) implies that if θ is a non-trivial endomorphism (not an automorphism), the $\hbar$ in the decomposition $f = g\hbar$ may not be monic. In the following we give an example of such case.

Example 3.4. *Consider the ring $\mathcal{A} = \mathbb{F}_2[u]$ where $u^2 = 0$, the non-trivial endomorphism $\theta_2(u) = 0$ of $\mathcal{A}$ and the θ_2-derivation $\delta_3(u) = u$ (Table 3.1b). Let $\mathcal{R} = \mathcal{A}[X;\theta_2,\delta_3]$. For the skew polynomials $g = X^2 + uX + u + 1$ and $f = X^4 + (u+1)X^3 + X + u + 1$, it can be verified that g is a left and right divisor of f. For the case g being a right divisor, we found that f can be decomposed into $f = hg$ with the unique monic $h = X^2 + (u+1)X + 1$. For the case g being a left divisor, f can be decomposed into $f = g\hbar$ with $\hbar = (u+1)X^2 + (u+1)X + u + 1$ or $\hbar = (u+1)X^2 + X + u + 1$, neither of which is monic.*

The following shows that the polynomial $\hbar$ plays a role as a "parity-check polynomial".

Lemma 3.1 (Parity-check polynomial of a (θ,δ)-code)**.** *Let $f \in \mathcal{R}$ be a monic polynomial of degree n, $g \in \mathcal{R}$ be a monic left and right divisor of f of degree $n-k$ ($f = hg = g\hbar$ for some $h, \hbar \in \mathcal{R}$), and $\mathcal{C}[n,k] \cong \mathcal{C}(g,f)$ be a (θ,δ)-code as defined in Definition 3.2. A word $\boldsymbol{c} = (c_0, c_1, \ldots, c_{n-1}) \in \mathcal{A}^n$ is a codeword of $\mathcal{C}[n,k]$ if and only if the corresponding skew polynomial $c = c_0 + c_1X + c_{n-1}X^{n-1}$ fulfills $c\hbar = 0$ in $\mathcal{R}/\langle f\rangle_l$.*

Proof. The statement is trivial for $\boldsymbol{c} = \boldsymbol{0}$. Consider any nonzero $\boldsymbol{c} \in \mathcal{A}^n$. If $\boldsymbol{c} \in \mathcal{C}[n,k]$, then $c = wg$ for some $w \in \mathcal{R}$. Then $c\hbar = wg\hbar = w(hg) = 0 \bmod f$, which is equivalent to say that $c\hbar = 0$ in $\mathcal{R}/\langle f\rangle_l$. Conversely, if $c\hbar = 0$ in $\mathcal{R}/\langle f\rangle_l$, then $c\hbar = \tilde{w}f = \tilde{w}(g\hbar)$ for some $\tilde{w} \in \mathcal{R}$. Since f is monic and $\hbar$ is a right divisor of f, $\hbar$ is not a zero divisor in $\mathcal{R}$ and we obtain $c = \tilde{w}g$, showing that $\boldsymbol{c} \in \mathcal{C}[n,k]$. □

Lemma 3.2 (Parity-check matrix of a (θ, δ)-code)**.** *We follow the notations in Lemma 3.1. There is a matrix $\boldsymbol{M} \in \mathcal{A}^{n\times n}$ such that $\boldsymbol{cM} = \boldsymbol{0}$, $\forall \boldsymbol{c} \in \mathcal{C}[n,k]$, where the entries of $\boldsymbol{M}$ are images under θ and δ of the coefficients of $\hbar$ and g.*

Proof. Note that $c = c_0 + c_1X + c_{n-1}X^{n-1} \in \mathcal{C}(g,f)$ is the corresponding skew polynomial of $\boldsymbol{c} \in \mathcal{C}[n,k]$. By Lemma 3.1, we have $c\hbar = 0$ in $\mathcal{R}/\langle f\rangle_l$, which is equivalent to

$$c\hbar = \left(\sum_{i=0}^{n-1} c_iX^i\right)\hbar \equiv 0 \bmod f \ . \tag{3.1}$$

By applying the multiplication and the right module by f, we get a system of linear equations

$$\sum_{i=0}^{n-1} c_iM_{ij}X^j = 0, \ \forall j \in [0, n-1] \ , \tag{3.2}$$

where M_{ij} is the coefficient of X^j in $X^i \cdot \hbar \bmod f$, which is the image under θ and δ of the coefficients of $\hbar$ and f, $\forall i,j = 0,\dots,n-1$. Since $f = g\hbar$, the coefficients of f can be replaced by the coefficients of $\hbar$ and g. Let $\boldsymbol{M}$ be an $n \times n$ matrix whose entries are M_{ij}'s. It can be shown that

$$(3.2) \iff \boldsymbol{cM} = \boldsymbol{0} \ .$$

□

The following toy example provides some insights on how such a matrix $\boldsymbol{M}$ may look like.

Example 3.5. *Consider a ring $\mathcal{A}$, a skew polynomial ring $\mathcal{R} = \mathcal{A}[X;\theta,\delta]$, $f = X^3 + \sum_{i=0}^{2} f_iX^i$, $g = X^2 + g_1X + g_0$ and $\hbar = \hbar_1X + \hbar_0$ in $\mathcal{R}$ such that $f = g\hbar$. According to Lemma 3.1, $\boldsymbol{c} = (c_0,c_1,c_2) \in \mathcal{C}[3,1] \cong \mathcal{C}(g,f)$ if and only if $c\hbar \equiv 0 \bmod f$, where $c = c_0 + c_1X + c_2X^2$. Since*

$$\begin{aligned} c\hbar \bmod f = &\Big(c_2(\hbar_1^{\theta\delta} + \hbar_1^{\delta\theta} + \hbar_0^{\theta^2} - \hbar_1^{\theta^2}f_2) + c_1\hbar_1^{\theta}\Big)X^2 \\ &+ \Big(c_2(\hbar_1^{\delta^2} + \hbar_0^{\theta\delta} + \hbar_0^{\delta\theta} - \hbar_1^{\theta^2}f_1) + c_1(\hbar_1^{\delta} + \hbar_0^{\theta}) + c_0\hbar_1\Big)X \\ &+ c_2(\hbar_0^{\delta^2} - \hbar_1^{\theta^2}f_0) + c_1\hbar_0^{\delta} + c_0\hbar_0 \ , \end{aligned} \tag{3.3}$$

we obtain the following matrix $\boldsymbol{M} \in \mathcal{A}^{3\times 3}$ such that $\boldsymbol{cM} = \boldsymbol{0}$.

$$\boldsymbol{M} = \begin{pmatrix} \hbar_0 & \hbar_1 & 0 \\ \hbar_0^{\delta} & \hbar_1^{\delta} + \hbar_0^{\theta} & \hbar_1^{\theta} \\ \hbar_0^{\delta^2} - \hbar_1^{\theta^2}f_0 & \hbar_1^{\delta^2} + \hbar_0^{\theta\delta} + \hbar_0^{\delta\theta} - \hbar_1^{\theta^2}f_1 & \hbar_1^{\theta\delta} + \hbar_1^{\delta\theta} + \hbar_0^{\theta^2} - \hbar_1^{\theta^2}f_2 \end{pmatrix} . \tag{3.4}$$

Note that the entry M_{ij} in $\boldsymbol{M}$ corresponds to the coefficient of the term c_iX^j in the polynomial $c\hbar \bmod f$, e.g., M_{11} is the coefficient of c_1X in (3.3), which is $(\hbar_1^{\delta} + \hbar_0^{\theta})$.

3.2.3 Dual Codes of (θ, δ)-Codes

Most dual codes studied in the literature consider the *Euclidean inner product* to define duality. In this section, we consider a more general notion of inner product and duality.

Definition 3.4 (Hermitian inner product and Hermitian dual)**.** *Let σ be an automorphism of $\mathcal{A}$ of order*[2] *at most* 2*. The* σ-Hermitian inner product *of $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{A}^n$ is defined as $\langle \boldsymbol{x}, \boldsymbol{y} \rangle_\sigma := \sum_{i=1}^{n} x_i \sigma(y_i)$. The* σ-Hermitian dual code *of a code $\mathcal{C}$ is defined as*

$$\mathcal{C}^{\perp_\sigma} := \{ \boldsymbol{v} \mid \langle \boldsymbol{v}, \boldsymbol{c} \rangle_\sigma = 0, \ \forall \boldsymbol{c} \in \mathcal{C} \} \ .$$

A code is σ-dual-containing *if $\mathcal{C}^{\perp_\sigma} \subseteq \mathcal{C}$, and* σ-self dual *if $\mathcal{C} = \mathcal{C}^{\perp_\sigma}$. In particular, if $\sigma = \mathrm{id}$, we obtain the* Euclidean inner product *and the* Euclidean dual*. In this case, we omit the σ in the notation.*

Frobenius rings are promoted by Wood [Woo99] as the most appropriate rings for coding theory over finite rings, because two classical theorems of MacWilliams – the extension theorem (also known as the equivalence theorem) and the MacWilliams identities – generalize to Frobenius rings. Szabo and Wood [SW17] showed that dual codes have complementary cardinality for codes over finite Frobenius rings. Note that the rings $\mathbb{F}_2[v]$, $\mathbb{F}_2[u]$ and $\mathbb{F}_2[\alpha]$ are all Frobenius rings.

Theorem 3.1 (MacWilliams theorems for codes over finite Frobenius rings [Woo99])**.** *Let $\mathcal{C}$ be a linear code over a finite Frobenius ring $\mathcal{A}$, then*

- (Extension theorem) *Every isometric map $\mathcal{C} \to \mathcal{A}^n$ that preserves the Hamming weight of the code can be extended to a monomial transformation.*
- (MacWilliams identities) *The weight enumerator of the dual code is determined by the weight enumerator of the code.*

Lemma 3.3 (Complementary cardinality of dual codes [SW17, Theorem 3.6])**.** *Let $\mathcal{A}$ be a finite frobenius ring and σ be an automorphism of $\mathcal{A}$ of order at most* 2*. For a linear code $\mathcal{C}$ over $\mathcal{A}$, its σ-Hermitian dual code $C^{\perp_\sigma}$ has the complementary cardinality of the code, i.e., $|\mathcal{C}| \cdot |\mathcal{C}^{\perp_\sigma}| = |\mathcal{A}|^n$.*

The following is devoted to derive a generator matrix of the Euclidean dual and σ-Hermitian dual of the (θ, δ)-codes, respectively. For the case $\delta = 0$, the dual codes have been studied for $\mathcal{A}$ being $\mathbb{Z}_4$ [BSU08], $\mathbb{F}_4$ [BU09a], $\mathbb{F}_q$ [BU09a] and general finite rings [BU11; BD18]. For $\delta \neq 0$, the dual codes are much less studied.

Theorem 3.2 (Generator matrix of the Euclidean dual)**.** *Let $\mathcal{A}$ be a finite Frobenius ring. Let $\mathcal{C} = \mathcal{C}[n, k]$ be the linear block (θ, δ)-code as defined in Definition 3.2. Then, the Euclidean dual code $\mathcal{C}^\perp$ of $\mathcal{C}$ is a linear free $\mathcal{A}$-module code of length n and dimension $n-k$. A generator matrix $\boldsymbol{G}^\perp \in \mathcal{A}^{(n-k)\times n}$ of $\mathcal{C}^\perp$ is composed of the transpose of the last $n-k$ columns of the matrix $\boldsymbol{M}$ defined in* (3.2)*.*

Proof. It follows from Lemma 3.2 that all the columns of the $n \times n$ matrix $\boldsymbol{M}$ are orthogonal to all the codewords in $\mathcal{C}$. In other words, all the columns of $\boldsymbol{M}$ are in $\mathcal{C}^\perp$. We denote by $\widetilde{\mathcal{C}}$ the code generated by the columns of the $n \times n$ matrix $\boldsymbol{M}$ described in (3.2) (see (3.4) for an

[2]The order of an automorphism σ is the minimum positive integer n such that $\sigma^n = \mathrm{id}$.

example). By construction we have that $\widetilde{\mathcal{C}} \subseteq \mathcal{C}^\perp$. In Lemma 3.2 we have shown that a vector $\boldsymbol{c} \in \mathcal{C}$ if and only if $\boldsymbol{c}\boldsymbol{M} = \boldsymbol{0}$, which is equivalent to $\boldsymbol{c}$ being orthogonal to all generators of $\widetilde{\mathcal{C}}$. Therefore, $\mathcal{C} = \widetilde{\mathcal{C}}^\perp$. Since $\widetilde{\mathcal{C}}$ is a linear code over $\mathcal{A}$, by Lemma 3.3 we have that $|\widetilde{\mathcal{C}}^\perp| \cdot |\widetilde{\mathcal{C}}| = |\mathcal{A}|^n$. From $\mathcal{C} = \widetilde{\mathcal{C}}^\perp$ and $|\mathcal{C}| = |\mathcal{A}|^k$ we then get $|\widetilde{\mathcal{C}}| = |\mathcal{A}|^{n-k}$. Since $\mathcal{C}$ is also a linear code over $\mathcal{A}$, by Lemma 3.3 we also have $|\mathcal{C}^\perp| \cdot |\mathcal{C}| = |\mathcal{A}|^n$, which implies that $|\mathcal{C}^\perp| = |\mathcal{A}|^{n-k}$. Since $\widetilde{\mathcal{C}} \subseteq \mathcal{C}^\perp$ and both codes have the same cardinality, we conclude that $\widetilde{\mathcal{C}} = \mathcal{C}^\perp$, i.e., $\mathcal{C}^\perp$ is generated by the columns of the matrix $\boldsymbol{M}$.

It can be shown from (3.2) that for $i = 0, \ldots, n-k-1$, the i-th row $\boldsymbol{M}$ correspond to $X^i\hbar$ (e.g., the first two rows (the green part) in (3.4)). This shows that the right-upper $(n-k) \times (n-k)$ submatrix of $\boldsymbol{M}$ is lower triangular with invertible diagonal elements $\hbar_k, \theta(\hbar_k), \ldots, \theta^{n-k-1}(\hbar_k)$ and the right-most $n-k$ columns of $\boldsymbol{M}$ are therefore linearly independent. Hence, the $\mathcal{A}$-submodule generated by the right-most $n-k$ columns of $\boldsymbol{M}$ contains $|\mathcal{A}|^{n-k}$ elements. This shows that $\mathcal{C}^\perp$ is a free $\mathcal{A}$-module generated by the right-most $n-k$ columns of $\boldsymbol{M}$ (e.g., the last two columns (the orange part) in (3.4)). □

Corollary 3.1. *The first k columns of $\boldsymbol{M}$ are $\mathcal{A}$-linear combinations of the last $n-k$ columns of $\boldsymbol{M}$.*

Proof. Using the fact that $|\mathcal{C}^\perp| = |\mathcal{A}|^{n-k}$ from Lemma 3.3 and Theorem 3.2, the statement is proven. □

Example 3.6. *According to Theorem 3.2, the Euclidean dual of the code in Example 3.5 is generated by the right-most two columns of $\boldsymbol{M}$ in* (3.4)*, i.e.,*

$$\boldsymbol{G}^\perp = \begin{pmatrix} \hbar_1 & \hbar_1^\delta + \hbar_0^\theta & \hbar_1^{\delta^2} + \hbar_0^{\theta\delta} + \hbar_0^{\delta\theta} - \hbar_1^{\theta^2} f_1 \\ 0 & \hbar_1^\theta & \hbar_1^{\theta\delta} + \hbar_1^{\delta\theta} + \hbar_0^{\theta^2} - \hbar_1^{\theta^2} f_2 \end{pmatrix} .$$

Theorem 3.3 (Generator matrix of the σ-Hermian dual)**.** *Let $\mathcal{C} = \mathcal{C}[n,k]$ be the linear block (θ,δ)-code as defined in Definition 3.2, and $\boldsymbol{G}^\perp \in \mathcal{A}^{(n-k)\times n}$ be a generator matrix of the Euclidean dual code $\mathcal{C}^\perp$ of the code $\mathcal{C}$ as derived in Theorem 3.2. Denote by $\sigma(\boldsymbol{G}^\perp)$ the matrix after applying σ to every entry of $\boldsymbol{G}^\perp$. Then, $\sigma(\boldsymbol{G}^\perp)$ is a generator matrix of the σ-Hermitian dual code $\mathcal{C}^{\perp_\sigma}$ of $\mathcal{C}$.*

Proof. For each row $\boldsymbol{g}_i = (g_{i,0}, \ldots, g_{i,n-1}), i \in \{0, \ldots, k-1\}$ of a generating matrix $\boldsymbol{G}$ of $\mathcal{C}$ and each row $\boldsymbol{g}_j^\perp = (g_{i,0}^\perp, \ldots, g_{i,n-1}^\perp), j \in \{0, \ldots, n-k-1\}$ of a generating matrix of $\boldsymbol{G}^\perp$ of $\mathcal{C}^\perp$, we have $\langle g_i, g_j^\perp \rangle = \sum_{l=0}^{n-1} g_{i,l}\, g_{j,l}^\perp = 0$. Since σ is of order at most 2, $\langle g_i, \sigma(g_j^\perp) \rangle_\sigma = \sum_{l=0}^{n-1} g_{i,l}\, \sigma(\sigma(g_{j,l}^\perp)) = \sum_{i=0}^{n-1} g_{i,l}\, g_{j,l}^\perp = 0$. Since $\boldsymbol{G}^\perp$ is an upper triangular matrix with $n-k$ rows as shown in the proof of Theorem 3.2 and σ is an automorphism, the $\sigma(\boldsymbol{G}^\perp)$ is also an upper triangular matrix with $n-k$ rows and therefore generates a free $\mathcal{A}$-module code of dimension $|\mathcal{A}|^{n-k}$. Finally, Lemma 3.3 implies that they generate $\mathcal{C}^{\perp_\sigma}$. □

Corollary 3.2. *Let $\sigma(\boldsymbol{M})$ be the matrix after applying the automorphism σ to every entry of $\boldsymbol{M}$ derived in Lemma 3.2. The first k columns of $\sigma(\boldsymbol{M})$ are $\mathcal{A}$-linear combinations of the last $n-k$ columns of $\sigma(\boldsymbol{M})$.*

Proof. Denote by $\boldsymbol{m}_i$ the i-th column of $\boldsymbol{M}$. It follows from Corollary 3.1 that for $i \in \{0, 1, \ldots, k-1\}$, we can write $\boldsymbol{m}_i = \sum_{j=k}^{n-k-1} a_j \boldsymbol{m}_j$ for some $a_j \in \mathcal{A}$. By the rules of an automorphism (Definition 2.6), we have $\sigma(\boldsymbol{m}_i) = \sigma(\sum_{j=k}^{n-k-1} a_j \boldsymbol{m}_j) = \sum_{j=k}^{n-k-1} \sigma(a\boldsymbol{m}_j) = \sum_{j=k}^{n-k-1} \sigma(a_j)\sigma(\boldsymbol{m}_j)$, which is an $\mathcal{A}$-linear combination of the last $n-k$ columns of $\sigma(\boldsymbol{M})$. □

3.3 Computing All σ-Dual-Containing (θ, δ)-Codes

It follows from the definition of a σ-dual-containing code $\mathcal{C}[n, k]$ in Definition 3.4 that, $k \geqslant n - k$. In the rest of this chapter, we implicitly assume this inequality holds when addressing dual-containing codes. By Definition 3.2, we can see that a (θ, δ)-code $\mathcal{C}(g, f)$ is determined by the polynomials $g, f \in \mathcal{R}$. If g is also a left divisor of f (i.e., $f = g\hbar$), then the code $\mathcal{C}(g, f)$ is determined by $g, \hbar$.

Our method to compute σ-dual-containing (θ, δ)-codes considers the coefficients of $g, \hbar$ as unknowns and solves a system of equations derived from the following constraints:

(i) $f = g\hbar$ is monic and $g \mid_r f$;

(ii) (For Euclidean dual, $\sigma = \mathrm{id}$) $\boldsymbol{M}^\top \boldsymbol{M} = \boldsymbol{0}$, where $\boldsymbol{M}$ is described in Lemma 3.2 and serves as a parity-check matrix of $\mathcal{C}(g, f)$;

(iii) (For σ-Hermitian dual) $\sigma(\boldsymbol{M})^\top \boldsymbol{M} = \boldsymbol{0}$, where $\sigma(\boldsymbol{M})$ is the matrix after applying σ to every entry of $\boldsymbol{M}$.

Lemma 3.4. *Let $g, \hbar \in \mathcal{R}$. If the coefficients of g and $\hbar$ fulfill the system of polynomials equations that are derived from the constraints (i) and (ii) (resp. (iii)), then the (θ, δ)-code $\mathcal{C}(g, f)$ is dual-containing (resp. σ-dual-containing), where $f = g\hbar$.*

Proof. The constraint (i) suffices that $\mathcal{C}(g, f)$ is a (θ, δ)-code, $\hbar$ serves as a parity-check polynomial (Lemma 3.1), and we can construct the matrix $\boldsymbol{M}$ that serves as a parity-check matrix (Lemma 3.2).

The constraint (ii) is equivalent to setting $\boldsymbol{G}^\perp \cdot \boldsymbol{M} = \boldsymbol{0}$ according to Corollary 3.1, where $\boldsymbol{G}^\perp$ is the transpose of the last $n-k$ columns of $\boldsymbol{M}$ (Theorem 3.2). This ensures that $\mathcal{C}^\perp \subseteq \mathcal{C}$.

Similarly, the constraint (iii) is equivalent to setting $\boldsymbol{G}^{\perp_\sigma} \boldsymbol{M} = \boldsymbol{0}$ according to Corollary 3.2, which ensures that $\mathcal{C}^{\perp_\sigma} \subseteq \mathcal{C}$. □

It can be seen in Example 3.5 that the entries in $\boldsymbol{M}$ are symbolic expressions in images under compositions of θ and δ of the coefficients of $\hbar$ and g (e.g., $\hbar_1^\theta$, $\hbar_1^\delta$). However, since some θ or δ maps are not always polynomial maps in $\mathcal{A}$ (see Example 3.1 and Example 3.2), it is difficult to solve for the coefficients of g and $\hbar$ over $\mathcal{A}$ by computers.

If $\mathcal{A} = \mathcal{B}[\beta_1, \ldots, \beta_s]$ is a free $\mathcal{B}$-algebra and the restriction of δ and θ to $\mathcal{B}$ are polynomial maps, then we can transform the symbolic expressions of images under θ and δ (e.g., $\hbar_1^\theta$) into polynomial expressions over $\mathcal{B}$. By representing the unknown coefficients of $\hbar$ and g as a linear combination of the algebra basis $(\beta_1, \ldots, \beta_s)$ (e.g., $\hbar_1 = \hbar_{1,1}\beta_1 + \cdots + \hbar_{1,s}\beta_s$), we obtain multivariate polynomial expressions over $\mathcal{B}$ for the entries of $\boldsymbol{M}$.

The following lemma shows that we can transform a system of symbolic equations over $\mathcal{A}$ into a system of polynomial equations over $\mathcal{B}$.

Lemma 3.5. *Let $\mathcal{E}_s$ be a finite system of symbolic equations over $\mathcal{A}$ that are in the images of θ and δ of a finite number of variables $y_1, \ldots, y_m$. If $\mathcal{A} = \mathcal{B}[\beta_1, \ldots, \beta_s]$ is a free $\mathcal{B}$-algebra and the restriction of θ and δ to $\mathcal{B}$ are polynomial maps, then all solutions in $\mathcal{A}^m$ to $\mathcal{E}_s$ correspond to the solutions in $\mathcal{B}^{ms}$ to a system $\mathcal{E}_p$ of polynomial equations over $\mathcal{B}$ in the variables $y_{1,1}, \ldots, y_{1,s}, \ldots, y_{m,1}, \ldots, y_{m,s}$, where $y_i = y_{i,1}\beta_1 + \cdots + y_{i,s}\beta_s, i \in [m]$.*

Proof. The image of the basis $(\beta_1, \ldots, \beta_s)$ of $\mathcal{A}$ over $\mathcal{B}$ under δ and θ are expressions of the form $\beta_i^\theta = \gamma_{i,1}\beta_1 + \cdots + \gamma_{i,s}\beta_s$ and $\beta_i^\delta = \xi_{i,1}\beta_1 + \cdots + \xi_{i,s}\beta_s$ for some $\gamma_{i,j}, \xi_{i,j} \in \mathcal{B}$. The symbolic

expressions in the images under θ and δ of the variables $y_i, i \in [m]$ can therefore be written as

$$\begin{aligned}
y_i^\theta &= (y_{i,1}\beta_1 + \cdots + y_{i,s}\beta_s)^\theta \\
&= y_{i,1}^\theta \beta_1^\theta + \cdots + y_{i,s}^\theta \beta_s^\theta \\
&= y_{i,1}^\theta (\gamma_{1,1}\beta_1 + \cdots + \gamma_{1,s}\beta_s) + \cdots + y_{i,s}^\theta (\gamma_{s,1}\beta_1 + \cdots + \gamma_{s,s}\beta_s) \ , \\
y_i^\delta &= (y_{i,1}\beta_1 + \cdots + y_{i,s}\beta_s)^\delta \\
&= (y_{i,1}\beta_1)^\delta + \cdots + (y_{i,s}\beta_s)^\delta \\
&= y_{i,1}^\delta \beta_1 + y_{i,1}^\theta \beta_1^\delta + \cdots + y_{i,s}^\delta \beta_s + y_{i,s}^\theta \beta_s^\delta \\
&= y_{i,1}^\delta \beta_1 + y_{i,1}^\theta (\xi_{1,1}\beta_1 + \cdots + \xi_{1,s}\beta_s) + \cdots + y_{i,s}^\delta \beta_s + y_{i,s}^\theta (\xi_{s,1}\beta_1 + \cdots + \xi_{s,s}\beta_s) \ .
\end{aligned}$$

Using

(i) the algebra relations $\beta_i\beta_j = \mu_{i,j,1}\beta_1 + \ldots + \mu_{i,j,s}\beta_s$ (where $\mu_{i,j,s} \in \mathcal{B}$ are given),

(ii) the additive and multiplicative rules of θ and δ (Definition 2.6),

(iii) the fact that the restriction of δ and θ to $\mathcal{B}$ are polynomial maps on $\mathcal{B}$, so that $y_{i,j}^\theta$ and $y_{i,j}^\delta$ are polynomials in $y_{i,j}$ over $\mathcal{B}$,

we can recursively transform the system $\mathcal{E}_s$ of symbolic equations in variables $y_1, \ldots, y_m$ into a system $\mathcal{E}_p$ of polynomial equations in the variables $y_{1,1}, \ldots, y_{1,s}, \ldots, y_{m,1}, \ldots, y_{m,s}$. For any solution $(\hat{y}_{1,1}, \ldots, \hat{y}_{m,s}) \in \mathcal{B}^{ms}$ to $\mathcal{E}_p$, we can construct the corresponding solution $(\hat{y}_1, \ldots, \hat{y}_m) \in \mathcal{A}^m$ to $\mathcal{E}_s$ by $\hat{y}_i = \hat{y}_{i,1}\beta_1 + \cdots + \hat{y}_{i,s}\beta_s, i \in [m]$. □

Now we can proceed by solving a system of polynomial equations over $\mathcal{B}$ to get the coefficients of g, f such that the code $\mathcal{C}(g, f)$ over $\mathcal{A}$ is a σ-dual-containing code. We use Gröbner bases (see Section 2.2.2) for ideals of multivariate polynomials to solve the system of polynomial equations. Algorithm 3.1 summarizes our implementation in Magma [BCP97] to search for σ-dual-containing (θ, δ)-codes. If a Gröbner basis of the ideal generated by the polynomials in the system of equations over $\mathcal{B}$ can be computed, we can find all σ-dual-containing codes $\mathcal{C}(g, f)$ over $\mathcal{A}$ for the given parameters $[n, k]$.

Using the results found by Algorithm 3.1, we can further look into the following properties of the (θ, δ)-codes $\mathcal{C}(g, f)$:

(i) Do nonzero derivations or non-trivial endomorphisms (not automorphisms) provide new σ-dual-containing codes that could not be constructed from skew polynomials with automorphisms or zero derivations?
(The answer is yes. In Table 3.2b and Table 3.6b, we give examples of Hamming weight distributions of dual-containing (θ, δ)-codes that could not be found without considering either nonzero derivations or non-trivial endomorphisms.)

(ii) Is existing a central f such that $g \mid_r f$ a necessary condition for the (θ, δ)-code $\mathcal{C}(g, f)$ being dual-containing?
(The answer is no. In Section 3.4.1 we give an example of a generating polynomial g of a $[6, 4]$ dual-containing (θ, δ)-code where all 8 polynomials $f = hg = g\hbar$ of degree 6 are non-central.)

Algorithm 3.1: Computing all σ-dual-containing (θ, δ)-codes for given n, k.

Input: A ring $\mathcal{B}$ where Gröbner bases can be computed, a ring $\mathcal{A} = \mathcal{B}[\beta_1, \ldots, \beta_s]$ which is a free $\mathcal{B}$-algebra, an endomorphism θ and a θ-derivation δ of $\mathcal{A}$ which are polynomial maps on $\mathcal{B}$, an automorphism σ of $\mathcal{A}$ of order at most 2, and code parameters n, k.

Output: A set of solutions $\mathcal{P} = \{\hat{g}, \hat{\hbar}, \hat{f} \mid \mathcal{C}(\hat{g}, \hat{f}) \text{ is } \sigma\text{-dual-containing}\}$

$P_1 \leftarrow \mathcal{B}[g_{0,1}, \ldots, g_{0,s}, \ldots, g_{n-k-1,1}, \ldots, g_{n-k-1,s}, \hbar_{0,1}, \ldots, \hbar_{0,s}, \ldots, \hbar_{k-1,1}, \ldots, \hbar_{k-1,s}]$; `/* multivariate polynomial ring over` $\mathcal{B}$ `*/`

$\mathcal{P} \leftarrow \{\}$; `/* Initialize a set to collect` σ`-dual-containing codes */`

foreach $\hbar_k = \sum_{j=1}^{s} \hbar_{k,j}\beta_j \in \{\textit{Invertible element of } \mathcal{A}\}$ **do**

- $\mathsf{LSEs} \leftarrow \{\text{Constraints s.t. } g_{i,j}, \hbar_{i,j} \in \mathcal{B}\}$; `/*` $g_{i,j}^p = g_{i,j}, \hbar_{i,j}^p = \hbar_{i,j}$ `if` $\mathcal{B} = \mathbb{F}_p$ `*/`
- $g \leftarrow \sum_{i=0}^{n-k-1}(\sum_{j=1}^{s} g_{i,j}\beta_j)X^i + X^{n-k}$; `/*` $g \in P_1[X; \theta, \delta]$ `*/`
- $\hbar \leftarrow \sum_{i=0}^{k-1}(\sum_{j=1}^{s} \hbar_{i,j}\beta_j)X^i + \hbar_k X^k$; `/*` $\hbar \in P_1[X; \theta, \delta]$ `*/`
- $f \leftarrow g \cdot \hbar$; `/*` $\mathrm{lc}(f)$ `may not be monic but does not contain variable */`
- $h, r \leftarrow$ quotient, remainder of right dividing f by g ; `/*` $h, r \in P_1[X; \theta, \delta]$ `*/`
- $\mathsf{LSEs} \xleftarrow{\text{Append}} \{\text{All coefficients of } r \text{ are } 0\}$; `/* implies` $g \mid_r f$ `*/`
- $\boldsymbol{M} \leftarrow$ the matrix constructed from $\hbar$ according to Lemma 3.2;
- $\mathsf{LSEs} \xleftarrow{\text{Append}} \{\text{All entries in } \sigma(\boldsymbol{M})^\top \cdot \boldsymbol{M} \text{ are } 0\}$; `/* implies` $\mathcal{C}^{\perp_\sigma} \subseteq \mathcal{C}$ `*/`
- $\mathcal{S} \leftarrow \{\text{Solutions of } g_{0,1}, \ldots, g_{n-k-1,s}, \hbar_{0,1}, \ldots, \hbar_{k-1,s} \text{ from a Gröbner basis of } \mathsf{LSEs}\}$;
- $\mathcal{P} \xleftarrow{\text{Append}} \{\hat{g}, \hat{\hbar}, \hat{f} \in \mathcal{A}[X; \theta, \delta] \mid \forall \text{ solution in } \mathcal{S}\}$;
- `/*` $\hat{g}, \hat{\hbar} \in \mathcal{A}[X; \theta, \delta]$ `are reconstructed by evaluating coefficients of` $g, \hbar \in P_1[X; \theta, \delta]$ `at each solution in` $\mathcal{S}$; $\hat{f} = \hat{g} \cdot \hat{\hbar}$ `*/`

return $\mathcal{P}$

3.3.1 Is the Dual Code of a (θ, δ)-Code also a (θ, δ)-Code?

By Definition 3.2, a (θ, δ)-code $\mathcal{C}$ is a principle left ideal $\langle g \rangle_l / \langle f \rangle_l$ in the quotient ring $\mathcal{R}/\langle f \rangle_l$. The algorithm we introduce in this section allows us to test whether the dual code $\mathcal{C}^\perp$ is also a (θ, δ)-code, in other words, whether there exists a generator polynomial $g^\perp \in \mathcal{R}$ such that $\mathcal{C}^\perp = \langle g^\perp \rangle_l / \langle f^\perp \rangle_l$ for some $f^\perp \in \mathcal{R}$ and $g^\perp \mid_r f^\perp$.

Note that the rows of a generator matrix $G^\perp$ of $\mathcal{C}^\perp$ derived in Theorem 3.3 correspond to skew polynomials $p_1, \ldots, p_k$ in $\mathcal{R}$, and they form a basis of the code $\mathcal{C}^\perp$ when we see it as an $\mathcal{A}$-module. If the dual code $\mathcal{C}^\perp$ has a monic generator polynomial $g^\perp$, then $g^\perp$ must be a right divisor of all the polynomials $p_1, \ldots, p_k$.

The method is similar to Algorithm 3.1. We see the coefficients of $g^\perp$ as unknowns and translate the constraints that $g^\perp$ right divides all $p_1, \ldots, p_k$ into polynomial equations. We then compute a Gröbner basis of the ideal generated by these polynomials. If the Gröbner basis is $\{1\}$, meaning that there is no solution for these constraints, we can conclude that $\mathcal{C}^\perp$ is not a (θ, δ)-code. Otherwise, the Gröbner basis gives the solution of the monic generator polynomial $g^\perp$ of the (θ, δ)-code $\mathcal{C}^\perp$. Algorithm 3.2 summarizes our implementation in Magma [BCP97]. The algorithm can be extended to σ-Hermitian dual of a (θ, δ)-code by changing the input $\boldsymbol{G}^\perp$ to a generator matrix $\boldsymbol{G}^{\perp_\sigma}$ of the σ-Hermitian dual code.

The results of Algorithm 3.2 show that dual codes of (θ, δ)-codes are in general not (θ, δ)-codes, see Table 3.3 and Table 3.7 for examples over $\mathbb{F}_2[v]$ and $\mathbb{F}_2[u]$, respectively.

Algorithm 3.2: Testing whether the dual code is a (θ, δ)-code.

Input: A ring $\mathcal{B}$ where Gröbner bases can be computed, a ring $\mathcal{A} = \mathcal{B}[\beta_1, \ldots, \beta_s]$ which is a free $\mathcal{B}$-algebra, an endomorphism θ and a θ-derivation δ of $\mathcal{A}$ which are polynomial maps on $\mathcal{B}$, and a generator matrix $\boldsymbol{G}^\perp \in \mathcal{A}^{(n-k)\times n}$ of the dual code $\mathcal{C}^\perp$.

Output: A monic generator polynomial $g^\perp$ of degree k of the dual code $\mathcal{C}^\perp$ or `False`.

$P_1 \leftarrow \mathcal{B}[g_{0,1}^\perp, \ldots, g_{0,s}^\perp, \ldots, g_{k-1,1}^\perp, \ldots, g_{k-1,s}^\perp]$; `/* multivariate ring over` $\mathcal{B}$ `*/`

$g^\perp \leftarrow \sum_{i=0}^{k-1}(\sum_{j=1}^{s} g_{i,j}^\perp \beta_j)X^i + X^k$; `/*` $g^\perp \in P_1[X;\theta,\delta]$ `*/`

$\mathsf{LSEs} \leftarrow \{$Constraints on $g_{i,j}^\perp$ such that they are in $\mathcal{B}\}$; `/*` $(g_{i,j}^\perp)^p = g_{i,j}^\perp$ `if` $\mathcal{B} = \mathbb{F}_p$ `*/`

foreach $p_l \in \mathcal{A}[X;\theta,\delta]$ *corresponding to the l-th row of* $\boldsymbol{G}^\perp$ **do**

  $h, r \leftarrow$ quotient, remainder of right dividing p_l by $g^\perp$; `/*` $h, r \in P_1[X;\theta,\delta]$ `*/`

  $\mathsf{LSEs} \xleftarrow{\text{Append}} \{$All coefficients of r are 0 $\}$; `/* implies` $g^\perp \mid_r p_l$ `*/`

$\mathsf{GB} \leftarrow$ a Gröebner basis of LSEs;

if $\mathsf{GB} = \{1\}$ **then**

  return `False`

else

  $\mathcal{S} \leftarrow \{$Solutions of $g_{0,1}^\perp, \ldots, g_{0,s}^\perp, \ldots, g_{k-1,1}^\perp, \ldots, g_{k-1,s}^\perp$ from GB of LSEs$\}$;

$\mathcal{P} \leftarrow \{g^\perp \in \mathcal{A}[X;\theta,\delta] \mid \forall$ solution in $\mathcal{S}\}$;

`/* the monic` $\hat{g}^\perp \in \mathcal{A}[X;\theta,\delta]$ `is reconstructed by evaluating coefficients of` $g^\perp \in P_1[X;\theta,\delta]$ `at each solution in` $\mathcal{S}$ `*/`

3.4 Computation Results on Dual-Containing (θ, δ)-Codes

In this section we present the (σ-)dual-containing codes over the rings $\mathbb{F}_2[v], \mathbb{F}_2[u]$ and $\mathbb{F}_2[\alpha] = \mathbb{F}_4$ found by Algorithm 3.1.

3.4.1 Results for $\mathcal{A} = \mathbb{F}_2[v]$ with $v^2 = v$

We keep the notation used in Table 3.1a and compute the (σ-)dual-containing (θ, δ)-code over the ring $\mathcal{A} = \mathbb{F}_2[v]$ with $v^2 = v$ using the method given in Section 3.3. Codes of small length over $\mathbb{F}_2[v]$ are classified in [Huf05]. We follow [DS01] and define the Lee weight of $0, 1, v, v+1$ respectively as $0, 2, 1, 1$ and the Bachoc weight respectively as $0, 1, 2, 2$.

Table 3.2a gives an overview of the best (in terms of minimum Hamming, Lee or Bachoc distance) dual-containing codes $\mathcal{C}(g, f) \subset \mathcal{R}/\langle f \rangle_l$ (Algorithm 3.1 found all such codes). Table 3.2b gives detailed Hamming weight distributions that could only been found by some specific (θ, δ) combinations.

Testing Results by Algorithm 3.2

We apply Algorithm 3.2 to verify whether the dual of a dual-containing (θ, δ)-code over $\mathbb{F}_2[v]$ is also a (θ, δ)-code. Table 3.3 presents the results. We list the following two examples to illustrate that the dual code of a dual-containing (θ, δ)-code is not always a (θ, δ)-code:

- For $n = 4, k = 3$, we found three $g \in \mathbb{F}_2[v][X;\theta_2,\delta_2]$ that generate dual-containing (θ, δ)-codes: $g_1 = X + v + 1$, $g_2 = X + 1$, $g_3 = X + v$ where only the dual of $g_2 = X + 1$ is a (θ, δ)-code, with $g_2^\perp = X^3 + X^2 + X + 1$.

Table 3.2: Results on dual-containing (θ, δ)-codes over $\mathbb{F}_2[v]$. The blue cells mark the code parameters or the weight distributions that could only been found with nonzero derivations. The gray cells mark the code parameters or the weight distributions that could only been found with nonzero derivations and non-trivial endomorphisms.

(a) The largest Hamming, Lee and Bachoc distance of dual-containing (θ, δ)-codes over $\mathbb{F}_2[v]$. The empty set $\varnothing$ indicates that no dual-containing (θ, δ)-code exists for the parameter $[n, k]$. A question mark indicates that such dual-containing codes exist, but we could not compute the minimal distance due to the computation limitation.

n \ k	2	3	4	5	6	7	8	9	10	11	12
3	$1,1,2$										
4	$2,2,4$	$2,2,2$									
5		$\varnothing$	$\varnothing$								
6		$2,2,2$	$2,2,2$	$2,2,2$							
7			$3,3,5$	$\varnothing$	$\varnothing$						
8			$4,4,7$	$2,2,4$	$2,2,2$	$2,2,2$					
9				$\varnothing$	$\varnothing$	$\varnothing$	$1,1,2$				
10				$2,2,2$	$2,2,2$	$\varnothing$	$\varnothing$	$2,2,2$			
11					$\varnothing$	$\varnothing$	$\varnothing$	$\varnothing$	$\varnothing$		
12					$4,4,6$	$3,3,4$	$2,2,?$	$2,?,?$	$?,?,?$	$?,?,?$	
13						$\varnothing$	$\varnothing$	$\varnothing$	$\varnothing$	$\varnothing$	$\varnothing$

(b) Hamming weight distributions of dual-containing (θ, δ)-codes over $\mathbb{F}_2[v]$.

$[n, k]$	Hamming Weight Distribution	Constructed with (θ, δ)
$[4, 2]$	$[1, 0, 6, 0, 9]$	all combinations (θ, δ)
	$[1, 0, 4, 4, 7]$	$(\theta_2, \delta_2), (\theta_3, \delta_3), (\theta_4, \delta_4)$
$[6, 3]$	$[1, 0, 9, 0, 27, 0, 27]$	all combinations (θ, δ)
[6,4]	$[1, 0, 9, 24, 99, 72, 51]$	all combinations (θ, δ)
	$[1, 0, 17, 24, 83, 72, 59]$	$(\theta_2, \delta_3), (\theta_2, \delta_3)$
	$[1, 2, 11, 28, 87, 66, 61]$	$(\theta_3, \delta_3), (\theta_4, \delta_4)$
	$[1, 0, 13, 24, 91, 72, 55]$	$(\theta_3, \delta_3), (\theta_4, \delta_4)$
[8,4]	$[1, 0, 12, 0, 54, 0, 108, 0, 81]$	all combinations (θ, δ)
	$[1, 0, 0, 0, 28, 56, 84, 56, 31]$	$(\theta_2, 0)$
	$[1, 0, 4, 0, 38, 32, 100, 32, 49]$	$(\theta_2, \delta_2), (\theta_3, \delta_3), (\theta_4, \delta_4)$

- For $n = 6, k = 4$, we found four $g \in \mathbb{F}_2[v][X; \theta_3, \delta_3]$ that generate dual-containing (θ, δ)-codes: $g_1 = X^2 + (v+1)X + v + 1$, $g_2 = X^2 + X + 1$, $g_3 = X^2 + X + v + 1$, $g_4 = X^2 + (v+1)X + 1$. Only the dual codes of g_2 and g_4 are (θ, δ)-codes, with $g_2^\perp = X^4 + X^3 + X + 1$ and $g_4^\perp = X^4 + (v+1)X^3 + X + v + 1$, respectively.

Table 3.3: Results over $\mathbb{F}_2[v]$ on whether the dual code of a dual-containing (θ, δ)-code is also a (θ, δ)-code.

(a) The entries indicates that None/Some/All of the $[n, k]$ dual-containing (θ, δ)-codes whose dual codes are also (θ, δ)-codes.

n \ k	2	3	4	5	6	7	8	9
3	None							
4	All	Some						
5		/	/					
6		All	Some	Some				
7			All	/	/			
8			All	Some	Some	Some		
9				/	/	/	None	
10				All	Some	/	/	All

(b) The number of dual-containing (θ, δ)-codes whose dual codes are also (θ, δ)-codes. Only the parameters marked with "Some" in Table 3.3a are listed.

$[n, k]$	# of dual-containing (θ, δ)-codes for each (θ, δ) # of dual-containing (θ, δ)-codes whose dual codes are also (θ, δ)-codes								
	$(\mathrm{Id}; 0)$	$(\theta_2, 0)$	(θ_2, δ_2)	(θ_2, δ_3)	(θ_2, δ_4)	$(\theta_3, 0)$	(θ_3, δ_3)	$(\theta_4, 0)$	(θ_4, δ_4)
$[4, 3]$	1	1	3	1	1	1	2	1	2
	1	1	1	1	1	1	1	1	1
$[6, 4]$	1	1	1	2	2	1	4	1	4
	1	1	1	1	1	1	2	1	2
$[6, 5]$	1	1	1	2	2	1	1	1	1
	1	1	1	1	1	1	1	1	1
$[8, 5]$	1	3	5	1	1	1	8	1	8
	1	3	1	1	1	1	1	1	1
$[8, 6]$	1	3	5	1	1	1	4	1	4
	1	3	3	1	1	1	2	1	2
$[8, 7]$	1	1	3	1	1	1	2	1	2
	1	1	1	1	1	1	1	1	1
$[10, 6]$	1	1	1	1	1	1	16	1	16
	1	1	1	1	1	1	2	1	2

Hermitian Dual-Containing (θ, δ)-Codes over $\mathbb{F}_2[v]$

It can be verified that the automorphism θ_2 of $\mathbb{F}_2[v]$ is of order 2. We hence use Algorithm 3.1 to find all θ_2-dual-containing (θ, δ)-codes over $\mathbb{F}_2[v]$. Table 3.4a gives an overview of the largest minimum Hamming, Lee or Bachoc distance of θ_2-dual-containing codes $\mathcal{C}(g, f) \subset \mathcal{R}/\langle f \rangle_l$. Table 3.4b gives some Hamming weight distributions that could only be found by some specific

(θ, δ) combinations.

Table 3.4: Results on θ_2-dual-containing (θ, δ)-codes over $\mathbb{F}_2[v]$. The colored cells and special symbols have the same indications as in Table 3.2.

(a) The largest Hamming, Lee, Bachoc distance of θ_2-dual-containing (θ, δ)-codes over $\mathbb{F}_2[v]$.

n \ k	2	3	4	5	6	7	8	9
4	$2,2,4$	$2,2,2$						
5		$2,2,2$	$1,1,2$					
6		$3,3,4$	$2,2,4$	$2,2,2$				
7			$3,3,5$	$1,1,2$	$1,1,2$			
8			$3,3,6$	$2,2,4$	$2,2,2$	$2,2,2$		
9				$1,1,2$	$\varnothing$	$\varnothing$	$\varnothing$	
10				$2,2,2$	$2,2,2$	$\varnothing$	$\varnothing$	$2,2,2$

(b) The Hamming weight distributions of θ_2-dual-containing (θ, δ)-codes over $\mathbb{F}_2[v]$.

$[n, k]$	Hamming Weight	Constructed with (θ, δ)
$[4, 2]$	$[1, 0, 6, 0, 9]$	all combinations (θ, δ)
	$[1, 0, 2, 8, 5]$	$(\theta_2, 0)$
$[4, 3]$	$[1, 0, 18, 24, 21]$	all combinations (θ, δ)
	$[1, 2, 16, 22, 23]$	$(\theta_2, \delta_2), (\theta_3, \delta_3), (\theta_4, \delta_4)$
	$[1, 2, 12, 30, 19]$	$(\theta_2, \delta_3), (\theta_2, \delta_4)$
$[5, 3]$	$[1, 0, 8, 14, 23, 18]$	$(\theta_2, \delta_3), (\theta_2, \delta_4)$
$[5, 4]$	$[1, 3, 22, 66, 105, 59]$	$(\theta_2, \delta_3), (\theta_2, \delta_4)$
$[6, 3]$	$[1, 0, 0, 8, 21, 24, 10]$	$(\theta_3, \delta_3), (\theta_4, \delta_4)$

An Example where f being Central is Not Necessary for Dual-Containing (θ, δ)-Codes $\mathcal{C}(g, f)$

In most studies on the dual-containing (θ, δ)-codes $\mathcal{C}(g, f)$, e.g., [BSU08; BU11; BD18], f is assumed to be a central polynomial, since it is easier to derive closed formulas of a generator polynomial g of the code. With the example below we intend to show that there are dual-containing codes $\mathcal{C}(g, f)$ where g is not a right factor of any central f.

Note that many $f = hg = g\hbar$ can exist for the same g, and all (g, f) pairs lead to the same code, whose generator matrix is determined only by g and the corresponding (θ, δ). To illustrate this, we present more details of the $[6, 4]$ code with the Hamming weight distribution $[1, 0, 13, 24, 91, 72, 55]$ in Table 3.2b. There are four possible generator polynomials g of this code and they are presented in Table 3.5. We consider the first $g = X^2 + X + v + 1 \in \mathbb{F}_2[v][X, \theta_3, \delta_3]$ in Table 3.5. There are only 8 non-central polynomials f such that $f = hg = g\hbar$ for some $h, \hbar \in \mathcal{R}$ (i.e., g is both a left and right divisor of f):

$$
\begin{aligned}
f_1 &= X^6 + vX^4 + vX^3 + vX + v + 1 = (X^4 + X^3 + vX^2 + X + v + 1) \cdot g \\
f_2 &= X^6 + X^5 + (v+1)X^4 + X^3 + vX + v + 1 = (X^4 + vX^2 + (v+1)X + 1) \cdot g \\
f_3 &= X^6 + (v+1)X^4 + vX^3 + vX^2 + X + v + 1 = (X^4 + X^3 + (v+1)X^2 + 1) \cdot g \\
f_4 &= X^6 + X^5 + vX^4 + X^3 + vX^2 + X + v + 1 = (X^4 + (v+1)X^2 + vX + v + 1) \cdot g \\
f_5 &= X^6 + vX^4 + vX^3 + X^2 + (v+1)X = (X^4 + X^3 + vX^2 + X + v) \cdot g
\end{aligned}
$$

Table 3.5: All possible generator polynomials/matrices of the $[6, 4]$ dual-containing (θ, δ)-codes over $\mathbb{F}_2[v]$ with Hamming weight distribution $[1, 0, 13, 24, 91, 72, 55]$.

Index	g	(θ, δ)	**G**
1	$g = X^2 + X + v + 1$	(θ_3, δ_3)	$\begin{pmatrix} v+1 & 1 & 1 & 0 & 0 & 0 \\ v & 1 & 1 & 1 & 0 & 0 \\ v & 0 & 1 & 1 & 1 & 0 \\ v & 0 & 0 & 1 & 1 & 1 \end{pmatrix}$
2	$g = X^2 + (v+1)X + 1$	(θ_3, δ_3)	$\begin{pmatrix} 1 & v+1 & 1 & 0 & 0 & 0 \\ 0 & v+1 & 1 & 1 & 0 & 0 \\ 0 & v & 1 & 1 & 1 & 0 \\ 0 & v & 0 & 1 & 1 & 1 \end{pmatrix}$
3	$g = X^2 + vX + 1$	(θ_4, δ_4)	$\begin{pmatrix} 1 & v & 1 & 0 & 0 & 0 \\ 0 & v & 1 & 1 & 0 & 0 \\ 0 & v+1 & 1 & 1 & 1 & 0 \\ 0 & v+1 & 0 & 1 & 1 & 1 \end{pmatrix}$
4	$g = X^2 + X + v$	(θ_4, δ_4)	$\begin{pmatrix} v & 1 & 1 & 0 & 0 & 0 \\ v+1 & 1 & 1 & 1 & 0 & 0 \\ v+1 & 0 & 1 & 1 & 1 & 0 \\ v+1 & 0 & 0 & 1 & 1 & 1 \end{pmatrix}$

$$
\begin{aligned}
f_6 &= X^6 + X^5 + (v+1)X^4 + X^3 + X^2 + (v+1)X = (X^4 + vX^2 + (v+1)X) \cdot g \\
f_7 &= X^6 + (v+1)X^4 + vX^3 + (v+1)X^2 = (X^4 + X^3 + (v+1)X^2) \cdot g \\
f_8 &= X^6 + X^5 + vX^4 + X^3 + (v+1)X^2 = (X^4 + (v+1)X^2 + vX + v) \cdot g
\end{aligned}
$$

For each $f_i, i = 1, \ldots, 8$, there is a unique h_i corresponding to $f_i = h_i g$ (see the decomposition above) and 16 distinct $\hbar_i$ such that $f_i = g\hbar_i$, where one of $\hbar_i$ is equal to h_i. In the following we present for f_1 the other 15 distinct $\hbar_1 \neq h_1$ such that $f_1 = g\hbar_1$:

$$
\begin{aligned}
f_1 = g \cdot \left(X^4 + X^3 + X + 1\right) &= g \cdot \left(X^4 + X^3 + vX^2 + (v+1)X + v + 1\right) \\
= g \cdot \left(X^4 + (v+1)X^3 + (v+1)X + v + 1\right) &= g \cdot \left(X^4 + X^3 + (v+1)X + v + 1\right) \\
= g \cdot \left(X^4 + (v+1)X^3 + vX^2 + X + v + 1\right) &= g \cdot \left(X^4 + (v+1)X^3 + X + v + 1\right) \\
= g \cdot \left(X^4 + X^3 + X + v + 1\right) &= g \cdot \left(X^4 + (v+1)X^3 + vX^2 + (v+1)X + 1\right) \\
= g \cdot \left(X^4 + X^3 + vX^2 + (v+1)X + 1\right) &= g \cdot \left(X^4 + (v+1)X^3 + (v+1)X + 1\right) \\
= g \cdot \left(X^4 + X^3 + (v+1)X + 1\right) &= g \cdot \left(X^4 + (v+1)X^3 + vX^2 + X + 1\right) \\
= g \cdot \left(X^4 + X^3 + vX^2 + X + 1\right) &= g \cdot \left(X^4 + (v+1)X^3 + X + 1\right) \\
= g \cdot \left(X^4 + (v+1)X^3 + vX^2 + (v+1)X + v + 1\right) &.
\end{aligned}
$$

3.4.2 Results for $\mathcal{A} = \mathbb{F}_2[u]$ with $u^2 = 0$

We keep the notations used in Table 3.1b and compute the dual-containing (θ, δ)-codes over the ring $\mathcal{A} = \mathbb{F}_2[u]$ with $u^2 = 0$ using the method presented in Section 3.3. We follow [DS01] and define the Lee weight of $0, 1, u, u + 1$ respectively as $0, 1, 2, 1$ and the Euclidean weight respectively as $0, 1, 4, 1$.

Table 3.6a gives an overview of the best (in terms of minimum Hamming, Lee and Euclidean

distance) dual-containing (θ, δ)-codes (Algorithm 3.1 found all such codes). Table 3.6b gives some Hamming weight distributions that could only been found by some specific (θ, δ) combinations.

Table 3.6: Results on dual-containing (θ, δ)-codes over $\mathbb{F}_2[u]$. The colored cells and special symbols have the same indications as in Table 3.2.

(a) The best Hamming, Lee, and Euclidean distances of dual-containing (θ, δ)-codes over $\mathbb{F}_2[u]$.

n \ k	2	3	4	5	6	7	8	9
4	$2,4,4$	$2,2,2$						
5		$\varnothing$	$1,2,2$					
6		$2,4,4$	$2,2,2$	$2,2,2$				
7			$3,3,3$	$\varnothing$	$1,2,2$			
8			$4,4,4$	$2,4,4$	$2,2,2$	$2,2,2$		
9				$\varnothing$	$\varnothing$	$\varnothing$	$1,2,2$	
10				$2,4,6$	$2,4,5$	$\varnothing$	$\varnothing$	$2,2,2$

(b) Hamming weight distributions of dual-containing (θ, δ)-codes over $\mathbb{F}_2[u]$.

$[n,k]$	Hamming Weight	Constructed with (θ, δ)
$[4,2]$	$[1,0,2,8,5]$	$(\mathrm{id},0), (\mathrm{id},\delta_2), (\mathrm{id},\delta_3), (\theta_2,\delta_2)$
	$[1,0,6,0,9]$	all combinations (θ,δ)
$[8,4]$	$[1,0,4,0,30,64,52,64,41]$	$(\mathrm{id},0), (\theta_2,\delta_2)$
	$[1,0,4,0,46,0,148,0,57]$	$(\mathrm{id},0)$
	$[1,0,4,16,14,32,84,80,25]$	$(\mathrm{id},0)$
	$[1,0,12,0,54,0,108,0,81]$	all combinations (θ,δ)
	$[1,0,0,0,26,64,72,64,29]$	(id,δ_2)
$[8,5]$	$[1,0,4,16,94,224,308,272,105]$	$(\mathrm{id},0), (\mathrm{id},\delta_2)$
	$[1,0,4,16,110,160,404,208,121]$	$(\mathrm{id},0)$
	$[1,0,12,0,102,192,396,192,129]$	all combinations (θ,δ)
	$[1,0,16,8,114,176,360,200,149]$	(id,δ_2)

Testing Results by Algorithm 3.2

We apply Algorithm 3.2 to verify whether the dual of the a dual-containing (θ, δ)-code over $\mathbb{F}_2[u]$ is also a (θ, δ)-code. Table 3.7 presents the results.

3.4.3 Results for $\mathcal{A} = \mathbb{F}_2[\alpha] = \mathbb{F}_4$

We keep the notations used in Table 3.1c. Note that $\theta_2 : a \mapsto a^2$ is an automorphism of order 2. We compute the θ_2-dual-containing (θ, δ)-codes over $\mathbb{F}_4$ by Algorithm 3.1. Following [DS01] we define the Lee weight of $0, 1, \alpha, \alpha+1$ as $0, 2, 1, 1$, respectively, and following [LS01] we define the Euclidean weight as $0, 1, 2, 1$, respectively.

Table 3.8a shows the existence and the best Hamming, Lee and Euclidean distance of the θ_2-Hermitian dual-containing (θ, δ)-codes over $\mathbb{F}_4$. Table 3.8b provides some examples of the Hamming weight distributions.

Table 3.7: Test results over $\mathbb{F}_2[u]$ on whether the dual of a dual-containing (θ, δ)-code is also a (θ, δ)-code.

n \ k	2	3	4	5	6	7	8	9
4	All	All						
5		/	(id, δ_2): None (id, δ_4): All					
6		All	All	All				
7			All	/	All			
8			All	All	All	All		
9				/	/	/	(id, δ_2): None (id, δ_4): Some	
10				All	All	/	/	All

Table 3.8: Results on θ_2-dual-containing (θ, δ)-codes over $\mathbb{F}_2[\alpha] = \mathbb{F}_4$. The colored cells and special symbols have the same indications as in Table 3.2.

(a) The best Hamming, Lee and Euclidean distance of θ_2-dual-containing codes over $\mathbb{F}_4$.

n \ k	2	3	4	5	6	7	8	9
4	$2,2,2$	$2,2,2$						
5		$3,3,3$	$1,1,1$					
6		$4,4,4$	$2,2,2$	$2,2,2$				
7			$3,3,3$	$\varnothing$	$1,1,1$			
8			$2,2,2$	$2,2,2$	$2,2,2$	$2,2,2$		
9				$\varnothing$	$\varnothing$	$\varnothing$	$1,1,1$	
10				$(4,4,4)$	$(3,3,3)$	$(2,2,2)$	$(2,2,2)$	$(2,2,2)$

(b) Weight distributions of θ_2-dual-containing (θ, δ)-codes over $\mathbb{F}_4$.

$[n, k]$	Hamming Weight Enumerator	Constructed with (θ, δ)
$[4, 3]$	$[1, 0, 18, 24, 21]$	all combinations (θ, δ) maps
	$[1, 6, 12, 18, 27]$	(θ_2, δ_2)
$[5, 4]$	$[1, 9, 30, 54, 81, 81]$	(θ_2, δ_2)
$[6, 5]$	$[1, 0, 45, 120, 315, 360, 183]$	all combinations (θ, δ)
	$[1, 12, 57, 144, 243, 324, 243]$	(θ_2, δ_2)

3.5 Summary and Outlooks

This chapter considers (θ, δ)-polycyclic codes that are constructed from principle ideals of skew polynomials with endomorphisms θ and θ-derivations δ. In particular, we focused on constructing dual-containing (θ, δ)-codes over rings. As a basis, we first derived a parity-check matrix of a (θ, δ)-code within the framework of skew polynomials. For a finite commutative Frobenius ring $\mathcal{A}$, we then derived generator matrices of the Euclidean dual and the σ-Hermian dual of a (θ, δ)-code. This implies that the (σ-Hermitian) dual codes are $\mathcal{A}$-modules. For $\mathcal{A} = \mathcal{B}[\beta_1, \ldots, \beta_s]$ being a free $\mathcal{B}$-algebra, we developed an algorithm using Gröbner bases to compute all the (σ-)dual-containing (θ, δ)-codes over $\mathcal{A}$. We also presented an algorithm to test whether the dual code is also a (θ, δ)-code, in other words, whether there is a generator polynomial in $\mathcal{A}[X; \theta, \delta]$ of the dual code.

With the computational results for several rings of order 4, we obtain the following observations:

- nonzero derivations and non-trivial endomorphisms (not automorphisms) do give new (Euclidean/Hermitian) dual-containing (θ, δ)-codes that could not been found by zero derivations or automorphisms. See Table 3.2, Table 3.4, Table 3.6, and Table 3.8.
- The monic generator polynomial g being a right factor of some central polynomial f is not a necessary condition for the (θ, δ)-code generated by g to be a dual-containing code. See the example in Section 3.4.1.
- The dual code of a dual-containing (θ, δ)-code is in general not a (θ, δ)-code. See Table 3.3 and Table 3.7.

It can be seen that there are dual-containing (θ, δ)-codes over rings with large minimum Hamming distances, e.g., the $[6, 3]_4$ code over $\mathbb{F}_2[v]$ with $\mathrm{d_H} = 3$ in Table 3.4, and the $[8, 4]_4$ code over $\mathbb{F}_2[u]$ with $\mathrm{d_H} = 4$ in Table 3.6. For future research, it would be helpful to compare these codes with other existing codes over rings or some bounds on the cardinality or the distance of codes over rings, such as Singleton-like bounds, sphere-packing bounds. Moreover, fast decoding algorithms for codes based on skew polynomials with automorphisms and zero derivations has been extensively studied lately, e.g., in [BJPR21; HB22]. Since both of the above codes with large Hamming distances can only be found by non-trivial endomorphisms or nonzero derivations, advanced decoding algorithms based on [Bou20] for such (θ, δ)-codes are relevant to be developed.

4

Support-Constrained Evaluation Codes based on Skew Polynomials

Gabidulin codes [Gab85], introduced by Ernest M. Gabidulin, were the first evaluation codes from a special class of skew polynomials, namely the linearized polynomials, where the Frobenius automorphism and zero derivation are used. Boucher and Ulmer [BU14a] extended the notion of evaluation codes to skew polynomials over finite fields with inner derivations. Liu, Manganiello and Kschischang [LMK15] defined *generalized skew-evaluation codes* that contain Gabidulin codes as a special class, which combine the maximum distance separable (MDS) and the maximum rank distance (MRD) properties via a pasting construction. Martínez-Peñas introduced in [Mar18] *linearized Reed-Solomon* (LRS) codes, a class of evaluation codes based on skew polynomials that achieve the maximum sum-rank distance (MSRD) property.

This chapter investigates the support-constrained MSRD codes motivated from multi-source network coding and the advantage of vector network coding compared to scalar network coding. We first give brief introductions to support-constrained codes and LRS codes in Section 4.1 and Section 4.2, respectively. In Section 4.3 we present a necessary and sufficient condition on a generator matrix $\boldsymbol{G}$ such that it generates an MSRD code. Moreover, if the required generator matrix $\boldsymbol{G}$ does not satisfy the condition, we give the largest possible sum-rank distance of the code generated by $\boldsymbol{G}$. Using these results, we give in Section 4.4 a scheme to design *distributed LRS codes* for distributed multi-source unicast networks. Finally, we turn our focus to a family of *multicast* networks, namely the *generalized combination networks*, in Section 4.5. We investigate the advantages of using vectors as coding coefficients at the relay nodes in the network compared to using scalars. The advantages are shown via the gap between the minimum required field size of vector coding solutions and that of scalar coding solutions.

The results in Sections 4.3 and 4.4 have been submitted to IEEE Transactions on Information Theory (TIT) and partly published in the proceeding of 2023 IEEE Information Theory Workshop (ITW) [LWWS23]. The results in Section 4.5 were published in IEEE TIT [LWP+21] and partly in the proceeding of 2020 IEEE Information Symposium on Information Theory (ISIT) [LWP+20].

4.1 Support-Constrained Codes

Support-constrained codes are codes that have some codewords having zeros at certain positions and these codewords form a basis of the code. In other words, a support-constrained

code has a generator matrix having zeros at certain entries. A formal definition is given as follows.

Definition 4.1 (Support-constrained codes)**.** *Given the support constraints $Z_1, \ldots, Z_k \subseteq \{1, \ldots, n\}$, a linear $[n,k]_q$ code is a* support-constrained code *w.r.t. $Z_1, \ldots, Z_k$ if it has a generator matrix $\boldsymbol{G} \in \mathbb{F}_q^{k \times n}$ fulfilling the support constraints, i.e.,*

$$G_{ij} = 0 \ , \quad \forall i \in \{1, \ldots, k\}, \ \forall j \in Z_i \ . \tag{4.1}$$

Designing support-constrained error-correcting codes was motivated by its application in weakly secure network coding for wireless cooperative data exchange [YS11; SWY+12; YSZ14; LG17], where each node in the network stores a subset of all messages and the nodes communicate via broadcast transmissions to disseminate the messages in the presence of an eavesdropper.

From both, the theoretical and the practical point of view, the objective is to design support-constrained codes achieving the largest possible minimum distance. In the Hamming metric, research focused on proving the following necessary and sufficient condition for the existence of MDS codes fulfilling the support constraints. The condition was first conjectured in [DSY14b] (known as the *GM–MDS conjecture*), further studied in [HS17; YH18a], and finally proven independently by Lovett [Lov18] and by Yildiz and Hassibi [YH18b].

Theorem 4.1 (GM–MDS condition [YH18b; Lov18])**.** *Let $Z_1, \ldots, Z_k \subseteq \{1, \ldots, n\}$. For any $q \geqslant n+k-1$, there exists an $[n,k]_q$ Reed–Solomon (RS) code with a generator matrix $\boldsymbol{G} \in \mathbb{F}_q^{k \times n}$ fulfilling the support constraints in* (4.1)*, if and only if, for any nonempty $\Omega \subseteq \{1, \ldots, k\}$,*

$$\left| \bigcap_{i \in \Omega} Z_i \right| + |\Omega| \leqslant k \ . \tag{4.2}$$

Moreover, if an MDS code has a generator matrix fulfilling the support constraint (4.1)*, then the sets Z_i's satisfy* (4.2)*.*

Yildiz and Hassibi adapted the approach for Gabidulin codes in [YH20] and derived the following GM–MRD condition.

Theorem 4.2 (GM–MRD condition [YH20, Theorem 1])**.** *Let $Z_1, \ldots, Z_k \subseteq \{1, \ldots, n\}$. For any prime power q and integer $m \geqslant \max\{n, k-1+\log_q k\}$, there exists an $[n,k]_{q^m}$ Gabidulin code with a generator matrix $\boldsymbol{G} \in \mathbb{F}_{q^m}^{k \times n}$ fulfilling* (4.1)*, if and only if, for any nonempty $\Omega \subseteq \{1, \ldots, k\}$, the inequality* (4.2) *holds.*

Recently, special cases of support-constrained MDS codes have also been studied. For instance, the work by Greaves and Syatriadi [GS19] considered the following two special cases of Z's:

(i) $\left|\bigcap_{i=1}^{s} Z_i\right| = k - s$ for all $s = 1, \ldots, k$. Note that when $s = k-1$, it is required that $\left|\bigcap_{i=1}^{k-1} Z_i\right| = 1$, which means that there is at least one column of the generator matrix $\boldsymbol{G}$ containing $k-1$ zeros. For this case an $[n,k]_q$ RS code generated by $\boldsymbol{G}$ exists if $q \geqslant n$.

(ii) $|Z_i| \leqslant i-1$ for all $i = 1, \ldots, k$. Note that when $i \leqslant k-1$, $|Z_i| \leqslant k-2$, which implies that less zeros are allowed in $\boldsymbol{G}$ than in (4.1). For this case, an $[n,k]_q$ RS code generated by $\boldsymbol{G}$ exists if $q \geqslant n+1$.

Driven by the requirement of balanced computation load during the encoding process in wireless sensor networks [DSDY13] and multiple access networks [DSY14a; HHYD14; HHD14], codes fulfilling *sparse and balanced* support constraints have been proposed, e.g., in [HLH16; HLD+18]. In [SC18; CZ22], the existence of an MDS code with a sparse and balanced generator matrix $\boldsymbol{G}$ has been studied. "Sparse" means that each row of $\boldsymbol{G}$ has the maximum number of zeros, i.e., $k-1$ zeros, and "balanced" means that the number of zeros in any two columns differs by at most one, i.e., the weight of each column is either $\lceil k(n-k+1)/n \rceil$ or $\lfloor k(n-k+1)/n \rfloor$. It was shown in [SC18] that for any $k \in [n]$, if $q \geqslant n + \lceil k(k-1)/n \rceil$, then there exists an $[n,k]_q$ generalized RS code with a sparse and balanced generator matrix. More recently, it is shown in [CZ22] that for any $k \geqslant 3$, $\frac{k}{n} \geqslant \frac{1}{2}$, and $q \geqslant n-1$, there exists an $[n,k]_q$ MDS code with a sparse and balanced generator matrix.

4.2 Linearized Reed-Solomon Codes

LRS codes [Mar18] are a class of evaluation codes based on skew polynomials [Ore33], achieving the Singleton bound in the sum-rank metric, and therefore known as MSRD codes. They are the first linear MSRD codes with sub-exponential field sizes (in contrast to Gabidulin codes, which are MRD codes but require exponential field sizes in the code length). LRS codes have been applied in network coding [MK19b], locally repairable codes [MK19c] and code-based cryptography [HBH22]. The decoding of LRS codes has been extensively studied recently, e.g., [Bou20; PR21; BJPR21; PRR22; HB22; HBP22; JHB22; BP22].

The definition of LRS codes adopted in this chapter follows from the *generalized skew evaluations codes* [LMK15, Section III] with particular choices of the evaluation points and column multipliers.

Definition 4.2 (Linearized Reed-Solomon (LRS) code)**.** *For a prime power q and integers m, n, let $\ell \in [q-1]$ and $(n_1, \ldots, n_\ell)$ be an ordered partition of n with $n_l \leqslant m, \forall l \in [\ell]$. Let $a_1, \ldots, a_\ell \in \mathbb{F}_{q^m}$ be from distinct σ-conjugacy classes of $\mathbb{F}_{q^m}$, called* block representatives*. Let*

$$\boldsymbol{b} = (\beta_{1,1}, \ldots, \beta_{1,n_1} \ , \ \ldots \ , \ \beta_{\ell,1}, \ldots, \beta_{\ell,n_\ell}) \in \mathbb{F}_{q^m}^n$$

be a vector of column multipliers*, where $\beta_{l,1}, \ldots, \beta_{l,n_l}$, called the columns multipliers of the l-th block, are linearly independent over $\mathbb{F}_q, \forall l \in [\ell]$.*

Let the set of code locators *be*

$$\mathcal{L} = \{a_1\beta_{1,1}^{q-1}, \ldots, a_1\beta_{1,n_1}^{q-1} \ , \ \ldots \ , \ a_\ell\beta_{\ell,1}^{q-1}, \ldots, a_\ell\beta_{\ell,n_\ell}^{q-1}\} \ . \tag{4.3}$$

An $[n,k]_{q^m}$ linearized Reed-Solomon *code is defined as*

$$\mathcal{C}_{\mathcal{L},\boldsymbol{b}}^{\sigma}[n,k] := \{\boldsymbol{b} \star (f(\alpha))_{\alpha \in \mathcal{L}} \mid \ f(X) \in \mathbb{F}_{q^m}[X;\sigma], \ \deg f(X) < k\},$$

where $\mathbb{F}_{q^m}[X;\sigma]$ is the skew polynomial ring with the Frobenius automorphism $\sigma : a \mapsto a^q$ of $\mathbb{F}_{q^m}$, the evaluation $f(\alpha) = \sum_{i=0}^{\deg f} f_i N_i(\alpha)$ is the remainder evaluation as in Theorem 2.6, and $\star$ is the entry-wise multiplication of two vectors.

The code locator set $\mathcal{L}$ of LRS codes has the following properties.

Proposition 4.1 ([LL88b, Theorem 4.5])**.** *Since $\beta_{l,1}, \ldots, \beta_{l,n_l}$ are linearly independent over*

$\mathbb{F}_q$, *the set of code locators in the* l*-th block, denoted by* $\mathcal{L}^{(l)} = \left\{a_l\beta_{l,1}^{q-1}, \ldots, a_l\beta_{l,n_l}^{q-1}\right\}$, *is P-independent (Definition 2.14).*

Proposition 4.2 ([MSK+22, Theorem 2.11])**.** *The union of P-independent sets which are subsets of different conjugacy classes is P-independent. Hence, the code locator set* $\mathcal{L}$ *given in* (4.3) *is P-independent.*

A generator matrix of the LRS code in Definition 4.2 is given by

$$\boldsymbol{G}^{(\mathsf{LRS})} = \begin{pmatrix} \boldsymbol{G}_1^{(\mathsf{LRS})} & \ldots & \boldsymbol{G}_\ell^{(\mathsf{LRS})} \end{pmatrix} \in \mathbb{F}_{q^m}^{k\times n} \tag{4.4}$$

where for each $l \in [\ell]$,

$$\begin{aligned}
\boldsymbol{G}_l^{(\mathsf{LRS})} &= \boldsymbol{V}_k^{\sigma}(\mathcal{L}^{(l)}) \cdot \operatorname{diag}(\boldsymbol{b}^{(l)}) \\
&= \begin{pmatrix} 1 & \ldots & 1 \\ N_1(a_l\beta_{l,1}^{q-1}) & \ldots & N_1(a_l\beta_{l,n_l}^{q-1}) \\ \vdots & \ddots & \vdots \\ N_{k-1}(a_l\beta_{l,1}^{q-1}) & \ldots & N_{k-1}(a_l\beta_{l,n_l}^{q-1}) \end{pmatrix} \cdot \begin{pmatrix} \beta_{l,1} & & \\ & \ddots & \\ & & \beta_{l,n_l} \end{pmatrix} \\
&= \begin{pmatrix} 1 & & \\ & \ddots & \\ & & N_{k-1}(a_l) \end{pmatrix} \cdot \begin{pmatrix} \beta_{l,1} & \beta_{l,2} & \ldots & \beta_{l,n_l} \\ \beta_{l,1}^q & \beta_{l,2}^q & \ldots & \beta_{l,n_l}^q \\ \vdots & \vdots & \ddots & \vdots \\ \beta_{l,1}^{q^{k-1}} & \beta_{l,2}^{q^{k-1}} & \ldots & \beta_{l,n_l}^{q^{k-1}} \end{pmatrix},
\end{aligned}$$

where $\mathcal{L}^{(l)} = \left\{a_l\beta_{l,1}^{q-1}, \ldots, a_l\beta_{l,n_l}^{q-1}\right\}$ and $\boldsymbol{b}^{(l)} = (\beta_{l,1}, \ldots, \beta_{l,n_l})$. The last equality holds because for $\sigma(a) = a^q$, we have that $N_i(\beta_{l,t}^{q-1}) \cdot \beta_{l,t} = \left(\beta_{l,t}^{q-1}\right)^{(q^i-1)/(q-1)} \cdot \beta_{l,t} = \beta_{l,t}^{q^i}$.

LRS codes $\mathcal{C} = \left\langle \boldsymbol{G}^{(\mathsf{LRS})} \right\rangle \subseteq \mathbb{F}_{q^m}^n$ are MSRD codes [MSK+22, Theorem 2.20] and the punctured codes $\mathcal{C}_l = \left\langle \boldsymbol{G}_l^{(\mathsf{LRS})} \right\rangle \subseteq \mathbb{F}_{q^m}^{n_l}$ at any block $l = 1, \ldots, \ell$ are MRD codes [LMK15, Section III.C].

Example 4.1 ([12, 3] LRS code over $\mathbb{F}_{4^4}$with 3 blocks)**.** *Let* $q = 4, m = 4, \ell = 3, n_1 = n_2 = n_3 = 4, k = 3$*. Denote by* γ *a primitive element of* $\mathbb{F}_{4^4}$*. We choose the block representatives to be* $\boldsymbol{a} = (1, \gamma, \gamma^2)$ *and the basis of each block to be* $\boldsymbol{b}_1 = (1, \gamma, \gamma^2, \gamma^3), \boldsymbol{b}_2 = (\gamma, \gamma^2, \gamma^3, \gamma^4), \boldsymbol{b}_3 = (\gamma^2, \gamma^3, \gamma^4, \gamma^5)$*. Then the code locators are*

$$\mathcal{L} = \{\underbrace{1, \gamma^3, \gamma^6, \gamma^9}_{\textit{first block}}, \underbrace{\gamma^4, \gamma^7, \gamma^{10}, \gamma^{13}}_{\textit{second block}}, \underbrace{\gamma^8, \gamma^{11}, \gamma^{14}, \gamma^{17}}_{\textit{third block}}\} .$$

The generator matrix of this LRS code is

$$\boldsymbol{G}^{(\mathsf{LRS})} = \left(\begin{array}{cccc:cccc:cccc} 1 & \gamma & \gamma^2 & \gamma^3 & \gamma & \gamma^2 & \gamma^3 & \gamma^4 & \gamma^2 & \gamma^3 & \gamma^4 & \gamma^5 \\ 1 & \gamma^4 & \gamma^8 & \gamma^{12} & \gamma^5 & \gamma^9 & \gamma^{13} & \gamma^{17} & \gamma^{10} & \gamma^{14} & \gamma^{18} & \gamma^{22} \\ 1 & \gamma^{16} & \gamma^{32} & \gamma^{48} & \gamma^{21} & \gamma^{37} & \gamma^{53} & \gamma^{69} & \gamma^{42} & \gamma^{59} & \gamma^{74} & \gamma^{90} \end{array}\right) .$$

In [Mar18, Definition 31], LRS codes are defined using linear operator evaluations with respect to the block representatives $\boldsymbol{a} = (a_1, \ldots, a_\ell)$ and block basis $\boldsymbol{b}_l = (\beta_{l,1}, \ldots, \beta_{l,n_l})$. It

was shown in [MSK+22, Theorem 2.18] that these two definitions are equivalent.

4.3 GM-MSRD Condition

Motivated by the practical interest in support-constrained codes and the theoretical research on MSRD codes (in particular, LRS codes), we investigate the existence of support-constrained MSRD codes in this section and prove the following result.

Theorem 4.3 (GM-MSRD condition)**.** *Let ℓ, n be positive integers and $(n_1, \ldots, n_\ell)$ be an ordered partition of n. Given $Z_1, \ldots, Z_k \subset [n]$, for any prime power $q \geqslant \ell + 1$ and integer $m \geqslant \max_{l\in[\ell]}\{k - 1 + \log_q k, n_l\}$, there exists an $[n, k]_{q^m}$ linearized Reed-Solomon code with ℓ blocks, and each block of length n_l, $l \in [\ell]$ such that it has a generator matrix $\boldsymbol{G} \in \mathbb{F}_{q^m}^{k\times n}$ fulfilling the support constraints $G_{ij} = 0,\ \forall i \in [k],\ \forall j \in Z_i$, if and only if, for any nonempty $\Omega \subseteq [k]$,*

$$\left|\bigcap_{i\in\Omega} Z_i\right| + |\Omega| \leqslant k \ . \qquad \text{(Recall (4.2))}$$

For the necessity, since the sum-rank weight of any vector in $\mathbb{F}_{q^m}^n$ is at most its Hamming weight by Lemma 2.4, an MSRD code is necessarily an MDS code. Therefore, (4.2) is also a necessary condition for $\boldsymbol{G}$ to generate an MSRD code.

Now we proceed to show the sufficiency of (4.2) for MSRD codes, in particular, via support-constrained LRS codes with sufficiently large alphabet size. Note that for any $\Omega = \{i\}$, we have $|Z_i| \leqslant k - 1$. One can add elements from $[n]$ to each Z_i until $|Z_i|$ reaches $k - 1$ while preserving (4.2) [YH20, Corollary 3]. This operation will only put more zero constraints on $\boldsymbol{G}$ but not remove any. This means that the code we design under the new Z_i's of size $k-1$ will also satisfy the original constraints. Therefore, without loss of generality, along with (4.2), we can assume that

$$|Z_i| = k - 1,\ \forall i \in [k] \ . \tag{4.5}$$

Let $\boldsymbol{G}^{(\mathsf{LRS})}$ be a generator matrix of an LRS code as in (4.4). Given the following matrix

$$\boldsymbol{G} = \boldsymbol{T} \cdot \boldsymbol{G}^{(\mathsf{LRS})} \ , \tag{4.6}$$

if $\boldsymbol{T} \in \mathbb{F}_{q^m}^{k\times k}$ has full rank, then $\boldsymbol{G}$ is another generator matrix of the LRS code. Recall that $a_1, \ldots, a_\ell \in \mathbb{F}_{q^m}$ are the block representatives, $\beta_{1,1}, \ldots, \beta_{1,n_1}, \ldots, \beta_{\ell,1}, \ldots, \beta_{\ell,n_\ell} \in \mathbb{F}_{q^m}$ are the column multipliers, and $\mathcal{L} = \{\alpha_1, \ldots, \alpha_n\}$ is the code locator set as defined in (4.3).

Let $n_0 = 0$. Define the following bijective map between the indices,

$$\begin{aligned} \varphi : \mathbb{N} \times \mathbb{N} &\to \mathbb{N} \\ l, t &\mapsto j = t + \sum_{r=0}^{l-1} n_r \ . \end{aligned} \tag{4.7}$$

Then $\alpha_j = a_l \beta_t^{q-1}$ for $j = \varphi(l, t)$. The inverse map $\varphi^{-1} : \mathbb{N} \to \mathbb{N} \times \mathbb{N}$ is $j \mapsto (l, t)$, where $l = \max\left\{ i \in [\ell] \mid \sum_{r=0}^{i} n_r \leqslant j \right\}$ and $t = j - \sum_{r=0}^{l-1} n_r$.

For all $i \in [k]$, define the skew polynomials

$$f_i(X) := \sum_{j=1}^{k} T_{i,j} X^{j-1} \ \in \mathbb{F}_{q^m}[X;\sigma] \ , \tag{4.8}$$

where $T_{i,j+1}$ is the entry at i-th ($i \in [k]$) row, j-th ($j \in [k]$) column in $\boldsymbol{T}$. The entries of $\boldsymbol{G}$ are $G_{ij} = f_i(a_l \beta_{l,t}^{q-1})\beta_{l,t}, i \in [k], j = \varphi(l,t) \in [n]$. Then, the zero constraints in (4.1) become the following root constraints on f_i's:

$$f_i(a_l \beta_{l,t}^{q-1}) = 0, \quad \forall i \in [k], \ \forall j = \varphi(l,t) \in Z_i \ . \tag{4.9}$$

For brevity, denote by

$$\mathcal{Z}_i := \{\alpha_j = a_l \beta_{l,t}^{q-1} \mid j = \varphi(l,t) \in Z_i\} \tag{4.10}$$

the corresponding set of code locators to the zero set Z_i. Since $\mathcal{L}$ is P-independent, any subset $\mathcal{Z}_i \subset \mathcal{L}$ is also P-independent [Lam86, Theorem 23]. Then the minimal polynomial $f_{\mathcal{Z}_i}(X)$ of $\mathcal{Z}_i$ is of degree $|\mathcal{Z}_i| = k-1$. By the properties of $f_i(X)$ in (4.8) and (4.9), it can be seen that $f_i(X) = f_{\mathcal{Z}_i}(X)$ by setting $T_{i,k} = 1$ (as minimal polynomials are monic polynomials). By the computation of the minimal polynomial in (2.8), the skew polynomials $f_i(X)$ fulfilling (4.9) can be written as

$$f_i(X) = f_{\mathcal{Z}_i}(X) = \mathrm{lclm}_{\alpha \in \mathcal{Z}_i}\{X - \alpha\} \ . \tag{4.11}$$

Since all $\mathcal{Z}_i \subset \mathcal{L}, i \in [k]$ are P-independent, it follows from Lemma 2.3 that $f_i(\alpha) \neq 0$, for all $\alpha \in \mathcal{L} \backslash \mathcal{Z}_i$. Hence, there is no other zero entry in $\boldsymbol{G}$ than the required positions in Z_i's. Moreover, with $k-1$ P-independent roots of $f_i(X)$ and setting $T_{i,k} = 1$, the coefficients $T_{i,j}$ of $f_i(X)$ in (4.8) are uniquely determined in terms of $a_1 \beta_{1,1}^{q-1}, \ldots, a_\ell \beta_{\ell,n_\ell}^{q-1}$.

In the following, we assume that $a_1, \ldots, a_\ell$ are fixed, nonzero, and from distinct σ-conjugacy classes. We see $\beta_{l,t}$'s as variables of the following commutative multivariate polynomial ring

$$R_n := \mathbb{F}_{q^m}[\beta_{1,1}, \ldots, \beta_{\ell,n_\ell}] \ , \tag{4.12}$$

and the coefficients $T_{i,j}$ of $f_i(X)$ are polynomials in R_n. Then, the problem of finding $\beta_{l,t}$'s such that $\boldsymbol{G}$ generates the same LRS code as $\boldsymbol{G}^{(\mathsf{LRS})}$ becomes finding $\beta_{l,t}$'s such that

$$P(\beta_{1,1}, \ldots, \beta_{\ell,n_\ell}) := P_{\boldsymbol{T}}(\beta_{1,1}, \ldots, \beta_{\ell,n_\ell}) \cdot \prod_{l=1}^{\ell} P_{\boldsymbol{M}_l}(\beta_{l,1}, \ldots, \beta_{l,n_l}) \neq 0 \ , \tag{4.13}$$

where $P_{\boldsymbol{T}}$ is the determinant of $\boldsymbol{T}$ in (4.6), whose entries are the coefficients of f_i's and therefore determined by the roots of f_i's, and

$$P_{\boldsymbol{M}_l} := \det \begin{pmatrix} \beta_{l,1} & \beta_{l,2} & \cdots & \beta_{l,n_l} \\ \beta_{l,1}^{q} & \beta_{l,2}^{q} & \cdots & \beta_{l,n_l}^{q} \\ \vdots & \vdots & \ddots & \vdots \\ \beta_{l,1}^{q^{n_l-1}} & \beta_{l,2}^{q^{n_l-1}} & \cdots & \beta_{l,n_l}^{q^{n_l-1}} \end{pmatrix} .$$

Since the coefficient of the monomial $\prod_{i=1}^{n_l} \beta_{l,i}^{q^{i-1}}$ in $P_{\boldsymbol{M}_l}$ is 1, $P_{\boldsymbol{M}_l}$ is a nonzero polynomial in R_n.

Claim 1. *If the condition in* (4.2) *is satisfied, then $P_{\boldsymbol{T}}$ is a nonzero polynomial in R_n.*

With Claim 1, we can conclude that $P(\beta_{1,1}, \ldots, \beta_{\ell,n_\ell})$ is a nonzero polynomial in R_n. We now proceed to prove Theorem 4.3 assuming that Claim 1 is true. A more general statement (Theorem 4.5) of the claim is given and proven in Section 4.3.1.

For a fixed $l \in [\ell], t \in [n_\ell]$, the $\beta_{l,t}$-degree of $P_{\boldsymbol{M}_l}$ is $\deg_{\beta_{l,t}} P_{\boldsymbol{M}_l} = q^{n_l-1}$ [LN97, Lemma 3.51]. Moreover, $\deg_{\beta_{l,t}} P_{\boldsymbol{T}} \leqslant (k-1)(q-1) \cdot q^{k-2}$, which is proven in Appendix A.1. Then, the $\beta_{l,t}$-degree of $P(\beta_{1,1}, \ldots, \beta_{\ell,n_\ell})$ in (4.13) is

$$\deg_{\beta_{l,t}} P \leqslant (k-1)(q-1) \cdot q^{k-2} + q^{n_l-1} \ .$$

Proof of Theorem 4.3. Claim 1 implies that $P(\beta_{1,1}, \ldots, \beta_{\ell,n_\ell})$ is a nonzero polynomial. By the Combinatorial Nullstellensatz [Alo99, Theorem 1.2](see also in Theorem 2.1), there exist $\hat{\beta}_{1,1}, \ldots, \hat{\beta}_{\ell,n_\ell}$ in $\mathbb{F}_{q^m}$ such that

$$P(\hat{\beta}_{1,1}, \ldots, \hat{\beta}_{\ell,n_\ell}) \neq 0 \ ,$$

if

$$\begin{aligned} q^m &> \max_{l\in[\ell], t\in[n_l]} \{\deg_{\beta_{l,t}} P\} \\ &= \max_{l\in[\ell]} \{(k-1)(q-1) \cdot q^{k-2} + q^{n_l-1}\} \ . \end{aligned} \tag{4.14}$$

If $m \geqslant \max_{l\in[\ell]}\{k - 1 + \log_q k \ , \ n_l\}$, we have

$$\begin{aligned} q^m &= (q-1)q^{m-1} + q^{m-1} \\ &\geqslant \max_{l\in[\ell]} \{k(q-1) \cdot q^{k-2} + q^{n_l-1}\} > (4.14) \ . \end{aligned}$$

To have $a_1, \ldots, a_\ell$ from different non-trivial σ-conjugacy class of $\mathbb{F}_{q^m}$, by Theorem 2.8, we require $q - 1 \geqslant \ell$. □

Remark 4.1. *Consider the extreme cases:*

(i) For $\ell = 1$, the sum-rank metric is the rank metric and LRS codes are Gabidulin codes. In this case, the field size in Theorem 4.3 coincides with [YH20, Theorem 1].

(ii) For $\ell = n$ and $n_l = 1, \forall l \in [\ell]$, the sum-rank metric is the Hamming metric. In addition, with $\sigma = \mathrm{id}$, LRS codes are GRS codes with distinct nonzero $a_1, \ldots, a_\ell$ as code locators and nonzero $\beta_{l,t}$'s as column multipliers (see [MSK+22, Theorem 2.17], [MK19b, Table II]). In this case, by adapting the setup in (4.8) *to $\sigma = \mathrm{id}$, and the proof in Appendix A.1 with the usual evaluation of commutative polynomials, one can obtain the same results as in [YH18b, Theorem 2].*

If the necessary and sufficient condition on $Z_1, \ldots, Z_k$ in (4.2) is not satisfied, then we cannot obtain an MSRD code fulfilling the support constraints. The following result derives the largest possible sum-rank distance that can be achieved with the given constraints. In fact, the largest sum-rank distance can be achieved by subcodes of LRS codes. This result

is an analogue to those for MDS codes [YH18b] and MSRD codes [YH20]. The following upper bound on the minimum Hamming distance of a support-constrained code $\mathcal{C}$ is given in [YH18b, Theorem 1],

$$\mathrm{d}_{\mathsf{H}}(\mathcal{C}) \leqslant n - \widetilde{k} + 1$$

where

$$\widetilde{k} := \max_{\varnothing \neq \Omega \subseteq [k]} \left| \bigcap_{i \in \Omega} Z_i \right| + |\Omega| \ . \tag{4.15}$$

Note that $\widetilde{k} > k$ if $Z_1, \ldots, Z_k$ do not satisfy the condition in (4.2). For any ordered partition $\boldsymbol{n}_\ell = (n_1, \ldots, n_\ell)$ of n, according to Lemma 2.4, we have

$$\mathrm{d}_{\mathsf{SR},\boldsymbol{n}_\ell}(\mathcal{C}) \leqslant \mathrm{d}_{\mathsf{H}}(\mathcal{C}) \leqslant n - \widetilde{k} + 1 \ . \tag{4.16}$$

Theorem 4.4. *Given $Z_1, \ldots, Z_k \subseteq [n]$, let $\widetilde{k}$ be as in (4.15). For any prime power $q \geqslant \ell + 1$ and integer $m \geqslant \max_{l \in [\ell]}\{\widetilde{k} - 1 + \log_q \widetilde{k}, n_l\}$, there exists a subcode of an $[n, \widetilde{k}]_{q^m}$ linearized Reed-Solomon code with ℓ blocks, and each block of length n_l, $l \in [\ell]$ such that it has a generator matrix $\boldsymbol{G} \in \mathbb{F}_{q^m}^{k \times n}$ fulfilling the support constraints $G_{ij} = 0, \ \forall i \in [k], \ \forall j \in Z_i$.*

Proof. Let $Z_{k+1} = \cdots = Z_{\widetilde{k}} = \varnothing$. For any nonempty $\Omega \subseteq [\widetilde{k}]$, we have

$$\left| \bigcap_{i \in \Omega} Z_i \right| + |\Omega| \leqslant \widetilde{k} \ .$$

Then, by Theorem 4.3, there exists an LRS code of dimension $\widetilde{k}$ with a generator matrix $\widetilde{\boldsymbol{G}} \in \mathbb{F}_{q^m}^{\widetilde{k} \times n}$ having zeros at the positions specified by $Z_1, \ldots, Z_{\widetilde{k}}$. Since it is an MSRD code, its sum-rank distance is $n - \widetilde{k} + 1$. The first k rows of $\widetilde{\boldsymbol{G}}$ will generate a subcode $\mathcal{C}$ whose sum-rank distance is as good as the LRS code, i.e., $\mathrm{d}_{\mathsf{SR},\boldsymbol{n}_\ell}(\mathcal{C}) \geqslant n - \widetilde{k} + 1$, where $\boldsymbol{n}_\ell = (n_1, \ldots, n_\ell)$. Hence, the subcode achieves the largest possible distance given in (4.16). □

4.3.1 A More General Result of Claim 1

Let R_n be the commutative multivariate polynomial ring defined in (4.12). Note that $R_0 = \mathbb{F}_{q^m}$. Let σ be the Frobenius automorphism of R_0, which we extend to any $a = \sum_{\boldsymbol{i} \in \mathbb{N}^n} a_{\boldsymbol{i}} \cdot \beta_{1,1}^{i_1} \cdots \beta_{\ell,n_\ell}^{i_n} \in R_n$ by

$$\begin{aligned} \sigma \ : \ R_n &\to R_n \\ \sum_{\boldsymbol{i} \in \mathbb{N}^n} a_{\boldsymbol{i}} \cdot \beta_{1,1}^{i_1} \cdots \beta_{\ell,n_\ell}^{i_n} &\mapsto \sum_{\boldsymbol{i} \in \mathbb{N}^n} a_{\boldsymbol{i}}^q \cdot (\beta_{1,1}^{i_1})^q \cdots (\beta_{\ell,n_\ell}^{i_n})^q \ . \end{aligned}$$

Let $R_n[X; \sigma]$ be the univariate skew polynomial ring with indeterminate X, whose coefficients are from R_n, i.e.,

$$R_n[X; \sigma] := \left\{ \sum_{i \in \mathbb{N}} c_i X^i \ \middle| \ c_i \in R_n \right\} \ .$$

The degree of $f = \sum_{i\in\mathbb{N}} c_i X^i \in R_n[X;\sigma]$ is $\deg f := \max\{i \in \mathbb{N} \mid c_i \neq 0\}$ and $\deg 0 := -\infty$ by convention.

Similar to skew polynomials over a finite field, addition is commutative and multiplication is defined using the commutation rule

$$X \cdot a = \sigma(a) \cdot X, \ \forall a \in R_n \ , \tag{4.17}$$

and naturally extended by distributivity and associativity. As in (2.2), the product of $f, g \in R_n[X;\sigma]$ with $\deg f = d_f$ and $\deg g = d_g$ is

$$f \cdot g = \sum_{i=0}^{d_f} \sum_{j=0}^{d_g} f_i \sigma^i(g_j) X^{i+j} \ , \tag{4.18}$$

and the degree of the product is $\deg(f \cdot g) = d_f + d_g$. Note that in general, $f \cdot g \neq g \cdot f$.

With a bit abuse of the notation, in the following, we also denote by

$$\mathcal{L} = \{a_1\beta_{1,1}^{q-1}, \ldots, a_1\beta_{1,n_1}^{q-1} \ , \ \ldots \ , \ a_\ell\beta_{\ell,1}^{q-1}, \ldots, a_\ell\beta_{\ell,n_\ell}^{q-1}\} \subseteq R_n$$

the P-independent set as a subset of R_n. Let $\mathcal{Z}_i \subseteq \mathcal{L}$ be the set as in (4.10) corresponding to Z_i and $f_{\mathcal{Z}_i} \in R_n[X;\sigma]$ be the minimal polynomial of $\mathcal{Z}_i$ as in (4.11).

We note the following properties of $R_n[X;\sigma]$, which will be useful for the proof of the more general result of Claim 1 in Theorem 4.5. The proofs of these properties can be found in Appendix A.2.

P1 $R_n[X;\sigma]$ is a ring without zero divisors.

P2 For any sets $\mathcal{Z}_1, \mathcal{Z}_2 \subseteq R_n$ s.t. $\mathcal{Z}_1 \cup \mathcal{Z}_2$ is P-independent, $\operatorname{gcrd}(f_{\mathcal{Z}_1}, f_{\mathcal{Z}_2}) = f_{\mathcal{Z}_1 \cap \mathcal{Z}_2}$. In particular, $\mathcal{Z}_1 \cap \mathcal{Z}_2 = \varnothing \iff \operatorname{gcrd}(f_{\mathcal{Z}_1}, f_{\mathcal{Z}_2}) = 1$.

P3 For $t \in \mathbb{N}$ and any $f \in R_n[X;\sigma]$, $X^t \mid_l f \iff X^t \mid_r f$. In this case, we write $X^t | f$.

P4 For $t \in \mathbb{N}$ and any $f_1, f_2 \in R_n[X;\sigma]$ such that $X \nmid f_2$, then $X^t \mid (f_1 \cdot f_2) \iff X^t \mid f_1$.

In the general result in Theorem 4.5, we are interested in skew polynomials in the following form: for any $Z \subseteq [n], \tau \geqslant 0$

$$\mathsf{f}(Z,\tau) := X^\tau \cdot \operatorname*{lclm}_{\alpha \in \left\{ a_l\beta_{l,t}^{q-1} \;\middle|\; \varphi(l,t)\in Z \right\}} \{X - \alpha\} \in R_n[X;\sigma] \ , \tag{4.19}$$

where $\varphi(l,t)$ is defined in (4.7).

Define the set of skew polynomials of the following form:

$$\mathcal{S}_{n,k} := \{\mathsf{f}(Z,\tau) \mid \ \tau \geqslant 0, Z \subseteq [n] \text{ s.t. } |Z| + \tau \leqslant k-1\} \subseteq R_n[X;\sigma] \ . \tag{4.20}$$

Note that $\deg f \leqslant k-1, \forall f \in \mathcal{S}_{n,k}$. We also note the following properties of polynomials in $\mathcal{S}_{n,k}$, whose proofs are given in Appendix A.2.

P5 For any $f_1 = \mathsf{f}(Z_1,\tau_1), f_2 = \mathsf{f}(Z_2,\tau_2) \in \mathcal{S}_{n,k}$, we have

$$\operatorname{gcrd}(f_1, f_2) = \mathsf{f}(Z_1 \cap Z_2, \min\{\tau_1, \tau_2\}) \in \mathcal{S}_{n,k} \ .$$

P6 Let $f = \mathsf{f}(Z,\tau) \in \mathcal{S}_{n,k}$ and let $f' = f|_{\beta_{\ell,n_\ell}=0} \in R_{n-1}[X;\sigma]$ (namely, we substitute $\beta_{\ell,n_\ell} = 0$ in each coefficient of f). Then $f' \in \mathcal{S}_{n-1,k}$ and

$$f' = \begin{cases} \mathsf{f}(Z,\tau) & n \notin Z \ , \\ \mathsf{f}(Z\backslash\{n\},\tau+1) & n \in Z \ . \end{cases}$$

The following theorem is a more general statement of Claim 1 and it is the analogue of [YH20, Theorem 3.A] for skew polynomials.

Theorem 4.5. *Let $k \geqslant s \geqslant 1$ and $n \geqslant 0$. For any $f_1, f_2, \ldots, f_s \in \mathcal{S}_{n,k}$, the following are equivalent:*

(i) For any $g_1, g_2, \ldots, g_s \in R_n[X;\sigma]$ such that $\deg(g_i \cdot f_i) \leqslant k-1$, we have

$$\sum_{i=1}^{s} g_i \cdot f_i = 0 \quad \Longrightarrow \quad g_1 = g_2 = \cdots = g_s = 0 \ .$$

(ii) For all nonempty $\Omega \subseteq [s]$, we have

$$k - \deg(\mathrm{gcrd}_{i\in\Omega} f_i) \geqslant \sum_{i\in\Omega}(k - \deg f_i) \ . \tag{4.21}$$

Before proving Theorem 4.5, we first show in Corollary 4.1 that Claim 1 is a special case of Theorem 4.5. For this purpose, we give an equivalence of Theorem 4.5 in terms of matrices with entries from R_n.

We first describe the multiplication between skew polynomials in matrix language. Let $u = \sum_{i\in\mathbb{N}} u_i X^i \in R_n[X;\sigma]$. For $b - a \geqslant \deg u$, define the following matrix

$$\boldsymbol{S}_{a\times b}(u) \coloneqq \begin{pmatrix} u_0 & \cdots & u_{b-a} & & \\ & \sigma(u_0) & \cdots & \sigma(u_{b-a}) & \\ & & \ddots & \ddots & \ddots \\ & & & \sigma^{a-1}(u_0) & \cdots & \sigma^{a-1}(u_{b-a}) \end{pmatrix} \in R_n^{a\times b} \ .$$

In particular, for $a = 1$, denote by $R_n[X;\sigma]_{<b}$ the set of skew polynomials of degree strictly less than b. The map

$$\begin{aligned} \boldsymbol{S}_{1\times b}(\cdot) \ : \ R_n[X;\sigma]_{<b} &\to R_n^b \\ u &\mapsto (u_0, \ldots, u_{b-1}) \end{aligned} \tag{4.22}$$

is bijective and $\boldsymbol{S}_{1\times b}(0) = \boldsymbol{0}, \forall b \in \mathbb{N}$. For any skew polynomial $v = \sum_i v_i X^i \in R_n[X;\sigma]$, we have

$$\boldsymbol{S}_{a\times b}(v \cdot u) = \boldsymbol{S}_{a\times c}(v) \cdot \boldsymbol{S}_{c\times b}(u) \ , \tag{4.23}$$

where $a, b, c \in \mathbb{N}$ are such that $c - a \geqslant \deg v, b - c \geqslant \deg u$. As a special case, when $v = X^\tau, \tau \in$

$\mathbb{N}$, we can write

$$\begin{aligned}\boldsymbol{S}_{a\times(b+\tau)}(X^\tau \cdot u) =& \boldsymbol{S}_{a\times(a+\tau)}(X^\tau) \cdot \boldsymbol{S}_{(a+\tau)\times(b+\tau)}(u) \\ =& \begin{pmatrix}\boldsymbol{0}_{a\times\tau} & \boldsymbol{I}_{a\times a}\end{pmatrix} \cdot \boldsymbol{S}_{(a+\tau)\times(b+\tau)}(u) \ . \end{aligned} \tag{4.24}$$

By the definition in (4.19), we have $\mathsf{f}(Z,\tau) = X^\tau \cdot u$ for some $u \in R_n[X;\sigma]$. It can be readily seen from (4.24) that the first τ columns of $\boldsymbol{S}_{a\times(b+\tau)}(\mathsf{f}(Z,\tau))$ are all zero.

For $s \in [k]$, $i \in [s]$, let $f_i = \mathsf{f}(Z_i,\tau_i) \in \mathcal{S}_{n,k}$. We write $\boldsymbol{S}(f_i)$ instead of $\boldsymbol{S}_{(k-\tau_i-|Z_i|)\times k}(f_i)$ for ease of notation. By (4.24), $\boldsymbol{S}(f_i)$ looks like

$$\boldsymbol{S}(f_i) = \left.\begin{pmatrix} 0 & \cdots & 0 & \times & \times & \cdots & \times & & \\ 0 & \cdots & 0 & & \times & \times & \cdots & \times & \\ \vdots & & \vdots & & & \ddots & \ddots & & \ddots \\ 0 & \cdots & 0 & & & & \times & \times & \cdots & \times \end{pmatrix}\right\} k-\tau_i-|Z_i| \ ,$$

$$\underbrace{}_{\tau_i} \quad \underbrace{}_{k-1-\tau_i-|Z_i|} \quad \underbrace{}_{|Z_i|+1}$$

where the $\times$'s represent possibly nonzero entries. Then, applying (4.23) to the expression $g_i \cdot f_i$ in Theorem 4.5 yields

$$\boldsymbol{S}_{1\times k}(g_i \cdot f_i) = \boldsymbol{u}_i \cdot \boldsymbol{S}(f_i) \ ,$$

where $\boldsymbol{u}_i = \boldsymbol{S}_{1\times(k-\tau_i-|Z_i|)}(g_i)$ is a row vector. Therefore, we can write

$$\boldsymbol{S}_{1\times k}(\sum_{i=1}^{s} g_i \cdot f_i) = (\boldsymbol{u}_1, \cdots, \boldsymbol{u}_s) \cdot \underbrace{\begin{pmatrix}\boldsymbol{S}(f_1)\\ \vdots \\ \boldsymbol{S}(f_s)\end{pmatrix}}_{=:\boldsymbol{M}(f_1,\ldots,f_s)} \ , \tag{4.25}$$

which is a linear combination of the rows of $\boldsymbol{M}(f_1,\ldots,fs)$.

The following theorem is equivalent to Theorem 4.5 in matrix language and is analogous to [YH20, Theorem 3.B].

Theorem 4.6. *Let $k \geqslant s \geqslant 1$ and $n \geqslant 0$. For $i \in [s]$, let $Z_i \in [n], \tau_i \geqslant 0$ such that $\tau_i + |Z_i| \leqslant k-1$ and $f_i = \mathsf{f}(Z_i,\tau_i) \in \mathcal{S}_{n,k}$. The matrix $\boldsymbol{M}(f_1,\ldots,f_s)$ defined in* (4.25) *has full row rank if and only if, for all nonempty $\Omega \subseteq [s]$,*

$$k - \left|\bigcap_{i\in\Omega} Z_i\right| - \min_{i\in\Omega}\tau_i \geqslant \sum_{i\in\Omega}(k-\tau_i-|Z_i|) \ . \tag{4.26}$$

Proof. For brevity, we write $\boldsymbol{M}$ instead of $\boldsymbol{M}(f_1,\ldots,f_s)$. The logic of the proof is as follows

$$\boldsymbol{M} \text{ has full row rank} \overset{(I)}{\iff} (i) \overset{\text{Theorem 4.5}}{\iff} (ii) \overset{(II)}{\iff} (4.26) \text{ holds}$$

where (i) and (ii) are shown to be equivalent in Theorem 4.5. We only need to show the equivalence *(I)* and *(II)*.

(I): Assuming $\boldsymbol{M}$ has full row rank, it is equivalent to writing

$$\forall \boldsymbol{u} \in R_n^{1\times\sum_{i=1}^{s}(k-\tau_i-|Z_i|)},\ \boldsymbol{u}\cdot\boldsymbol{M}=\boldsymbol{0} \Longrightarrow \boldsymbol{u}=\boldsymbol{0}\ . \tag{4.27}$$

Partition $\boldsymbol{u}$ into s blocks $(\boldsymbol{u}_1,\ldots,\boldsymbol{u}_s)$, where $\boldsymbol{u}_i \in R_n^{1\times(k-\tau_i-|Z_i|)}$. Note that $\boldsymbol{u}=\boldsymbol{0} \iff \forall i \in [s], \boldsymbol{u}_i=\boldsymbol{0}$. For each $i\in[s]$, the set $\{g_i \mid \boldsymbol{S}_{1\times(k-\tau_i-|Z_i|)}(g_i)=\boldsymbol{u}_i,\ \forall \boldsymbol{u}_i \in R_n^{1\times(k-\tau_i-|Z_i|)}\}$ is $R_n[X;\sigma]_{<(k-\tau_i-|Z_i|)}$, which is the set of skew polynomials of degree less than $(k-\tau_i-|Z_i|)$, since the map $\boldsymbol{S}_{1\times *}(\cdot)$ defined in (4.22) is bijective. Therefore, $\boldsymbol{u}_i=\boldsymbol{0} \iff g_i=0,\ \forall i\in[s]$. It can be further inferred that every $\boldsymbol{u}\in R_n^{1\times\sum_{i=1}^{s}(k-\tau_i-|Z_i|)}$ corresponds to a unique tuple $(g_1,\ldots,g_s) \in R_n[X;\sigma]_{<(k-\tau_1-|Z_1|)}\times\cdots\times R_n[X;\sigma]_{<(k-\tau_s-|Z_s|)}$. We denote the Cartesian product by $\mathcal{G}$. Since $\deg f_i=\tau_i+|Z_i|,\ \forall i\in[s]$, for any tuple $(g_1,\ldots,g_s)\in\mathcal{G}$, $\deg(g_i\cdot f_i)\leqslant k-1,\ \forall i\in[s]$.

By the equality in (4.25), $\boldsymbol{u}\cdot\boldsymbol{M}=\boldsymbol{S}_{1\times k}(\sum_{i=1}^{s} g_i\cdot f_i)$ and $\boldsymbol{S}_{1\times k}(\sum_{i=1}^{s} g_i\cdot f_i)=\boldsymbol{0} \iff \sum_{i=1}^{s} g_i\cdot f_i=0$. Hence (4.27) can be equivalently written as

$$\forall g_1,\ldots,g_s\in R_n[X;\sigma] \text{ such that } \deg(g_i\cdot f_i)\leqslant k-1\ ,$$
$$\sum_{i=1}^{s} g_i\cdot f_i=0 \implies g_i=0,\ \forall i\in[s]\ ,$$

which is exactly the statement (i).

(II): It follows from **P5** that for any nonempty set $\Omega\subseteq[s]$,

$$\deg(\operatorname{gcrd}_{i\in\Omega} f_i)=\left|\bigcap_{i\in\Omega} Z_i\right|+\min_{i\in\Omega}\tau_i\ .$$

Then the left hand side of (4.21), $k-\deg(\operatorname{gcrd}_{i\in\Omega} f_i)$, is equal to $k-|\bigcap_{i\in\Omega} Z_i|-\min_{i\in\Omega}\tau_i$, which is the left hand side of (4.26). By the definition of $f_i=\mathsf{f}(Z_i,\tau_i)$ in (4.19), the right hand side of (4.21), $\sum_{i\in\Omega}(k-\deg f_i)$, is equal to $\sum_{i\in\Omega}(k-(|Z_i|+\tau_i))$, which is the right hand side of (4.26). □

As a special case, when $s=k$, $\tau_i=0$ and $|Z_i|=k-1,\ \forall i\in[k]$, each block $\boldsymbol{S}(f_i)$ becomes a row vector with entries being the coefficients of $f_i=\mathsf{f}(Z_i,0)=\sum_{j=1}^{k} f_{i,j}X^{j-1}\in R_n[X;\sigma]$ and

$$\boldsymbol{M}(f_1,\ldots,f_k)=\begin{pmatrix} f_{11} & f_{12} & \cdots & f_{1k}\\ f_{21} & f_{22} & \cdots & f_{2k}\\ \vdots & \vdots & \ddots & \vdots\\ f_{k1} & f_{k2} & \cdots & f_{kk}\end{pmatrix}\in R_n^{k\times k}\ . \tag{4.28}$$

Note that $\boldsymbol{M}(f_1,\ldots,f_k)$ coincides with the matrix $\boldsymbol{T}$ in (4.6). Hence, we have Corollary 4.1 below, which is exactly Claim 1.

Corollary 4.1. *For $i\in[k]$, let $Z_i\subseteq[n]$ with $|Z_i|=k-1$. Then, $\det\boldsymbol{M}(f_1,\ldots,f_k)$ is a nonzero polynomial in R_n, if and only if, for all nonempty $\Omega\subseteq[k]$, $k-|\bigcap_{i\in\Omega} Z_i|\geqslant|\Omega|$.*

Proof of Theorem 4.5

Denote $f_\Omega:=\operatorname{gcrd}_{i\in\Omega} f_i$. By **P2**, f_Ω is equal to the minimal polynomial of the set $Z_\Omega:=\bigcap_{i\in\Omega} Z_i$.

We first show the direction (i) $\Longrightarrow$ (ii). Suppose (ii) does not hold and w.l.o.g., assume that for $\Omega = \{1, 2, \ldots, \nu\} \subseteq [k]$, $k - \deg f_\Omega < \sum_{i\in\Omega}(k - \deg f_i)$. For $i \in \Omega$, let $f_i = q_i \cdot f_\Omega$ for some $q_i \in R_n[X;\sigma]$. Then, for $g_1, \ldots, g_\nu \in R_n[X;\sigma]$ such that $\deg(g_i \cdot f_i) \leqslant k-1$, the equation $\sum_{i\in\Omega} g_i \cdot q_i = 0$ gives a homogeneous linear system of equations in the unknowns which are the coefficients of the g_i's. Since the g_i's are such that $\deg(g_i \cdot f_i) \leqslant k-1$, the number of unknowns is at least $\sum_{i\in\Omega}(k - \deg f_i)$. The number of equations is at most $k - \deg f_\Omega$, which is smaller than the number of unknowns by the assumption. Therefore, one can find $g_1, \ldots, g_\nu$, not all zero, solving the linear system of equations, which contradicts (i).

We then show the direction (ii) $\Longrightarrow$ (i) by induction. We do induction on the parameters (k, s, n) considered in the lexicographical order $\prec$ (page 6).

For the induction basis, when $(k \geqslant s = 1, n \geqslant 0)$, (i) always holds due to **P1**, i.e., $g_1 \cdot f_1 = 0$ implies $g_1 = 0$.

For $(k \geqslant s \geqslant 2, n = 0)$, both (i) and (ii) never hold therefore they are equivalent. Note that $n = 0 \implies f_i = X^{\tau_i}$ for all $i \in [k]$. For any $f_i = X^{\tau_i}$ and $f_j = X^{\tau_j}$ with $\tau_i \neq \tau_j$ (w.l.o.g. assuming $\tau_i > \tau_j$), there exist $g_i = 1$ and $g_j = -X^{\tau_i - \tau_j}$ such that $g_i f_i + g_j f_j = 0$ and hence (i) never holds. Suppose $\tau_1 \leqslant \tau_2$, then for $\Omega = \{1, 2\}$, (4.21) becomes $k - \tau_1 \geqslant (k - \tau_1) + (k - \tau_2)$, which contradicts that $\deg f_i = |Z_i| + \tau_i \leqslant k - 1$. Hence, (ii) never holds.

For $(k \geqslant s \geqslant 2, n \geqslant 1)$, we do the induction with the following hypotheses:

H1 Assume that (ii) $\Longrightarrow$ (i) is true for all parameters $(k', s', n') \prec (k, s, n)$.

H2 Take any $f_1, \ldots, f_s \in \mathcal{S}_{n,k}$ for which (ii) is true for (k, s, n).

The logic of the proof is summarized in Figure 4.1.

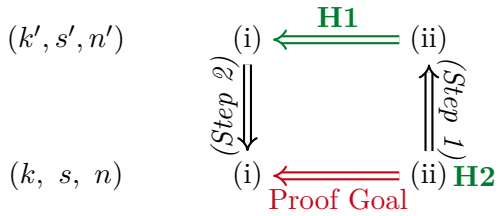


Figure 4.1: Proof logic for $(ii) \implies (i)$ with initial hypothesis **H1** and **H2**.

Starting from **H2**, we have that for all the subsets $\varnothing \neq \Omega \subseteq [s]$, the inequality (4.21) in (ii) holds. We prove that (i) is true for (k, s, n) via *(Step 1)* $\rightarrow$ **H1** $\rightarrow$ *(Step 2)* under different cases:

Case 1 For $s \geqslant 3$ and $n \geqslant 2$,

- ***Case 1a*** $\forall i \in [s]$, $\tau_i \geqslant 1$ (i.e., $|Z_i| \leqslant k-2$). *(In this case, we do induction by reducing k.)*
- ***Case 1b*** $\exists$ a unique $i \in [s]$ such that $\tau_i = 0$. *(In this case, we do induction by reducing k. We may need to reduce s as well.)*
- ***Case 1c*** $\exists\ \Omega \subset [s]$ with $2 \leqslant |\Omega| \leqslant s-1$ such that (4.21) holds with equality. *(In this case, we do induction by reducing s.)*
- ***Case 1d*** $\forall\ \Omega \subset [s]$ with $2 \leqslant |\Omega| \leqslant s-1$, (4.21) holds strictly and $\exists$ at least two $i \in [s]$ such that $\tau_i = 0$. *(In this case, we do induction by reducing n.)*

Case 2 For $s = 2$ and $n \geqslant 2$,

Case 2a $\forall i \in \{1,2\}$, $\tau_i \geqslant 1$ (i.e., $|Z_i| \leqslant k-2$). *(The same as* ***Case 1a****.)*

Case 2b $\exists$ a unique $i \in \{1,2\}$ such that $\tau_i = 0$. *(The same as* ***Case 1b****.)*

Case 2c $\forall i \in \{1,2\}$, $\tau_i = 0$. *(In this case, we do induction by reducing n.)*

Case 3 For $s \geqslant 2$ and $n = 1$,

Case 3a $\forall i \in [s]$, $\tau_i \geqslant 1$ (i.e., $|Z_i| \leqslant k-2$). *(The same as* ***Case 1a****.)*

Case 3b $\exists$ a unique $i \in \{1,2\}$ such that $\tau_i = 0$. *(The same as* ***Case 1b****.)*

Case 3c $\exists$ at least two $i \in [s]$, $\tau_i = 0$. *(We show that this case cannot happen if (ii) is true for $(k \geqslant s \geqslant 2, n = 1)$.)*

We illustrate the reduction of s and n in the induction under these cases in Fig. 4.2. We omitted the parameter k for clarity and simplicity, since only s, n are essential in classifying the different cases. The elaborated proofs for each case are presented in Appendix A.3.

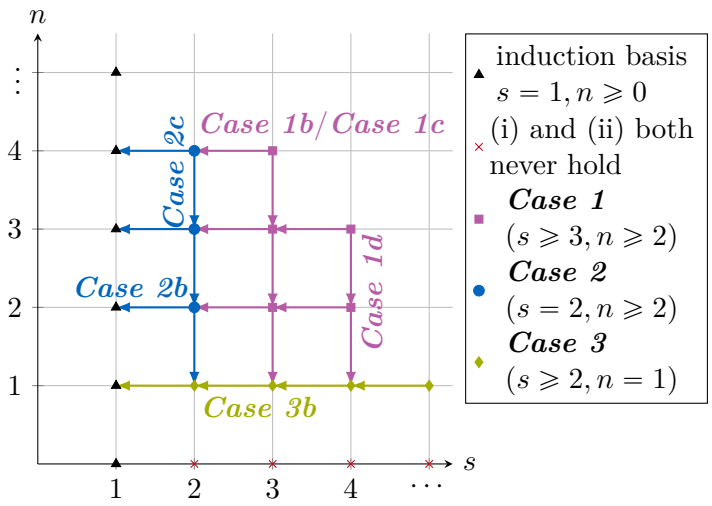


Figure 4.2: Illustration of the induction for (ii) $\Longrightarrow$ (i) under difference cases.

4.4 Applications in Multi-Source Network Coding

Distributed multi-source networks were studied in [HHYD14; HHD14], where supported-constrained error-correcting codes were used to achieve reliable communication against malicious (or failed) nodes in the network. In this section, we introduce a scheme to design *distributed LRS codes* for any such network instance. The scheme illustrates how the necessary and sufficient conditions derived in Section 4.3 can be used as constraints in a linear programming problem to design the parameters of desired distributed LRS codes.

Consider a distributed multi-source network as illustrated in Fig. 4.3. The receiver at the sink intends to obtain all the messages in a set $\mathcal{M}$ by downloading through an $\mathbb{F}_q$-linear network from multiple source nodes. Each source node has access to only a few messages in $\mathcal{M}$. This access is assumed to have unlimited link capacity (e.g., the source nodes store the subset of $\mathcal{M}$ locally). The topology of the $\mathbb{F}_q$-linear network is not known to the source nodes nor to the sink; therefore, it is a *non-coherent* communication scenario. This model can find

its applications in data sharing platforms, sensor networks, satellite communication networks and MIMO (multiple-input multiple-output) antenna communication systems, etc.

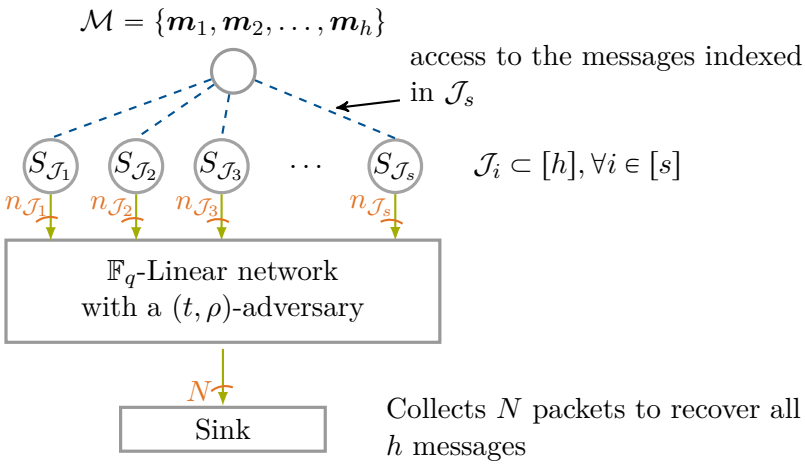


Figure 4.3: Illustration of the distributed multi-source network model.

The set $\mathcal{M}$ contains h messages. The message $\boldsymbol{m}_j, j \in [h]$ is composed of r_j symbols over $\mathbb{F}_{q^m}$, i.e., $\boldsymbol{m}_j \in \mathbb{F}_{q^m}^{r_j}$. The source node $S_{\mathcal{J}_i}, i \in [s]$ has access only to the messages indexed in $\mathcal{J}_i$, e.g., if $\mathcal{J}_2 = \{3, 6\}$, then $S_{\mathcal{J}_2}$ only has the access to the messages $\boldsymbol{m}_3$ and $\boldsymbol{m}_6$. Let $\mathcal{S} = \{\mathcal{J}_1, \ldots, \mathcal{J}_s\}$. For any $\mathcal{J} \in \mathcal{S}$, the source node $S_{\mathcal{J}}$ encodes the messages $\boldsymbol{m}_j, j \in \mathcal{J}$, into $n_{\mathcal{J}}$ symbols over $\mathbb{F}_{q^m}$, denoted by $\boldsymbol{c}_{\mathcal{J}} \in \mathbb{F}_{q^m}^{n_{\mathcal{J}}}$. It then extends them to their matrix representation over $\mathbb{F}_q$, denoted by $\boldsymbol{C}_{\mathcal{J}} \in \mathbb{F}_q^{m \times n_{\mathcal{J}}}$, and then generates $\boldsymbol{X}_{\mathcal{J}} = (\boldsymbol{0} \ \cdots \ \boldsymbol{I}_{n_{\mathcal{J}}} \ \boldsymbol{0} \ \cdots \ \boldsymbol{C}_{\mathcal{J}}^{\top}) \in \mathbb{F}_q^{n_{\mathcal{J}} \times (n+m)}$, where $n = \sum_{\mathcal{J} \in \mathcal{S}} n_{\mathcal{J}}$. We call each row of $\boldsymbol{X}_{\mathcal{J}}$ a *packet*. Denote $\boldsymbol{X} = \begin{pmatrix} \boldsymbol{X}_{\mathcal{J}_1} \\ \vdots \\ \boldsymbol{X}_{\mathcal{J}_s} \end{pmatrix} \in \mathbb{F}_q^{n \times (n+m)}$, where the rows are the packets transmitted by all the source nodes into the $\mathbb{F}_q$-linear network. The task is to design $n_{\mathcal{J}}$ for all $\mathcal{J} \in \mathcal{S}$ such that the sink can recover all the messages $\boldsymbol{m}_i$. The goal of the design is to minimize the total number of packets n. A concrete example is given in Section 4.4.2.

In the $\mathbb{F}_q$-linear network, whenever there is a transmission opportunity, a relay node in the network produces and sends an arbitrary $\mathbb{F}_q$-linear combination of all the incoming packets they have received. Suppose that there are at most t *malicious* nodes that inject erroneous packets and at most ρ *frozen* nodes that do not send any packet, which we refer as a (t, ρ)-*adversary*. The sink collects $N \geqslant n - \rho$ packets, which are represented by the rows of $\boldsymbol{Y} \in \mathbb{F}_q^{N \times (n+m)}$. The transmitted packets (rows of $\boldsymbol{X}$) and the received packets (rows of $\boldsymbol{Y}$) can be related via the following network equation:

$$\boldsymbol{Y} = \boldsymbol{A}\boldsymbol{X} + \boldsymbol{E} \ , \tag{4.29}$$

where $\boldsymbol{A} \in \mathbb{F}_q^{N \times n}$ is the *transfer* matrix of the network and the difference between the number of columns and its row-rank is at most ρ. In other words, $n - \mathrm{rank}(\boldsymbol{A}) \leqslant \rho$. $\boldsymbol{E} \in \mathbb{F}_q^{N \times M}$ is an error matrix of $\mathrm{rank}(\boldsymbol{E}) \leqslant t$. Note that the matrices $\boldsymbol{A}$ and $\boldsymbol{E}$ are not known to any of the source nodes or the sink since we consider a non-coherent communication scenario.

The *capacity region* of a multi-source network with h messages is a set $\{(r_1, \ldots, r_h)\} \subseteq \mathbb{N}^h$

such that the receiver at the sink can recover all the messages $\boldsymbol{m}_j \in \mathbb{F}_{q^m}^{r_j}, j \in [h]$. The capacity region of a multi-source network against a (t,ρ)-adversary has been given in [DHJ+11, Theorem 2] (for $\rho = 0$) and [RK18, Corollary 66]. To present the result, we require the following definitions of *min-cut*.

Definition 4.3 (Min-cut between a set of nodes and another node)**.** *For a directed graph $\mathcal{G}(\mathcal{V},\mathcal{E})$ composed of a set of nodes $\mathcal{V}$ and a set of edges $\mathcal{E}$, a* cut *between a set of nodes $\mathcal{V}' \subset \mathcal{V}$ and another node $t \in \mathcal{V}\backslash\mathcal{V}'$ is a subset of edges $\mathcal{E}_{\mathcal{V}',t} \subseteq \mathcal{E}$ such that, after removing the edges in $\mathcal{E}_{\mathcal{V}',t}$, there is no path from any of the nodes in $\mathcal{V}'$ to t. The* min-cut *between $\mathcal{V}'$ and t is the smallest cardinality of a cut between $\mathcal{V}'$ and t.*

Definition 4.4 (Min-cut between a subset of messages and the sink)**.** *Consider the distributed multi-source network with h messages as above. Given a subset of messages, $\mathcal{J}' \subseteq [h]$, consider the set of source nodes $\mathcal{V}'_{\mathcal{J}'}$ that contain messages in $\mathcal{J}'$, namely,*

$$\mathcal{V}'_{\mathcal{J}'} = \left\{ S_{\mathcal{J}} \in \mathcal{S} \mid \mathcal{J} \cap \mathcal{J}' \neq \varnothing \right\} .$$

We define the min-cut between $\mathcal{J}'$ and the sink *as the min-cut between $\mathcal{V}'_{\mathcal{J}'}$ and the sink, and denote it by $w_{\mathcal{J}'}$.*

Theorem 4.7 ([DHJ+11; RK18])**.** *Consider a multi-source network with h messages. For any $(r_1,\ldots,r_h) \in \mathbb{N}^h$ in the capacity region against a (t,ρ)-adversary, we have*

$$\forall \varnothing \neq \mathcal{J}' \subseteq [h],\ \sum_{i\in\mathcal{J}'} r_i \leqslant w_{\mathcal{J}'} - 2t - \rho \ , \tag{4.30}$$

where $w_{\mathcal{J}'}$ is the min-cut between the set $\mathcal{J}'$ of messages and the sink.

In addition to the general settings, we further assume the following setup of the non-coherent network:

- The communication capacity of the non-coherent linear network is large enough so that the min-cut $w_{\mathcal{J}'}$ for all $\mathcal{J}' \subseteq [h]$ is determined by the number of encoded symbols $n_{\mathcal{J}}$ sent by the source node $S_{\mathcal{J}}$ for all $\mathcal{J} \in \mathcal{S}$, i.e.,

$$w_{\mathcal{J}'} = n - \sum_{\substack{\mathcal{J}\in\mathcal{S} \\ \mathcal{J}\subseteq[h]\backslash\mathcal{J}'}} n_{\mathcal{J}} \ .$$

  Note that the term $\sum_{\substack{\mathcal{J}\in\mathcal{S} \\ \mathcal{J}\subseteq[h]\backslash\mathcal{J}'}} n_{\mathcal{J}}$ is the total number of encoded symbols that do not contain any information about the messages in $\mathcal{J}'$.

- Although the encoding is distributed (since each source node may access only a few messages), there is a centralized coordination unit designing the overall code, and the sink knows the distributed code.

4.4.1 Sum-Rank Weight of Error and Erasure with Constrained Rank Weight

In the following, we intend to use LRS codes for the distributed multi-source linear network model. Note that the errors and erasures in the (t,ρ)-adversarial model are measured in the rank metric. However, LRS codes are used to deal with errors and erasures in the sum-rank

metric. Hence, we first look into the sum-rank deficiency of the network transfer matrix $\boldsymbol{A} \in \mathbb{F}_q^{N\times n}$ and the sum-rank weight of the error matrix $\boldsymbol{E} \in \mathbb{F}_q^{N\times M}$.

Let $\ell \in \mathbb{N}$ and $\boldsymbol{n}_\ell = (n_1, \ldots, n_\ell)$ be an ordered partition of n. By Lemma 2.5, we have

$$\mathrm{wt}_{\mathsf{SR},\boldsymbol{n}_\ell}(\boldsymbol{A}) \geqslant \mathrm{rank}(\boldsymbol{A}) \geqslant n - \rho \ . \tag{4.31}$$

Hence the sum-rank weight of the erasure induced by the rank-deficient $\boldsymbol{A}$ is at most ρ.

For the error $\boldsymbol{E}$, consider an ordered partition $\boldsymbol{N}_\ell = (N_1, \ldots, N_\ell)$ of N such that

$$\boldsymbol{A}\boldsymbol{X} + \boldsymbol{E} = \begin{matrix} N_1\{ \\ N_2\{ \\ \\ N_\ell\{ \end{matrix} \underbrace{\begin{pmatrix} \boldsymbol{A}_{1,1} & \boldsymbol{A}_{1,2} & \cdots & \boldsymbol{A}_{1,\ell} \\ \boldsymbol{A}_{2,1} & \boldsymbol{A}_{2,2} & \cdots & \boldsymbol{A}_{2,\ell} \\ \vdots & \vdots & \vdots & \vdots \\ \boldsymbol{A}_{\ell,1} & \boldsymbol{A}_{\ell,2} & \cdots & \boldsymbol{A}_{\ell,\ell} \end{pmatrix}}_{n_1 \quad n_2 \quad\quad n_\ell} \underbrace{\begin{pmatrix} \boldsymbol{X}_1 \\ \boldsymbol{X}_2 \\ \vdots \\ \boldsymbol{X}_\ell \end{pmatrix}}_{M} \begin{matrix} \} n_1 \\ \} n_2 \\ \\ \} n_4 \end{matrix} + \underbrace{\begin{pmatrix} \boldsymbol{E}_1 \\ \boldsymbol{E}_2 \\ \vdots \\ \boldsymbol{E}_\ell \end{pmatrix}}_{M} \begin{matrix} \} N_1 \\ \} N_2 \\ \\ \} N_\ell \end{matrix} .$$

Given $\mathrm{rank}(\boldsymbol{E}) \leqslant t$, by Lemma 2.5, we have

$$\mathrm{wt}_{\mathsf{SR},\boldsymbol{N}_\ell}(\boldsymbol{E}) = \sum_{i=1}^{\ell} \mathrm{rank}(\boldsymbol{E}_i) \leqslant \sum_{i=1}^{\ell} \mathrm{rank}(\boldsymbol{E}) = \ell t \ . \tag{4.32}$$

This upper bound holds for any arbitrary ordered partition $\boldsymbol{N}_\ell$ of N. A lower bound on $\Pr[\mathrm{wt}_{\mathsf{SR},\boldsymbol{N}_\ell}(\boldsymbol{E}) = \ell t \mid \mathrm{rank}(\boldsymbol{E}) = t]$ (i.e., the probability that (4.32) is tight) for small t ($t \leqslant N_i, \forall i \in [\ell]$) is given in [SL23, Theorem 1]. In particular, if $q \geqslant \ell + 1$, then $\Pr[\mathrm{wt}_{\mathsf{SR},\boldsymbol{N}_\ell}(\boldsymbol{E}) = \ell t \mid \mathrm{rank}(\boldsymbol{E}) = t] > 1/4$ [SL23, Corollary 1].

It can been seen from (4.31) and (4.32) that the network model in (4.29) results in an erasure of sum-rank weight at most ρ and an error of sum-rank weight at most ℓt. It has been shown in [MK19b, Theorem 1, Eq.(4), Proposition 2] that a code with sum-rank distance d can guarantee reliable communication against errors of sum-rank weight at most ℓt and erasures with sum-rank weight at most ρ in the non-coherent communication if $d \geqslant 2\ell t + \rho + 1$. Therefore, an LRS code with sum-rank distance $d \geqslant 2\ell t + \rho + 1$ can correct any error and erasure in the (t, ρ)-adversarial model.

4.4.2 Example of Distributed LRS codes

We first give a toy example to show the usage of LRS codes in a distributed multi-source network, and then provide a general scheme to design *distributed LRS codes* for arbitrary distributed multi-source networks in Section 4.4.3.

Consider the following toy example of the network illustrated in Fig. 4.3: There are $h = 4$ messages in $\mathcal{M}$. The lengths of messages are $(r_1, r_2, r_3, r_4) = (1, 3, 2, 3)$. There are 4 source nodes and each can access to only 3 messages, i.e., $\mathcal{J}_1 = \{1, 2, 3\}, \mathcal{J}_2 = \{1, 2, 4\}, \mathcal{J}_3 = \{1, 3, 4\}, \mathcal{J}_4 = \{2, 3, 4\}$. Suppose there is a $(2, 2)$-adversary in the $\mathbb{F}_q$-linear network.

The number of encoded packets from each source node is $(n_{\mathcal{J}_1}, n_{\mathcal{J}_2}, n_{\mathcal{J}_3}, n_{\mathcal{J}_4}) = (6, 7, 2, 8)$ (see (i) in Section 4.4.3 for the computation of these values) and $n = \sum_{i=1}^{4} n_{\mathcal{J}_i} = 23$. Let $\boldsymbol{m} = (\boldsymbol{m}_1, \boldsymbol{m}_2, \boldsymbol{m}_3, \boldsymbol{m}_4)$ be a concatenated vector of all the messages. Some entries in a encoding matrix $\boldsymbol{G}$ are forced to be 0, as shown in Fig. 4.4, so that $\boldsymbol{m} \cdot \boldsymbol{G}$ represents the overall encoding at all source nodes. For example, the first 6 columns of $\boldsymbol{G}$, corresponding to

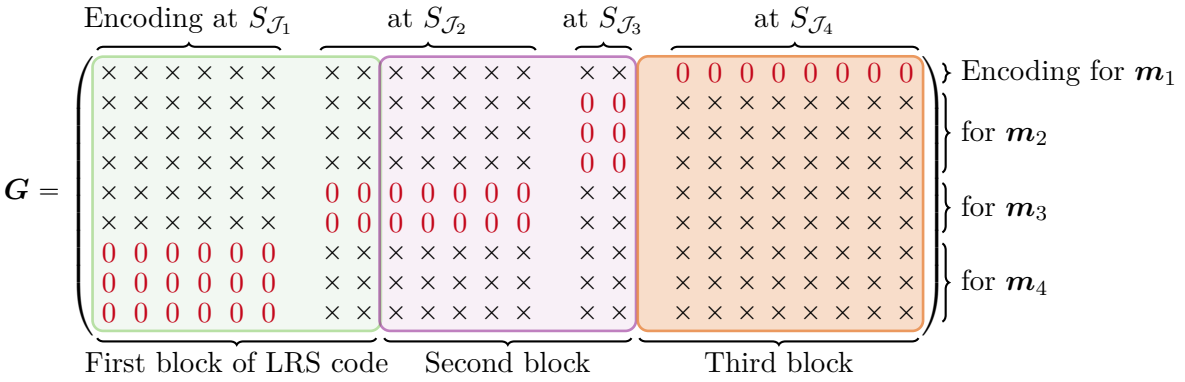


Figure 4.4: Illustration of the required encoding matrix for the instance of distributed multi-source networks with $h = 4$ messages. This support-constrained matrix is a generator matrix of a $[23, 9]$ LRS over $\mathbb{F}_{4^9}$ with $\ell = 3$ blocks.

the encoding at $S_{\mathcal{J}_1}$, have zero entries in the last 3 rows. This indicates that $S_{\mathcal{J}_1}$ does not encode $\boldsymbol{m}_4$ since it does not have the access to $\boldsymbol{m}_4$.

We can obtain the support-constrained encoding matrix $\boldsymbol{G}$ from a generator matrix of a $[23, 9]_{4^9}$ LRS code with $\ell = 3$ blocks. The lengths of the blocks are $(n_1, n_2, n_3) = (8, 7, 8)$ (see (i) in Section 4.4.3 for the computation of these parameters). Let γ be a primitive element of $\mathbb{F}_{4^9}$. The block representatives of the LRS code are $(a_1, a_2, a_3) = (1, \gamma, \gamma^2)$ and the column multipliers are $\boldsymbol{b} = (1, \gamma, \ldots, \gamma^7, \gamma, \gamma^2, \ldots, \gamma^7, \gamma^2, \gamma^3, \ldots, \gamma^9)$. We construct a generator matrix $\boldsymbol{G}^{(\mathsf{LRS})}$ of the LRS code according to (4.4) and find a full-rank matrix $\boldsymbol{T} \in \mathbb{F}_{4^9}^{9\times 9}$ such that the support-constrained encoding matrix $\boldsymbol{G}$ is given by $\boldsymbol{G} = \boldsymbol{T} \cdot \boldsymbol{G}^{(\mathsf{LRS})}$. It can be verified by Theorem 4.3 that such a matrix $\boldsymbol{T}$ exists over $\mathbb{F}_{4^9}$ and it can be found by solving a linear system of equations. For this example, we found the following $\boldsymbol{T}$ as a solution (see Section 4.4.3 (iv) for the computation method that we used here).

$$\boldsymbol{T} = \begin{pmatrix}
\gamma^{29883} & \gamma^{208968} & \gamma^{19488} & \gamma^{27791} & \gamma^{137529} & \gamma^{135128} & \gamma^{142532} & \gamma^{123564} & \gamma^{199506} \\
\gamma^{8272} & \gamma^{137891} & \gamma^{117682} & \gamma^{134830} & \gamma^{175546} & \gamma^{199273} & \gamma^{233167} & \gamma^{13175} & \gamma^{75587} \\
\gamma^{171183} & \gamma^{60863} & \gamma^{88547} & \gamma^{152810} & \gamma^{183852} & \gamma^{129008} & \gamma^{223733} & \gamma^{220778} & \gamma^{215911} \\
\gamma^{136657} & \gamma^{53725} & \gamma^{187129} & \gamma^{236279} & \gamma^{244758} & \gamma^{124656} & \gamma^{163100} & \gamma^{222367} & \gamma^{245041} \\
\gamma^{67172} & \gamma^{31331} & \gamma^{217264} & \gamma^{133630} & \gamma^{190037} & \gamma^{228340} & \gamma^{210873} & \gamma^{222699} & \gamma^{102082} \\
\gamma^{180377} & \gamma^{78748} & \gamma^{71136} & \gamma^{170404} & \gamma^{251773} & \gamma^{44364} & \gamma^{188627} & \gamma^{44347} & \gamma^{145983} \\
\gamma^{82368} & \gamma^{167072} & \gamma^{210000} & \gamma^{110692} & \gamma^{24773} & \gamma^{69984} & \gamma^{182180} & \gamma^{211569} & \gamma^{24237} \\
\gamma^{78461} & \gamma^{249391} & \gamma^{68483} & \gamma^{120459} & \gamma^{140206} & \gamma^{243029} & \gamma^{126875} & \gamma^{75641} & \gamma^{12289} \\
\gamma^{33368} & \gamma^{98307} & \gamma^{247550} & \gamma^{210053} & \gamma^{223247} & \gamma^{103052} & \gamma^{160318} & \gamma^{69947} & \gamma^{42305}
\end{pmatrix}$$

Remark 4.2. *With the choice of ℓ and $(n_1, \ldots, n_\ell)$ for this toy LRS code we intend to show that the number of blocks ℓ does not need to be the same as the number of source nodes s. The value of ℓ determines the upper bound in (4.32) on the sum-rank weight of $\boldsymbol{E}$. We listed several other parameters of the LRS codes in Table 4.1 that can be used for this network example. It can be seen that, the larger ℓ is, the larger error-correction capability is required, which results in larger sum-rank distance of the LRS code and hence larger total length n and field size q^m.*

However, larger ℓ may result in a smaller field size. For instance, suppose that the messages $\boldsymbol{m}_i$ are over $\mathbb{F}_{3^{11}}$. According to Table 4.1, setting $\ell = 1$ (i.e., using a distribued Gabidulin code [HHD14]) requires a field size $q^m = 3^{15}$ while using the distributed LRS codes with $\ell = 2$ requires a field size $q^m = 3^{11}$ (note that the field size of the messages is 3^{11} hence the code should be over $\mathbb{F}_{3^{11}}$).

Table 4.1: Parameters of distributed LRS codes for the toy network example while increasing ℓ. The q and m are the minimal parameters of the required field over which the $[n, \widetilde{k}, d]$ distributed LRS code can be constructed, where $d = 2\ell t + \rho + 1$ is the sum-rank distance of the distributed LRS code.

ℓ	q	m	$[n, \widetilde{k}, d]$	$(n_1, \ldots, n_\ell)$	$(n_{\mathcal{J}_1}, n_{\mathcal{J}_2}, n_{\mathcal{J}_3}, n_{\mathcal{J}_4})$
1 (Gabidulin code)	2	15	$[15, 9, 7]$	(15)	$(6, 1, 0, 8)$
2	3	10	$[19, 9, 11]$	$(10, 9)$	$(6, 5, 0, 8)$
3 (Fig. 4.4)	4	9	$[23, 9, 15]$	$(8, 7, 8)$	$(6, 7, 2, 8)$
4	5	9	$[27, 9, 19]$	$(7, 7, 7, 6)$	$(6, 7, 6, 8)$
5	7	11	$[33, 11, 23]$	$(7, 6, 7, 7, 6)$	$(8, 9, 6, 10)$
6	7	12	$[38, 12, 27]$	$(7, 6, 6, 7, 6, 6)$	$(9, 10, 8, 11)$
7	8	13	$[43, 13, 31]$	$(6, 6, 6, 7, 6, 6, 6)$	$(10, 11, 10, 12)$

In Table 4.2, we list the parameters of LRS codes for several different $\mathcal{S} = \{\mathcal{J}_1, \mathcal{J}_2, \mathcal{J}_3, \mathcal{J}_4\}$. It can be seen that encoding each message independently requires a longer code (hence, a larger alphabet size) than jointly encoding subsets of messages.

Table 4.2: Parameters of distributed LRS codes for the toy example while changing $\mathcal{S}$.

$\mathcal{S}$	ℓ	q	m	$[n, \widetilde{k}, d]$	$(n_1, \ldots, n_\ell)$	$(n_{\mathcal{J}_1}, n_{\mathcal{J}_2}, n_{\mathcal{J}_3}, n_{\mathcal{J}_4})$
$\{\{1\}, \{2\}, \{3\}, \{4\}\}$	1	2	33	$[33, 27, 7]$	(33)	$(7, 9, 8, 9)$
	2	3	39	$[49, 39, 11]$	$(25, 24)$	$(11, 13, 12, 13)$
	3	4	51	$[65, 51, 15]$	$(22, 22, 21)$	$(15, 17, 16, 17)$
$\{\{1, 2\}, \{1, 3\}, \{2, 4\}, \{3, 4\}\}$	1	2	17	$[17, 11, 7]$	(17)	$(6, 1, 3, 7)$
	2	3	15	$[25, 15, 11]$	$(13, 12)$	$(10, 1, 3, 11)$
	3	4	19	$[33, 19, 15]$	$(11, 11, 11)$	$(14, 1, 3, 15)$

Now we proceed to apply the *lifting* technique [SKK08] to deal with the non-coherent situation. Supposing $(\boldsymbol{c}_{\mathcal{J}_1}, \boldsymbol{c}_{\mathcal{J}_2}, \boldsymbol{c}_{\mathcal{J}_3}, \boldsymbol{c}_{\mathcal{J}_4}) = (\boldsymbol{m}_1, \boldsymbol{m}_2, \boldsymbol{m}_3, \boldsymbol{m}_4) \cdot \boldsymbol{G}$. Each source node $S_{\mathcal{J}_i}$ generates $\boldsymbol{C}_{\mathcal{J}_i} = \mathrm{ext}_\beta(\boldsymbol{c}_{\mathcal{J}_i}) \in \mathbb{F}_q^{m \times n_{\mathcal{J}_i}}$ by the map defined below and lifts the $\boldsymbol{C}_{\mathcal{J}_i}^\top$ by adding the identity and zero matrices as in (4.33) to obtain the transmitted packets (rows of $\boldsymbol{X}$).

$$\boldsymbol{X} = \underbrace{\begin{pmatrix} \boldsymbol{I}_{n_{\mathcal{J}_1}} & & & & \boldsymbol{C}_{\mathcal{J}_1}^\top \\ & \boldsymbol{I}_{n_{\mathcal{J}_2}} & & & \boldsymbol{C}_{\mathcal{J}_2}^\top \\ & & \boldsymbol{I}_{n_{\mathcal{J}_3}} & & \boldsymbol{C}_{\mathcal{J}_3}^\top \\ & & & \boldsymbol{I}_{n_{\mathcal{J}_4}} & \boldsymbol{C}_{\mathcal{J}_4}^\top \end{pmatrix}}_{n} \Bigg\} n = \textstyle\sum_{i=1}^4 n_{\mathcal{J}_i} \qquad (4.33)$$

Each row is a packet of length $n + m$ ($= 23 + 9 = 31$ for the toy example) over $\mathbb{F}_q$ ($\mathbb{F}_4$) transmitted into the network. Note that for the lifting step, the centralized coordination unit is also needed to instruct the source nodes where to put the identity matrix in their packets.

4.4.3 The General Scheme: Distributed LRS Codes

In the following we present the general scheme at the centralized coordination unit to design the overall distributed LRS codes, given:

- the total number of messages h and their lengths $r_1, \ldots, r_h$;
- the set $\mathcal{S} = \{\mathcal{J}_1, \ldots, \mathcal{J}_s\}$, where each $\mathcal{J}_i \subset [h]$ contains the indices of the messages that the source node $S_{\mathcal{J}_i}$ has access to;
- the (t, ρ)-adversarial model: the maximum number t of malicious nodes and the maximum number ρ of frozen nodes in the network;
- the number of blocks ℓ of the LRS code.

The task is to design the $n_{\mathcal{J}}$, for all $\mathcal{J} \in \mathcal{S}$, such that the sink can recover all h messages. The goal of the design is to minimize n, the total number of the encoded symbols.

The general scheme contains the following steps:

(i) Solving the following integer linear programming problem for $(n_{\mathcal{J}_1}, \ldots, n_{\mathcal{J}_s})$

$$\begin{aligned} \text{minimize} \quad & n = n_{\mathcal{J}_1} + \cdots + n_{\mathcal{J}_s} \\ \text{subject to} \quad & \forall \varnothing \neq \mathcal{J}' \subseteq [h], \ \sum_{i \in \mathcal{J}'} r_i + 2t + \rho \leqslant n - \sum_{\substack{\mathcal{J} \in \mathcal{S} \\ \mathcal{J} \subseteq [h] \setminus \mathcal{J}'}} n_{\mathcal{J}} \ , \qquad (4.34) \\ & \forall \varnothing \neq \Omega \subseteq [h], \ \sum_{\substack{\mathcal{J} \in \mathcal{S} \\ [h] \setminus \mathcal{J} \supseteq \Omega}} n_{\mathcal{J}} + \sum_{i \in \Omega} r_i \leqslant n - 2\ell t - \rho \ , \qquad (4.35) \\ & \forall \mathcal{J} \in \mathcal{S}, \qquad n_{\mathcal{J}} \geqslant 0 \ . \end{aligned}$$

Remark: Recall that we assume that the min-cut $w_{\mathcal{J}'} = n - \sum_{\substack{\mathcal{J} \in \mathcal{S} \\ \mathcal{J} \subseteq [h] \setminus \mathcal{J}'}} n_{\mathcal{J}}$, *for all* $\varnothing \neq \mathcal{J}' \subseteq [h]$. *With the constraints in* (4.34), *the choice of* $(n_{\mathcal{J}_1}, \ldots, n_{\mathcal{J}_s})$ *guarantees that the message lengths* $(r_1, \ldots, r_h)$ *are in the capacity region given in Theorem 4.7. Let*

$$\widetilde{k} := \max_{\varnothing \neq \Omega \subseteq [h]} \sum_{\substack{\mathcal{J} \in \mathcal{S} \\ [h] \setminus \mathcal{J} \supseteq \Omega}} n_{\mathcal{J}} + \sum_{i \in \Omega} r_i \ .$$

By Theorem 4.4, there exist a subcode of an $[n, \widetilde{k}]$ *LRS code whose generator matrix fulfills the support constraints of the encoding matrix* $\boldsymbol{G}$ *for the distributed multi-source network. The constraints in* (4.35) *guarantee that* $\widetilde{k} \leqslant n - 2\ell t - \rho$, *which ensures that the* $[n, \widetilde{k}]$ *LRS code can decode the rank-metric errors and erasures induced by the* (t, ρ)*-adversarial model (see Section 4.4.1).*

(ii) Determine the field size q^m required for the $[n, \widetilde{k}]$ LRS code with ℓ blocks according to Theorem 4.4.

Remark: The total length should be distributed as evenly as possible into ℓ *blocks so that the extension degree* m *is minimized.*

(iii) Construct a generator matrix $\boldsymbol{G}^{(\mathsf{LRS})}$ of the $[n, \widetilde{k}]_{q^m}$ LRS code according to (4.4).

(iv) Find a full-rank matrix $\boldsymbol{T} \in \mathbb{F}_{q^m}^{k \times \widetilde{k}}$ (where $k = \sum_{i=1}^{h} r_i$) such that the support-constrained encoding matrix $\boldsymbol{G}$ can be obtained from $\boldsymbol{G} = \boldsymbol{T} \cdot \boldsymbol{G}^{(\mathsf{LRS})}$.

Remark: This can be done by solving a linear system of equations for the entries of $\boldsymbol{T}$. For example, in our implementation, we see the entries of $\boldsymbol{T}$ as variables in a multivariate polynomial ring $\mathcal{R} = \mathbb{F}_{q^m}[T_{11}, \ldots, T_{kk}]$ and translate the constraints (the zero entries in $\boldsymbol{G}$) into a system of linear equations. We use the facilities (Gröbner bases, variate, etc.) for multivariate polynomials embedded in SageMath [The22] to solve the system. Note that the `variate()` function in SageMath avoids computing the whole solution space when the system is underdetermined. If this is the case, let λ be the degree of freedom of the system. We assign random values in $\mathbb{F}_{q^m}$ to λ variables, so that it becomes a determined system that is solvable by `variate()`.

4.5 Vector Network Coding for Generalized Combination Networks

A *multicast network* is a network with exactly one source and multiple receivers demanding all the messages from the source. In networks that apply routing, every relay node can only pass on their received data. *Network coding* has been attracting increasing attention since the seminal paper by Ahlswede, et. al. [ACLY00], which showed that the throughput of the network can be increased significantly by not just forwarding packets but also performing operations on them. We formulate the *network coding problem* as follows: for each node in the network, find an encoding function of its incoming messages for each of its outgoing links. A *solution* is a set of the encoding functions at all the nodes in the network, such that each receiver can recover all (or a predefined subset of all) the messages. A network is *solvable* if a solution exists.

Although network coding has the advantage of good throughput, the encoding at relay nodes incurs extra delay and memory occupation than routing. This section considers these costs from the aspect of the required alphabet size to utilize network coding. Reducing the alphabet size of the coding operations results in less complexity, hence less delay, and less memory occupation for practical implementations of network coding [LSB06; LS09; GSRM19].

In the following, we first formally introduce the concepts that are considered in this section.

Generalized Combination Networks

The main object that we study in this section is the class of *generalized combination networks*. An $(\varepsilon, \ell) - \mathcal{N}_{h,r,\alpha\ell+\varepsilon}$ generalized combination network is illustrated in Figure 4.5 (see also [EW18]).

The network has three layers. The first layer consists of a source with h source messages. The source is connected to each of r middle nodes in the second layer via ℓ parallel links (solid lines). Any α middle nodes are connected to a unique receiver in the third layer, each via ℓ parallel links. This implies that there are $N = \binom{r}{\alpha}$ receivers that receive distinct packets from the second layer. In addition, each receiver is also connected to the source via ε direct links (dashed lines). It was shown in [EW18, Theorem 8] that the $(\varepsilon, \ell) - \mathcal{N}_{h,r,\alpha\ell+\varepsilon}$ network has a trivial solution if $h \leqslant \ell + \varepsilon$ and it has no solution if $h > \alpha\ell + \varepsilon$. We only consider non-trivially solvable networks, hence we assume that $\ell + \varepsilon < h \leqslant \alpha\ell + \varepsilon$ throughout the section.

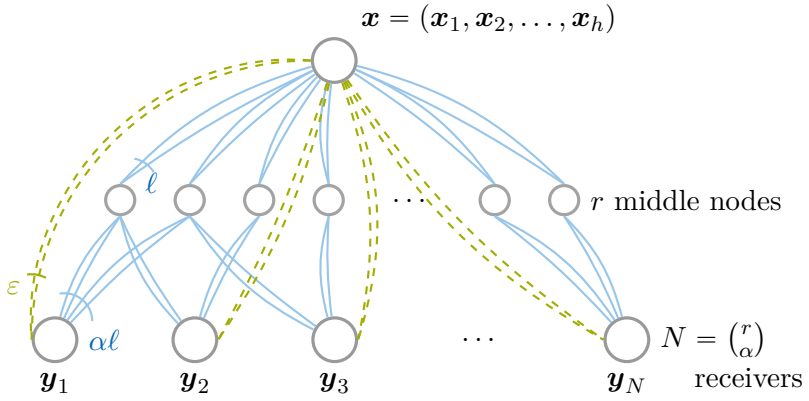


Figure 4.5: Illustration of an $(\varepsilon, \ell) - \mathcal{N}_{h,r,\alpha\ell+\varepsilon}$ network.

Vector/Scalar Linear Solutions of Network Coding

In linear network coding, the outgoing links of each relay node carry linear functions of the messages from the incoming links. The linear functions are called *coding coefficients*. The set of the coding coefficients is the solution of the linear network.

If the messages are scalars in $\mathbb{F}_q$ and the coding coefficients are vectors over $\mathbb{F}_q$, then a solution is called a *scalar linear solution*, denoted by $(q, 1)$-linear solution. If the messages are vectors in $\mathbb{F}_q^t$, and the coding coefficients are matrices over $\mathbb{F}_q$, then a solution is called a *vector linear solution*, denoted by (q, t)-linear solution.

We now formulate a solution to the $(\varepsilon, \ell) - \mathcal{N}_{h,r,\alpha\ell+\varepsilon}$ generalized combination network illustrated in Fig. 4.5. W.l.o.g., we only formulate it with the notations for a vector linear solution; a scalar linear solution can be obtained by simply setting $t = 1$.

Denote by $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_h \in \mathbb{F}_q^t$ the h source messages and by $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_N \in \mathbb{F}_q^{(\varepsilon+\alpha\ell)t}$ the packets received by each receiver. For each $i \in [N]$, $\boldsymbol{y}_i$ is the concatenation of all the packets that the i-th receiver gets from the α middle nodes and the source node. Since each middle node has ℓ incoming links and $\alpha\ell$ outgoing links, we assume w.l.o.g. that the middle nodes just forward their incoming packets and the encoding is done at the source node.

We denote by $\boldsymbol{A}_1, \ldots, \boldsymbol{A}_r \in \mathbb{F}_q^{\ell t \times ht}$ the coding coefficients used by the source node for the messages transmitted to the r middle nodes, and by $\boldsymbol{B}_1, \ldots, \boldsymbol{B}_N \in \mathbb{F}_q^{\varepsilon t \times ht}$ the coding coefficients used by the source node for the messages transmitted directly to the receivers. Then, for each $i \in [N]$,

$$\boldsymbol{y}_i = \underbrace{\begin{pmatrix} \boldsymbol{A}_{i_1} \\ \vdots \\ \boldsymbol{A}_{i_\alpha} \\ \boldsymbol{B}_i \end{pmatrix}}_{(\varepsilon+\alpha\ell)t \times ht} \cdot \underbrace{\begin{pmatrix} \boldsymbol{x}_1 \\ \vdots \\ \boldsymbol{x}_h \end{pmatrix}}_{ht \times 1},$$

where $\{\boldsymbol{A}_{i_1}, \ldots, \boldsymbol{A}_{i_\alpha}\} \subset \{\boldsymbol{A}_1, \ldots, \boldsymbol{A}_r\}$.

Note that the receivers can recover the h source messages $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_h$ if and only if

$$\operatorname{rank} \begin{pmatrix} \boldsymbol{A}_{i_1} \\ \vdots \\ \boldsymbol{A}_{i_\alpha} \end{pmatrix} \geqslant (h-\varepsilon)t, \ \forall i \in [N] \ . \tag{4.36}$$

Hence a solution to the $(\varepsilon, \ell)-\mathcal{N}_{h,r,\alpha\ell+\varepsilon}$ network is a set of the coding coefficients $\{\boldsymbol{A}_1, \ldots, \boldsymbol{A}_r\}$ such that (4.36) holds. The coding coefficients for the direct links $\boldsymbol{B}_1, \ldots, \boldsymbol{B}_N$ can be determined once $\{\boldsymbol{A}_1, \ldots, \boldsymbol{A}_r\}$ is given.

Gap between Required Alphabet Sizes for Scalar and Vector Solutions

The goal of this section is to investigate the gap between the minimum required alphabet size for scalar and vector solutions of the generalized combination networks. This gap was shown to be positive for generalized combination networks [EW18]. We further quantify the advantage of vector linear solutions versus scalar linear solutions of the generalized combination networks. For this purpose, we need to fix a metric.

We follow the notations from [CCE+20] to distinguish between optimal scalar and vector solutions. Given a generalized combination network $\mathcal{N}$, let

$$q_s(\mathcal{N}) := \min\{q \mid \mathcal{N} \text{ has a } (q,1)\text{-linear solution}\} \ .$$

The $(q_s(\mathcal{N}), 1)$-linear solution is said to be *scalar-optimal*. Similarly, let

$$q_v(\mathcal{N}) := \min\{q^t \mid \mathcal{N} \text{ has a } (q,t)\text{-linear solution}\} \ .$$

Note that $q_v(\mathcal{N})$ is defined by the size of the vector space, rather than the field size. For $q^t = q_v(\mathcal{N})$, the (q,t)-linear solution is called *vector-optimal*. We define the *gap* as

$$\operatorname{gap}_2(\mathcal{N}) := \log_2(q_s(\mathcal{N})) - \log_2(q_v(\mathcal{N})) \ ,$$

which intuitively measures the advantage of vector network coding by the amount of extra bits per transmitted symbol that an optimal scalar linear solution has to pay compared to an optimal vector linear solution.

Overview of Results

In Section 4.5.1, we give two upper bounds on $r_{\max}$, the maximal number of nodes in the middle layer of a generalized combination network (Corollary 4.2 (valid for $h \geqslant 2\ell + \varepsilon$) and Corollary 4.3 (a better bound for $\alpha = 2$)). In Section 4.5.2, we give two lower bounds on $r_{\max}$ (Theorem 4.10 and Corollary 4.4 ($h \leqslant 2\ell + \varepsilon$)). In Section 4.5.3, we provide an upper bound on $\operatorname{gap}_2(\mathcal{N})$ for any fixed generalized combination network $\mathcal{N}$ (Theorem 4.13), and a lower bound on $\operatorname{gap}_2(\mathcal{N})$ (Theorem 4.14). We compare the new bounds with some existing bounds on $r_{\max}$ in Section 4.5.4 and summarize the best known bound on $r_{\max}$ in Table 4.3.

4.5.1 Upper Bounds on the Maximum Number of Middle Layer Nodes

The Grassmannian of dimension k is a set of all k-dimensional subspaces of $\mathbb{F}_q^n$. Recall that its cardinality is the well-known q-binomial coefficient:

$$|\mathcal{G}_q(n,k)| = \begin{bmatrix} n \\ k \end{bmatrix}_q := \prod_{i=0}^{k-1} \frac{q^n - q^i}{q^k - q^i} = \prod_{i=0}^{k-1} \frac{q^{n-i} - 1}{q^{k-i} - 1} \ .$$

A good approximation of the q-binomial coefficient can be found in [KK08, Lemma 4]:

$$q^{k(n-k)} \leqslant \begin{bmatrix} n \\ k \end{bmatrix}_q < \gamma \cdot q^{k(n-k)} \ , \tag{4.37}$$

where $\gamma \approx 3.48$.

Lemma 4.1. *Let $\alpha \geqslant 2$, $h, \ell, t \geqslant 1$, $\varepsilon \geqslant 0$, $h - \varepsilon \geqslant 2\ell$, and let $\mathcal{T}$ be a collection of subspaces of $\mathbb{F}_q^{(h-\varepsilon)t}$ such that*

(i) each subspace has dimension at most ℓt, and

(ii) any subset of α subspaces spans $\mathbb{F}_q^{(h-\varepsilon)t}$.

Then, we have $\alpha\ell \geqslant h - \varepsilon$ and

$$|\mathcal{T}| \leqslant \left(\left\lfloor \frac{h-\varepsilon}{\ell} \right\rfloor - 2 \right) + \left(\alpha - \left\lfloor \frac{h-\varepsilon}{\ell} \right\rfloor + 1 \right) \begin{bmatrix} \ell t + 1 \\ 1 \end{bmatrix}_q .$$

Proof. Take arbitrarily $\left\lfloor \frac{h-\varepsilon}{\ell} \right\rfloor - 2$ subspaces from $\mathcal{T}$ and a subspace $W \subset \mathbb{F}_q^{(h-\varepsilon)t}$ of dimension $(h-\varepsilon)t - \ell t - 1$ which contains all these $\left\lfloor \frac{h-\varepsilon}{\ell} \right\rfloor - 2$ subspaces. Then, for any subspace $T \in \mathcal{T}$, there is a *hyperplane* (an $((h-\varepsilon)t - 1)$-dimensional subspace) of $\mathbb{F}_q^{(h-\varepsilon)t}$ containing both W and T. Note that there are $\begin{bmatrix} \ell t+1 \\ \ell t \end{bmatrix} = \begin{bmatrix} \ell t+1 \\ 1 \end{bmatrix}$ hyperplanes of $\mathbb{F}_q^{(h-\varepsilon)t}$ containing W and each of them contains at most $\alpha - 1$ subspaces from $\mathcal{T}$. Thus,

$$\begin{aligned} |\mathcal{T}| &\leqslant \left(\left\lfloor \frac{h-\varepsilon}{\ell} \right\rfloor - 2 \right) + \begin{bmatrix} \ell t + 1 \\ \ell t \end{bmatrix}_q \left(\alpha - 1 - \left(\left\lfloor \frac{h-\varepsilon}{\ell} \right\rfloor - 2 \right) \right) \\ &= \left(\left\lfloor \frac{h-\varepsilon}{\ell} \right\rfloor - 2 \right) + \left(\alpha - \left\lfloor \frac{h-\varepsilon}{\ell} \right\rfloor + 1 \right) \begin{bmatrix} \ell t + 1 \\ 1 \end{bmatrix}_q . \end{aligned}$$

□

Theorem 4.8. *Let $\alpha \geqslant 2$, $h, \ell, t \geqslant 1$, $\varepsilon \geqslant 0$, $h - \varepsilon \geqslant 2\ell$, and let $\mathcal{S}$ be a collection of subspaces of $\mathbb{F}_q^{ht}$ such that*

(i) each subspace has dimension at most ℓt, and

(ii) any subset of α subspaces spans a subspace of dimension at least $(h - \varepsilon)t$.

Then, we have $\alpha\ell \geqslant h-\varepsilon$ *and*

$$\begin{aligned}|\mathcal{S}| &\leqslant \begin{bmatrix}(\varepsilon+\ell)t\\ \varepsilon t\end{bmatrix}_q\left(\left(\alpha-\left\lfloor\frac{h-\varepsilon}{\ell}\right\rfloor+1\right)\frac{q^{\ell t+1}-1}{q-1}-1\right)+\left\lfloor\frac{h-\varepsilon}{\ell}\right\rfloor-1\\ &< \gamma\left(\alpha-\left\lfloor\frac{h-\varepsilon}{\ell}\right\rfloor+1\right)q^{\ell t(\varepsilon t+1)}+\left\lfloor\frac{h-\varepsilon}{\ell}\right\rfloor-1\ .\end{aligned} \tag{4.38}$$

Proof. Take arbitrarily $\left\lfloor\frac{h-\varepsilon}{\ell}\right\rfloor-1$ subspaces from $\mathcal{S}$ and a subspace $W\subset\mathbb{F}_q^{ht}$ of dimension $(h-\varepsilon)t-\ell t$ such that W contains all these $\left\lfloor\frac{h-\varepsilon}{\ell}\right\rfloor-1$ subspaces. Then, for any subspace $S\in\mathcal{S}$ there is a subspace of dimension $(h-\varepsilon)t$ containing both W and S.

Let $m:=\begin{bmatrix}(\varepsilon+\ell)t\\ \varepsilon t\end{bmatrix}_q$. Then, there are m subspaces of dimension $(h-\varepsilon)t$ containing W, say $W_1,W_2,\ldots,W_m$. Note that every subset of α subspaces in $W_i\cap\mathcal{S}$ span the subspace W_i. According to Lemma 4.1, we have

$$|W_i\cap\mathcal{S}|\leqslant\left(\left\lfloor\frac{h-\varepsilon}{\ell}\right\rfloor-2\right)+\left(\alpha-\left\lfloor\frac{h-\varepsilon}{\ell}\right\rfloor+1\right)\begin{bmatrix}\ell t+1\\ 1\end{bmatrix}_q\ .$$

Hence,

$$\begin{aligned}|\mathcal{S}| &\leqslant \sum_{i=1}^{m}\left(|W_i\cap\mathcal{S}|-\left(\left\lfloor\frac{h-\varepsilon}{\ell}\right\rfloor-1\right)\right)+\left\lfloor\frac{h-\varepsilon}{\ell}\right\rfloor-1\\ &\leqslant \begin{bmatrix}(\varepsilon+\ell)t\\ \varepsilon t\end{bmatrix}_q\left(\left(\alpha-\left\lfloor\frac{h-\varepsilon}{\ell}\right\rfloor+1\right)\frac{q^{\ell t+1}-1}{q-1}-1\right)+\left\lfloor\frac{h-\varepsilon}{\ell}\right\rfloor-1\ .\end{aligned}$$

The inequality (4.38) is derived from (4.37). □

The following corollary rephrases Theorem 4.8 with network parameters.

Corollary 4.2. *Let* $\alpha\geqslant 2$, $h,\ell,t\geqslant 1$, $\varepsilon\geqslant 0$, *and* $h-\varepsilon\geqslant 2\ell$. *If* $(\varepsilon,\ell)-\mathcal{N}_{h,r,\alpha\ell+\varepsilon}$ *has a* (q,t)*-linear solution then*

$$r\leqslant r_{\max}<\gamma\theta q^{\ell t(\varepsilon t+1)}+\alpha-\theta\ ,$$

where $\theta:=\alpha-\left\lfloor\frac{h-\varepsilon}{\ell}\right\rfloor+1$ *and* $\gamma\approx 3.48$.

Proof. If a (q,t)-linear solution exists, then each of the r nodes in the middle layer gets a subspace of dimension ℓt of the source messages space. Since all receivers are able to recover the entire source message space, every α-subset of the middle nodes span a subspace of dimension at least $(h-\varepsilon)t$. The statement then follows from Theorem 4.8. □

Theorem 4.8 and Corollary 4.2 are valid for all $\alpha\geqslant 2$. However, we derive a tighter upper bound for $\alpha=2$, as shown in the following theorem.

Theorem 4.9. *Let* $\alpha=2,h,\ell,t\geqslant 1$, $\varepsilon\geqslant 0$, *and let* $\mathcal{S}$ *be a collection of subspaces of* $\mathbb{F}_q^{ht}$ *such that*

(i) each subspace has dimension at most ℓt, *and*

(ii) the sum of any two subspaces has dimension at least $(h-\varepsilon)t$.

Then, we have

$$|\mathcal{S}| \leqslant \frac{\begin{bmatrix} ht \\ 2\ell t-(h-\varepsilon)t+1 \end{bmatrix}_q}{\begin{bmatrix} \ell t \\ 2\ell t-(h-\varepsilon)t+1 \end{bmatrix}_q} < \gamma \cdot q^{(h-\ell)(2\ell+\varepsilon-h)t^2+(h-\ell)t} \ .$$

Proof. We may assume that each subspace has dimension ℓt. Since the sum of every two subspaces has dimension at least $(h-\varepsilon)t$, their intersection has dimension at most $2\ell t-(h-\varepsilon)t$. It follows that any subspace of dimension $2\ell t-(h-\varepsilon)t+1$ is contained in at most one subspace of $\mathcal{S}$. Note that there are $\begin{bmatrix} ht \\ 2\ell t-(h-\varepsilon)t+1 \end{bmatrix}_q$ subspaces of dimension $2\ell t-(h-\varepsilon)t+1$ and each subspace of dimension ℓt contains $\begin{bmatrix} \ell t \\ 2\ell t-(h-\varepsilon)t+1 \end{bmatrix}_q$ such spaces. We then have

$$|\mathcal{S}| \leqslant \begin{bmatrix} ht \\ 2\ell t-(h-\varepsilon)t+1 \end{bmatrix}_q \Big/ \begin{bmatrix} \ell t \\ 2\ell t-(h-\varepsilon)t+1 \end{bmatrix}_q \ .$$

□

The following corollary rephrases Theorem 4.9 with network parameters.

Corollary 4.3. *Let $\alpha = 2$, $h, \ell, t \geqslant 1$, $\varepsilon \geqslant 0$. If $(\varepsilon, \ell) - \mathcal{N}_{h,r,\alpha\ell+\varepsilon}$ has a (q,t)-linear solution then*

$$r \leqslant r_{\max} < \gamma \cdot q^{(h-\ell)(2\ell+\varepsilon-h)t^2+(h-\ell)t} \ ,$$

where $\gamma \approx 3.48$.

Proof. If a (q,t)-linear solution exists, then each of the r nodes in the middle layer gets a subspace of dimension ℓt of the source messages space. Since all receivers are able to recover the entire source message space, any two subset of the middle nodes span a subspace of dimension at least $(h-\varepsilon)t$. We then use Theorem 4.9. □

4.5.2 Lower Bounds on the Maximum Number of Middle Layer Nodes

We now turn to study a lower bound on $r_{\max}$ with the parameters $\alpha, \ell, \varepsilon, h$ being fixed. The main results are summarized in Theorem 4.10 and Corollary 4.4.

A Lower Bound by Lovász-Local Lemma

Lemma 4.2 (Lovász-Local-Lemma [AS08, Ch. 5], [Bec91])**.** *Let $\mathcal{E}_1, \mathcal{E}_2, \ldots, \mathcal{E}_k$ be a sequence of events. Each event occurs with probability at most p and each event is independent of all the other events except for at most d of them. If $epd \leqslant 1$, where $e \approx 2.718$ is the base of natural logarithms, then there is a nonzero probability that none of the events occurs.*

Recall that a solution to the $(\varepsilon, \ell) - \mathcal{N}_{h,r,\alpha\ell+\varepsilon}$ network is a set of the coding coefficients $\{\boldsymbol{A}_1, \ldots, \boldsymbol{A}_r\}$ such that (4.36) holds. We choose the matrices $\boldsymbol{A}_1, \ldots, \boldsymbol{A}_r \in \mathbb{F}_q^{\ell t \times ht}$ independently and uniformly at random. For $1 \leqslant i_1 < \cdots < i_\alpha \leqslant r$, we define the event

$$\mathcal{E}_{i_1,\ldots,i_\alpha} := \left\{ (\boldsymbol{A}_{i_1}, \ldots, \boldsymbol{A}_{i_\alpha}) \mid \operatorname{rank} \begin{pmatrix} \boldsymbol{A}_{i_1} \\ \vdots \\ \boldsymbol{A}_{i_\alpha} \end{pmatrix} < (h-\varepsilon)t \right\} .$$

Lemma 4.3. *Let $\alpha \geqslant 2$, $h, \ell, t \geqslant 1$, $\varepsilon \geqslant 0$. Fixing $1 \leqslant i_1 < \cdots < i_\alpha \leqslant r$, we have*

$$\Pr(\mathcal{E}_{i_1,\ldots,i_\alpha}) \leqslant 2\gamma \cdot q^{(h-\alpha\ell-\varepsilon)\varepsilon t^2+(h-\alpha\ell-2\varepsilon)t-1} ,$$

where $\gamma \approx 3.48$.

Proof. The number of matrices $\boldsymbol{A} \in \mathbb{F}_q^{m\times n}$ of rank s is

$$M(m,n,s) := \prod_{j=0}^{s-1} \frac{(q^m-q^j)(q^n-q^j)}{q^s-q^j} \leqslant \gamma \cdot q^{(m+n)s-s^2}. \tag{4.39}$$

Then,

$$\begin{aligned}
\Pr(\mathcal{E}_{i_1,\ldots,i_\alpha}) &= \frac{\sum_{i=0}^{(h-\varepsilon)t-1} M(\alpha\ell t, ht, i)}{q^{\alpha\ell h t^2}} \\
&\leqslant \frac{\sum_{i=0}^{(h-\varepsilon)t-1} \gamma \cdot q^{(h+\alpha\ell)ti-i^2}}{q^{\alpha\ell h t^2}} && (4.40)\\
&\leqslant \gamma \cdot \frac{q}{q-1} \cdot q^{\max_i\{(h+\alpha\ell)ti-i^2\}-\alpha\ell h t^2} && (4.41)\\
&= \gamma \cdot \frac{q}{q-1} \cdot q^{(h+\alpha\ell)ti-i^2|_{i=(h-\varepsilon)t-1}-\alpha\ell h t^2} && (4.42)\\
&\leqslant \gamma \cdot 2 \cdot q^{(h-\alpha\ell-\varepsilon)\varepsilon t^2+(h-\alpha\ell-2\varepsilon)t-1} ,
\end{aligned}$$

where (4.40) holds due to (4.39), (4.41) follows from a geometric sum, and (4.42) follows by maximizing $(h+\alpha\ell)ti - i^2$. □

Lemma 4.4. *Let $\alpha \geqslant 2$, $h, \ell, t \geqslant 1$, $\varepsilon \geqslant 0$. Fixing $1 \leqslant i_1 < \cdots < i_\alpha \leqslant r$, the event $\mathcal{E}_{i_1,\ldots,i_\alpha}$ is statistically independent of all the other events $\mathcal{E}_{i'_1,\ldots,i'_\alpha}$ ($1 \leqslant i'_1 < \cdots < i'_\alpha \leqslant r$), except for at most $\alpha\binom{r-1}{\alpha-1}$ of them.*

Proof. For $1 \leqslant i_1 < \cdots < i_\alpha \leqslant r$ and $1 \leqslant i'_1 < \cdots < i'_\alpha \leqslant r$, the events $\mathcal{E}_{i_1,\ldots,i_\alpha}$ and $\mathcal{E}_{i'_1,\ldots,i'_\alpha}$ are statistically independent if and only if $\{i_1,\ldots,i_\alpha\} \cap \{i'_1,\ldots,i'_\alpha\} = \varnothing$. Thus, having chosen $1 \leqslant i_1 < \cdots < i_\alpha \leqslant r$, there are at most $\binom{\alpha}{1}\binom{r-1}{\alpha-1}$ ways of choosing $\{i'_1,\ldots,i'_\alpha\}$ such that it is not independent from $\{i_1,\ldots,i_\alpha\}$ (including the case $\{i'_1,\ldots,i'_\alpha\} = \{i_1,\ldots,i_\alpha\}$). □

Remark 4.3. *Lemma 4.4 is a union-bound argument on the number of dependent events. The exact number is $\binom{r}{\alpha} - \binom{r-\alpha}{\alpha}$. However the exact expression makes it harder to resolve for r later thus we use the bound instead.*

Theorem 4.10. *Let $\alpha \geqslant 2$, $\varepsilon \geqslant 0$, $\ell, t \geqslant 1$, and $1 \leqslant h \leqslant \alpha\ell + \varepsilon$ be fixed integers. If*

$$r \leqslant \beta \cdot q^{\frac{f(t)}{\alpha-1}} , \tag{4.43}$$

where $\beta := \left(\frac{(\alpha-1)!}{2e\gamma\alpha}\right)^{\frac{1}{\alpha-1}}, \gamma \approx 3.48$ and $f(t) := (\alpha\ell+\varepsilon-h)\varepsilon t^2 + (\alpha\ell+2\varepsilon-h)t + 1$, then $(\varepsilon,\ell) - \mathcal{N}_{h,r,\alpha\ell+\varepsilon}$ has a (q,t)-linear solution.

Namely, for an $(\varepsilon, \ell) - \mathcal{N}_{h,r,\alpha\ell+\varepsilon}$ *that has a* (q,t)*-linear solution, the maximum number of middle nodes satisfies*

$$r_{\max} \geqslant \beta \cdot q^{\frac{f(t)}{\alpha-1}} \ .$$

Proof. Let $p = \Pr(\mathcal{E}_{i_1,\ldots,i_\alpha})$ and denote by d the number of other events $\mathcal{E}_{i'_1,\ldots,i'_\alpha}$ that are dependent on $\mathcal{E}_{i_1,\ldots,i_\alpha}$. We have shown that $p \leqslant 2\gamma \cdot q^{(h-\alpha\ell-\varepsilon)\varepsilon t^2+(h-\alpha\ell-2\varepsilon)t-1}$ in Lemma 4.3 and $d \leqslant \alpha\binom{r-1}{\alpha-1}$ in Lemma 4.4. By the Lovász Local Lemma, it suffices to show that $epd \leqslant 1$. Noting that $d \leqslant \alpha\binom{r-1}{\alpha-1} \leqslant \alpha \cdot \frac{(r-1)^{\alpha-1}}{(\alpha-1)!}$, we shall require

$$e \cdot 2\gamma q^{(h-\alpha\ell-\varepsilon)\varepsilon t^2+(h-\alpha\ell-2\varepsilon)t-1} \cdot \alpha\frac{(r-1)^{\alpha-1}}{(\alpha-1)!} \leqslant 1 \ .$$

Namely, if $r \leqslant \beta \cdot q^{\frac{(\alpha\ell+\varepsilon-h)\varepsilon}{\alpha-1}t^2+\frac{\alpha\ell+2\varepsilon-h}{\alpha-1}t+\frac{1}{\alpha-1}} + 1$, then $(\varepsilon,\ell) - \mathcal{N}_{h,r,\alpha\ell+\varepsilon}$ has a (q,t)-linear solution. We omit the plus one for simplicity. □

Remark 4.4. *For any* $\alpha \geqslant 7$*,* (4.43) *can be simplified to*

$$r \leqslant q^{\frac{f(t)}{\alpha-1}} \ ,$$

since the prefactor $\beta > 1$ *for all* $\alpha \geqslant 7$*.*

Remark 4.5. *For* $t \geqslant 3$*,* $\alpha \geqslant 5$ *or* $q \geqslant 4$*, it can be seen from numerical analysis that* $\beta \cdot q^{\frac{\alpha\ell+2\varepsilon-h}{\alpha-1}t+\frac{1}{\alpha-1}} \geqslant 1$*. Thus,* (4.43) *can be simplified to a looser upper bound*

$$r \leqslant q^{\frac{(\alpha\ell+\varepsilon-h)\varepsilon}{\alpha-1}t^2} \ .$$

However, omitting the term $\beta \cdot q^{\frac{\alpha\ell+2\varepsilon-h}{\alpha-1}t+\frac{1}{\alpha-1}}$ *will cause a loss in estimating the maximum achievable number of middle nodes. Nevertheless, the loss is negligible when* $t \to \infty$*.*

A Lower Bound by α-Covering Grassmannian Codes

Definition 4.5 (Covering Grassmannian Codes [EZ19])**.** *An* α*-*$(n,k,\delta)_q^c$ covering Grassmannian code $\mathcal{C}$ *is a subset of* $\mathcal{G}_q(n,k)$ *such that each subset with* α *codewords of* $\mathcal{C}$ *spans a subspace whose dimension is at least* $\delta + k$ *in* $\mathbb{F}_q^n$*.*

The following theorem from [EZ19] shows the connection between covering Grassmannian codes and linear network coding solutions.

Theorem 4.11 ([EZ19, Thm. 4])**.** *The* $(\varepsilon,\ell) - \mathcal{N}_{h,r,\alpha\ell+\varepsilon}$ *network is solvable with a* (q,t)*-linear solution if and only if there exists an* α*-*$(ht, \ell t, ht-\ell t-\varepsilon t)_q^c$ *code with* r *codewords.*

Let $\mathcal{B}_q(n,k,\delta;\alpha)$ denote the maximum possible size of an α-$(n,k,\delta)_q^c$ covering Grassmannian code. Let $\boldsymbol{A}$ be a $k \times (n-k)$ matrix, and let $\boldsymbol{I}_k$ be a $k \times k$ identity matrix. The matrix $[\boldsymbol{I}_k\ \boldsymbol{A}]$ can be viewed as a generator matrix of a k-dimensional subspace of $\mathbb{F}_q^n$, and it is called the *lifting* of $\boldsymbol{A}$. When all the codewords of an MRD code $\mathcal{C}$ are lifted to k-dimensional subspaces, the result is called *lifted MRD code*, denoted by $\mathcal{C}^{\text{lifted}}$.

Theorem 4.12. *Let n, k, δ and α be positive integers such that $1 \leqslant \delta \leqslant k$, $\delta + k \leqslant n$ and $\alpha \geqslant 2$. Then*

$$\mathcal{B}_q(n, k, \delta; \alpha) \geqslant (\alpha - 1) q^{\max\{k, n-k\}(\min\{k, n-k\} - \delta + 1)} .$$

Proof. Let $m = n - k$ and $K = \max\{m, n-m\}(\min\{m, n-m\} - \delta + 1)$. Since $\delta \leqslant \min\{m, n-m\}$, an $[m \times (n-m), K, \delta]_q$ MRD code $\mathcal{C}$ exists. Let $\mathcal{C}^{\text{lifted}}$ be the lifted code of $\mathcal{C}$. Then $\mathcal{C}^{\text{lifted}}$ is a subspace code of $\mathbb{F}_q^n$, which contains q^K m-dimensional subspaces as codewords and its minimum subspace distance is 2δ [SKK08].

Hence, for any two different codewords $C_1, C_2 \in \mathcal{C}^{\text{lifted}}$ we have

$$\dim(C_1 \cap C_2) \leqslant m - \delta .$$

Now, let $\mathcal{D} := \left\{ C^\perp \mid C \in \mathcal{C}^{\text{lifted}} \right\}$. Take $\alpha - 1$ copies of $\mathcal{D}$ and denote their multiset union by $\mathcal{D}^{\alpha-1}$. We show that $\mathcal{D}^{\alpha-1}$ is an α-$(n, k, \delta)_q^c$ covering Grassmannian code. Each subspace $D \in \mathcal{D}^{\alpha-1}$ has dimension $n - m$, since it is the dual of a codeword in $\mathcal{C}^{\text{lifted}}$. For any α subspaces $D_1, D_2, \ldots, D_\alpha \in \mathcal{D}^{\alpha-1}$, there exist $1 \leqslant i < j \leqslant \alpha$ such that $D_i \neq D_j$. Let $C_i = D_i^\perp$ and $C_j = D_j^\perp$. By definition, C_i and C_j are two distinct codewords of $\mathcal{C}^{\text{lifted}}$. We then have

$$\begin{aligned} \dim\left(\sum_{\ell=1}^{\alpha} D_\ell \right) &\geqslant \dim(D_i + D_j) = n - \dim\left(D_i^\perp \cap D_j^\perp\right) \\ &= n - \dim(C_i \cap C_j) \geqslant n - m + \delta = k + \delta . \end{aligned}$$

So far we have shown that $\mathcal{D}^{\alpha-1}$ is an α-$(n, k, \delta)_q^c$ covering Grassmannian code. Then the statement follows by

$$\mathcal{B}_q(n, k, \delta; \alpha) \geqslant |\mathcal{D}^{\alpha-1}| = (\alpha - 1)|\mathcal{D}| = (\alpha - 1)|\mathcal{C}^{\text{lifted}}| = (\alpha - 1) q^{\max\{k, n-k\}(\min\{k, n-k\} - \delta + 1)} .$$

□

The following corollary rephrases Theorem 4.11 using the result in Theorem 4.12.

Corollary 4.4. *Let $\alpha \geqslant 2$, $h, \ell, t \geqslant 1$, $\varepsilon \geqslant 0$, $h \leqslant 2\ell + \varepsilon$. For an $(\varepsilon, \ell) - \mathcal{N}_{h,r,\alpha\ell+\varepsilon}$ which has a (q, t)-linear solution, the maximum number of middle nodes is*

$$r_{\max} \geqslant (\alpha - 1) q^{g(t)} ,$$

where

$$\begin{aligned} g(t) &:= \max\{\ell t, (h-\ell)t\} \cdot (\min\{\ell t, (h-\ell)t\} - (h - \ell - \varepsilon)t + 1) \\ &= \begin{cases} \ell\varepsilon t^2 + \ell t & h \leqslant 2\ell \\ (h-\ell)(2\ell + \varepsilon - h)t^2 + (h-\ell)t & \text{otherwise} \end{cases} . \end{aligned}$$

4.5.3 Bounds on the Gap on Field Size

In the last section, we presented bounds on $r_{\max}$. The main results in this section are the upper and lower bounds on $\text{gap}_2(\mathcal{N})$ in Theorem 4.13 and Theorem 4.14, respectively. To discuss $\text{gap}_2(\mathcal{N})$, we first need the following conditions on the smallest field size $q_s(\mathcal{N})$ or $q_v(\mathcal{N})$, under which a network $\mathcal{N}$ is solvable.

Lemma 4.5. *Let $\alpha \geqslant 2$, $r, h, \ell, t \geqslant 1$, $\varepsilon \geqslant 0$. If $(\varepsilon, \ell) - \mathcal{N}_{h,r,\alpha\ell+\varepsilon}$ has a (q,t)-linear solution then*

$$q^t \geqslant \begin{cases} \left(\frac{r+\theta-\alpha}{\gamma\cdot\theta}\right)^{\frac{1}{\ell(\varepsilon t+1)}} & h \geqslant 2\ell+\varepsilon \\ \left(\frac{r}{\gamma(\alpha-1)}\right)^{\frac{1}{\ell(\varepsilon t+1)}} & \text{otherwise} \end{cases},$$

where $\theta := \alpha - \left\lfloor \frac{h-\varepsilon}{\ell} \right\rfloor + 1$ and $\gamma \approx 3.48$.

Proof. The first case follows from Corollary 4.2 that for $h \geqslant 2\ell+\varepsilon$, $q^t \geqslant \left(\frac{r+\theta-\alpha}{\gamma\cdot\theta}\right)^{\frac{1}{\ell(\varepsilon t+1)}}$. The second case is derived from an upper bound on r in [EZ19] (recalled in Corollary 4.7) in a similar manner. □

Lemma 4.6. *Let $\alpha \geqslant 2$, $r, h, \ell, t \geqslant 1$, $\varepsilon \geqslant 0$. There exists a (q,t)-linear solution to $(\varepsilon, \ell) - \mathcal{N}_{h,r,\alpha\ell+\varepsilon}$ when*

$$q^t \geqslant \begin{cases} \left(\frac{r}{\beta}\right)^{\frac{(\alpha-1)t}{f(t)}} & h \geqslant 2\ell+\varepsilon \\ \left(\frac{r}{\alpha-1}\right)^{\frac{t}{g(t)}} & \text{otherwise} \end{cases},$$

where β and $f(t)$ are defined as in Theorem 4.10, and $g(t)$ is defined as in Corollary 4.4.

Proof. The proof is similar to that in Lemma 4.5 and the cases follow from Theorem 4.10 and Corollary 4.4, respectively. □

Lemmas 4.5 and 4.6 can be seen as the necessary and the sufficient conditions respectively on the pair (q,t) such that a (q,t)-linear solution exists.

In the following, we use the lemmas above to derive bounds on the $\mathrm{gap}_2(\mathcal{N})$ for a given network $\mathcal{N}$. The bounds are determined only by the network parameters.

Theorem 4.13. *Let $\alpha \geqslant 2$, $r, h, \ell \geqslant 1$, $\varepsilon \geqslant 0$. Then for the $(\varepsilon, \ell) - \mathcal{N}_{h,r,\alpha\ell+\varepsilon}$ network,*

$$\mathrm{gap}_2(\mathcal{N}) \leqslant \begin{cases} \frac{\alpha-1}{f(1)} \log_2\left(\frac{r}{\beta}\right) - A & h \geqslant 2\ell+\varepsilon \\ \frac{1}{g(1)} \log_2\left(\frac{r}{\alpha-1}\right) - B & \text{otherwise} \end{cases},$$

where $\theta = \alpha - \left\lfloor \frac{h-\varepsilon}{\ell} \right\rfloor + 1$, β and $f(t)$ are defined as in Theorem 4.10, $g(t)$ is defined as in Corollary 4.4, and

$$A := \min\left\{ \log_2(q^t) \mid q^t \geqslant \left(\frac{r+\theta-\alpha}{\gamma\theta}\right)^{\frac{1}{\ell(\varepsilon t+1)}} \right\},$$
$$B := \min\left\{ \log_2(q^t) \mid q^t \geqslant \left(\frac{r}{\gamma(\alpha-1)}\right)^{\frac{1}{\ell(\varepsilon t+1)}} \right\}.$$

Furthermore, for $t_A := \min\left\{ t \mid 2^t \geqslant \left(\frac{r+\theta-\alpha}{\gamma\theta}\right)^{\frac{1}{\ell(\varepsilon t+1)}} \right\} > 2$, we have

$$A \geqslant \min\left\{ t_A, \frac{1}{\ell(\varepsilon(t_A-2)+1)} \log_2\left(\frac{r+\theta-\alpha}{\gamma\theta}\right) \right\} \geqslant t_A - 1,$$

and for $t_B := \min\left\{t \mid 2^t \geqslant \left(\frac{r}{\gamma(\alpha-1)}\right)^{\frac{1}{\ell(\varepsilon t+1)}}\right\} > 2$*, we have*

$$B \geqslant \min\left\{t_B, \frac{1}{\ell(\varepsilon(t_B-2)+1)} \log_2\left(\tfrac{r+\theta-\alpha}{\gamma\theta}\right)\right\} \geqslant t_B - 1 \ .$$

Proof. We only prove the bound for the case $h \geqslant 2\ell + \varepsilon$. The other case follows analogously. Lemma 4.6 implies that

$$q_s(\mathcal{N}) \leqslant \left(\frac{r}{\beta}\right)^{\frac{\alpha-1}{f(1)}} \ .$$

By the definition of $q_v(\mathcal{N})$ and Lemma 4.5, $q^t = q_v(\mathcal{N})$ must fulfill

$$q^t \geqslant \left(\tfrac{r+\theta-\alpha}{\gamma\theta}\right)^{\frac{1}{\ell(\varepsilon t+1)}} \ . \tag{4.44}$$

Hence, we get a lower bound on $q_v(\mathcal{N})$ by determining the smallest q^t that fulfills (4.44), i.e., the constraint in A. Note that the left-hand side of the inequality is a strictly monotonically increasing function in t (for a fixed prime power q), and the right side is monotonically decreasing in t, which imply that A and t_A are well-defined.

For the lower bound on A for $t_A > 2$, consider the case that there is a prime power $q > 2$ and a positive integer t with $2^{t_A} \geqslant q^t \geqslant \left(\frac{r+\theta-\alpha}{\gamma\theta}\right)^{\frac{1}{\ell(\varepsilon t+1)}}$. Then we have $t \leqslant t_A - 2$ since $q \geqslant 3$ and $t_A \geqslant 3$. Hence,

$$q^t \geqslant \left(\tfrac{r+\theta-\alpha}{\gamma\theta}\right)^{\frac{1}{\ell(\varepsilon(t_A-2)+1)}} \geqslant \left(\tfrac{r+\theta-\alpha}{\gamma\theta}\right)^{\frac{1}{\ell(\varepsilon(t_A-1)+1)}} \geqslant 2^{t_A-1} \ ,$$

which proves the claim. □

Corollary 4.5. *Let* $\alpha \geqslant 2$*,* $r, h, \ell \geqslant 1$*,* $\varepsilon \geqslant 1$*. Then for the* $(\varepsilon, \ell) - \mathcal{N}_{h,r,\alpha\ell+\varepsilon}$ *network,*

$$\operatorname{gap}_2(\mathcal{N}) \leqslant \begin{cases} \frac{\alpha-1}{f(1)} \log_2\left(\frac{r}{\beta}\right) - \max\left\{\sqrt{\frac{1}{\ell\varepsilon}\log_2\left(\frac{r+\theta-\alpha}{\gamma\theta}\right) + \frac{1}{4\varepsilon^2}} - \frac{2\varepsilon+1}{2\varepsilon}, 1\right\} & h \geqslant 2\ell+\varepsilon \\ \frac{1}{g(1)} \log_2\left(\frac{r}{\alpha-1}\right) - \max\left\{\sqrt{\frac{1}{\ell\varepsilon}\log_2\left(\frac{r}{\gamma(\alpha-1)}\right) + \frac{1}{4\varepsilon^2}} - \frac{2\varepsilon+1}{2\varepsilon}, 1\right\} & \text{otherwise} \end{cases} .$$

In particular, if all parameters are constants except for $r \to \infty$*, then* $\operatorname{gap}_2(\mathcal{N}) \in O(\log r)$*.*

Proof. We only prove the bound for the case $h \geqslant 2\ell + \varepsilon$. The other case follows analogously. We determine t_A as defined in Theorem 4.13. Note that 2^t is strictly monotonically increasing in t and $\left(\frac{r+\theta-\alpha}{\gamma\theta}\right)^{\frac{1}{\ell(\varepsilon t+1)}}$ is strictly monotonically decreasing. Hence, we have $t_A = \lceil t' \rceil$, where t' is the unique (positive) solution of

$$2^{t'} = \left(\tfrac{r+\theta-\alpha}{\gamma\theta}\right)^{\frac{1}{\ell(\varepsilon t'+1)}} \ .$$

By rewriting this equation into a quadratic equation in t', we obtain the following positive solution for $\varepsilon > 0$:

$$t' = \sqrt{\frac{1}{\ell\varepsilon}\log_2\left(\frac{r+\theta-\alpha}{\gamma\theta}\right) + \frac{1}{4\varepsilon^2}} - \frac{1}{2\varepsilon}.$$

Using the bound $A \geqslant t_A - 1$ for $t_A > 2$ (Theorem 4.13) and the trivial bound $A \geqslant 1$ otherwise, the claim follows. The asymptotic statement is an immediate consequence. □

Theorem 4.14. *Let $\alpha \geqslant 2$, $r, h, \ell \geqslant 1$, $\varepsilon \geqslant 0$. Then for the $(\varepsilon, \ell) - \mathcal{N}_{h,r,\alpha\ell+\varepsilon}$ network,*

$$\operatorname{gap}_2(\mathcal{N}) \geqslant \begin{cases} \frac{1}{\ell(\varepsilon+1)} \log_2\left(\frac{r+\theta-\alpha}{\gamma\theta}\right) - t_\Delta & h \geqslant 2\ell + \varepsilon \\ \frac{1}{\ell(\varepsilon+1)} \log_2\left(\frac{r}{\gamma(\alpha-1)}\right) - t_\star & \textit{otherwise} \end{cases},$$

where t_Δ is the smallest positive integer such that $2^{\frac{f(t_\Delta)}{\alpha-1}} \geqslant \frac{r}{\beta}$ and $t_\star$ is the smallest positive integer such that $2^{g(t_\star)} \geqslant \frac{r}{\alpha-1}$. Here, β and $f(t)$ are defined as in Theorem 4.10, and $g(t)$ is defined as in Corollary 4.4.

Proof. Let us only consider the first case $h \geqslant 2\ell + \varepsilon$. The other case can be proved in the same manner. According to Lemma 4.5, we have the lower bound on the smallest field size of a scalar solution,

$$q_s(\mathcal{N}) \geqslant \left(\frac{r + \theta - \alpha}{\gamma \cdot \theta}\right)^{\frac{1}{\ell(\varepsilon+1)}}.$$

For vector solutions, according to Lemma 4.6, we want to find (q, t) such that $q^{\frac{f(t)}{\alpha-1}} \geqslant \frac{r}{\beta}$. Since t_Δ is the smallest positive integer t such that $2^{\frac{f(t_\Delta)}{\alpha-1}} \geqslant \frac{r}{\beta}$, it is guaranteed that a $(2, t_\Delta)$-linear solution exists. Therefore, $q_v(\mathcal{N})$ (the smallest value of q^t) should be at most $q_v(\mathcal{N}) \leqslant 2^{t_\Delta}$. The lower bound then follows directly from the definition of $\operatorname{gap}_2(\mathcal{N})$. □

By carefully bounding $t_\star$ and t_Δ, the following result is obtained.

Corollary 4.6. *Let $\alpha \geqslant 2$, $r, h, \ell, \varepsilon \geqslant 1$. Then, for the $(\varepsilon, \ell) - \mathcal{N}_{h,r,\alpha\ell+\varepsilon}$ network,*

$$\operatorname{gap}_2(\mathcal{N}) \geqslant \begin{cases} \frac{\log_2\left(\frac{r+\theta-\alpha}{\gamma\theta}\right)}{\ell(\varepsilon+1)} - \sqrt{\frac{(\alpha-1)\log_2(\frac{r}{\beta})}{(\alpha\ell+\varepsilon-h)\varepsilon}} & h \geqslant 2\ell + \varepsilon, \\ \frac{\log_2\left(\frac{r}{\alpha-1}\right)-2}{\ell(\varepsilon+1)} - \sqrt{\frac{\log_2(\frac{r}{\alpha-1})}{\ell\varepsilon}} & \textit{otherwise.} \end{cases}$$

In particular, if all parameters are constants except for $r \to \infty$, then $\operatorname{gap}_2(\mathcal{N}) \in \Omega(\log r)$.

Proof. When $h \geqslant 2\ell + \varepsilon$, noting that $\alpha\ell + 2\varepsilon - h > 0$, we may choose

$$t = \left(\frac{(\alpha-1)\log_2(\frac{r}{\beta})}{(\alpha\ell+\varepsilon-h)\varepsilon}\right)^{1/2},$$

such that $2^{f(t)} \geqslant 2^{(\alpha\ell+\varepsilon-h)\varepsilon t^2} = (\frac{r}{\beta})^{\alpha-1}$. Then we have that

$$\begin{aligned} \operatorname{gap}_2(\mathcal{N}) &\geqslant \frac{\log_2\left(\frac{r+\theta-\alpha}{\gamma\theta}\right)}{\ell(\varepsilon+1)} - \left(\frac{(\alpha-1)\log_2(\frac{r}{\beta})}{(\alpha\ell+\varepsilon-h)\varepsilon}\right)^{1/2} \\ &\geqslant \frac{\log_2(r+\theta-\alpha) - \log_2\theta - 2}{\ell(\varepsilon+1)} - \left(\frac{\log_2 r - \log_2\beta}{(\ell - \frac{h-\ell-\varepsilon}{\alpha-1})\varepsilon}\right)^{1/2}. \end{aligned}$$

Recall that β and θ are determined by α, h, ε, and ℓ. Thus, if α, h, ε, and ℓ are fixed, $\mathrm{gap}_2(\mathcal{N}) \in \Omega(\log r)$.

When $h < 2\ell + \varepsilon$, we may choose

$$t = \left(\frac{\log_2(\frac{r}{\alpha-1})}{\ell\varepsilon}\right)^{1/2},$$

such that $2^{g(t)} \geqslant 2^{\ell\varepsilon t^2} = \frac{r}{\alpha-1}$. It follows that

$$\begin{aligned}\mathrm{gap}_2(\mathcal{N}) &\geqslant \frac{\log_2\left(\frac{r}{\gamma(\alpha-1)}\right)}{\ell(\varepsilon+1)} - \left(\frac{\log_2(\frac{r}{\alpha-1})}{\ell\varepsilon}\right)^{1/2} \\ &\geqslant \frac{\log_2\left(\frac{r}{\alpha-1}\right) - 2}{\ell(\varepsilon+1)} - \left(\frac{\log_2(\frac{r}{\alpha-1})}{\ell\varepsilon}\right)^{1/2}.\end{aligned}$$

This shows that $\mathrm{gap}_2(\mathcal{N}) \in \Omega(\log r)$. □

Corollary 4.5 and Corollary 4.6 show that for fixed network parameters, the gap size grows as

$$\mathrm{gap}_2(\mathcal{N}) = \Theta(\log r) \quad (r \to \infty) .$$

Example 4.2. *We illustrate the proof of Theorem 4.13 and Theorem 4.14 by two network examples with $r = 8 \times 10^5$ in Figure 4.6a and $r = 8 \times 10^6$ in Figure 4.6b. Note that the curves in the figures are not bounds on the gap size. They are the necessary (blue curve) and the sufficient (green curve) condition on q^t such that a (q,t)-linear solution exists. Namely, there is no (q,t)-linear solution in the region below the blue curve and there must be a (q,t)-linear solution in the region above the green curve. Thus, the* minimum *gap of the network $(2,1)-\mathcal{N}_{12,r,20}$ is determined by the difference between the necessary condition with $t = 1$ and the minimum 2^t that is in the region above the sufficient condition. Similarly, the* maximum *gap of the network is determined by the difference between the sufficient condition with $t = 1$ and the minimum 2^t that is in the region above the necessary condition.*

By comparing the two plots it can be seen that the gap increases as the number of middle node in the network increases.

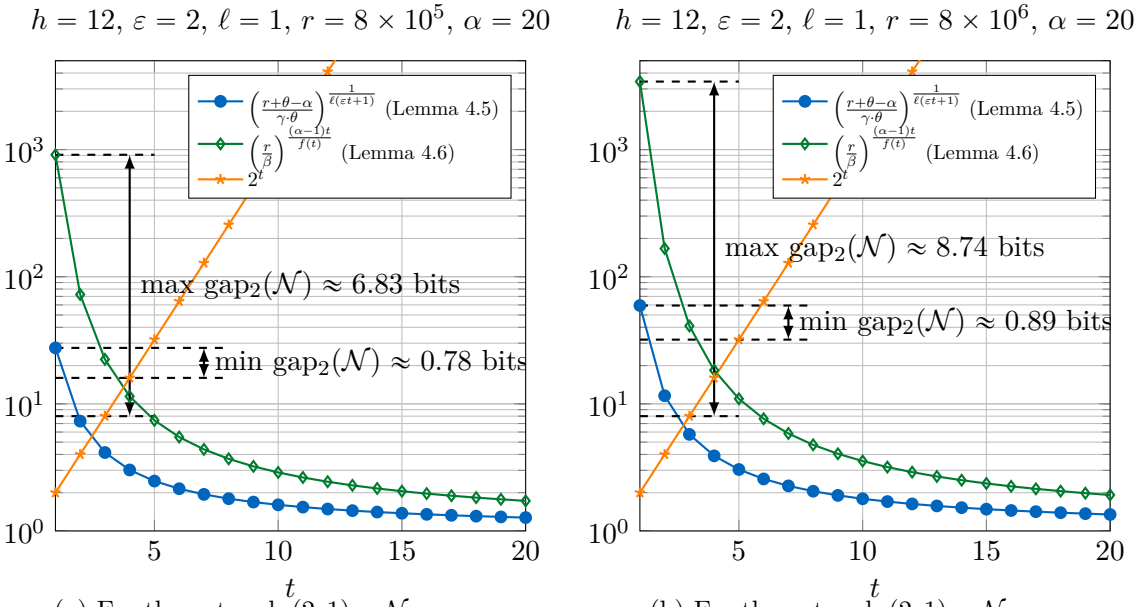
4.5.4 Comparisons of Bounds on $r_{\max}$

In the following we compare our upper and lower bound on $r_{\max}$ with previously known bounds.

Other Upper Bound on $r_{\max}$

We recall the result from [EZ19, Corollary 3] and compare it with our upper bound in Corollary 4.3.

Theorem 4.15 ([EZ19, Corollary 3])**.** *If n, k, δ, and α, are positive integers such that $1 < k < n$, $1 \leqslant \delta \leqslant n-k$ and $2 \leqslant \alpha \leqslant \begin{bmatrix} k+\delta-1 \\ k \end{bmatrix}_q + 1$, then for an $\alpha - (n,k,\delta)_q^c$ covering*



(a) For the network $(2,1) - \mathcal{N}_{12,8e5,20}$. (b) For the network $(2,1) - \mathcal{N}_{12,8e6,20}$.

Figure 4.6: An illustration of proofs of Theorem 4.13 and Theorem 4.14.

Grassmannian code $\mathcal{C}$, we have

$$|\mathcal{C}| \leqslant \left\lfloor (\alpha - 1) \frac{\begin{bmatrix} n \\ \delta+k-1 \end{bmatrix}_q}{\begin{bmatrix} n-k \\ \delta-1 \end{bmatrix}_q} \right\rfloor .$$

By combining Theorem 4.15 and Theorem 4.11, the following corollary can be derived.

Corollary 4.7. *If the $(\varepsilon, \ell) - \mathcal{N}_{h,r,\alpha\ell+\varepsilon}$ network has a (q,t)-linear solution, then*

$$\begin{aligned} r \leqslant r_{\max} &\leqslant \left\lfloor (\alpha - 1) \frac{\begin{bmatrix} ht \\ ht-\varepsilon t-1 \end{bmatrix}_q}{\begin{bmatrix} ht-\ell t \\ ht-\ell t-\varepsilon t-1 \end{bmatrix}_q} \right\rfloor \\ &< (\alpha - 1) \frac{\gamma q^{(\varepsilon t+1)(ht-\varepsilon t-1)}}{q^{(\varepsilon t+1)(ht-\ell t-\varepsilon t-1)}} \\ &= \gamma(\alpha - 1) q^{\ell t(\varepsilon t+1)} , \end{aligned}$$

with $1 < \ell t < ht$, $0 \leqslant \varepsilon \leqslant h - \ell - \frac{1}{t}$, $2 \leqslant \alpha \leqslant \begin{bmatrix} ht-\varepsilon t-1 \\ \ell t \end{bmatrix}_q + 1$.

Comparison Between the Upper Bounds

We first show that for some parameters, the upper bound in Corollary 4.2 can be tighter than that in Corollary 4.7. The upper bounds in Corollary 4.2 and Corollary 4.7 can be respectively written as

$$U_A := \begin{bmatrix} (\varepsilon + \ell)t \\ \varepsilon t \end{bmatrix}_q \left(\theta \cdot \frac{q^{\ell t+1} - 1}{q - 1} - 1 \right) + \alpha - \theta ,$$

where $\theta = \left(\alpha - \left\lfloor \frac{h-\varepsilon}{\ell} \right\rfloor + 1\right)$, and

$$U_B := (\alpha-1)\frac{\begin{bmatrix} ht \\ ht-\varepsilon t-1 \end{bmatrix}_q}{\begin{bmatrix} ht-\ell t \\ ht-\ell t-\varepsilon t-1 \end{bmatrix}_q} = (\alpha-1)q^{\ell t(\varepsilon t+1)} \prod_{i=0}^{\varepsilon t} \frac{q^{ht-i}-1}{q^{ht-i}-q^{\ell t}} \ .$$

Lemma 4.7. *Let $h \geqslant 2\ell+\varepsilon$ and $2 \leqslant \alpha \leqslant \begin{bmatrix} ht-\varepsilon t-1 \\ \ell t \end{bmatrix}_q + 1$. Assume $\begin{bmatrix} (\varepsilon+\ell)t \\ \varepsilon t \end{bmatrix}_q \leqslant \alpha$, then*

$$\log_q U_A - \log_q U_B < \log_q \frac{2\theta\alpha}{\alpha-1} - \ell\varepsilon t^2 \ .$$

Particularly, if $\frac{2\theta\alpha}{\alpha-1} \leqslant q^{\ell\varepsilon t^2}$, then $U_A < U_B$ (the upper bound in Corollary 4.2 is tighter than that in Corollary 4.7).

Proof. Under the assumption $\begin{bmatrix} (\varepsilon+\ell)t \\ \varepsilon t \end{bmatrix}_q \leqslant \alpha$, we have

$$\begin{aligned}
\log_q U_A &\leqslant \log_q\left(\alpha\left(\theta\cdot\frac{q^{\ell t+1}-1}{q-1}-1\right)+\alpha-\theta\right) \\
&= \log_q\left(\alpha\theta\cdot\frac{q^{\ell t+1}-1}{q-1}-\alpha+\alpha-\theta\right) \\
&= \log_q\theta + \log_q\left(\alpha\cdot\frac{q^{\ell t+1}-1}{q-1}-1\right) \\
&< \log_q\theta + \log_q\left(\alpha\cdot\frac{q^{\ell t+1}-1}{q-1}\right) \\
&\overset{(*)}{<} \log_q\theta + \log_q\alpha + \log_q\left(2\cdot q^{\ell t}\right) \\
&= \log_q\theta + \log_q\alpha + \ell t + \log_q 2 \ .
\end{aligned}$$

The inequality $(*)$ holds because $\frac{q^{\ell t+1}-1}{q-1} = \sum_{i=0}^{\ell t} q^i < 2\cdot q^{\ell t}$. With the bounds on the q-binomial coefficient in (4.37), we have

$$\log_q U_B > \log(\alpha-1) + \ell t(\varepsilon t+1) \ ,$$

and therefore

$$\log_q U_A - \log_q U_B < \log_q \frac{2\theta\alpha}{\alpha-1} - \ell\varepsilon t^2 \ .$$

Together with the assumption $\frac{2\theta\alpha}{\alpha-1} \leqslant q^{\ell\varepsilon t^2}$, the statement follows. □

Lemma 4.8. *Let $h \geqslant 2\ell+\varepsilon$ and $2 \leqslant \alpha \leqslant \begin{bmatrix} ht-\varepsilon t-1 \\ \ell t \end{bmatrix}_q + 1$. Assume $\begin{bmatrix} (\varepsilon+\ell)t \\ \varepsilon t \end{bmatrix}_q \geqslant \alpha$. If $h \geqslant 2\varepsilon$, then*

$$\frac{U_A}{U_B} \leqslant \frac{8\theta}{\alpha-1} \ .$$

Particularly, if $8\theta < \alpha-1$, we have $U_A < U_B$ (the upper bound in Corollary 4.2 is tighter than that in Corollary 4.7).

Proof. Since $\begin{bmatrix}(\varepsilon+\ell)t\\ \varepsilon t\end{bmatrix}_q \geqslant \alpha$, we have that

$$U_A \leqslant \theta \cdot \begin{bmatrix}(\varepsilon+\ell)t\\ \varepsilon t\end{bmatrix}_q \frac{q^{\ell t+1}-1}{q-1} \ .$$

Then,

$$\begin{aligned}
\frac{U_A}{U_B} &\leqslant \frac{\theta}{\alpha-1}\cdot\frac{q^{\ell t+1}-1}{q-1}\begin{bmatrix}(\varepsilon+\ell)t\\ \varepsilon t\end{bmatrix}_q \cdot \begin{bmatrix}(h-\ell)t\\ (h-\ell-\varepsilon)t-1\end{bmatrix}_q \begin{bmatrix}ht\\ (h-\varepsilon)t-1\end{bmatrix}_q^{-1}\\
&= \frac{\theta}{\alpha-1}\cdot\frac{q^{\ell t+1}-1}{q-1}\begin{bmatrix}(\varepsilon+\ell)t\\ \varepsilon t\end{bmatrix}_q \begin{bmatrix}(h-\ell)t\\ \varepsilon t+1\end{bmatrix}_q \begin{bmatrix}ht\\ \varepsilon t+1\end{bmatrix}_q^{-1}\\
&= \frac{\theta}{\alpha-1}\cdot\frac{q^{\ell t+1}-1}{q-1}\cdot\frac{(q^{(\varepsilon+\ell)t}-1)\cdots(q^{\ell t+1}-1)}{(q^{\varepsilon t}-1)\cdots(q-1)}\cdot\frac{(q^{(h-\ell)t}-1)\cdots(q^{(h-\ell-\varepsilon)t}-1)}{(q^{ht}-1)\cdots(q^{(h-\varepsilon)t}-1)}\\
&< \frac{\theta}{\alpha-1}\cdot\frac{q^{\ell t+1}}{q-1}\cdot\frac{q^{(\varepsilon+\ell)t}\cdots q^{\ell t+1}}{(q^{\varepsilon t}-1)\cdots(q-1)}\cdot\frac{q^{(h-\ell)t}\cdots q^{(h-\ell-\varepsilon)t}}{(q^{ht}-1)\cdots(q^{(h-\varepsilon)t}-1)}\\
&= \frac{\theta}{\alpha-1}\cdot\frac{q}{q-1}\cdot\prod_{i=1}^{\varepsilon t}\left(1-\frac{1}{q^i}\right)^{-1}\cdot\prod_{i=ht-\varepsilon t}^{ht}\left(1-\frac{1}{q^i}\right)^{-1}\\
&\leqslant \frac{\theta}{\alpha-1}\cdot\left(1+\frac{1}{q-1}\right)\prod_{i=1}^{ht}\left(1-\frac{1}{q^i}\right)^{-1} \quad \text{(assume } 2\varepsilon\leqslant h\text{)}\\
&< \frac{8\cdot\theta}{\alpha-1} \ ,
\end{aligned}$$

and the statement follows. □

Now, we compare the upper bound in Corollary 4.3 with that in Corollary 4.7 for $\alpha = 2$.

Lemma 4.9. *Denote* $U_C := \gamma q^{(h-\ell)(2\ell+\varepsilon-h)t^2+(h-\ell)t}$ *and* $U_D := \gamma q^{\ell t(\varepsilon t+1)}$. *Then,*

$$\log_q U_C - \log_q U_D = [(h-\ell)(2\ell+\varepsilon-h)-\varepsilon\ell]t^2 + (h-2\ell)t \ .$$

Particularly, if one of the following three conditions is satisfied,

- $\varepsilon t+1 < \ell t$, *and either* $h > 2\ell$ *or* $h < \ell+\varepsilon+\frac{1}{t}$;
- $\varepsilon t+1 > \ell t$, *and either* $h > \ell+\varepsilon+\frac{1}{t}$ *or* $h < 2\ell$;
- $\varepsilon t+1 = \ell t$ *and* $h \neq 2\ell$,

then,

$$\log_q U_C - \log_q U_D < 0 \ .$$

In other words, the upper bound in Corollary 4.3 is tighter than the upper bound in Corollary 4.7 for $\alpha = 2$, *if one of the conditions above holds.*

Proof. Denote $C = (h-\ell)(2\ell+\varepsilon-h)t+(h-\varepsilon)$ and $D = \ell(\varepsilon t+1)$. Then $\log_q U_C - \log_q U_D = Ct - Dt$. It suffices to show that $C < D$. Note that $C = -th^2 + 3\ell + \varepsilon th + h + \cdots$ is a quadratic function in h which is symmetric about $h = \frac{(3\ell+\varepsilon)t+1}{2t}$. We proceed in three cases, according to the position of the axis of symmetry.

(i) If $\varepsilon t + 1 < \ell t$, then $\frac{(3\ell+\varepsilon)t+1}{2t} < 2\ell$, i.e., the axis of symmetry is on the left of $h = 2\ell$. In this case, C is decreasing when $h \geqslant 2\ell$. It follows that $C < D$ for $h > 2\ell$ as $C = D$ when $h = 2\ell$. Furthermore, according to the symmetry, $C < D$ also holds for $h < \ell + \varepsilon + \frac{1}{t}$.

(ii) If $\varepsilon t + 1 > \ell t$, then $\frac{(3\ell+\varepsilon)t+1}{2t} > 2\ell$. Using the same argument, we can see that $C < D$ holds for $h < 2\ell$ and $h > \ell + \varepsilon + \frac{1}{t}$.

(iii) If $\varepsilon t + 1 = \ell t$, then $\frac{(3\ell+\varepsilon)t+1}{2t} = 2\ell$. The maximal value of $C - D$ is taken at $h = 2\ell$, which is 0. So $C < D$ for all $h \neq 2\ell$.

□

The following example shows that, in some cases, the upper bound in Corollary 4.3 matches a lower bound from [EKOO20] within a factor of $\gamma \approx 3.48$.

Example 4.3. *Let $\alpha = 2$, $\varepsilon = \ell$, and $h = 2\ell + 1$. A lower bound from [EKOO20] is*

$$q^{(\ell^2-1)t^2+(\ell+1)t} \leqslant r \ .$$

For the upper bound, Corollary 4.3 shows that

$$r \leqslant \gamma q^{(\ell^2-1)t^2+(\ell+1)t} \ ,$$

agreeing with the lower bound up to a factor of γ. In contrast, Corollary 4.7 shows that

$$r \leqslant \gamma q^{\ell^2 t^2+\ell t} \ ,$$

which differs from the lower bound by a factor of γq^{t^2-t}.

Other Lower Bounds on $r_{\max}$

Let $\mathcal{B}_q(n, k, \delta; \alpha)$ denote the maximum possible size of an α-$(n, k, \delta)_q^c$ covering Grassmannian code. The following lower bounds were proposed on $\mathcal{B}_q(n, k, \delta; \alpha)$ for $\delta \leqslant k$ in [EKOO20].

Theorem 4.16 ([EKOO20, Theorem 21])**.** *Let $1 \leqslant \delta \leqslant k$, $k + \delta \leqslant n$ and $2 \leqslant \alpha \leqslant q^k + 1$ be integers.*

(i) If $n < k + 2\delta$, then

$$\mathcal{B}_q(n, k, \delta; \alpha) \geqslant (\alpha - 1)q^{\max\{k,n-k\}(\min\{k,n-k\}-\delta+1)}.$$

(ii) If $n \geqslant k + 2\delta$, then for each t such that $\delta \leqslant t \leqslant n - k - \delta$, we have

a) If $t < k$, then

$$\mathcal{B}_q(n, k, \delta; \alpha) \geqslant (\alpha - 1)q^{k(t-\delta+1)}\mathcal{B}_q(n - t, k, \delta; \alpha).$$

b) If $t \geqslant k$, then

$$\mathcal{B}_q(n, k, \delta; \alpha) \geqslant (\alpha - 1)q^{t(k-\delta+1)}\mathcal{B}_q(n - t, k, \delta; \alpha) + \mathcal{B}_q(t + k - \delta, k, \delta; \alpha).$$

Discussion of Lower Bounds

Theorem 4.12 improves the lower bounds in Theorem 4.16 [EKOO20] by removing the conditions $\alpha \leqslant q^k + 1$ and $n < k + 2\delta$. For $n \geqslant k + 2\delta$, the numerical results show that either could be tighter, depending on the parameters. The theoretical comparison between the two lower bounds is complicated due to the recursive function.

In the following, we compare the lower bound on $r_{\max}$ in Corollary 4.4 with the upper bounds in the previous sections.

- When $h \leqslant 2\ell$, Corollary 4.4 gives

$$r_{\max} \geqslant (\alpha - 1)q^{\ell t(\varepsilon t+1)} ,$$

  which coincides with the upper bound (up to a constant factor of $\gamma \approx 3.48$) in Corollary 4.7, $r_{\max} < \gamma(\alpha - 1)q^{\ell t(\varepsilon t+1)}$.

- When $h \geqslant 2\ell$ and $\alpha = 2$, Corollary 4.4 gives

$$r_{\max} \geqslant q^{(h-\ell)(2\ell+\varepsilon-h)t^2+(h-\ell)t} ,$$

  which coincides with the upper bound (up to a constant factor of γ) in Corollary 4.3, $r_{\max} < \gamma q^{(h-\ell)(2h+\varepsilon-h)t^2+(h-\ell)t}$.

- The upper bound in Corollary 4.2 cannot be applied here as $(h - \varepsilon)/\ell \leqslant 2$.

Based on the comparisons above, we summarize the best known bounds on $r_{\max}$ for different parameter ranges, in Table 4.3.

Table 4.3: Upper bounds (UBs) and lower bounds (LBs) on $r_{\max}$ of the $(\varepsilon, \ell) - \mathcal{N}_{\alpha,r,\alpha\ell+\varepsilon}$ network with (q, t)-linear solutions. The bounds are valid for $\alpha \geqslant 2, h, \ell \geqslant 1, \varepsilon \geqslant 0$. For non-trivially solvable generalized combination networks, one should consider $\ell+\varepsilon \leqslant h \leqslant \alpha\ell+\varepsilon$. The other parameters are $\gamma \approx 3.48$, $\beta = ((\alpha - 1)!/(2e\gamma\alpha))^{1/(\alpha-1)}$, $f(t) = (\alpha\ell + \varepsilon - h)\varepsilon t^2 + (\alpha\ell + 2\varepsilon - h)t + 1$, $\theta = \alpha - \lfloor (h - \varepsilon)/\ell \rfloor + 1$, and $g(t) = \max\{\ell t, (h - \ell)t\} \cdot (\min\{\ell t, (h - \ell)t\} - (h - \ell - \varepsilon)t + 1)$.

UB	$h < 2\ell + \varepsilon$	Reference	$h \geqslant 2\ell + \varepsilon$	Reference
$\alpha > 2$	$r_{\max} < \gamma(\alpha - 1) \cdot q^{\ell t(\varepsilon t+1)}$	[EZ19] (cf. Corollary 4.7)	$r_{\max} < \gamma\theta q^{\ell t(\varepsilon t+1)} + \alpha - \theta$	Corollary 4.2
$\alpha = 2$	$r_{\max} < \gamma q^{\min\{\ell t(\varepsilon t+1),(h-\ell)(2\ell+\varepsilon-h)t^2+(h-\ell)t\}}$			[EZ19] & Corollary 4.3 (Comparison in Lemma 4.9)
LB	$h < 2\ell + \varepsilon$	Reference	$h \geqslant 2\ell + \varepsilon$	Reference
$\alpha \geqslant 2$	$r_{\max} \geqslant (\alpha - 1)q^{g(t)}$	Corollary 4.4	$r_{\max} \geqslant \beta \cdot q^{\frac{f(t)}{\alpha-1}}$	Theorem 4.10

4.6 Summary and Outlooks

The contributions in this chapter are of two-fold. We first investigated the minimum required field size to construct an MSRD code (in particular, LRS codes) from a support-constrained

generator matrix. For this purpose, we proved that the condition on the support constraints such that a support-constrained MDS/MRD code exists is also a necessary and sufficient condition for a support-constrained MSRD code, via the framework of skew polynomials. Given support constraints fulfilling this condition, an $[n,k]_{q^m}$ support-constrained LRS code exists for any prime power $q \geqslant \ell+1$ and integer $m \geqslant \max_{l\in[\ell]}\{k-1+\log_q k, n_l\}$, where ℓ is the number of blocks and n_l is the length of the l-th block of the LRS code. If the desired support constraints do not fulfill the necessary condition, the maximum sum-rank distance of a code fulfilling these constraints is given. With these results, we proposed a network coding scheme using support-constrained LRS codes for the distributed multi-source networks. The key of the scheme is to formulate all the technical requirements into an integer linear programming (ILP) problem. However, the ILP problem has $\Omega(2^h)$ constraints, where h is the number of messages to be cast. For large h, solving (even constructing the constraints) the ILP problem is computationally expensive. In future research, more specific distributed networks should be investigated so that the scheme may become more practical while considering other properties of the networks.

In the second part of this chapter, we quantified the advantage of vector network coding compared to scalar network coding in a family of multicast networks – generalized combination networks. By studying necessary and sufficient conditions for the existence of (q,t)-linear solutions to the generalized combination network (ε,ℓ)-$\mathcal{N}_{h,r,\alpha\ell+\varepsilon}$. We derived upper and lower bounds on $r_{\max}$, the maximum number of nodes in the middle layer. The lower bounds coincide (up to a constant factor of $\gamma \approx 3.48$) with the upper bounds for $h \leqslant 2\ell$ or $h \geqslant 2\ell, \alpha = 2$. With these results, we obtained upper and lower bounds on $\mathrm{gap}_2(\mathcal{N})$, which is the number of extra bits that a scalar solution has to pay compared to a vector solution of a generalized combination network $\mathcal{N}$. The asymptotic behavior of the upper and lower bound shows that $\mathrm{gap}_2(\mathcal{N}) = \Theta(\log(r))$. Namely, for large generalized combination networks, using a scalar linear solution over-pays an order of $\log(r)$ extra bits per symbol, than using a vector linear solution. A notable observation is, the novel upper and lower bounds on $\mathrm{gap}_2(\mathcal{N})$ holds for all parameters range of the generalized combination network, except $\varepsilon = 0$. This may imply that the direct links between the source and the receivers are crucial for vector network coding to have an advantage in generalized combination networks. For future research, the role of the direct links for a nonzero $\mathrm{gap}_2(\mathcal{N})$ can be further investigated.

5

Locally Recoverable Evaluation Codes based on Multivariate Polynomials

Reed-Muller (RM) codes are a class of well-studied evaluation codes of low-degree multivariate polynomials. The restrictions to the evaluation points that fall on one line in the evaluation space can be readily seen to be equivalent to the evaluation of a low-degree univariate polynomial. This property gives RM codes the desired properties of being *locally testable*[1] [RS96] and *locally decodable*[2] [Lip90; BFLS91], which have been a subject of extensive studies, e.g., [AS97; AKK$^+$05; BKS$^+$10; RS12; MZ23], in the last years. However, the obvious drawback of RM codes with the nice local properties is their rather low rate. Concretely, for an RM code based on m-variate polynomials, the rate is $\leqslant \frac{1}{m!}$. In recent years, several new families of codes were proposed to overcome the rate bottleneck of RM codes while preserving the local properties, such as, multiplicity codes [KSY14; Kop15], lifted codes [GKS13; KSY14], expander codes [HOW15; HW18] and tensor product codes [Vid10; KRR$^+$20]. Among these new code constructions, multiplicity codes and lifted codes are also evaluation codes based on multivariate polynomials.

This chapter concerns constructions of codes with local properties based on evaluation codes. We give a brief introduction of lifted codes based on multivariate polynomials in Section 5.1. In Section 5.2, we introduce a new class bivariate evaluation codes, the *quadratic-lifted Reed-Solomon* (QLRS) codes and study their dimension, distance and local recovery property. Motivated by a class codes with local properties – *batch codes*, we introduce a family of subspaces that can be used to construct such codes and propose a construction of such subspaces based on the best-known evaluation codes – Reed-Solomon (RS) codes, in Section 5.3. Aside from the explicit construction, we give an upper and a lower bound on the growth of the cardinally of such family of subspaces.

The results in Section 5.2 have been published in the proceedings of 12th International Workshop on Coding and Cryptography (WCC) [LHP$^+$22] and the results in Section 5.3 have been published in Finite Fields and Their Applications [LPVW21].

[1] Locally testable codes are codes where the membership in the code is verifiable with a constant number of queries (independent from the code length).

[2] Locally decodable codes are codes that allow a single bit of the original message to be decoded with high probability by only examining (or querying) a small number of bits of a possible corrupted codeword.

5.1 Lifted Codes on Multivariate Polynomials

Lifted codes were introduced by Guo, Kopparty and Sudan [GKS13] as evaluation codes obtained from multivariate polynomials over finite fields. It is required that the restrictions of every codeword on subsets of coordinates are codewords of a *base* code, e.g., an RS code. By construction, lifted codes are equipped with local properties. A setting of particular interest, the *lifted Reed-Solomon* codes [GKS13], is the case where the restrictions of every codeword on all the lines in the evaluation space form codewords of an RS code. A surprising advantage of lifted Reed-Solomon codes is that they achieve much larger asymptotic rate compared to RM codes as the field size grows. The work [GKS13] derived good bounds on the rate of lifted Reed-Solomon for the bi-variate case. Tight asymptotic bounds for general cases were derived in [PV19b; HPPV20].

Degree-lifted codes are a class of evaluation codes introduced in [BGK$^+$13]. The codes are composed of low-weighted-degree polynomials and codewords are evaluations of such polynomials at the rational points of an algebraic curve. A class of lifted codes based on code automorphisms was introduced in [Guo16]. The works [LW19; Wu15; HPP$^+$21] studied lifted multiplicity codes. *Hermitian-lifted codes*, proposed in [LMM$^+$21], are constructed from evaluating bivariate polynomials at points on Hermitian curves. The restriction of the codewords on any line for a codeword of an RS code. *Wedge-lifted codes* [HKLW21] utilize the trace operation to obtain binary codes with good locality properties. *Weighted η-lifted codes* introduced in [LN20] are a class of multivariate evaluation codes where the restriction on the points on a higher degree curve (instead of a line) forms a codeword of an RS code. This gives a more general definition of QLRS codes investigated in Section 5.2.

5.2 Quadratic-Lifted Reed-Solomon Codes

We first give some notations and preliminaries needed to define the code and to present the results. Our goal is to study the properties of QLRS codes, specifically, dimension, minimum Hamming distance and local recovery capability, which are respectively investigated in Section 5.2.1, Section 5.2.2 and Section 5.2.3.

For non-negative integers $a, b \in \mathbb{N}$ with *binary representations* $a = (a_1, \ldots, a_\ell)_2$, $b = (b_1, \ldots, b_\ell)_2$, we say that *$a$ lies in the 2-shadow of b*, denoted by $a \leqslant_2 b$, if $a_i \leqslant b_i$, $\forall i \in [\ell]$. The bit a_ℓ is the most significant bit in the binary representation of a, i.e., $a = a_1 + a_2 \cdot 2 + a_3 \cdot 2^2 + \cdots + a_\ell \cdot 2^{\ell-1}$.

We call a *quadratic curve* on $\mathbb{F}_q^2$ the set of zeros of a bivariate polynomial $p(x, y) = y - \phi(x)$, where $\phi \in \mathbb{F}_q[x]$ with $\deg \phi \leqslant 2$ is a *quadratic function*. We denote by Φ the set of all quadratic functions over $\mathbb{F}_q$,

$$\Phi := \left\{ \phi(x) = \alpha x^2 + \beta x + \gamma \;\middle|\; \forall \alpha, \beta, \gamma \in \mathbb{F}_q \right\} . \tag{5.1}$$

For a bivariate polynomial $f \in \mathbb{F}_q[x, y]$ and a quadratic function $\phi \in \mathbb{F}_q[x]$, we define the *restriction* of f on ϕ as

$$f|_\phi := f(x, \phi(x)) \in \mathbb{F}_q[x] .$$

Note that the definition of the *restriction* here can be also written as the restriction of the vector $(f(a))_{a \in \mathbb{F}_q} \in \mathbb{F}_q^q$ on the quadratic curve corresponding to ϕ, so that this is the same underlying technique as used for the other lifted codes mentioned in Section 5.1.

Define an operation $(\mathrm{mod}^*\ q)$ that takes a non-negative integer and maps it to an element in $[0, q-1]$ as follows:

$$a\ (\mathrm{mod}^*\ q) := \begin{cases} a, & \text{if } a \leqslant q-1 \\ q-1, & \text{if } a\ (\mathrm{mod}\ q-1) = 0,\ a \neq 0 \\ a\ (\mathrm{mod}\ q-1), & \text{else} \end{cases}$$

It can be readily seen that if $a\ (\mathrm{mod}^*\ q) = b$, then $x^a = x^b\ (\mathrm{mod}\ x^q - x)$ in $\mathbb{F}_q[x]$.

Lemma 5.1 (Lucas' Theorem [Luc78])**.** *Let p be a prime and $a, b \in \mathbb{N}$ be written in p-ary representations $a = (a_1, \ldots, a_\ell)_p$, $b = (b_1, \ldots, b_\ell)_p$. Then*

$$\binom{a}{b} = \prod_{i=1}^{\ell} \binom{a_i}{b_i} \bmod p\ .$$

If $p = 2$, then $\binom{a}{b} = 1$ if and only if $b \leqslant_2 a$.

QLRS codes generalize lifted Reed-Solomon codes by considering the restriction of bivariate polynomials to quadratic functions rather than only linear functions. This definition is coincidentally identical to the weighted η-lifted Reed-Solomon codes [LN20, Def. IV.1] with $\eta = 2$. The formal definition is the following.

Definition 5.1 (Quadratic-lifted Reed-Solomon (QLRS) codes)**.** *Let q be a power of 2, $r \in [q-1]$ and Φ be a set of quadratic functions as given in Eq.* (5.1)*. A quadratic-lifted Reed-Solomon (QLRS) code is defined as*

$$\mathcal{C}_q(\Phi, q-r) := \{f \in \mathbb{F}_q[x,y] \mid \deg(f|_\phi) < q-r,\ \forall \phi \in \Phi\}\ .$$

The integer r can be seen as the *local redundancy* since it is the redundancy of the Reed-Solomon code $\mathsf{RS}_q(q-r)$ such that $f|_\phi$ is a codeword polynomial in $\mathsf{RS}_q(q-r)$.

5.2.1 Dimension of Quadratic-Lifted Reed-Solomon Codes

The dimension of the lifted Reed-Solomon codes is analyzed via the number of *good monomials*, which are the multivariate monomials whose restriction to a line results in a codeword of an RS code. The linear span of these good monomials is shown to generate the lifted Reed-Solomon code [GKS13; HPPV20]. Similarly, we investigate the dimension of QLRS codes using the *good monomials* as the tool. We first derive a necessary and sufficient condition (Lemma 5.2) on the good monomials for any QLRS code via similar approaches as in [HKLW21]. Then we show in Theorem 5.1 that these good monomials form a basis of the QLRS code as in [LW19].

Definition 5.2 ($(\Phi, q-r)^*$-good monomial)**.** *Given a set Φ of quadratic functions, a monomial $m(x,y) = x^a y^b$ is $(\Phi, q-r)^*$-good if $\deg(m|_\phi) < q-r, \forall \phi \in \Phi$. The monomial is $(\Phi, q-r)^*$-bad otherwise.*

Lemma 5.2. *A monomial $m(x,y) = x^a y^b$ is $(\Phi, q-r)^*$-good if and only if*

$$2i + j + a\ (\mathrm{mod}^*\ q) < q-r,\ \forall i \leqslant_2 b,\ j \leqslant_2 b-i\ . \tag{5.2}$$

Proof. Let $\phi(x) = \alpha x^2 + \beta x + \gamma \in \Phi$. The restriction of the monomial $m(x,y) = x^a y^b$ on ϕ is

$$\begin{aligned} m|_\phi(x) &= x^a(\alpha x^2 + \beta x + \gamma)^b \\ &= x^a \sum_{i=0}^{b} \binom{b}{i} \alpha^i x^{2i} \cdot (\beta x + \gamma)^{b-i} \\ &= \sum_{i=0}^{b} \binom{b}{i} \alpha^i x^{2i+a} \cdot \sum_{j=0}^{b-i} \binom{b-i}{j} \beta^j x^j \cdot \gamma^{b-i-j} \\ &\overset{(*)}{=} \sum_{i \leqslant_2 b} \alpha^i x^{2i+a} \cdot \sum_{j \leqslant_2 b-i} \beta^j x^j \cdot \gamma^{b-i-j} \\ &= \sum_{i \leqslant_2 b} \sum_{j \leqslant_2 b-i} \alpha^i \cdot \beta^j \cdot \gamma^{b-i-j} \cdot x^{2i+j+a} \ , \end{aligned}$$

where the equality $(*)$ follows from the Lucas' Theorem (Lemma 5.1). If the condition (5.2) in the statement holds, then $\deg m_\phi(x) < q - r$. Therefore the sufficiency is proven.

Let $m|^*_\phi(x) := m|_\phi(x) \bmod x^q - x$. The coefficient of x^s in $m|^*_\phi(x)$ is

$$[x^s]m|^*_\phi = \sum_{\substack{i \leqslant_2 b,\ j \leqslant_2 b-i \\ 2i+j+a \pmod{q}=s}} \alpha^i \cdot \beta^j \cdot \gamma^{b-i-j} \ ,$$

which can be seen as a multivariate polynomial in $\mathbb{F}_q[\alpha, \beta, \gamma]$. Assume the condition (5.2) does not hold but $m(x,y)$ is $(\Phi, q-r)^*$-good, i.e., for any $s \geqslant q - r$, $[x^s]m|^*_\phi$ is not a zero polynomial but the evaluations at all $(\alpha, \beta, \gamma) \in \mathbb{F}_q^3$ equal to 0. However, by Theorem 2.1, since the exponents $i, j, b - i - j < q$, there exists some $(\alpha_0, \beta_0, \gamma_0) \in \mathbb{F}_q^3$ such that the evaluation of $[x^s]m|^*_\phi$ at $(\alpha_0, \beta_0, \gamma_0)$ is nonzero. By contradiction we have proven that the condition (5.2) is also a necessary condition. □

The first important result is that the dimension of the code $\mathcal{C}_q(\Phi, q-r)$ is exactly the number of $(\Phi, q-r)^*$-good monomials, which is presented in Theorem 5.1. In order to prove this, we first discuss in the following lemma a special case that will be excluded in the proof of Theorem 5.1.

Lemma 5.3. *Consider two monomials $m_1(x,y) = x^{q-1}y^b$ and $m_2(x,y) = y^b$ with $b \in [0, q-1]$ and a polynomial $P(x,y)$ containing m_1 and m_2, i.e.,*

$$P(x,y) = (\xi_1 x^{q-1} y^b + \xi_2 y^b) + P'(x,y)$$

where $\xi_1, \xi_2 \neq 0$ and $P'(x,y)$ does not contain m_1 or m_2. Then, P is $(\Phi, q-r)^*$-bad for any $r \in [q-1]$.

Proof. Consider the restriction of P on the function $\phi(x) = \gamma$ for some $\gamma \in \mathbb{F}_q$ and $\alpha = \beta = 0$,

$$P|_\phi(x) = P(x, y = \gamma) = \gamma^b(\xi_1 x^{q-1} + \xi_2) + P'(x, y = \gamma) \ .$$

First, observe that for this choice of α, β, $P|_\phi(x)$ is of degree at most $q-1$ and we are only interested in the coefficient of x^{q-1}. Further, the only monomials of $P'(x, y = \gamma)$ that contribute to this coefficient are of the form $\xi' x^{q-1} \gamma^{b'}$ with $\xi' \neq 0$. Since $P'(x,y)$ does not contain the monomials m_1, m_2 by definition, we conclude that $b' \neq b$. Now consider the

coefficient of x^{q-1} in $P|_\phi(x)$:

$$[x^{q-1}]P|_\phi = \underbrace{\gamma^b \xi_1}_{\text{from } m_1+m_2} + \underbrace{\gamma^{b'}\xi' + \dots}_{\text{from } P'(x,y)} .$$

We view this as a polynomial in $\mathbb{F}_q[\gamma]$. Since $b \neq b'$ and $\xi_1, \xi' \neq 0$, this is not a zero polynomial. Also, as $b, b' \in [0, q-1]$ this is a polynomial in $\mathbb{F}_q[\gamma]$ of degree $\leqslant q-1$. By Theorem 2.1, there exists $\gamma \in \mathbb{F}_q$ such that $[x^{q-1}]P|_\phi \neq 0$, which means $P|_\phi(x)$ is of degree $q-1$ for some γ. Therefore, P is $(\Phi, q-r)^*$-bad according to Definition 5.2 for any $q-r \leqslant q-1$ (equivalently, for any $r \in [q-1]$). □

Theorem 5.1. *Let q be a power of 2, $r \in [q-1]$ and Φ be the set of all quadratic functions. The dimension of the QLRS code $\mathcal{C}_q(\Phi, q-r)$ is the number of $(\Phi, q-r)^*$-good monomials.*

Proof. Assume a polynomial $P \in \mathbb{F}_q[x, y]$ containing $(\Phi, q-r)^*$-bad monomials is $(\Phi, q-r)^*$-good. Let $\mathcal{G}$ and $\mathcal{B}$ be subsets of indices of all $(\Phi, q-r)^*$-good and -bad monomials, respectively (assuming the monomials are ordered according to some order). We can write P as

$$P = \sum_{c \in \mathcal{G}} \xi_c x^{a_c} y^{b_c} + \sum_{c \in \mathcal{B}} \xi_c x^{a_c} y^{b_c} ,$$

with $\xi_c \in \mathbb{F}_q^*$. Restricting P on the quadratic function $\phi(x) = \alpha x^2 + \beta x + \gamma$ gives the following univariate polynomial

$$\begin{aligned} P|_\phi &= \sum_{c \in \mathcal{G} \cup \mathcal{B}} \xi_c x^{a_c} (\alpha x^2 + \beta x + \gamma)^{b_c} \\ &= \sum_{c \in \mathcal{G} \cup \mathcal{B}} \xi_c \sum_{i=0}^{b_c} \sum_{j=0}^{b_c - i} \binom{b_c}{i} \binom{b_c - i}{j} \alpha^i \cdot \beta^j \cdot \gamma^{b_c - i - j} \cdot x^{2i+j+a_c} . \end{aligned}$$

Let $P|_\phi^* := P|_\phi \bmod (x^q - x)$. Denote by $[x^s]P|_\phi^*$ the coefficient of x^s in $P|_\phi^*$. By Lemma 5.1, we have

$$[x^s]P|_\phi^* = \sum_{c \in \mathcal{G} \cup \mathcal{B}} \sum_{\substack{i \leqslant_2 b_c,\ j \leqslant_2 b_c - i \\ 2i+j+a_c \pmod{q} = s}} \xi_c \cdot \alpha^i \cdot \beta^j \cdot \gamma^{b_c - i - j} .$$

The $(\Phi, q-r)^*$-good monomials do not contribute to the coefficients for $s \geqslant q-r$ (see Definition 5.2), therefore,

$$[x^s]P|_\phi^* = \sum_{c \in \mathcal{B}} \sum_{\substack{i \leqslant_2 b_c, j \leqslant_2 b_c - i \\ 2i+j+a_c \pmod{q} = s}} \xi_c \cdot \alpha^i \cdot \beta^j \cdot \gamma^{b_c - i - j} \quad \text{for } s \geqslant q-r . \tag{5.3}$$

We view $[x^s]P|_\phi^*$ as a trivariate polynomial in $\mathbb{F}_q[\alpha, \beta, \gamma]$. Note that P is $(\Phi, q-r)^*$-good only if

$$[x^s]P|_\phi^*\ (\alpha, \beta, \gamma) = 0 , \quad \forall \alpha, \beta, \gamma \in \mathbb{F}_q, \forall s \geqslant q-r . \tag{5.4}$$

Now consider two bad monomials $x^{a_c} y^{b_c}$ and $x^{a_d} y^{b_d}$ with $c, d \in \mathcal{B}$. Then the corresponding terms in (5.3) contributed by them can be added up only if $\alpha^{i_c} \beta^{j_c} \gamma^{b_c - i_c - j_c} = \alpha^{i_d} \beta^{j_d} \gamma^{b_d - i_d - j_d}$,

which is true if and only if

$$\iff \begin{cases} i_c = i_d \\ j_c = j_d \\ b_c - i_c - j_c = b_d - i_d - j_d \\ 2i_c + j_c + a_c \pmod{q}^* = 2i_d + j_d + a_d \pmod{q}^* \end{cases}$$
$$\implies \begin{cases} b_c = b_d \\ |a_c - a_d| = 0 \text{ or } q-1 \ . \end{cases}$$

For the case $|a_c - a_d| = q - 1$, such polynomials are bad according to Lemma 5.3. For the case $|a_c - a_d| = 0$, we can conclude that (5.3) is in its simplest form[3].

Assume $\mathcal{B}$ is non-empty. Since $\xi_c \neq 0$ for all c, (5.3) is a nonzero polynomial in $\mathbb{F}_q[\alpha, \beta, \gamma]$. By Theorem 2.1, since the exponents $i, j, b_c - i - j < q$, there exists some $\alpha_0, \beta_0, \gamma_0 \in \mathbb{F}_q$, such that $[x^s]P|^*_\phi(\alpha_0, \beta_0, \gamma_0) \neq 0$. This contradicts the assumption that P is $(\Phi, q-r)^*$-good and implies that (5.4) can be fulfilled only if $[x^s]P|^*_\phi$ is a zero polynomial, i.e., $\mathcal{B}$ is empty. Hence, a polynomial P is $(\Phi, q-r)^*$-good if and only if it only consists of good monomials. □

Counting $(\Phi, q-r)^*$-Bad Monomials

By Theorem 5.1, we can calculate the dimension by

$$\begin{aligned} k &= \text{the number of } (\Phi, q-r)^*\text{-good monomials} \\ &= q^2 - \text{the number of } (\Phi, q-r)^*\text{-bad monomials.} \end{aligned}$$

Since it is hard to directly analyze the $(\Phi, q-r)^*$-bad monomials, we first consider a slightly different notion of $(\Phi, q-r)$-*bad monomials* as given in Definition 5.3. Then, we derive upper and lower bounds on the number of $(\Phi, q-r)^*$-bad monomials in Theorem 5.2 and finally establish the results on the rate of QLRS codes in Corollary 5.1.

Definition 5.3 ($(\Phi, q-r)$-bad monomials)**.** *Let $q = 2^\ell$ and $r \in [q-1]$. A monomial $m(x,y) = x^a y^b$ (or the exponents (a,b)) is* $(\Phi, q-r)$-bad *if there exist $i \leqslant_2 b$ and $j \leqslant_2 b - i$ such that $2i + j + a \pmod{q} \geqslant q - r$. For an integer $t \geqslant 0$, we define*

$$S_t(\ell) := \left\{ (a,b) \in \mathbb{Z}_q^2 \quad \middle| \quad \begin{array}{l} \exists\ i \leqslant_2 b,\ j \leqslant_2 b - i,\ \textit{such that} \\ 2i + j + a = q - r' + tq,\ \textit{for some } r' \in [r] \end{array} \right\}. \tag{5.5}$$

For $r \in [q-1]$ and $t \geqslant 3$, the set $S_t(\ell)$ is empty as $2i + j + a \leqslant i + b + a \leqslant 2b + a \leqslant 3(q-1) < q - r + tq$. Hence, if $x^a y^b$ is $(\Phi, q-r)$-bad, then $(a,b) \in S_0(\ell) \cup S_1(\ell) \cup S_2(\ell)$.

In the following, we attempt to derive some recursive relations on $S_0(\ell)$, $S_1(\ell)$ and $S_2(\ell)$ for $r \in [q-1]$. We have two observations in Lemma 5.4 and Lemma 5.5.

Lemma 5.4. *Consider $q = 2^\ell$, $r < \frac{q}{2}$, $a = (a_1, \ldots, a_\ell)_2$ and $b = (b_1, \ldots, b_\ell)_2$. Let $a' := (a_1, \ldots, a_{\ell-1})_2$ and $b' := (b_1, \ldots, b_{\ell-1})_2$. If $(a,b) \in S_0(\ell) \cup S_1(\ell) \cup S_2(\ell)$, then $(a', b') \in S_0(\ell-1) \cup S_1(\ell-1) \cup S_2(\ell-1)$.*

Proof. The condition $(a,b) \in S_0(\ell) \cup S_1(\ell) \cup S_2(\ell)$ implies that there exist $i = (i_1, \ldots, i_\ell)_2 \leqslant_2 b$ and $j = (j_1, \ldots, j_\ell)_2 \leqslant_2 b - i$ such that $2i + j + a = q - r' \pmod{q}$ for some $r' \in [r]$.

[3] A polynomial is in its simplest form if no terms can be further combined.

Let $i' := (i_1, \ldots, i_{\ell-1})_2$ and $j' := (j_1, \ldots, j_{\ell-1})_2$. Clearly $i' \leqslant_2 b'$ and $j' \leqslant_2 b' - i'$ and $2i' + j' + a' = \frac{q}{2} - r' \pmod{\frac{q}{2}}$. □

Lemma 5.5. *For $t = 1, 2$, if $(a, b) \in S_t(\ell)$, then $(a, b) \in S_{t-1}(\ell)$.*

Proof. We first prove for $t = 1$. The condition $(a, b) \in S_1(\ell)$ implies that there exists an $i \leqslant_2 b$ and an $j \leqslant_2 b - i$ with $2i + j + a = 2q - r'$ for some $r' \in [r]$. The statement $(a, b) \in S_0(\ell)$ means that there exists $i' \leqslant_2 i$ and $j' \leqslant_2 j$ such that $2i' + j' + a \in [q - r, q - 1]$. Note that for $q - r \leqslant a \leqslant q - 1$ the statement holds with $i' = j' = 0$. Assuming $a < q - r$, we claim that there exists of a pair $(i' \leqslant_2 i, j' \leqslant_2 j)$ such that $2i' + j' + a = q - r'$. This claim would imply the required statement. Such i', j' can be found by Algorithm 5.1 where we replace some ones in the binary representations of i and j by zeros to get i' and j' so that $(2i + j) - (2i' + j') = q$. Note that the algorithm outputs the correct i', j' for $2i + j > q$ if it enters the $\delta \leqslant 0$ else-part (Line 9) at some point. Assume the contrary that this does not happen, resulting in that the algorithm output the all-zero i', j' at the end. However, this contrary implies that $\delta = q - (2i + j) > 0$ which contradicts the condition that $2i + j + a \geqslant 2q - r$ while $a < q - r$.

For $t = 2$, given i, j such that $2i + j + a = 3q - r'$, which implies that $2i + j > q$, we can find i', j' by Algorithm 5.1 such that $2i' + j' + a = 2q - r'$. This completes the proof. □

Algorithm 5.1: Deduct $q = 2^\ell$ from $2i + j$

Input: $\ell \geqslant 1, i, j$
Output: i', j' such that $i' \leqslant_2, j' \leqslant_2$

1 **Init:** $i' \leftarrow i, j' \leftarrow j, h \leftarrow \ell, \Delta \leftarrow 1$
2 **if** $h = 1$ **then**
3 | **return** i', j'
4 Let $h \leftarrow h - 1$ and $\Delta \leftarrow 2\Delta$
5 Compute $\delta \leftarrow \Delta - i'_h - j'_{h+1}$
6 **if** $\delta > 0$ **then**
7 | $i'_h \leftarrow 0,\ j'_{h+1} \leftarrow 0$
8 | **Go back to Line** 2
9 **else**
10 | Let $\begin{cases} i'_h \leftarrow 0, & \textbf{if } \Delta - i'_h = 0 \\ j'_{h+1} \leftarrow 0, & \textbf{if } \Delta - j'_{h+1} = 0 \\ i'_h \leftarrow 0, j'_{h+1} \leftarrow 0, & \textbf{if } \Delta - i'_h - j_{h+1} = 0 \end{cases}$
11 | **return** i', j'

Example 5.1 (A toy example of Lemma 5.5)**.** *Consider the parameters $q = 2^\ell = 2^4, r = 2$. It can be seen that $(a, b) = (12, 14) = ((0011)_2, (0111)_2)$ is in $S_1(4)$, since with $i = 2 = (0010)_2, j = 5 = (0101)_2$ we have $i \leqslant_2 b, j \leqslant_2 b - i$ and $2i + j + a = 2q - r' = 30 = (01111)_2$ with $r' = 2 \in [r]$. We now apply Algorithm 5.1 to find $i' \leqslant_2 i, j' \leqslant_2 j$ such that $2i' + j' + a = q - r'$ for some $r' \in [r]$:*

Init: *$i' \leftarrow (0010)_2, j' \leftarrow (0101)_2$, $\Delta = 1$ and $h \leftarrow 4$.*

Line 4: *Let $h \leftarrow 3, \Delta \leftarrow 2$.*

Line 5: *Compute $\delta \leftarrow \Delta - i'_3 - j'_4 = 2 - 1 - 1 = 0$.*

***Line 9**: Since $\delta \not> 0$ and $\Delta - i'_3 - j'_4 = 0$,*

***Line 10**: $i'_3 \leftarrow 0$, $j'_4 \leftarrow 0$ and output $i' = (0000)_2 = 0, j' = (0100)_2 = 2$.*

Note that $i' \leqslant_2 i \leqslant_2 b$, $j' \leqslant_2 j \leqslant_2 b-i$ and $2i'+j'+a = (0111)_2 = 14 = q-r'$ with $r' = 2 \in [r]$. Hence, (a, b) is also in $S_0(4)$.

It follows from Lemma 5.5 that $x^a y^b$ is $(\Phi, q-r)$-bad if and only if $(a, b) \in S_0(\ell)$. Together with the observation in Lemma 5.4, we can provide a recursive formula for computing the size of $S_t(\ell)$ for $t = 0, 1, 2$.

Lemma 5.6. *For $1 \leqslant r < \frac{q}{2}$, it holds that*

$$\begin{aligned}
|S_0(\ell)| &= 3|S_0(\ell-1)| + |S_1(\ell-1)|, \\
|S_1(\ell)| &= |S_0(\ell-1)| + |S_1(\ell-1)| + |S_2(\ell-1)|, \\
|S_2(\ell)| &= |S_2(\ell-1)|.
\end{aligned}$$

Proof. We follow the notations in Lemma 5.5. To obtain a valid $S_t(\ell-1)$, we require $r < q^{\ell-1} = \frac{q}{2}$ due to the condition in Lemma 5.4. According to Lemma 5.4 and Lemma 5.5, we know that if $(a, b) \in S_0(\ell)$, then $(a', b') \in S_0(\ell-1) \cup S_1(\ell-1) \cup S_2(\ell-1)$. The statement can be proven by counting how many ways to add the most significant bits a_ℓ and b_ℓ for a' and b' to obtain a and b. Denote them by $a = [a', a_\ell], b = [b', b_\ell]$. Recall the definition in (5.5), given $(a', b') \in S_t(\ell-1)$, there exist $i' \leqslant_2 b', j' \leqslant_2 b' - i'$ such that $2i' + j' + a' = \frac{q}{2} - r' + t\frac{q}{2}$ with some $r' \in [r]$. Construct i, j by appending one most significant bit to i', j', i.e., $i = [i', i_\ell]$ and $j = [j', j_\ell]$ with $i_\ell \leqslant_2 b_\ell$ and $j_\ell \leqslant_2 b_\ell$. To obtain $(a, b) \in S_t(\ell)$, we require $2i+j+a = q-r''+tq$ with some $r'' \in [r]$. We can write

$$2i + j + a = 2i' + j' + a' + (2i_\ell + j_\ell + a_\ell)\frac{q}{2} \ .$$

Since the difference between $2i+j+a$ and $2i'+j'+a'$ is always some multiple of $\frac{q}{2}$, we obtain $r'' = r'$. By Lemma 5.5, we have $S_2(\ell-1) \subset S_1(\ell-1) \subset S_0(\ell-1)$.

We first prove $|S_0(\ell)|$. To have $(a, b) \in S_0(\ell)$, we require $2i + j + a = q - r'$. Consider three cases,

- Given $(a', b') \in S_0(\ell-1) \backslash S_1(\ell-1)$, it means $2i' + j' + a' = \frac{q}{2} - r'$. To obtain $2i + j + a = q - r'$, we require $2i_\ell + j_\ell + a_\ell = 1$. There are three options of (a_ℓ, b_ℓ) that this can be fulfilled, i.e., $(a_\ell, b_\ell) = (1, 0), (0, 1)$ or $(1, 1)$.
- Given $(a', b') \in S_1(\ell-1) \backslash S_2(\ell-1)$, we have $2i'+j'+a' = q-r'$. The option $(a_\ell, b_\ell) = (0, 0)$ makes $(a, b) \in S_0(\ell)$. Since $S_1(\ell-1) \subset S_0(\ell-1)$, we can find $i'' \leqslant_2 i' \leqslant_2 b'$ and $j'' \leqslant_2 j' \leqslant_2 b - i$ such that $2i'' + j'' + a' = \frac{q}{2} - r'$ (by Algorithm 5.1). So all the other three options in the first case are also valid for this case.
- Given $(a', b') \in S_2(\ell-1)$, we have $2i' + j' + a' = \frac{3}{2}q - r'$. Since $S_2(\ell-1) \subset S_1(\ell-1)$, all four options of (a_ℓ, b_ℓ) allow to get $(a, b) \in S_0(\ell)$.

Then we show $|S_1(\ell)|$. We require $2i + j + a = 2q - r'$. Again, consider the three cases,

- Given $(a', b') \in S_0(\ell-1) \backslash S_1(\ell-1)$, we have $2i' + j' + a' = \frac{q}{2} - r'$. This means that $2i_\ell + j_\ell + a_\ell = 3$ is required. $(a_\ell, b_\ell) = (1, 1)$ is the only way to add the most significant bits.

- Given $(a', b') \in S_1(\ell-1) \backslash S_2(\ell-1)$, we have $2i'+j'+a' = q-r'$. We need $2i_\ell+j_\ell+a_\ell = 2$ to obtain $2i + j + a = 2q - r'$. The two options $(0, 1)$ and $(1, 1)$ allow this.

- Given $(a', b') \in S_2(\ell - 1)$, we have $2i' + j' + a' = \frac{3}{2}q - r'$. We require $2i_\ell + j_\ell + a_\ell = 1$ to obtain $2i + j + a = 2q - r'$. The three options $(a_\ell, b_\ell) = (1, 0), (0, 1)$ or $(1, 1)$ can fulfill this.

Now we show $|S_2(\ell)|$. We require $2i + j + a = 3q - r'$. Consider the three cases,

- Given $(a', b') \in S_0(\ell - 1) \backslash S_1(\ell - 1)$, we have $2i' + j' + a' = \frac{q}{2} - r'$. This means that $2i_\ell + j_\ell + a_\ell = 5$ is required. However, this cannot happen since all i_ℓ, j_ℓ, a_ℓ are in $\mathbb{F}_2$.

- Given $(a', b') \in S_1(\ell-1) \backslash S_2(\ell-1)$, we have $2i'+j'+a' = q-r'$. We need $2i_\ell+j_\ell+a_\ell = 4$. However, due to $j_\ell \leqslant_2 b_\ell - i_\ell$, this cannot happen since i_ℓ and j_ℓ cannot be one at the same time.

- Given $(a', b') \in S_2(\ell - 1)$, we have $2i' + j' + a' = \frac{3}{2}q - r'$. We require $2i_\ell + j_\ell + a_\ell = 3$. $(a_\ell, b_\ell) = (1, 1)$ is the only option.

To sum up, the statements follow from

$$
\begin{aligned}
|S_0(\ell)| &= 3(|S_0(\ell-1) \backslash S_1(\ell-1)|) + 4(|S_1(\ell-1) \backslash S_2(\ell-1)|) + 4|S_2(\ell-1)| \\
|S_1(\ell)| &= |S_0(\ell-1) \backslash S_1(\ell-1)| + 2|S_1(\ell-1) \backslash S_2(\ell-1)| + 3|S_2(\ell-1)| \\
|S_2(\ell)| &= |S_2(\ell-1)| \ .
\end{aligned}
$$

□

Lemma 5.6 yields a recurrence relation for $|S_0(\ell)|$, $|S_1(\ell)|$ and $|S_2(\ell)|$. For a given r, the initial value ℓ_0 should be chosen such that $S_i(\ell_0), i = 0, 1, 2$ is a valid set according to the definition in (5.5). Denote by $\boldsymbol{s}(\ell) = (|S_0(\ell)|, |S_1(\ell)|, |S_2(\ell)|)^\top$. We then have

$$
\boldsymbol{s}(\ell) = \boldsymbol{A}^{\ell-\ell_0} \cdot \boldsymbol{s}(\ell_0), \quad \text{where } \boldsymbol{A} = \begin{pmatrix} 3 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix} . \tag{5.6}
$$

The recursion enables us to find the asymptotic behavior of the number of $(\Phi, q - r)$-bad monomials, which is exactly $|S_0(\ell)|$. Note that the order of $|S_j(\ell)|, j = 0, 1, 2$ is controlled by λ_1^ℓ, where $\lambda_1 = 2 + \sqrt{2}$ is the largest eigenvalue of $\boldsymbol{A}$ in (5.6). Hence,

$$
|S_0(\ell)| = \Theta((2 + \sqrt{2})^\ell) \ . \tag{5.7}
$$

For different r, the exact values of $|S_0(\ell)|$ can be different, since the initial value $|S_0(\ell_0)|$ depends on r. However, the asymptotic behavior is the same for any fixed r.

We provide the exact expressions of $|S_0(\ell)|$ for $r = 1$ and $r = 3$, denoted by $|S_0^{(1)}(\ell)|$ and $|S_0^{(3)}(\ell)|$ respectively, which we use later to derive upper and lower bound on the number of

$(\Phi, q-r)^*$-bad monomials:

$$\begin{aligned}|S_0^{(1)}(\ell)| =& \frac{5\sqrt{2}+7}{2(3\sqrt{2}+4)} \cdot \lambda_1^\ell + \frac{5\sqrt{2}-7}{2(3\sqrt{2}-4)} \cdot \lambda_2^\ell \\ \approx& 0.8536 \cdot \lambda_1^\ell + 0.1464 \cdot \lambda_2^\ell \end{aligned} \tag{5.8}$$

$$\begin{aligned}|S_0^{(3)}(\ell)| =& \frac{65\sqrt{2}+92}{4(12\sqrt{2}+17)} \cdot \lambda_1^\ell + \frac{65\sqrt{2}-92}{4(12\sqrt{2}-17)} \cdot \lambda_2^\ell - \lambda_3^\ell \\ \approx& 1.3536 \cdot \lambda_1^\ell + 0.6465 \cdot \lambda_2^\ell - 1 \end{aligned} \tag{5.9}$$

where $\lambda_1 = 2+\sqrt{2}, \lambda_2 = 2-\sqrt{2}, \lambda_3 = 1$ are the three distinct eigenvalues of the matrix $\boldsymbol{A}$.

Recall from Lemma 5.2 that a monomial $m(x,y) = x^a y^b$ is $(\Phi, q-r)^*$-bad if and only if there exist $i \leqslant_2 b$ and $j \leqslant_2 b-i$ such that $2i+j+a \pmod{^*}\ q \geqslant q-r$. For $q = 2^\ell$ and $r \in [q-1]$, we define the following set

$$S^*(\ell) := \left\{ (a,b) \in \mathbb{Z}_q^2 \;\middle|\; \begin{array}{r} \exists\ i \leqslant_2 b, j \leqslant_2 b-i, \text{ s.t. } 2i+j+a = q-r'+t(q-1), \\ \text{for some } r' \in [r], t \geqslant 0 \end{array} \right\}. \tag{5.10}$$

It is clear that $(a,b) \in S^*(\ell)$ if and only if $x^a y^b$ is $(\Phi, q-r)^*$-bad.

We first relate $|S^*(\ell)|$ with $|S_0(\ell)|$ in Lemma 5.7 and Lemma 5.8.

Lemma 5.7. *Let $\ell \geqslant 2, q = 2^\ell$, $1 \leqslant r \leqslant \frac{q}{4}$, $s = \lceil \log_2(r) \rceil$ and $q' = 2^{\ell-s}$. Denote by $S_0^{(3)}(\ell-s)$ the set of (a,b) such that $x^a y^b$ is $(\Phi, q'-3)$-bad. Then*

$$|S^*(\ell)| < 4r^2 \cdot |S_0^{(3)}(\ell-s)| \ .$$

If r is a power of 2, then

$$|S^*(\ell)| \leqslant r^2 \cdot |S_0^{(3)}(\ell-s)| \ .$$

Proof. By definition, we require $\ell - s \geqslant 2$ to have a valid $S_0^{(3)}(\ell-s)$. Therefore, we require $\ell \geqslant 2$ and $r \leqslant \frac{q}{4}$. Let $x^a y^b$ be an arbitrary $(\Phi, q-r)^*$-bad monomial. By definition, this means that there exist $i \leqslant_2 b$ and $j \leqslant_2 b-i$ such that $2i+j+a = q-r'+(q-1)t$ for some $r' \in [r]$ and $t \in [0,2]$.[4] We drop $s = \lceil \log(r) \rceil$ least significant bits in a, b, i and j to obtain a', b', i' and j', and write

$$\begin{aligned} i &= i' \cdot 2^s + r_i \\ 2i &= 2i' \cdot 2^s + 2r_i \\ j &= j' \cdot 2^s + r_j \\ a &= a' \cdot 2^s + r_a \end{aligned}$$

[4] Note that $2i+j+a \leqslant 2b+a \leqslant 3(q-1) < q-r+(q-1)t$ for any $t \geqslant 3$ and $r < q$.

where the remainders $0 \leqslant r_i, r_j, r_a < 2^s$. Recall that $q' = q/2^s = 2^{\ell-s}$, it is clear that

$$\begin{aligned} 2i' + j' + a' &= \frac{2i + j + a}{2^s} - \frac{2r_i + r_j + r_a}{2^s} \\ &= \frac{q - r' + (q-1)t}{2^s} - \frac{2r_i + r_j + r_a}{2^s} \\ &= q'(t+1) - \frac{r' + t}{2^s} - \frac{2r_i + r_j + r_a}{2^s} \ . \end{aligned}$$

Since the bits in i, j cannot be both one at the same position, $2r_i + r_j \leqslant 2(2^s - 1)$ and therefore $0 \leqslant 2r_i + r_j + r_a \leqslant 3(2^s - 1)$. In addition, since $1 \leqslant r' + t \leqslant r + 2 \leqslant 2^s + 2$, we have

$$q'(t+1) - 4 < 2i' + j' + a' \leqslant q'(t+1) - \frac{1}{2^s} \ .$$

As $2i' + j' + a'$ can only be an integer, we have

$$q'(t+1) - 3 \leqslant 2i' + j' + a' \leqslant q'(t+1) - 1 \ .$$

This implies that (a', b') is $(\Phi, q' - 3)$-bad. Therefore, adding arbitrary s least significant bits to a pair $(a', b') \in S_0^{(3)}(\ell - s)$ results in a pair (a, b) that may be $(\Phi, q - r)^*$-bad. The number of $(\Phi, q - r)^*$-bad monomials is therefore bounded from above by

$$2^s \cdot 2^s \cdot |S_0^{(3)}(\ell - s)| = (2^{\lceil \log_2(r) \rceil})^2 \cdot |S_0^{(3)}(\ell - s)| < (2r)^2 \cdot |S_0^{(3)}(\ell - s)| \ .$$

If r is a power of 2, we can set $s = \log_2 r$ and obtain the tighter bound. □

Lemma 5.8. *Let $\ell \geqslant 1, q = 2^\ell$, $1 \leqslant r \leqslant \frac{q}{2}$, $s = \lfloor \log_2 r \rfloor$ and $q' = 2^{\ell - s}$. Denote by $S_0^{(1)}(\ell - s)$ the set of (a, b) such that $x^a y^b$ is $(\Phi, q' - 1)$-bad. Then*

$$|S^*(\ell)| > \frac{r^2}{4} \cdot |S_0^{(1)}(\ell - s)| \ .$$

If r is a power of 2, then

$$|S^*(\ell)| \geqslant r^2 \cdot |S_0^{(1)}(\ell - s)| \ .$$

Proof. By definition, we require $\ell - s \geqslant 1$ to have a valid $S_0^{(1)}(\ell - s)$. Therefore, we require $\ell \geqslant 1$ and $r \leqslant \frac{q}{2}$. Consider a pair $(a', b') \in S_0^{(1)}(\ell - s)$. According to the definition in (5.5), there exist $i' \leqslant_2 b, j' \leqslant_2 b - i$ such that $2i' + j' + a' = q' - 1$.

Construct integers a, b, i, j as $a = a'' + 2^s \cdot a'$, $b = b'' + 2^s \cdot b'$, $i = 2^s \cdot i'$ and $j = 2^s \cdot j'$, where $a'', b'' \in [0, 2^s - 1]$. Note that this is equivalent to appending the binary representation of a'' (resp. b'') on the left to the binary representations of a' (resp. b'), and appending s zeros on the left to the binary representations of i' and j'. It can be seen that $i \leqslant_2 b$, $j \leqslant_2 b - i$ by construction, and

$$\begin{aligned} 2i + j + a &= (2i' + j' + a')2^s + a'' \\ &= (q' - 1) \cdot 2^s + a'' \\ &= q - 2^s + a'' \ . \end{aligned} \tag{5.11}$$

Since $s = \lfloor \log_2 r \rfloor$, $q - r \leqslant (5.11) \leqslant q - 1$. Hence any choice of $a'', b'' \in [0, 2^s - 1]$ results in a pair (a, b) such that $x^a y^b$ is $(\Phi, q-r)^*$-bad. In total there are $(2^s)^2 > \left(\frac{r}{2}\right)^2$ ways of choosing a'', b''. If r is a power of 2, we can set $s = \log_2 r$ and obtain a tighter lower bound. □

In the following theorem we provide upper and lower bounds on the number $|S^*(\ell)|$ of $(\Phi, q-r)^*$-bad monomials in terms of ℓ and r.

Theorem 5.2. *Let $\ell \geqslant 2, q = 2^\ell$, $1 \leqslant r \leqslant \frac{q}{4}$, $s = \log_2 r$ and $S^*(\ell)$ be the set of $(\Phi, q-r)^*$-bad monomials as in* (5.10)*. Then*

$$\frac{0.8536 \cdot \lambda_1^{\ell - \lfloor s \rfloor} + 0.1464 \cdot \lambda_2^{\ell - \lfloor s \rfloor}}{4} < \frac{|S^*(\ell)|}{r^2} < 4(1.3536 \cdot \lambda_1^{\ell - \lceil s \rceil} + 0.6465 \cdot \lambda_2^{\ell - \lceil s \rceil} - 1) \ ,$$

where $\lambda_1 = 2 + \sqrt{2}$ and $\lambda_2 = 2 - \sqrt{2}$.

If r is a power of 2, we obtain

$$0.8536 \cdot \lambda_1^{\ell - s} + 0.1464 \cdot \lambda_2^{\ell - s} \leqslant \frac{|S^*(\ell)|}{r^2} \leqslant 1.3536 \cdot \lambda_1^{\ell - s} + 0.6465 \cdot \lambda_2^{\ell - s} - 1 \ .$$

Proof. It follows directly from the estimation of $|S_0(\ell)|$ in (5.8)–(5.9) and the bounds in Lemma 5.7 and Lemma 5.8. □

For an illustration, we plot in Fig. 5.1 the rate of the code $\mathcal{C}_q(\Phi, q-r)$ with $q = 2^5$, which is done by computer-search according to the necessary and sufficient condition in Lemma 5.2. The lower and upper bounds on the rates for $r \in [1, \frac{q}{4}]$ are calculated from the bounds on $|S^*(\ell)|$ in Theorem 5.2.

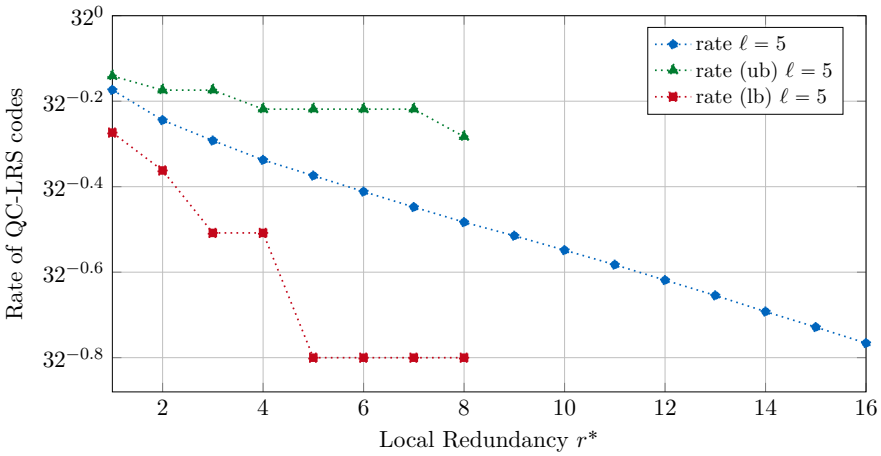


Figure 5.1: The dimension of QLRS code $\mathcal{C}_q(\Phi, q-r)$ with $q = 2^5$ along with the corresponding upper bound (ub) and lower bound (lb) for $r \in [1, \frac{q}{4}]$ calculated by $1 - |S^*(\ell)|/q^2$. The lower and upper bound on $|S^*(\ell)|$ are given in Theorem 5.2.

Corollary 5.1. *Let $\mu = \log_2(2+\sqrt{2})$. For $q \to \infty$ and $1 \leqslant r \leqslant \frac{q}{4}$, the number of $(\Phi, q-r)^*$-bad monomials is*

$$|S^*(\ell)| = \Theta(r^{2-\mu} q^\mu) \ .$$

Moreover, the QLRS code $\mathcal{C}_q(\Phi, q-r)$ has rate

$$R = 1 - \Theta\left((q/r)^{\mu-2}\right) = 1 - \Theta\left((q/r)^{-0.2284}\right) \ .$$

Proof. It can be seen from Theorem 5.2 that the order of $|S^*(\ell)|$ is controlled by λ_1^ℓ. Its asymptotic estimation is obtained by neglecting the other terms and the constant coefficients. The rate is calculated by dividing the number of good monomials, $q^2 - |S^*(\ell)|$ by the number of all bi-variate monomials, q^2. □

Remark 5.1. *Recall that the rate of bivariate lifted Reed-Solomon codes given in [HPPV20] is*

$$R = 1 - \Theta((q/r)^{\log_2 3 - 2} = 1 - \Theta((q/r)^{-0.4150}) \ .$$

We compare the performance of QLRS codes with lifted Reed-Solomon codes in terms of local recovery in Section 5.2.3.

5.2.2 Minimum Hamming Distance of Quadratic-Lifted Reed-Solomon Codes

Similar to RS codes and RM codes, a QLRS code $\mathcal{C}_q(\Phi, q-r)$ can be written as a block code with length $n = q^2$ and dimension $k = q^2 - |S^*(\ell)|$, where $\ell = \log_2 q$, via evaluations:

$$\mathsf{QC}_q[n,k] := \left\{ \left(f(\boldsymbol{v})\right)_{\boldsymbol{v}\in\mathbb{F}_q^2} \;\middle|\; f \in \mathcal{C}_q(\Phi, q-r) \right\} \ .$$

The minimum Hamming distance of a block code is given in Definition 2.16. We define the minimum Hamming distance of $\mathcal{C}_q(\Phi, q-r)$ as the minimum Hamming distance of $\mathsf{QC}_q[n,k]$, i.e.,

$$\mathrm{d}_\mathsf{H}(\mathcal{C}_q(\Phi, q-r)) := \mathrm{d}_\mathsf{H}(\mathsf{QC}_q[n,k]) \ .$$

We provide upper and lower bounds on the minimum Hamming distance of QLRS codes in the following theorem.

Theorem 5.3 (Bounds on minimum Hamming distance)**.** *Let q be a power of 2, $r \in [q-1]$ and Φ be the set of all quadratic functions. The QLRS code $\mathcal{C}_q(\Phi, q-r)$ has minimum Hamming distance*

$$qr + 1 \leqslant \mathrm{d}_\mathsf{H}(\mathcal{C}_q(\Phi, q-r)) \leqslant qr + q \ .$$

Proof. We first show the upper bound. Let $\mathcal{A} \subset \mathbb{F}_q$ be a subset with $|\mathcal{A}| = q - r - 1$. Consider a bivariate polynomial $f(x,y) = \prod_{\alpha\in\mathcal{A}}(x-\alpha) \in \mathbb{F}_q[x,y]$. It can be seen that $\deg(f|_\phi) = q - r - 1$ for any $\phi \in \Phi$ therefore $f(x,y)$ is in the code $\mathcal{C}_q(\Phi, q-r)$. The zeros of $f(x,y)$ in $\mathbb{F}_q^2$ are $\{(x,y) \mid x \in \mathcal{A}, y \in \mathbb{F}_q\}$. Therefore, the evaluations of $f(x,y)$ in $\mathbb{F}_q^2$ are of weight $q^2 - q(q-r-1) = qr + q$. Due to the linearity of the code, the upper bound on the minimum distance is proven.
Now we prove the lower bound. For any nonzero $f \in \mathcal{C}_q(\Phi, q-r)$ consider a point $\boldsymbol{p} = (a,b) \in \mathbb{F}_q^2$ such that $f(a,b) \neq 0$. Denote by $\mathcal{L}_{\boldsymbol{p}} \subset \Phi$ the set of lines (i.e., quadratic functions with $\alpha = 0$) in Φ intersecting with each other only at $\boldsymbol{p}$. It can be seen that $|\mathcal{L}_{\boldsymbol{p}}| = q$. By definition, for any line $L \in \mathcal{L}_{\boldsymbol{p}}$, $\deg(f|_L) < q - r$. Therefore there are at least $r+1$ nonzero evaluations of f at the points on L. Denote by $\mathrm{wt}_\mathsf{H}(f)$ the number of nonzero evaluations of f on $\mathbb{F}_q^2$ and by

$\mathrm{wt}_{\mathsf{H}}(f|_L)$ the number of nonzero evaluations of f on L, then

$$\mathrm{wt}_{\mathsf{H}}(f) \geqslant \sum_{L \in \mathcal{L}_{\boldsymbol{p}}} \big(\mathrm{wt}_{\mathsf{H}}(f|_L) \underbrace{-1}_{\text{excluding } f(\boldsymbol{p})} \big) \underbrace{+1}_{\text{including } f(\boldsymbol{p})} \geqslant qr + 1$$

Note that the bounds are derived in a similar manner as for lifted Reed-Solomon codes in [GKS13, Theorem 5.1]. □

5.2.3 Local Recovery of Erasures

Consider an erasure channel with erasure probability τ. A *local recovery set* of an erasure (i.e., a missing codeword symbol) in a codeword is a set of indices where the erasure can be recovered by accessing only the other codeword symbols whose indices are in the set. Given a QLRS code over $\mathbb{F}_q$, the number of local recovery sets of any codeword symbol is the number of quadratic functions over $\mathbb{F}_q$ passing through the evaluation point of that codeword symbol, which is q^2. For a lifted Reed-Solomon code, the number of local recovery sets of any codeword symbol is $q + 1$.

In this section, we are interested in correcting a certain erasure within any local recovery set by QLRS codes. We say a local recovery fails if the erasure cannot be recovered from any of its local recovery sets. By the structure of a QLRS code $\mathcal{C}_q(\Phi, q-1)$, this happens if and only if there are at least r other erasures in each local recovery set of the erased symbol.

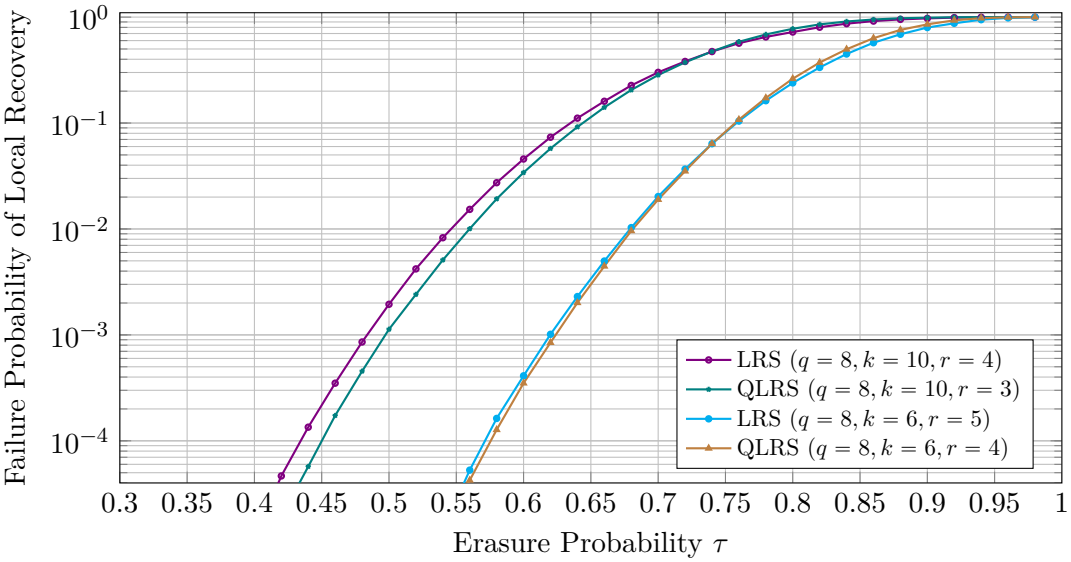


Figure 5.2: Local recovery performance of lifted Reed-Solomon (LRS) codes and QLRS codes of length $n = q^2 = 64$ and dimension $k = 10$ (rate $= k/n = 0.15625$) or $k = 6$ (rate $= k/n = 0.09375$).

For a lifted Reed-Solomon code, since all the local recovery sets of a certain codeword symbol are disjoint, the failure probability of a local recovery for an erasure, denoted by $P_{f,LRS}$, is

$$P_{f,LRS} = \left(\sum_{i=r}^{q-1} \binom{q-1}{i} \tau^i (1-\tau)^{q-1-i} \right)^{q+1} .$$

For QC-LRS codes, since the local recovery sets may intersect with each other, a closed form of the failure probability is still an open problem.

In order to compare the performance of these two codes, we run simulations with both codes of length $n = 64$, dimension $k = 10$ and $k = 6$, respectively. It can be seen from Corollary 5.1 and Remark 5.1 that for the same local redundancy r, the dimension of the lifted Reed-Solomon code is larger than that of QLRS. To have a fair comparison, we choose different local redundancy r for lifted Reed-Solomon codes and QLRS codes such that their dimensions are the same. For instance, to obtain the same dimension $k = 10$, we need to set the local redundancy $r = 4$ for the lifted Reed-Solomon code and $r = 3$ for the QLRS code.

The simulation results are presented in Fig. 5.2. We can see that for both $k = 10$ and $k = 6$, the failure probability of a local recovery with QLRS is smaller than or similar to that with lifted Reed-Solomon codes for $\tau \leqslant 0.7$.

5.3 Almost Affinely Disjoint Subspace Design based on Reed-Solomon Codes

In this section we show an application of the most well-known evaluation codes – Reed-Solomon codes – in constructing a family of k-dimensional subspaces of $\mathbb{F}_q^n$. This family of subspaces was motivated by *batch codes*, which is a class of local recovery codes with *availability* for distributed storage systems. We point out the connection after introducing the necessary notations.

Definition 5.4 (Almost affinely disjoint (AAD) subspace family)**.** *Given positive integers k and n such that $n > 2k$, let $\mathcal{F}$ be a family of k-dimensional linear subspaces in $\mathbb{F}_q^n$. This family is said to be* L-almost affinely disjoint*, denoted by* $[n, k, L]_q$-AAD*, if the following two properties hold:*

(i) The family $\mathcal{F}$ is a partial k-spread of $\mathbb{F}_q^n$, i.e., a collection of k-dimensional subspaces with pairwise trivial intersection.

(ii) For any $\mathcal{S} \in \mathcal{F}$ and $\boldsymbol{u} \in \mathbb{F}_q^n \backslash \mathcal{S}$, the affine subspace of $\mathcal{S}$ w.r.t. $\boldsymbol{u}$,

$$\boldsymbol{u} + \mathcal{S} := \{\boldsymbol{u} + \boldsymbol{v} \mid \boldsymbol{v} \in S\}$$

intersects at most L subspaces from the family $\mathcal{F}$.

We denote the *maximal size* of an $[n, k, L]_q$-AAD family by $m_q^{AAD}(n, k, L)$ and define the *polynomial growth* of the maximal size of an AAD family as

$$p^{AAD}(n, k, L) := \limsup_{q \to \infty} \log_q(m_q^{AAD}(n, k, L)) \ . \tag{5.12}$$

AAD subspace families with $n = 2k + 1$ were first introduced by Polyanskii and Vorobyev in [PV19a] to construct *primitive batch codes* [IKOS04, Definition 2.3]. A binary primitive $[N, K, s]_q$-batch code encodes an information vector $\boldsymbol{x} \in \mathbb{F}_2^K$ into a codeword $\boldsymbol{c} \in \mathbb{F}_2^N$, such that for any multiset $\{\{i_1, \ldots, i_s\}\} \subseteq [K]$, there exists s mutually disjoint sets $\mathcal{S}_1, \ldots, \mathcal{S}_s \subseteq [N]$ (referred to as *recovering sets*) such that each $x_{i_j}, j \in [s]$ can be recovered by the bits in $\boldsymbol{c}$ whose indices are in $\mathcal{S}_j$. The parameter s is usually called *availability* and it plays an important role in supporting high throughput of the distributed storage system.

It has been shown in [PV19a, Lemma 2] that a systematic $[N, K, s]_q$-batch code can be constructed from an $[n, k, L]_q$-AAD subspace family $\mathcal{F}$, where $N = q^n + |\mathcal{F}|q^{n-k}$, $K = q^n$ and $s = \lfloor |\mathcal{F}|/L \rfloor$. The explicit encoding procedure is as follows: we associate $K = q^n$ information bits with K points in $\mathbb{F}_q^n$ and let the first K bits in $\boldsymbol{c}$ equal to $\boldsymbol{x}$. For every affine subspace $\boldsymbol{u} + \mathcal{S}$ with $\mathcal{S} \in \mathcal{F}$ and $\boldsymbol{u} \in \mathbb{F}_q^n$, we compute a parity-check bit as a sum of information bits associated with the points lying in this affine subspace and append it to $\boldsymbol{c}$. As the number of distinct affine subspaces of such a form is $|\mathcal{F}|q^{n-k}$, the constructed systematic code has length $N = q^n + |\mathcal{F}|q^{n-k}$. By the definition of the $[n, k, L]_q$-AAD family, it can be seen that for every bit in $\boldsymbol{x}$, each of its recovery sets (composed of a parity-check bit c_i for some $i > K$ and the information bits that are the other sumands of c_i) intersects with at most L recovery sets of any other bit. Hence, $s = \lfloor |\mathcal{F}|/L \rfloor$.

A naive way to construct AAD families is by exploiting constructions of long linear codes $\mathcal{C}$ with fixed $\mathsf{d_H}(\mathcal{C})$. Let $\boldsymbol{H}$ be a parity-check matrix of a linear $[N, K]_q$ code $\mathcal{C}$ with $\mathsf{d_H}(\mathcal{C}) = 3k+1$. Let the subspace $\mathcal{S}_i$ be the $\mathbb{F}_q$-linear span of k consecutive columns, from the $(ik+1)$-th to the $(i+1)k$-th column, of $\boldsymbol{H}$. Then $\mathcal{F} = \{\mathcal{S}_1, \ldots, \mathcal{S}_{\lfloor N/k \rfloor}\}$ is an $[N-K, k, 1]_q$-AAD family. Thus, for a fixed minimum Hamming distance, the longer the code $\mathcal{C}$, the larger the constructed AAD family. Yekhanin and Dumer have developed a class of long non-binary codes with fixed Hamming distance in [YD04]. For $k = 1$, linear $[N, K]_q$ codes with $\mathsf{d_H} = 3k + 1 = 4$ are known to be equivalent to caps in projective geometries and have been studied extensively under this name, e.g., in [Muk78; EB99; HS01]. From the results in [EB99; YD04], for fixed k and large enough n, it holds that $p^{ADD}(n, k, 1) \geqslant (3k-1)(n+1)/(9k^2 - 9k + 1)$.

In the rest of the section, we present a construction of $[n, k, L]_q$-AAD families based on Reed-Solomon codes for $k = 1, 2$ in Section 5.3.1, and new upper and lower bounds on the polynomial growth $p^{ADD}(n, k, L)$ of the maximal size of an AAD family for general $L \geqslant 1$ in Section 5.3.2.

5.3.1 Explicit Constructions

Construction 5.1. *Let $q \geqslant nk$, $m = q^{n-2k}$ and γ be a primitive element of $\mathbb{F}_q$. For $i \in [m]$, let $\mathcal{S}_i$ be a subspace spanned by the vectors $\{\boldsymbol{v}_{i,1}, \ldots, \boldsymbol{v}_{i,k}\}$ with*

$$\boldsymbol{v}_{i,t} := \begin{pmatrix} \boldsymbol{e}_t & \Gamma_t(\boldsymbol{c}_i) & h_t(\boldsymbol{c}_i) \end{pmatrix} \in \mathbb{F}_q^n, \ t \in [k] \ , \tag{5.13}$$

where $\boldsymbol{e}_t$ is a unit vector $\in \mathbb{F}_q^k$ with a one at the t-th position, $\boldsymbol{c}_i$ is a codeword of an $\mathsf{RS}_q[n-k-1, n-2k]$ code having a parity-check matrix as the following

$$\boldsymbol{H}_{\mathsf{RS}} := \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & \gamma & \gamma^2 & \cdots & \gamma^{n-k-2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \gamma^{k-2} & \gamma^{2(k-2)} & \cdots & \gamma^{(n-k-2)(k-2)} \end{pmatrix} , \tag{5.14}$$

$\Gamma_t(\cdot)$ is a map

$$\begin{aligned} \Gamma_t(\cdot) : \mathbb{F}_q^{n-k-1} &\to \mathbb{F}_q^{n-k-1} \\ \boldsymbol{x} &\mapsto \begin{pmatrix} \gamma^{t-1}x_1 & \gamma^{2(t-1)}x_2 & \gamma^{3(t-1)}x_3 & \cdots & \gamma^{(n-k-1)(t-1)}x_{n-k-1} \end{pmatrix} , \end{aligned}$$

and $h_t(\cdot)$ is a function $h_t(\boldsymbol{x}) := \sum_{p=1}^{n-k-1} x_p^{(t-1)(n-k-1)+p+1}$. Then let $\mathcal{F}_{n,k} := \{\mathcal{S}_1, \ldots, \mathcal{S}_m\}$.

Theorem 5.4. *The family $\mathcal{F}_{n,k}$ from Construction 5.1 is a partial k-spread in $\mathbb{F}_q^n$. Moreover, for $k = 1$ and $k = 2$, $\mathcal{F}_{n,k}$ is $[n, k, L(n,k)]_q$-AAD with $L(n,1) = n-1$ and $L(n,2) = 1 + 2(n-2)(2n-6)$.*

Proof. The vectors $\boldsymbol{v}_{i,1}, \ldots, \boldsymbol{v}_{i,k} \in \mathbb{F}_q^n$ are linearly independent as the restriction to the first k coordinates are unit vectors. Hence, their span defines a k-dimensional subspace in $\mathbb{F}_q^n$. Suppose that $\mathcal{S}_i$ and $\mathcal{S}_j$ have a non-trivial intersection. Then,

$$\operatorname{rank}\begin{pmatrix} \boldsymbol{v}_{i_1} \\ \vdots \\ \boldsymbol{v}_{i,k} \\ \boldsymbol{v}_{j,1} \\ \vdots \\ \boldsymbol{v}_{j,k} \end{pmatrix} = \operatorname{rank}\begin{pmatrix} \boldsymbol{e}_1 & \Gamma_1(\boldsymbol{c}_i) & h_1(\boldsymbol{c}_i) \\ \vdots & & \\ \boldsymbol{e}_k & \Gamma_k(\boldsymbol{c}_i) & h_k(\boldsymbol{c}_i) \\ \boldsymbol{e}_1 & \Gamma_1(\boldsymbol{c}_j) & h_1(\boldsymbol{c}_j) \\ \vdots & & \\ \boldsymbol{e}_k & \Gamma_k(\boldsymbol{c}_j) & h_k(\boldsymbol{c}_j) \end{pmatrix} \leqslant 2k-1 \ ,$$

which yields that

$$\operatorname{rank}\begin{pmatrix} \Gamma_1(\boldsymbol{c}_i - \boldsymbol{c}_j) \\ \Gamma_2(\boldsymbol{c}_i - \boldsymbol{c}_j) \\ \vdots \\ \Gamma_k(\boldsymbol{c}_i - \boldsymbol{c}_j) \end{pmatrix} < k \ . \tag{5.15}$$

Denote by $c_{i,j}$ the jth entry of $\boldsymbol{c}_i$. Since $\boldsymbol{c}_i - \boldsymbol{c}_j$ is a nonzero codeword of the $\mathsf{RS}_q[n-k-1, n-2k]$ code with $\mathsf{d_H} = k$, there exist k coordinates $p_1, \ldots, p_k \in [n-k-1]$ such that $u_t := c_{i,p_t} - c_{j,p_t} \neq 0$ for $t \in [k]$. Thus, after restricting each row of the matrix in (5.15) to coordinates $p_1, \ldots, p_k$, the rank deficiency of the matrix in (5.15) is equivalent to

$$\det\begin{pmatrix} u_1 & u_2 & \cdots & u_k \\ \gamma^{p_1}u_1 & \gamma^{p_2}u_2 & \cdots & \gamma^{p_k}u_k \\ \vdots & \vdots & \ddots & \vdots \\ \gamma^{p_1(k-1)}u_1 & \gamma^{p_2(k-1)}u_2 & \cdots & \gamma^{p_k(k-1)}u_k \end{pmatrix} = \prod_{t=1}^{k} u_t \prod_{1 \leqslant s < r \leqslant k} (\gamma^{p_r} - \gamma^{p_s}) = 0 \ .$$

However, since γ is a primitive element of the field $\mathbb{F}_q$ with $q \geqslant nk$ and all u_t's for $t \in [k]$ are nonzero, the determinant cannot be zero, which contradicts the assumption that S_i and S_j intersect non-trivially. Hence, the family $\mathcal{F}_{n,k}$ is a partial k-spread.

Suppose that for the $\mathcal{S}_i \in \mathcal{F}_{n,k}$ and some $\boldsymbol{v} \in \mathbb{F}_q^n \notin \mathcal{S}_i$, the affine space $\mathcal{V} := \boldsymbol{v} + \mathcal{S}_i$ intersects more than $L_{n,k}$ subspaces from the family $\mathcal{F}_{n,k}$. Let $\boldsymbol{v}$ be a nonzero vector from one of the $L_{n,k}+1$ subspaces intersecting $\mathcal{V}$. E.g., assume $\boldsymbol{v} \in \mathcal{S}_j \in \mathcal{F}_{n,k}$, for some $j \in [m] \backslash \{i\}$. Then we can write $\boldsymbol{v}$ as a linear combination of the basis of $\mathcal{S}_j$. Namely,

$$\boldsymbol{v} = \boldsymbol{v}_j(\boldsymbol{\alpha}) := \sum_{t=1}^{k} \alpha_t \boldsymbol{v}_{j,t} \text{ for some } \boldsymbol{\alpha} \in \mathbb{F}_q^k \backslash \{\boldsymbol{0}\} \ ,$$

where $\boldsymbol{v}_{j,t}$'s are given in (5.13). In what follows, we estimate the number of other subspaces

in $\mathcal{F}_{n,k}$ such that there exists a linear combination of their basis in $\mathcal{V}$, i.e.,

$$\left|\left\{\ell \in [m]\backslash\{i,j\} \mid \boldsymbol{v}_\ell(\boldsymbol{\beta}) \in \mathcal{V}, \text{ for some } \mathbf{0} \neq \boldsymbol{\beta} \in \mathbb{F}_q^k\right\}\right| . \tag{5.16}$$

It can be seen that this is the number of subspaces in $\mathcal{F}_{n,k}$ that intersect $\mathcal{V}$. Note that $\boldsymbol{v}_\ell(\boldsymbol{\beta}) \in \mathcal{V}$ is is equivalent to the property that

$$\operatorname{rank}\begin{pmatrix} \boldsymbol{v}_{i,1} \\ \vdots \\ \boldsymbol{v}_{i,k} \\ \boldsymbol{v}_j(\boldsymbol{\alpha}) \\ \boldsymbol{v}_\ell(\boldsymbol{\beta}) \end{pmatrix} \leqslant k+1 .$$

By the structure of the vectors $\boldsymbol{v}_{i,t}$'s in (5.13), the rank deficiency above implies that

$$\operatorname{rank} \underbrace{\begin{pmatrix} \sum_{t=1}^k \alpha_t \Gamma_t(\boldsymbol{c}_j - \boldsymbol{c}_i) & \sum_{t=1}^k \alpha_t (h_t(\boldsymbol{c}_j) - h_t(\boldsymbol{c}_i)) \\ \sum_{t=1}^k \beta_t \Gamma_t(\boldsymbol{c}_\ell - \boldsymbol{c}_i) & \sum_{t=1}^k \beta_t (h_t(\boldsymbol{c}_\ell) - h_t(\boldsymbol{c}_i)) \end{pmatrix}}_{=:\boldsymbol{R}_{\boldsymbol{\beta},\ell} \in \mathbb{F}_q^{2\times(n-k)}} = 1 .$$

Denote

$$\mathcal{T} := \left\{ \boldsymbol{c}_\ell \in \mathsf{RS}_q[n-k-1, n-2k] \mid \operatorname{rank}(\boldsymbol{R}_{\boldsymbol{\beta},\ell}) = 1 \text{ for some } \mathbf{0} \neq \boldsymbol{\beta} \in \mathbb{F}_q^k \right\} . \tag{5.17}$$

Then, (5.16)$= |\mathcal{T}|$.

Now we show that at least one entry in the first $n-k-1$ entries of the first row of $\boldsymbol{R}_{\boldsymbol{\beta},\ell}$ is nonzero. Observe that for each $p \in [n-k-1]$, the p-th entry in $\sum_{t=1}^k \alpha_t \Gamma_t(\boldsymbol{c}_j - \boldsymbol{c}_i)$ has the form $(c_{j,p} - c_{i,p}) \sum_{t=1}^k \alpha_t \gamma^{(t-1)p}$. Since $\boldsymbol{c}_i$ and $\boldsymbol{c}_j$ are codewords of the $\mathsf{RS}_q[n-k-1, n-2k]$ code with minimum Hamming distance k, there are at least k indices $p \in [n-k-1]$ such that $c_{j,p} - c_{i,p} \neq 0$. We see $\sum_{t=1}^k \alpha_t \gamma^{(t-1)p}$ as a polynomial of degree $k-1$ in the unknown $x = \gamma^p$. Then the polynomial has at most $k-1$ distinct roots in $\mathbb{F}_q$. For each root x_0, there is at most one $p \in [n-k-1]$ such that $\gamma^p = x_0$, since γ is a primitive element in $\mathbb{F}_q$ with $q \geqslant nk$. Then for any nonzero $\boldsymbol{\alpha}$, there are at most $k-1$ distinct p so that $\sum_{t=1}^k \alpha_t \gamma^{(t-1)p} = 0$. Hence, there is at least one entry $(c_{j,p_*} - c_{i,p_*}) \sum_{t=1}^k \alpha_t \gamma^{(t-1)p_*}$ in the vector $\sum_{t=1}^k \alpha_t \Gamma_t(\boldsymbol{c}_j - \boldsymbol{c}_i)$ being nonzero.

To continue, we need the following lemma, whose proof is given later.

Lemma 5.9. *Given a nonzero vector $\boldsymbol{\beta} \in \mathbb{F}_q^k$, there are at most $k(n-k)$ distinct codeword $\boldsymbol{c}_\ell \in \mathsf{RS}_q[n-k-1, n-2k]$ such that* $\operatorname{rank}(\boldsymbol{R}_{\boldsymbol{\beta},\ell}) = 1$.

Let us proceed with proving the remaining statement of this theorem, i.e., the value of $L_{n,k}$ for $k = 1, 2$. For this purpose, we estimate the number of possible $\boldsymbol{\beta}$'s such that the first $n-k-1$ columns of $\boldsymbol{R}_{\beta,\ell}$, are collinear to the p_*-th column and then apply Lemma 5.9.

For $k = 1$, note that for any multiple of $\boldsymbol{\beta} = (1)$, the set of $\boldsymbol{c}_\ell$ such that $\operatorname{rank}(\boldsymbol{R}_{\boldsymbol{\beta},\ell}) = 1$ is the same. By Lemma 5.9, the number of distinct appropriate ℓ's is at most $n-1$. Thus, $\mathcal{F}_{n,1}$ is an $[n, 1, n-1]_q$-AAD family.

Now we discuss the case $k = 2$. Note that any nonzero vector $\boldsymbol{\beta} \in \mathbb{F}_q^2$ is either collinear to

$(1, q-1)$ or $(\beta_1, 1-\beta_1)$, where $\beta_1 \in \mathbb{F}_q$. Define a set

$$\mathcal{B} := \left\{(\beta_1, 1-\beta_1) \in \mathbb{F}_q^2 \mid \beta_1 + (1-\beta_1)\gamma^p = 0, \text{ for some } p \in [n-3]\right\} ,$$

and it can be readily seen that $|\mathcal{B}| \leqslant n-3$. We assume that for $\boldsymbol{\beta} = (1, q-1)$ or $\boldsymbol{\beta} \in \mathcal{B}$, the set $\mathcal{T}$ in (5.17) is not empty. We now estimate the number of *suspicious* $\widetilde{\boldsymbol{\beta}} = (\widetilde{\beta}_1, 1-\widetilde{\beta}_1) \notin \mathcal{B}$ such that the set $\mathcal{T}$ may not be empty. If $\boldsymbol{R}_{\widetilde{\boldsymbol{\beta}},\ell}$ has rank 1, then two rows of $\boldsymbol{R}_{\widetilde{\boldsymbol{\beta}},\ell}$ are collinear and there exists some $\lambda \in \mathbb{F}_q^*$ such that

$$\left(\widetilde{\beta}_1 + (1-\widetilde{\beta}_1)\gamma^p\right)(c_{\ell,p} - c_{i,p}) = \underbrace{\lambda(\alpha_1 + \alpha_2\gamma^p)(c_{j,p} - c_{i,p})}_{=:w_p}, \ \forall p \in [n-3] . \tag{5.18}$$

From the parity-check equation $\sum_{p=1}^{n-3}(c_{\ell,p} - c_{i,p}) = 0$ imposed by the first column of (5.14), we have

$$\sum_{p=1}^{n-3} \frac{w_p}{\widetilde{\beta}_1 + (1-\widetilde{\beta}_1)\gamma^p} = 0 \quad \Longleftrightarrow \quad \sum_{p=1}^{n-3} w_p \prod_{\substack{t=1 \\ t\neq p}}^{n-3} (\widetilde{\beta}_1 + (1-\widetilde{\beta}_1)\gamma^t) = 0 .$$

Since there is some $p \in [n-3]$ such that $w_p \neq 0$ and $\widetilde{\beta}_1 + (1-\widetilde{\beta}_1)\gamma^p \neq 0$ for all $p \in [n-3]$, the left-hand side of the above equation is a nonzero polynomial in $\mathbb{F}_q[\widetilde{\beta}_1]$ of degree at most $n-4$. Therefore, there are at most $n-4$ suspicious $\widetilde{\boldsymbol{\beta}}$'s such that the vector $\sum_{t=1}^{2} \widetilde{\beta}_t \Gamma_t(\boldsymbol{c}_\ell - \boldsymbol{c}_i)$ is collinear to $\sum_{t=1}^{2} \alpha_t \Gamma_t(\boldsymbol{c}_j - \boldsymbol{c}_i)$. Let $\mathcal{D}$ be the union of the suspicious $\widetilde{\boldsymbol{\beta}}$'s, the set $\mathcal{B}$ and the set $\{(1, q-1)\}$. It can be seen that

$$\begin{aligned} |\mathcal{D}| &= \left|\left\{\widetilde{\boldsymbol{\beta}}\right\}\right| + |\mathcal{B}| + |\{(1, q-1)\}| \\ &\leqslant (n-4) + (n-3) + 1 \leqslant 2n-6 . \end{aligned}$$

Hence, by Lemma 5.9, $\mathcal{F}_{n,2}$ is an $[n, 2, L(n,2)]_q$-AAD family with $L(n,2) = 1 + 2(n-2)(2n-6)$. □

Proof of Lemma 5.9. Recall that the entry at the first row, p_*-th column of $\boldsymbol{R}_{\boldsymbol{\beta},\ell}$ is nonzero. If $\boldsymbol{R}_{\boldsymbol{\beta},\ell}$ has rank 1, then each of the first $n-k-1$ columns of $\boldsymbol{R}_{\boldsymbol{\beta},\ell}$ is linearly dependent on the p_*-th column. Moreover, the dependency is determined by the first row of $\boldsymbol{R}_{\boldsymbol{\beta},\ell}$. Fix a $c_{\ell,*} \in \mathbb{F}_q$. For any $p \in [n-k-1]$, let

$$\phi_p = \frac{(c_{j,p} - c_{i,p}) \sum_{t=1}^{k} \alpha_t \gamma^{(t-1)p}}{(c_{j,p_*} - c_{i,p_*}) \sum_{t=1}^{k} \alpha_t \gamma^{(t-1)p_*}} .$$

Then, having the p-th column collinear to the p_*-th column gives the following system of equations on the unknowns $c_{\ell,p}, p \in [n-k-1]\backslash\{p_*\}$:

$$(c_{\ell,p} - c_{i,p}) \sum_{t=1}^{k} \beta_t \gamma^{(t-1)p} = \phi_p (c_{\ell,p_*} - c_{i,p_*}) \sum_{t=1}^{k} \beta_t \gamma^{(t-1)p_*} . \tag{5.19}$$

Note that $\sum_{t=1}^{k} \beta_t \gamma^{(t-1)p} = 0$ for at most $k-1$ distinct $p \in [n-k-1]\backslash\{p_*\}$. Therefore, (5.19) provides at least $n-k-2-(k-1) = n-2k-1$ equations on the unknown $c_{\ell,p}$'s.

Since $c_{\ell,p}$ are entries of a codeword of the $\mathsf{RS}_q[n-k-1, n-2k]$ code with a parity-check

matrix (5.14), we also have the following equations:

$$\sum_{p=1}^{n-k-1} \gamma^{(p-1)(t-1)}(c_{\ell,p} - c_{i,p}) = 0, \ \forall t \in [k-1] \ . \tag{5.20}$$

Thus, the system of equations (5.19)-(5.20) gives at least $n-k-2$ linearly independent equations on the $n-k-2$ unknowns $\{c_{\ell,p}, p \in [n-k-1] \backslash \{p_*\}\}$. This system of equations has at most one solution. W.l.o.g., for any $p \in [n-k-1]\backslash\{p_*\}$, we write $c_{\ell,p} = a_p c_{\ell,p_*} + b_p$ with some $a_p, b_p \in \mathbb{F}_q$ for later use. So far, we have shown that given the c_{ℓ,p_*}, the $\boldsymbol{c}_\ell$ is uniquely determined if $\operatorname{rank}(\boldsymbol{R}_{\boldsymbol{\beta},\ell}) = 1$.

To have $\boldsymbol{R}_{\boldsymbol{\beta},\ell}$ has rank 1, we also require that the last column of $\boldsymbol{R}_{\boldsymbol{\beta},\ell}$ is collinear to the p_*-th column, which implies that

$$\det \begin{pmatrix} \sum_{t=1}^{k} \alpha_t \gamma^{(t-1)p_*}(c_{j,p_*} - c_{i,p_*}) & \sum_{t=1}^{k} \alpha_t (h(\boldsymbol{c}_j) - h(\boldsymbol{c}_i)) \\ \sum_{t=1}^{k} \beta_t \gamma^{(t-1)p_*}(c_{\ell,p_*} - c_{i,p_*}) & \sum_{t=1}^{k} \beta_t (h(\boldsymbol{c}_\ell) - h(\boldsymbol{c}_i)) \end{pmatrix} = 0 \ .$$

Note that the right-bottom entry

$$\sum_{t=1}^{k} \beta_t (h(\boldsymbol{c}_\ell) - h(\boldsymbol{c}_i)) = \sum_{t=1}^{k} \beta_t \sum_{p=1}^{n-k-1} \left((a_p c_{\ell,p_*} + b_p)^{(t-1)(n-k-1)+p+1} - c_{i,p}^{(t-1)(n-k-1)+p+1} \right)$$

is a polynomial in c_{ℓ,p_*} of degree at least p_*+1 and at most $k(n-k-1)+1 \leqslant k(n-k)$. Therefore, the determinant is a nonzero polynomial in c_{ℓ,p_*} of degree at least p_*+1 and at most $k(n-k)$. Since $q \geqslant nk$, there are at most $k(n-k)$ solutions for c_{ℓ,p_*} resulting in a zero determinant. □

Corollary 5.2. *For $k = 1, 2, n > 2k$ and $L = L(n,k)$, the polynomial growth of the maximum cardinality of an $[n,k,L]_q$-AAD family is*

$$p^{AAD}(n,k,L) \geqslant n - 2k \ .$$

Proof. The statement follows from Theorem 5.5 and the cardinality of the family $\mathcal{F}_{n,k}$ given in Construction 5.1. □

5.3.2 Bounds on Polynomial Growth of the Cardinality

In this section, we give an upper bound and a lower bound on the polynomial growth $p^{AAD}(n,k,L)$ of the maximal size of an AAD family, which is defined as in (5.12).

An Upper Bound

Theorem 5.5 (Upper bound)**.** *Fix arbitrary positive integers L, k, n such that $2k < n$ and a prime power q. Let $\mathcal{F}$ be an $[n,k,L]_q$-AAD family. Then*

$$|\mathcal{F}| \leqslant 1 + L\frac{q^{n-k}-1}{q^k-1} \ . \tag{5.21}$$

For $L = q^{o(1)}$, it follows that $p^{AAD}(n,k,L) \leqslant n - 2k$.

Proof. Let $m := |\mathcal{F}|$ and $\mathcal{S}_i$ be the i-th subspace in $\mathcal{F}$. For all $i \in [m]$, let $\boldsymbol{G}_i \in \mathbb{F}_q^{k\times n}$ be a matrix such that $\langle \boldsymbol{G}_i \rangle_r = \mathcal{S}_i$ and let $\boldsymbol{H}_i \in \mathbb{F}_q^{(n-k)\times n}$ be a matrix such that $\langle \boldsymbol{H}_i \rangle_r = \mathcal{S}_i^{\perp}$, where $\langle \cdot \rangle_r$ denotes the row span. Hence, for all $i \in [m]$, $\boldsymbol{H}_i \boldsymbol{G}_i^{\top} = \boldsymbol{0}$. Note that for any $j \in [m-1]$, $\boldsymbol{H}_m \boldsymbol{G}_j^{\top} \in \mathbb{F}_q^{(n-k)\times k}$ has full column rank because $\mathcal{S}_m$ and $\mathcal{S}_j$ have only trivial intersection by Definition 5.4 of the AAD family $\mathcal{F}$. For any $j \in [m-1]$, let $\hat{\boldsymbol{G}}_j \in \mathbb{F}_q^{(n-2k)\times(n-k)}$ be a full-rank matrix such that

$$\hat{\boldsymbol{G}}_j \cdot (\boldsymbol{H}_m \boldsymbol{G}_j^{\top}) = \boldsymbol{0} \ . \tag{5.22}$$

Now, we prove via contradiction that for any nonzero vector $\boldsymbol{w}^{\top} \in \mathbb{F}_q^{(n-k)\times 1}$, $\hat{\boldsymbol{G}}_j \boldsymbol{w}^{\top} = \boldsymbol{0}$ holds for at most L different $j \in [m-1]$. Suppose that for some set $\mathcal{J} := \{j_1, \ldots, j_{L+1}\} \subset [m-1]$, we have $\hat{\boldsymbol{G}}_{j_t} \boldsymbol{w}^{\top} = \boldsymbol{0}$ for every $j_t \in \mathcal{J}$. This implies that $\boldsymbol{w}^{\top}$ is in the column span of $\boldsymbol{H}_m \boldsymbol{G}_{j_t}^{\top}$, i.e., $\boldsymbol{w}^{\top} = \boldsymbol{H}_m \boldsymbol{G}_{j_t}^{\top} \boldsymbol{y}_{j_t}^{\top}$ for some nonzero $\boldsymbol{y}_{j_t}^{\top} \in \mathbb{F}_q^{k\times 1}$.

Let $\boldsymbol{v}^{\top} := \boldsymbol{G}_{j_1}^{\top} \boldsymbol{y}_{j_1}^{\top} \in \mathbb{F}_q^{n\times 1}$. Then,

$$\forall j_t \in \mathcal{J},\ \boldsymbol{H}_m \boldsymbol{G}_{j_t}^{\top} \boldsymbol{y}_{j_t}^{\top} = \boldsymbol{w}^{\top} = \boldsymbol{H}_m \boldsymbol{G}_{j_1}^{\top} \boldsymbol{y}_{j_1}^{\top} = \boldsymbol{H}_m \boldsymbol{v}^{\top} \ ,$$

which means

$$\forall j_t \in \mathcal{J},\ \boldsymbol{G}_{j_t}^{\top} \boldsymbol{y}_{j_t}^{\top} = \boldsymbol{v}^{\top} + \boldsymbol{G}_m^{\top} \boldsymbol{x}_{j_t}^{\top},\ \text{for some } \boldsymbol{x}_{j_t}^{\top} \in \mathbb{F}_q^{k\times 1} \ .$$

But this implies that $\boldsymbol{v} + \mathcal{S}_m$ and $\mathcal{S}_{j_t}$ intersect non-trivially, for all $j_t \in \mathcal{J}$. By Definition 5.4, there are at most L different j's so that $\boldsymbol{v} + \mathcal{S}_m$ and $\mathcal{S}_j$ intersect. This leads to a contradiction.

We have derived above a necessary condition for a collection of subspaces to form an $[n, k, L]_q$-AAD family $\mathcal{F}$, that is, for any nonzero vector $\boldsymbol{w}^{\top} \in \mathbb{F}_q^{(n-k)\times 1}$, $\hat{\boldsymbol{G}}_j \boldsymbol{w}^{\top} = \boldsymbol{0}$ holds for at most L different $j \in [m-1]$, where the $\hat{\boldsymbol{G}}_j$'s are defined in (5.22) and $m = |\mathcal{F}|$. It can be seen that the expectation

$$\mathbb{E}\left|\left\{ j \in [m-1] \mid \hat{\boldsymbol{G}}_j \boldsymbol{w}^{\top} = \boldsymbol{0} \right\}\right| = (m-1) \Pr\left\{ \hat{\boldsymbol{G}}_j \boldsymbol{w}^{\top} = \boldsymbol{0} \right\} = (m-1) \frac{q^k - 1}{q^{n-k} - 1} \ .$$

If the condition in (5.21) is not fulfilled, i.e., $m > L\frac{q^{n-k}-1}{q^k-1} + 1$, then the above expectation is at least $L+1$, violating the necessary condition. □

Remark 5.2. *Note that if we change the definition of an AAD subspace family by dropping the first property (being a partial spread) in Definition 5.4, then the matrices $\boldsymbol{H}_m \boldsymbol{G}_j^{\top}$ would have full rank for at least $m-L-1$ different j's. This results in the bound $m \leqslant L+1+L\frac{q^{n-k}-1}{q^k-1}$.*

A Lower Bound by Random Construction

The construction proposed in Section 5.3.1 gives a lower bound for $k = 1, 2$. For general $k \geqslant q$, we give a lower bound via another subspace family, the *almost sparse subspace family* defined below.

Definition 5.5 (Almost sparse subspace family)**.** *Given positive integers k and n such that $2k < n$, let $\mathcal{F}$ be a family of k-dimensional linear subspaces in $\mathbb{F}_q^n$. This family is said to be* L-almost sparse*, denoted by $[n, k, L]_q$-*AS*, if the two properties hold:*

(i) The family $\mathcal{F}$ is a partial k-spread of $\mathbb{F}_q^n$.

(ii) Any $(k+1)$-dimensional subspace in $\mathbb{F}_q^n$ intersects non-trivially at most L subspaces from the family $\mathcal{F}$.

It can be readily seen that an $[n,k,L]_q$-AS family is also an $[n,k,L-1]_q$-AAD family. Hence the following lower bound on the cardinality of AS families holds naturally for AAD families. Similar to (5.12), we denote the *maximal size* of an $[n,k,L]_q$-AS family by $m_q^{AS}(n,k,L)$ and define the *polynomial growth* of $m_q^{AS}(n,k,L)$ as

$$p^{AS}(n,k,L) := \limsup_{q\to\infty} \log_q(m_q^{AS}(n,k,L)) \ .$$

Theorem 5.6 (Lower bound for AS families)**.** *For arbitrary positive integers L,n,k such that $2k<n$, and a prime power q, there exists an $[n,k,L]_q$-AS family $\mathcal{F}$ of size*

$$|\mathcal{F}| \leqslant m_q^*(n,k,L) := q^{n-2k-\frac{(n-k)(k+1)}{(L+1)}}(1-o(1)) \ .$$

For fixed L, it follows that $p^{AS}(n,k,L) \geqslant n-2k-\frac{(n-k)(k+1)}{(L+1)}$.

Proof. The number of k-dimensional subspaces in $\mathbb{F}_q^n$ is

$$\begin{bmatrix} n \\ k \end{bmatrix}_q = \prod_{i=0}^{k-1} \frac{q^n-q^i}{q^k-q^i} = \prod_{i=0}^{k-1} \frac{q^{n-i}-1}{q^{k-i}-1} = \Theta(q^{k(n-k)}) \ .$$

We form a family of k-dimensional subspaces, $\mathcal{F}=\{\mathcal{S}_1,\ldots,\mathcal{S}_m\}$, of size

$$m = q^{n-2k-\frac{(n-k)(k+1)}{(L+1)}}$$

by choosing each subspace $\mathcal{S}_i$ independently and uniformly with probability $1/\begin{bmatrix} n \\ k \end{bmatrix}_q$. Note that it is possible that $\mathcal{S}_i=\mathcal{S}_j$ for some $i\neq j$.

Define $\xi := |\{(i,j) \mid i,j\in[m],\ i<j,\ |\mathcal{S}_i\cap\mathcal{S}_j|\neq 1\}|$ as the number of pairs (i,j) such that $\mathcal{S}_i$ and $\mathcal{S}_j$ intersect non-trivially. We now estimate the expectation of ξ. The number of k-dimensional subspaces that only trivially intersect with a fixed k-dimensional subspace is equal to

$$g_k := \prod_{i=1}^{k-1} \frac{q^n-q^{k+i}}{q^k-q^i} \ .$$

Thus, the probability of two random k-dimensional subspaces having only the trivial intersection is $g_k/\begin{bmatrix} n \\ k \end{bmatrix}_q$. The expectation of ξ is then upper bounded as follows:

$$\begin{aligned}
\mathbb{E}(\xi) &\leqslant \sum_{1\leqslant i<j\leqslant m} \Pr\{|\mathcal{S}_i\cap\mathcal{S}_j|\neq 1\} \\
&= \sum_{1\leqslant i<j\leqslant m} \left(1-\frac{g_k}{\begin{bmatrix} n \\ k \end{bmatrix}_q}\right) \\
&\leqslant \binom{m}{2}\left(1-\prod_{i=0}^{k-1}\frac{q^n-q^{k+i}}{q^n-q^i}\right)
\end{aligned}$$

$$< m^2 \left(1 - \left(q^{nk} - q^{n(k-1)} \sum_{i=k}^{2k-1} q^i\right) \Big/ q^{nk}\right)$$
$$= m^2 \Big(q^{2k-1-n} + o\Big(q^{2k-1-n}\Big)\Big) < m\big(q^{-1} + o\big(q^{-1}\big)\big) \ .$$

By the Markov inequality, we have

$$\Pr\big\{\xi > q^{0.5}\mathbb{E}(\xi)\big\} < \frac{\mathbb{E}(\xi)}{q^{0.5}\mathbb{E}(\xi)} = o(1) \ .$$

Since $q^{0.5}\mathbb{E}(\xi) < m\big(q^{-0.5} + o\big(q^{-0.5}\big)\big) = o(m)$, we obtain that with probability at least $1-o(1)$, there exists a family $\mathcal{F}$ of size m, which contains at most $o(m)$ pairs of subspaces with non-trivial intersections. If we delete one of the intersecting subspaces for each pair, then we obtain a family $\mathcal{F}' \subset \mathcal{F}$ of subspaces of size at least $m - o(m)$ satisfying the first property in Definition 5.5 for AS subspace families.

Now we compute the probability of the second property in Definition 5.5 being violated. For a fixed $(k+1)$-dimensional subspace $\mathcal{V}$, the number of k-dimensional subspaces that only trivially intersect with $\mathcal{V}$ is equal to

$$u_k := \prod_{i=0}^{k-1} \frac{q^n - q^{k+1+i}}{q^k - q^i} \ .$$

Thus, the probability that $\mathcal{S}_i$ only trivially intersects $\mathcal{V}$ is $u_k/\begin{bmatrix} n \\ k \end{bmatrix}_q$. Let $\mathcal{F}_\mathcal{V} \subseteq \mathcal{F}$ be the set of subspaces in $\mathcal{F}$ that non-trivially intersect $\mathcal{V}$, i.e., $\mathcal{F}_\mathcal{V} := \{\mathcal{S} \in \mathcal{F} \mid \mathcal{S} \cap \mathcal{V}| \neq 1\}$. Applying the union bound, we can bound from above the probability that $\mathcal{V}$ non-trivially intersects at least $L+1$ subspaces in $\mathcal{F}$ by

$$\begin{aligned}
\Pr[|\mathcal{F}_\mathcal{V}| \geqslant L+1] &\leqslant \binom{m}{L+1} \left(1 - \frac{u_k}{\begin{bmatrix} n \\ k \end{bmatrix}_q}\right)^{L+1} \\
&= \binom{m}{L+1} \left(1 - \prod_{i=0}^{k-1} \frac{q^n - q^{k+1+i}}{q^n - q^i}\right)^{L+1} \\
&< m^{L+1} \left(1 - \left(q^{nk} - q^{n(k-1)} \sum_{i=k+1}^{2k} q^i\right) \Big/ q^{nk}\right)^{L+1} \\
&= m^{L+1} \Big(q^{2k-n} + o\Big(q^{2k-n}\Big)\Big)^{L+1} \\
&< q^{-(n-k)(k+1)}(1 + o(1)) \ .
\end{aligned}$$

Recall that the total number of $(k+1)$-dimensional subspaces is $\begin{bmatrix} n \\ k+1 \end{bmatrix}_q = \Theta(q^{(k+1)(n-k-1)})$. By the union bound,

$$\begin{bmatrix} n \\ k+1 \end{bmatrix}_q \cdot \Pr[|\mathcal{F}_\mathcal{V} \geqslant L+1|] < q^{-1-k} + o\Big(q^{-1-k}\Big) \ .$$

Hence, the probability that the second property is violated is $o(1)$. This completes the proof of the existence of an $[n, k, L]$-AS family of size $m - o(m)$. □

Corollary 5.3 (Lower bound for AAD families)**.** *For arbitrary positive integers L, n, k such that $2k < n$, and $q \to \infty$, there exists an $[n, k, L]_q$-AAD family of size*

$$|\mathcal{F}| \leqslant q^{n-2k-\frac{(n-k)(k+1)}{(L+2)}}(1 + o(1)) \ .$$

For fixed L, $p^{ADD}(n, k, L) \geqslant n - 2k - \frac{(n-k)(k+1)}{(L+2)}$.

Proof. The statements follows from the fact that an $[n, k, L+1]_q$-AS family is also an $[n, k, L]_q$-AAD family. □

The relation between an $[n, k, L + 1]_q$-AS family and an $[n, k, L]_q$-AAD family holds only in one direction in general. Nevertheless, for $k = 1$, an $[n, 1, L]_q$-AAD family is also an $[n, 1, L + 1]_q$-AS family. To see this, note that any 2-dimension subspace (a plane) that intersects with a 1-dimensional subspace (a line) from an $[n, k = 1, L]_q$-ADD family, must contain the line. Therefore, any $[n, 1, L]_q$-AAD family is also an $[n, 1, L + 1]_q$-AS family and Construction 5.1 with $k = 1$ also gives an $[n, 1, L + 1]_q$-AS family. Hence, $p^{AS}(n, 1, n) = p^{AAD}(n, 1, n - 1) = n - 2$.

Remark 5.3 (Related work on the AS families)**.** *The concept of almost sparse subspace families is closely related to the* weak subspace design *introduced by Guruswami and Xing in [GX13] and further studied in [GK16b; GXY18]. A collection $\mathcal{F}$ of subspaces in $\mathbb{F}_q^n$ is an $[n, k, L]_q$-*weak subspace design *if every k-dimensional subspace in $\mathbb{F}_q^n$ intersects non-trivially at most L subspaces from $\mathcal{F}$. Despite not being required by definition, many known constructions of weak subspace design contain subspaces with a fixed co-dimension at least k. Weak subspace design and AS families have the following (trivial) relations:*

- *An $[n, k, L]_q$-AS family is also an $[n, k + 1, L]_q$-weak subspace design.*
- *For $n \geqslant 2k + 1$, a partial k-spread of $\mathbb{F}_q^n$ is an $[n, k + 1, L]_q$-weak subspace family if and only if it is also an $[n, k, L]_q$-AS family.*

By the explicit constructions presented in [GK16b], we can derive that $p^{AS}(n, k, L) \geqslant \left\lfloor \frac{n-k}{k+1} \right\rfloor$ for $L \geqslant \frac{(n-1)(k+1)}{[(n-k)/(k+1)]}$.

5.4 Summary and Outlooks

The first part of this chapter introduces a new class bivariate evaluation codes, QLRS codes, which have the local property that the codeword symbols, whose coordinates are lying on a quadratic curve, form a codeword of an RS code. Hence, for any coordinate in a codeword, every quadratic curve passing through this coordinate gives a local recovery set of it. For a QLRS code over $\mathbb{F}_q$, there are q^2 local recovery sets for each codeword symbol. We have presented a necessary and sufficient condition on the monomials which form a basis of the code. Based on the condition, we give upper and lower bounds on the dimension and show that the asymptotic rate of a QLRS code over $\mathbb{F}_q$ with local redundancy r is $1-\Theta(q/r)^{-0.2284}$. Moreover, we have provided lower and upper bounds on the minimum distance of this class of codes and compared QLRS codes with lifted Reed-Solomon codes by simulations in terms of the probability that a certain erasure cannot be recovered locally. The simulation results showed that for short block lengths (e.g., $n = 64$) and under the same code dimension, QLRS

codes have better performance than lifted Reed-Solomon codes when the erasure probability $\tau \leqslant 0.7$.

For future research, error-correcting algorithms of QLRS codes can be developed and analyzed. A promising candidate is the *randomized list decoding algorithm* which has been used for RM codes [AS97; STV99; GRS00; KK16], lifted Reed-Solomon codes [GK16a] and lifted affine-invariant codes [HP21]. The algorithm uses list decoding in local correction and aggregates the local decoding results with appropriate weight to make a final decision on a symbol. Moreover, QLRS codes were originally motivated by potential applications in *coded caching*. The task of coded caching problems is to find *caching* and *delivery* strategies to minimize delay in a communication system, where K users, each has a cache capability for M files, demand some of $N > M$ files stored at a server. QLRS codes have the property that any two local recovery sets for one coordinate (a_1, b_1) intersect at another coordinate (a_2, b_2). If we see the symbol at the (a_1, b_1) as a segment of the file that a user demands from the server, then symbol at the intersection (a_2, b_2) can be placed as the cached content and the rest of symbols in one of the recovery sets should be downloaded to reconstruct the file. New coded caching schemes can be developed and analyzed based on QLRS codes, or the generalization – the weighted η-lifted codes [LN20].

The second part studies the AAD family of k-dimensional subspaces, motivated by constructions for batch codes – a class of local recoverable codes with availability. The subspaces in an AAD family form a partial spread and any $(k+1)$-dimensional subspace containing a subspace from the family non-trivially intersects with at most L subspaces from the family. We have presented an explicit construction of the AAD family for $k = 1, 2$ based on RS codes. The construction gives $[n, k, L]_q$-AAD families with cardinality q^{n-2k} and L being polynomial in n and k. Another question discussed is the polynomial growth $p^{AAD}(n, k, L)$ (with $q \to \infty$) of the maximal cardinality of these families. The upper and lower bounds on this quantity derived in this section show that

$$n - 2k - \frac{(k+1)(n-k)}{L+2} \leqslant p^{AAD}(n, k, L) \leqslant n - 2k, \quad \text{for all } 2k < n \ .$$

With the explicit constructions based on RS codes, we have $p^{ADD}(n, 1, L) = n-2$ for $L \geqslant n-1$ and $p^{AAD}(n, 2, L) = n - 4$ for $L \geqslant 1 + 2(n-2)(2n-6)$, respectively. For future work, the analysis on the explicit construction (Construction 5.1) can be done for $k \geqslant 3$ to show whether the upper bound on $p^{AAD}(n, k, L)$ is also tight for $k \geqslant 3$.

6

Joint Decoding of Interleaved Evaluation Codes

An s-interleaved code is the direct sum of s possible different codes (called *constituent codes*) of the same length n, and its codewords can be represented as $s \times n$ matrices. A common error model for these codes are *burst errors* [Ows88; CO92], where the errors are concentrated in several columns. As a distance metric, the Hamming weight of such an $s \times n$ matrix is defined as the number of nonzero columns of the matrix.

To decode an interleaved code, a naive approach is to simply decode each constituent codeword, i.e., each row of the $s \times n$ matrix independently. The error-correction capability of the interleaved code is then $\lfloor (\mathrm{d}_{\mathsf{H}} - 1)/2 \rfloor$, where d_{H} is the smallest minimum Hamming distance of the constituent codes. However, for various algebraic interleaved codes, it is possible to correct a larger fraction of errors by adopting a joint approach. For this reason, interleaved codes have many applications in which burst errors occur naturally or artificially, for instance, replicated file disagreement location [MK90], correcting burst errors in data-storage applications [KL97; HPW19; HLMV24], outer codes in concatenated codes [MK90; KL98; HV99; JTH04; SSB05; SSB09], ALOHA-like random-access schemes [HV99], decoding non-interleaved codes beyond half-the-minimum distance by power decoding [SSB10; Kam14; Ros18; PRB19; CP20; PRS21], distributed computing [SHN19] and code-based cryptography [EWZ18; HLPW19; RPW21; PHL+22; AAD+22; ADG+23].

Generalized Reed–Solomon (GRS) codes are among the most-studied classes of constituent codes for interleaved codes. There are several decoders for interleaved GRS codes [KL97; BKY03; BMS04; SSB09; Nie13; YL18a] that decode up to $t_{\mathsf{Int}} := \frac{s}{s+1}(n - \bar{k})$ errors, where $\bar{k}$ is the mean dimension of the constituent codes. All of these decoders fail for *some* error patterns of weight larger than the unique decoding radius of the constituent code with the smallest minimum Hamming distance. For errors of a given weight t, the fraction of errors leading to a unsuccessful decoding is roughly q^{-m} at the maximal decoding radius t_{Int} (where q^m is the field size of the GRS code), and decreases exponentially in $t_{\mathsf{Int}} - t$, the difference between the maximal decoding radius and the actual error weight.

There are also other decoding algorithms for interleaved GRS codes that decode beyond the radius $\frac{s}{s+1}(n - \bar{k})$, and even beyond the Johnson radius: [CS03; PV04; Par07; SSB07; CH13; WZB14; PR17; HLH+22]. For some of these decoders, simulation results suggest that these decoders can successfully decode a large fraction of error matrices of weight up to the claimed maximal radius, and in some very special cases, it is possible to derive bounds on this fraction. However, in general, only little is known about the fraction of decodable errors

by these decoders, which are therefore not considered in this chapter. Other code classes that have been considered as constituent codes of interleaved codes are one-point Hermitian codes [Kam14; PRB19; MMS21] and, more generally, algebraic-geometry codes [BMS05; Nak10].

For interleaved decoders of high order, i.e., where s is at least the number of corrupted columns in received codewords, a simple linear-algebraic decoder was proposed in [MK90; HV99] and generalized in [HV00; RV14]. Unlike all decoders mentioned above, this decoder works for interleaved codes with an arbitrary linear constituent code and guarantees to correct any error of weight up to $d-2$ that has full rank, where d is the minimum distance of the constituent code. This generic decoding approach has been applied to interleaved codes with arbitrary constituent codes, whose parity-check matrix is known, in the Hamming metric [Sen11; PHL$^+$22], the rank metric [CS96; OJ02; GRS15; AGHT18; BBB$^+$20] and the sum-rank metric [PRR22; SZHW23; ABD$^+$23], respectively, to enhance the security of the proposed post-quantum cryptosystems based on codes in these metrics.

This chapter is devoted to the joint decoding approaches applied to interleaved codes with two classes of evaluation codes as constituent codes. In Section 6.1, we present a syndrome-based joint decoding algorithm for interleaved RS codes and a necessary and sufficient condition on a successful decoding. In Section 6.2, we apply the joint decoding algorithm to interleaved *alternant* codes, where the constituent codes are subfield subcodes of GRS codes and give a condition of successful decoding tailored for alternant codes. In Section 6.3, we present lower and upper bounds on the probability of successful decoding of interleaved alternant codes by the joint decoding algorithm. Finally, we give brief summaries on the other results on joint decoding interleaved evaluation codes in Section 6.4.

The results in Section 6.1 and Section 6.2 were published in IEEE TIT [HLN$^+$21a] and partly in the proceeding of 2021 Information Theory Workshop [HLN$^+$21b].

6.1 Joint Decoding of Interleaved Reed-Solomon Codes

We first formally introduce the concept of interleaved codes and briefly recap the joint decoding algorithm for interleaved RS codes from [FT91; SSB09].

Definition 6.1 (Interleaved codes)**.** *The s-interleaved code $\mathcal{IC}^{(s)}$ with constituent code $\mathcal{C}$ is defined as*

$$\mathcal{IC}^{(s)} := \left\{ \begin{pmatrix} \boldsymbol{c}^{(1)} \\ \vdots \\ \boldsymbol{c}^{(s)} \end{pmatrix} \;\middle|\; \boldsymbol{c}^{(i)} \in \mathcal{C},\ i \in [s] \right\} .$$

The parameter s is referred to as the interleaving order *of the interleaved code.*

Consider a channel where burst errors of Hamming weight at most t occur. We transmit a codeword $\boldsymbol{C}$ of an s-interleaved code $\mathcal{IC}^{(s)}$. The received word is given by

$$\boldsymbol{R} = \boldsymbol{C} + \widetilde{\boldsymbol{E}} \in \mathbb{F}_q^{s\times n} ,$$

where each row of $\boldsymbol{C} \in \mathbb{F}_q^{s\times n}$ is a codeword of $\mathcal{C}$ and $\widetilde{\boldsymbol{E}} \in \mathbb{F}_q^{s\times n}$ has at most t nonzero columns. An illustration of a corrupted codeword of $\mathcal{IC}$ is given in Fig. 6.1.

In the following let $\mathcal{C}$ be a (generalized) RS code $\mathsf{RS}_q[n,k]$ with nonzero code locators $\alpha_1,\dots,\alpha_n$. The joint decoding algorithm that we present is also known as a *syndrome-based*

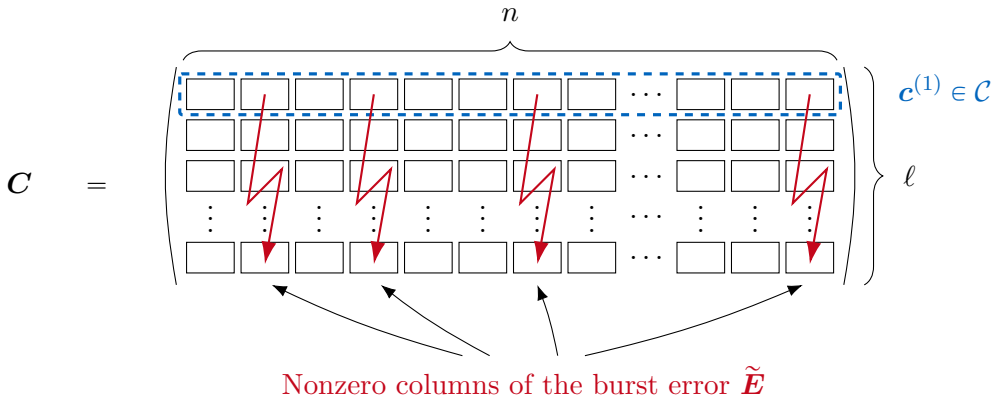


Figure 6.1: An illustration of a corrupted codeword of an s-code by a burst error $\widetilde{\boldsymbol{E}}$.

joint decoding algorithm for interleaved RS codes. Such algorithms, to name a few, can be found in [FT91] for BCH codes and [KL97; BKY03; SSB09] for interleaved RS code. We briefly recapitulate the decoding method below and summarize a naive version of [SSB09, Algorithm 2] in Algorithm 6.1.

Let $\boldsymbol{H}$ be a parity-check matrix of the constituent code $\mathsf{RS}_q[n,k]$ and $d = n-k+1$. From the received matrix $\boldsymbol{R}$, we are able to calculate the syndromes of each row of $\boldsymbol{R}$ by

$$\begin{pmatrix} \boldsymbol{s}_1 \\ \boldsymbol{s}_2 \\ \vdots \\ \boldsymbol{s}_s \end{pmatrix} = \boldsymbol{R} \cdot \boldsymbol{H}^\top = \widetilde{\boldsymbol{E}} \cdot \boldsymbol{H}^\top \,, \tag{6.1}$$

where $\boldsymbol{s}_i = (s_{i,1}, \ldots, s_{i,d-1}) \in \mathbb{F}_{q^m}^{d-1}$, for each $i \in [s]$.

Assuming that there are exactly t nonzero columns in $\widetilde{\boldsymbol{E}}$, we define the *error locator polynomial* as[1]

$$\Lambda(x) := \prod_{i=1}^{t}(1 - \alpha_{j_i}^{-1} x) = 1 + \Lambda_1 x + \cdots + \Lambda_t x^t \,, \tag{6.2}$$

where the t roots $\alpha_{j_1}, \ldots, \alpha_{j_t}$ of $\Lambda(x)$ are the code locators corresponding to the error positions. The coefficients of $\Lambda(x)$ fulfill the following linear equations (cf. [Pet60]),

$$\underbrace{\begin{pmatrix} s_{i,1} & s_{i,2} & \cdots & s_{i,t} \\ s_{i,2} & s_{i,3} & \cdots & s_{i,t+1} \\ \vdots & \vdots & & \vdots \\ s_{i,d-1-t} & s_{i,d-1-t+1} & \cdots & s_{i,d-2} \end{pmatrix}}_{\boldsymbol{S}^{(i)}(t)} \begin{pmatrix} \Lambda_t \\ \Lambda_{t-1} \\ \vdots \\ \Lambda_1 \end{pmatrix} = \underbrace{\begin{pmatrix} -s_{i,t+1} \\ -s_{i,t+2} \\ \vdots \\ -s_{i,d-1} \end{pmatrix}}_{\boldsymbol{T}^{(i)}(t)}, \quad \forall i \in [s] \,. \tag{6.3}$$

Thus, determining the error positions in $\widetilde{\boldsymbol{E}}$ is equivalent to solving the following linear system

[1] Since $\alpha_i \neq 0$, the error locator polynomial is well-defined.

of equations $\mathfrak{S}(t)$ for t unknowns,

$$\underbrace{\begin{pmatrix} \boldsymbol{S}^{(1)}(t) \\ \boldsymbol{S}^{(2)}(t) \\ \vdots \\ \boldsymbol{S}^{(s)}(t) \end{pmatrix}}_{\boldsymbol{S}(t)} \underbrace{\begin{pmatrix} \Lambda_t \\ \Lambda_{t-1} \\ \vdots \\ \Lambda_1 \end{pmatrix}}_{\boldsymbol{\Lambda}} = \underbrace{\begin{pmatrix} \boldsymbol{T}^{(1)}(t) \\ \boldsymbol{T}^{(2)}(t) \\ \vdots \\ \boldsymbol{T}^{(s)}(t) \end{pmatrix}}_{\boldsymbol{T}(t)} . \tag{6.4}$$

After determining $\boldsymbol{\Lambda}$ from (6.4), we may use a standard method for error evaluation such as *Forney's algorithm* [For65] (cf. [Rot06, Section 6.6]) to calculate the error values $\hat{\boldsymbol{E}}$. Then, by subtracting the calculated error $\hat{\boldsymbol{E}}$ from $\boldsymbol{R}$, we obtain the estimated codeword $\hat{\boldsymbol{C}} = \boldsymbol{R} - \hat{\boldsymbol{E}}$.

Algorithm 6.1: Syndrome-based Joint Decoding Algorithm

Input: received word $\boldsymbol{R}$
Output: $\hat{\boldsymbol{C}}$ or `decoding failure`

1 Calculate the syndromes $\boldsymbol{s}_{i,:}, \forall i \in [s]$ `// See` (6.1)
2 **if** $\boldsymbol{s}_{i,:} = \boldsymbol{0}$ *for all* i **then return** $\hat{\boldsymbol{C}} = \boldsymbol{R}$
3 Find minimal $t^\star$ for which $\boldsymbol{S}(t^\star) \cdot \boldsymbol{\Lambda}^\star = \boldsymbol{T}(t^\star)$ has a solution $\boldsymbol{\Lambda}^\star$ `// See` (6.4)
4 **if** the solution $\boldsymbol{\Lambda}^\star$ is not unique **then return** `decoding failure`
5 **if** $\Lambda^\star(x)$ has $t^\star$ *distinct* roots in $\mathbb{F}_{q^m}$ **then**
6 | Evaluate the errors $\hat{\boldsymbol{E}}$ by Forney's algorithm [For65][Rot06, Section 6.6]
7 | **return** $\hat{\boldsymbol{C}} = \boldsymbol{R} - \hat{\boldsymbol{E}}$
8 **else**
9 | **return** `decoding failure`
10 **end**

For the channel where the burst errors occur, the joint decoding algorithm given in Algorithm 6.1 may yield three different results:

- The algorithm returns the correct result, i.e., $\hat{\boldsymbol{C}} = \boldsymbol{C}$, with *success probability* P_{suc}.
- The algorithm returns an erroneous result, i.e., $\hat{\boldsymbol{C}} \neq \boldsymbol{C}$, with *miscorrection probability* P_{misc}.
- The algorithm returns a `decoding failure`, with *failure probability* P_{fail}.

Remark 6.1 (Practical Implementations)**.** *Algorithm 6.1 is a naive approach. It is mainly meant for the proof of the successful probability, instead of for an efficient implementation.*

For practical implementations, one can use some fast algorithm for Line 3, for instance, 1) [SS11, Algorithm 3] with the complexity of $O(sd^2)$ operations in $\mathbb{F}_{q^m}$, 2) the currently fastest algorithm [RS21] with complexity $O^{\sim}(s^{\omega-1}d)$ where $O^{\sim}$ omits the log*-factors in d and ω is the matrix multiplication exponent, for which the best algorithm allow $\omega < 2.38$ [CW90; LG14].*

Algorithm 6.1 yields a *bounded distance* decoder which can decode beyond half of the minimum distance with high probability. Clearly, the solution $\boldsymbol{\Lambda}^\star$ cannot be unique if the number of equations in (6.4) is less than the number of unknowns. Thus, the following maximum decoding radius of Algorithm 6.1 can be derived.

Theorem 6.1 ([SSB09, Theorem 3])**.** *Let $\mathcal{IC}^{(s)}$ be an s-interleaved code with $\mathcal{C} = \mathsf{RS}_q[n,k]$. For a received word $\boldsymbol{R} = \boldsymbol{C} + \widetilde{\boldsymbol{E}}$, where $\boldsymbol{C} \in \mathcal{IC}^{(s)}$ and the error $\widetilde{\boldsymbol{E}}$ has t nonzero columns, Algorithm 6.1 may only succeed, i.e., return $\hat{\boldsymbol{C}} = \boldsymbol{C}$, if*

$$t \leqslant t_{\max,\mathsf{RS}} := \frac{s}{s+1}(d-1) \ . \tag{6.5}$$

By the nature of a bounded distance decoder, where the correction balls of each codeword inevitably overlap for some error patterns of weight $t > \left\lfloor \frac{\mathrm{d_H}-1}{2} \right\rfloor$, Algorithm 6.1 is unsuccessful with some probability when $t > \left\lfloor \frac{\mathrm{d_H}-1}{2} \right\rfloor$. The following lemma gives a necessary and sufficient condition such that Algorithm 6.1 is unsuccessful, i.e., returning an erroneous result or a `decoding failure`. This will be the foundation to bound the success probability of Algorithm 6.1 in decoding interleaved alternant codes in Section 6.2. The sufficiency has been shown in the proof of [SSB09, Lemma 2]. The proof below completes the necessity.

Lemma 6.1 (Necessary and sufficient condition on unsuccessful decoding)**.** *Let $\mathcal{IC}^{(s)}$ be an s-interleaved code with $\mathcal{C} = \mathsf{RS}_q[n,k]$. For a received word $\boldsymbol{R} = \boldsymbol{C} + \widetilde{\boldsymbol{E}}$, where $\boldsymbol{C} \in \mathcal{IC}^{(s)}$ and the error $\widetilde{\boldsymbol{E}}$ has $t > 0$ nonzero columns, Algorithm 6.1 is not* successful*, i.e., returns $\hat{\boldsymbol{C}} \neq \boldsymbol{C}$ or a* `decoding failure`*, if and only if* $\mathrm{rank}(\boldsymbol{S}(t)) < t$.

Proof. Denote by $\Lambda(x)$ the *true* error locator polynomial corresponding to the t error positions (indices of nonzero columns) in $\widetilde{\boldsymbol{E}}$. Then $\Lambda(x)$ has t distinct roots in $\mathbb{F}_{q^m}$ and $\boldsymbol{\Lambda}$ is a solution of the linear system of equations $\mathfrak{S}(t)$ as in (6.4).

Sufficiency: We show that $\mathrm{rank}(\boldsymbol{S}(t)) < t$ implies unsuccessful decoding. Consider two cases, $\mathrm{rank}(\boldsymbol{S}(t)) = 0$ and $0 < \mathrm{rank}(\boldsymbol{S}(t)) < t$. For $\mathrm{rank}(\boldsymbol{S}(t)) = 0$, the algorithm outputs $\hat{\boldsymbol{C}} = \boldsymbol{R}$ at Line 2. However, since $t > 0$, this is apparently a miscorrection. For the latter case, assume $\mathrm{rank}(\boldsymbol{S}(t)) = t^\star$, which is found at Line 3. Then $t^\star < t$ by assumption. Suppose the algorithm runs until Line 7, then $\mathrm{wt_H}(\hat{\boldsymbol{E}}) = t^\star < t = \mathrm{wt_H}(\widetilde{\boldsymbol{E}})$ and hence $\hat{\boldsymbol{E}} \neq \widetilde{\boldsymbol{E}}$. Then the resulting $\hat{\boldsymbol{C}}$ is not the sent codeword $\boldsymbol{C}$. Other termination (Line 4 or Line 9) of the algorithm results in a `decoding failure`.

Necessity: We show that unsuccessful decoding implies $\mathrm{rank}(\boldsymbol{S}(t)) < t$. The algorithm returns `decoding failure` only on Line 4 or 9. Line 3 determines the *minimal* $t^\star$ such that $\mathfrak{S}(t^\star)$ has at least one solution $\boldsymbol{\Lambda}^\star$, hence $t^\star \leqslant t$. Note that a solution to $\mathfrak{S}(t^\star)$ is also a solution to $\mathfrak{S}(t)$. If the algorithm fails on Line 4, i.e., the system $\mathfrak{S}(t^\star)$ has many distinct solutions, then $\mathfrak{S}(t)$ also has many solutions and therefore $\mathrm{rank}(\boldsymbol{S}(t)) < t$. The failure occurs on Line 9 if $\Lambda^\star(x)$ does not have $t^\star$ different roots, which implies $\Lambda^\star(x) \neq \Lambda(x)$. This means that the system $\mathfrak{S}(t)$ has at least two solutions $\boldsymbol{\Lambda}$ and $\boldsymbol{\Lambda}^\star$. Hence $\mathrm{rank}(\boldsymbol{S}(t)) < t$.

The algorithm returns a miscorrected codeword only at Line 2 or 7. If the decoder outputs $\hat{\boldsymbol{C}}$ on Line 2, we have $\hat{\boldsymbol{C}} \neq \boldsymbol{C}$ as $t > 0$. In this case $\boldsymbol{S}(t) = \boldsymbol{0}$, so $\mathrm{rank}(\boldsymbol{S}(t)) = 0 < t$. Note that $\boldsymbol{S}(t) = \boldsymbol{0}$ will not occur if $0 < t < \mathrm{d_H}(\mathcal{C})$, since the error $\widetilde{\boldsymbol{E}}$ with $\mathrm{wt_H}(\widetilde{\boldsymbol{E}}) = t$ cannot result in $\boldsymbol{R}$ being another codeword in $\mathcal{C}$. Assume the algorithm runs to Line 7 and returns a $\hat{\boldsymbol{C}} \neq \boldsymbol{C}$. This implies that either $t^\star < t$ or, $t^\star = t$ and Line 3 has two distinct solutions $\Lambda^\star$. The former directly implies $\mathrm{rank}(\boldsymbol{S}(t)) < t$ and the latter will cause a `decoding failure` at Line 4, contradicting the assumption. □

Remark 6.2. *In the proof of [SSB09, Theorem 7],* $\Pr[\mathrm{rank}(\boldsymbol{S}(t)) < t]$ *is used to bound the probability of a* `decoding failure`*. Since we have shown in Lemma 6.1 that* $\mathrm{rank}(\boldsymbol{S}(t)) < t$ *is not only a sufficient condition but also a necessary condition for an unsuccessful decoding*

(either a miscorrection or a `decoding failure`*), the upper bound [SSB09, Theorem 7] is in fact an upper bound on* $1 - P_{\mathsf{suc}} = P_{\mathsf{misc}} + P_{\mathsf{fail}}$.

6.2 Joint Decoding of Interleaved Alternant Codes

GRS codes are generalizations of Reed-Solomon codes $\mathsf{RS}_q[n, k]$ with nonzero *column multipliers. Alternant code* are *subfield subcodes* of GRS codes. This code family contains some of the best-known and most-often used algebraic codes over small fields, including the BCH [Hoc59; BR60] and the Goppa codes [Gop70; Ber73; SKHN76].

6.2.1 Generalized Reed-Solomon Codes and Alternant Codes

We begin by formally defining the GRS codes and the alternant codes, then discuss some applications and special cases of alternant codes.

Definition 6.2 (Generalized Reed-Solomon codes)**.** *For positive integers d and n, let $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_n) \in (\mathbb{F}_{q^m}^*)^n$ be a vector of distinct* code locators *and $\boldsymbol{v} \in (\mathbb{F}_{q^m}^*)^n$ be a vector of* column multipliers*. A* generalized Reed-Solomon *(GRS) code $\mathsf{GRS}_{\boldsymbol{\alpha},\boldsymbol{v}}^d$ of length $n = |\boldsymbol{\alpha}|$, dimension $k = n - d + 1$ and minimum Hamming distance d is defined as*[2]

$$\mathsf{GRS}_{\boldsymbol{\alpha},\boldsymbol{v}}^d = \{\boldsymbol{c} \in \mathbb{F}_{q^m}^n \mid \boldsymbol{H} \cdot \mathrm{diag}(\boldsymbol{v}) \cdot \boldsymbol{c} = \boldsymbol{0}\} \ ,$$

with

$$\boldsymbol{H} = \begin{pmatrix} 1 & 1 & \ldots & 1 \\ \alpha_1 & \alpha_2 & \ldots & \alpha_n \\ \vdots & \vdots & & \vdots \\ \alpha_1^{d-2} & \alpha_2^{d-2} & \ldots & \alpha_n^{d-2} \end{pmatrix} \in \mathbb{F}_{q^m}^{(d-1)\times n} \ .$$

For a fixed $\boldsymbol{\alpha}$, denote by $\mathbb{G}_{\boldsymbol{\alpha}}^d$ the multiset of GRS codes with different column multipliers, i.e.,

$$\mathbb{G}_{\boldsymbol{\alpha}}^d := \{\{\mathsf{GRS}_{\boldsymbol{\alpha},\boldsymbol{v}}^d \mid \boldsymbol{v} \in (\mathbb{F}_{q^m}^*)^n\}\} \ .$$

Note that the most general definitions of GRS codes allow for the $\alpha_i = 0$ to be element of $\boldsymbol{\alpha}$, but for consistency with [SSB09] and as this complicates the decoding process, we restrict ourselves to $\alpha_i \neq 0$ here. GRS codes are well-known to be MDS codes, i.e., they achieve $\mathrm{d}_{\mathsf{H}} = n - k + 1$, where k is the dimension of the code.

By design, GRS codes must be defined over finite fields $\mathbb{F}_{q^m}$ with $q^m - 1 \geqslant n$ (or $q^m \geqslant n$ if $\alpha_i = 0$ is allowed as a code locator). In many applications it is desirable to work with codes of smaller field size, which can be obtained, e.g., by taking subfield subcodes of codes with good minimum distance.

Definition 6.3 (Subfield subcode)**.** *Let $\mathcal{C}$ be an $[n, k]_{q^m}$ code. We define the $\mathbb{F}_q$-subfield subcode of $\mathcal{C}$ as*

$$\mathcal{C} \cap \mathbb{F}_q^n = \{\boldsymbol{c} \in \mathbb{F}_q^n \mid \boldsymbol{c} \in \mathcal{C}\} \ .$$

[2] The results in this section are mostly dependent the minimum Hamming distance, code locators and column multipliers of GRS codes. Hence, we use the notation depend on d, $\boldsymbol{\alpha}$ and $\boldsymbol{v}$.

Equivalently, let $\boldsymbol{H} \in \mathbb{F}_{q^m}^{(n-k)\times n}$ *be a parity-check matrix of* $\mathcal{C}$*. Then* $\mathcal{C} \cap \mathbb{F}_q^n$ *is given by the* $\mathbb{F}_q$*-kernel of* $\boldsymbol{H}$*, i.e.,*

$$\mathcal{C} \cap \mathbb{F}_q^n = \{\boldsymbol{c} \in \mathbb{F}_q^n \mid \boldsymbol{H} \cdot \boldsymbol{c} = \boldsymbol{0}\} \ .$$

The subfield subcode of a GRS code is referred to as an *alternant code* [MS77, Ch. 12.2]. For a fixed $\boldsymbol{\alpha}$ and a *designed* distance d, we denote by $\mathbb{A}_{\boldsymbol{\alpha}}^d$ the multiset of alternant codes, i.e.,

$$\mathbb{A}_{\boldsymbol{\alpha}}^d := \{\{\mathcal{C} \cap \mathbb{F}_q^n \mid \mathcal{C} \in \mathbb{G}_{\boldsymbol{\alpha}}^d\}\} \ . \tag{6.6}$$

We define $\mathbb{A}_{\boldsymbol{\alpha}}^d$ as a multiset, as the multiplicities will be important for the bounds on the probability of successful decoding. An additional advantage is that for a given $\boldsymbol{\alpha}$ of length $n = |\boldsymbol{\alpha}|$, we know the cardinality of $\mathbb{A}_{\boldsymbol{\alpha}}^d$ is

$$|\mathbb{A}_{\boldsymbol{\alpha}}^d| = (q^m - 1)^n \ . \tag{6.7}$$

For GRS codes it is known (cf. [Del75]) that for a fixed vector $\boldsymbol{\alpha}$ of code locators, it holds that $\mathsf{GRS}_{\boldsymbol{\alpha},\boldsymbol{v}}^d = \mathsf{GRS}_{\boldsymbol{\alpha},\boldsymbol{u}}^d$ if and only if $\boldsymbol{v}$ is an $\mathbb{F}_{q^m}$-multiple of $\boldsymbol{u}$, i.e., any code $\mathcal{C} \in \mathbb{G}_{\boldsymbol{\alpha}}^d$ occurs with multiplicity exactly $\delta_{\mathbb{G}_{\boldsymbol{\alpha}}^d}^{\mathcal{C}} = q^m - 1$ in $\mathbb{G}_{\boldsymbol{\alpha}}^d$. This gives a lower bound on the multiplicity of alternant codes as

$$\delta_{\mathbb{A}_{\boldsymbol{\alpha}}^d}^{\mathcal{A}} \geqslant q^m - 1, \ \forall \mathcal{A} \in \mathbb{A}_{\boldsymbol{\alpha}}^d \ . \tag{6.8}$$

We give some general well-known bounds on the dimension of the $\mathbb{F}_q$-subcode of an $\mathbb{F}_{q^m}$-linear code $\mathcal{C}$ in terms of the parameters of $\mathcal{C}$.

Lemma 6.2. *Let* $\mathcal{C}$ *be an* $[n,k]_{q^m}$ *code with minimum Hamming distance* d*. Then,*

$$\max\{n - m(n-k), 0\} \leqslant \dim_q(\mathcal{C} \cap \mathbb{F}_q^n) \leqslant \min\{k, k_q^{\mathsf{opt.}}(n,d)\} \ ,$$

where $k_q^{\mathsf{opt.}}(n,d)$ *is an upper bound on the dimension of a* q*-ary linear code given the length* n *and the minimum Hamming distance* d *(e.g., Singleton, Griesmer, Hamming bounds, etc.)*

Proof. The lower bound 0 is trivial. The lower bound $n - m(n-k)$ follows from expanding the $n-k$ rows of any parity-check matrix of $\mathcal{C}$ via some basis of $\mathbb{F}_{q^m}$ over $\mathbb{F}_q$. The resulting $m(n-k) \times n$ matrix is a parity-check matrix of the $\mathbb{F}_q$-subcode of $\mathcal{C} \cap \mathbb{F}_q^n$ and the bound follows. The upper bound k is trivial because $\dim_q(\mathcal{C} \cap \mathbb{F}_q^n) \leqslant \dim_q(\mathcal{C}) = k$ and $k_q^{\mathsf{opt.}}(n,d)$ is an upper bound by definition. □

Other Applications of Alternant Codes

In principle, alternant codes can be used as constituent codes in any of the applications of interleaved codes mentioned at the beginning of this chapter. Several concrete reasons to specifically consider interleaved alternant codes are also worthy mentioning:

- Alternant codes (especially BCH codes) are widely used in practice, including data storage and communications. Any system that already uses these codes and is prone to burst errors may be retroactively upgraded to enable a larger error-correction capability. For instance, in NOR and NAND flash memory, Hamming and BCH codes are considered as the standard error-correction approach [LRS06; CLS09; WDPZ11]. Traditionally,

Hamming codes are used in single-level flash memories to correct single errors as they have a simple decoding algorithm and use only a small circuit area. For multi-level flash memories, however, single-error correction is not sufficient and BCH codes with larger distance are employed. In [SRZ06], the scenario of more than four levels (i.e., storing more than two bits per flash memory cell) was investigated and it was shown that BCH codes of larger correction capability are needed. To address the fact that errors in flash memories might occur over whole bit or word lines, in [YEC11] product codes with BCH codes were used. This motivates the use of *interleaved* alternant and in particular interleaved BCH codes.

- In applications where the cost of *encoding* is dominant (e.g., in storage systems where writing occurs more often than reading an erroneous codeword), encoding in a subfield reduces the complexity. Hence, it might be advantageous to use alternant codes instead of GRS codes in some of the above mentioned applications of interleaved codes. Note that *decoding* is usually done in the field of the corresponding GRS code, so the reduction in complexity is less significant.

- In some applications, such as code-based cryptography, GRS and algebraic-geometry codes cannot be used due to their vast structure, which can be turned into structural attacks on the cryptosystem. However, their subfield subcodes are in many cases unbroken, e.g., see in [CMP17, Conclusion] and [CR20, Section 7.5.3]. In particular, the codes proposed in McEliece's original paper [McE78], binary Goppa codes, have withstood efficient attacks for more than 40 years. In a McEliece-type system, the ciphertext is the sum of a codeword of a public code and a randomly chosen "error" which hides the codeword from the attacker. If we encrypt multiple codewords in parallel, we may consider them as an interleaved code and align the errors in bursts of larger weight. This approach has the potential to increase the designed security parameter, or in turn reduce the key size [EWZ18; HLPW19]. This comes at the cost of a (hopefully very small) probability of unsuccessful decryption/decoding, which corresponds to the probability of unsuccessful decoding of the interleaved decoder.

Dimension vs. Hamming Distance of Binary BCH and Wild Goppa Codes

Wild Goppa codes [SKHN76; Wir88], which include *binary square-free Goppa codes* [Gop70; Gop71; Ber73], are a subclass of Goppa Codes. Along with BCH codes [Hoc59; BR60], Goppa codes are the best known class of alternant codes, due to their good distance properties in the Hamming metric. Binary BCH and q-ary wild Goppa codes have been shown to be subfield subcodes of GRS codes in $\mathbb{G}_{\boldsymbol{\alpha}}^{d}$ for some $\boldsymbol{\alpha}$ and d.

Consider a binary BCH code that is a subfield subcode of some GRS code in $\mathbb{G}_{\boldsymbol{\alpha}}^{d}$ with length $n = |\alpha|$ and dimension $k = n - d + 1$ over $\mathbb{F}_{2^m}$. It is well-known (cf. [MS77, Ch. 7]) that the dimension of the binary BCH code is $k_{\mathsf{BCH}} \geqslant n - m\frac{n-k}{2}$, which exceeds the generic lower bound in Lemma 6.2.

Wild Goppa codes are often considered as subclasses of alternant codes of $\mathbb{A}_{\boldsymbol{\alpha}}^{d}$, but with an increased lower bound on the distance $d_{\mathsf{Goppa}} \geqslant \frac{q}{q-1}d$. However, the bounds on the probability of decoding success that are studied in Section 6.3 depend only on the properties of the corresponding GRS and, in particular, its distance d, but not on the *actual* dimension or distance of the alternant code itself. Therefore, instead of viewing wild Goppa codes as alternant codes in $\mathbb{A}_{\boldsymbol{\alpha}}^{d}$ with increased distance, it is convenient to view them as alternant

codes of $\mathbb{A}_{\boldsymbol{\alpha}}^{d_{\mathsf{Goppa}}}$ with a larger *dimension* than guaranteed by the lower bound in Lemma 6.2. This is possible because the general improvements of wild Goppa codes compared to alternant codes were shown by an equivalence between the Goppa codes obtained from different Goppa polynomials (cf. [SKHN76], [BLP11, Theorem 4.1]). In other words, the wild Goppa code is in $\mathbb{A}_{\boldsymbol{\alpha}}^{d} \cap \mathbb{A}_{\boldsymbol{\alpha}}^{d_{\mathsf{Goppa}}}$. It can be shown that the wild Goppa codes with distance d have an increased lower bound on the dimension compared to the generic lower bound in Lemma 6.2, i.e., $k_{\mathsf{Goppa}} \geqslant n - m\frac{q-1}{q}(d-1) = n - m\frac{q-1}{q}(n-k)$.

6.2.2 Condition on Successful Decoding of Interleaved Alternant Codes

The joint decoder given in Algorithm 6.1 immediately applies to interleaved alternant codes as well, i.e., the subfield subcodes of interleaved Reed–Solomon codes, but the fraction of decodable error matrices differs, since the error is now over the subfield. Due to this, the bounds on the probability of unsuccessful decoding of interleaved GRS codes does not hold for interleaved alternant codes, which has been shown in the simulation results in [HLPW19].

With the help of Lemma 6.1, we now present the crux in bounding the success probability of decoding interleaved alternant codes by Algorithm 6.1, which is the basis of the bounds presented in Section 6.3.

In the rest of this chapter, we denote by $\mathbb{E}_q^{(a,b)}$ the set of matrices $\boldsymbol{E}$ without zero columns in $\mathbb{F}_q^{a\times b}$.

Lemma 6.3. *Let $\mathcal{IC}^{(s)}$ be an s-interleaved alternant code with $\mathcal{C} \in \mathbb{A}_{\boldsymbol{\alpha}}^{d}$, $n = |\boldsymbol{\alpha}|$ and $\mathcal{E} = \{j_1, j_2, \ldots, j_t\} \subset [n]$ be a set of $|\mathcal{E}| = t$ error positions. For a codeword $\boldsymbol{C} \in \mathcal{IC}^{(s)}$, an error matrix $\widetilde{\boldsymbol{E}} \in \mathbb{F}_q^{s\times n}$ with $\operatorname{supp}(\widetilde{\boldsymbol{E}}) = \mathcal{E}$ and $\boldsymbol{E} := \widetilde{\boldsymbol{E}}|_{\mathcal{E}} \in \mathbb{E}_q^{(s,t)}$, and a received word $\boldsymbol{R} = \boldsymbol{C} + \widetilde{\boldsymbol{E}}$, Algorithm 6.1* succeeds*, i.e., returns $\hat{\boldsymbol{C}} = \boldsymbol{C}$, if and only if*

$$\nexists \boldsymbol{v} \in \mathbb{F}_{q^m}^t \backslash \{\boldsymbol{0}\} \text{ such that } \boldsymbol{H} \cdot \operatorname{diag}(\boldsymbol{v}) \cdot \boldsymbol{E}^\top = \boldsymbol{0} \ , \tag{6.9}$$

where $\boldsymbol{H} \in \mathbb{F}_{q^m}^{d-t-1\times t}$ is a parity-check matrix of the $[t, 2t-d+1, d-t]_{q^m}$ code $\mathsf{GRS}_{\boldsymbol{\alpha}|_{\mathcal{E}},\boldsymbol{1}}^{d-t}$.

Proof. We extend and adapt the proof for interleaved GRS codes from [SSB09].

According to Lemma 6.1, Algorithm 6.1 yields a `decoding failure` or a miscorrection $\hat{\boldsymbol{C}} \neq \boldsymbol{C}$ if and only if $\operatorname{rank}(\boldsymbol{S}(t)) < t$, with $\boldsymbol{S}(t)$ as in (6.4). In other words, the decoding may only be unsuccessful, if there exists a nonzero vector $\boldsymbol{u} \in \mathbb{F}_{q^m}^t$ such that $\boldsymbol{S}(t) \cdot \boldsymbol{u} = \boldsymbol{0}$, i.e.,

$$\exists \boldsymbol{u} \in \mathbb{F}_{q^m}^t \backslash \{\boldsymbol{0}\} \text{ such that } \boldsymbol{S}^{(i)}(t) \cdot \boldsymbol{u} = \boldsymbol{0} \ , \ \forall i \in [s] \ . \tag{6.10}$$

It is known (cf. [PW72, Theorem 9.9][SSB09]) that a syndrome matrix $\boldsymbol{S}^{(i)}(t)$ can be decomposed into

$$\boldsymbol{S}^{(i)}(t) = \boldsymbol{H} \cdot \boldsymbol{F}^{(i)} \cdot \boldsymbol{D} \cdot \boldsymbol{V} \ ,$$

where $\boldsymbol{H}$ is a parity-check matrix of the $[t, 2t-d+1, d-t]_{q^m}$ code $\mathsf{GRS}^{d-t}_{\boldsymbol{\alpha}|_{\mathcal{E}},\mathbf{1}}$ as in Definition 6.2,

$$\boldsymbol{V} = \begin{pmatrix} 1 & 1 & \dots & 1 \\ \alpha_{j_1} & \alpha_{j_2} & \dots & \alpha_{j_t} \\ \alpha_{j_1}^2 & \alpha_{j_2}^2 & \dots & \alpha_{j_t}^2 \\ \vdots & \vdots & & \vdots \\ \alpha_{j_1}^{t-1} & \alpha_{j_2}^{t-1} & \dots & \alpha_{j_t}^{t-1} \end{pmatrix}^\top \in \mathbb{F}_{q^m}^{t\times t} ,$$
$$\boldsymbol{F}^{(i)} = \operatorname{diag}(\boldsymbol{e}_i) \in \mathbb{F}_q^{t\times t} ,$$
$$\boldsymbol{D} = \operatorname{diag}(\boldsymbol{v}'|_{\mathcal{E}}) \in \mathbb{F}_{q^m}^{t\times t} ,$$

and $\boldsymbol{v}' \in \left(\mathbb{F}_{q^m}^*\right)^n$ is the vector of column multipliers of the GRS code corresponding to the alternant code $\mathcal{C}$, i.e., $\mathsf{GRS}^d_{\boldsymbol{\alpha},\boldsymbol{v}'} \cap \mathbb{F}_q = \mathcal{C}$.

We observe that the matrices $\boldsymbol{D}$ and $\boldsymbol{V}$ are both square and of full rank. Therefore, the product $\boldsymbol{v} := \boldsymbol{D}\cdot\boldsymbol{V}\cdot\boldsymbol{u}$ defines a one-to-one mapping $\boldsymbol{u} \to \boldsymbol{v}$, such that $\mathbf{0} \to \mathbf{0}$. Consequently, the statement (6.10) is equivalent to

$$\exists \boldsymbol{v} \in \mathbb{F}_{q^m}^t \backslash \{\mathbf{0}\} \text{ such that } \boldsymbol{H}\cdot\operatorname{diag}(\boldsymbol{e}_i)\cdot\boldsymbol{v} = \mathbf{0} \ , \ \forall i \in [s]$$
$$\Updownarrow$$
$$\exists \boldsymbol{v} \in \mathbb{F}_{q^m}^t \backslash \{\mathbf{0}\} \text{ such that } \boldsymbol{H}\cdot\operatorname{diag}(\boldsymbol{v})\cdot\boldsymbol{e}_i = \mathbf{0} \ , \ \forall i \in [s] ,$$

and the statement follows. □

Note that the upper bound on the probability of unsuccessful decoding interleaved GRS codes from [SSB09] applies to error matrices $\widetilde{\boldsymbol{E}}$ over $\mathbb{F}_{q^m}$ (the field of the GRS code). However, for interleaved alternant codes, $\widetilde{\boldsymbol{E}}$ is over $\mathbb{F}_q$ (the *subfield* of RS codes) and the bound from [SSB09] is not valid in this case.

6.3 Bounds on Success Probability of Decoding Interleaved Alternant Codes

In this section we present lower and upper bounds on the probability of successful decoding of interleaved alternant codes by Algorithm 6.1. Lemma 6.3 gives a necessary and sufficient condition for Algorithm 6.1 to succeed for an error $\widetilde{\boldsymbol{E}}$ with fixed $\mathcal{E} = \operatorname{supp}(\widetilde{\boldsymbol{E}})$ and $\widetilde{\boldsymbol{E}}|_{\mathcal{E}} \in \mathbb{E}_q^{(s,t)}$. With this as the basis, we bound the probability of successful decoding for a random error matrix $\widetilde{\boldsymbol{E}}$ where $\widetilde{\boldsymbol{E}}|_{\mathcal{E}}$ is i.i.d. in $\mathbb{E}_q^{(s,t)}$.

6.3.1 Technical Preliminary Results

Before deriving the bounds, we establish some technical preliminary results which are needed to prove the bounds.

Maximization of Integer Distributions

To begin, we derive a simple upper bound on the maximization of a sum of integer powers, under a restriction on the base of the power.

Definition 6.4 (Majorization relation)**.** *Let $\mathcal{M} = \{\{m_1, m_2, \ldots, m_c\}\}$ and $\mathcal{K} = \{\{k_1, k_2, \ldots, k_c\}\}$ be two (finite) multisets of real numbers with the same cardinality. We say that the set $\mathcal{M}$* majorizes *the set $\mathcal{K}$ and write*

$$\mathcal{M} \succ \mathcal{K} \quad \text{or} \quad \mathcal{K} \prec \mathcal{M}$$

if, after a possible renumeration, $\mathcal{M}$ and $\mathcal{K}$ satisfy the following conditions:

(1) $m_1 \geqslant m_2 \geqslant \cdots \geqslant m_c$ *and* $k_1 \geqslant k_2 \geqslant \cdots \geqslant k_c$*;*

(2) $\sum_{i=1}^{j} m_i \geqslant \sum_{i=1}^{j} k_i, \ \forall\, 1 \leqslant j \leqslant c.$

We recap the following well-known result on multisets with this majorization relation.

Lemma 6.4 (Karamata's inequality [KDLM05, Theorem 1])**.** *Let $\mathcal{M} = \{\{m_1, m_2, \ldots, m_c\}\}$ and $\mathcal{K} = \{\{k_1, k_2, \ldots, k_c\}\}$ be two multisets of real numbers from an interval $[a, b]$. If the set $\mathcal{M} \succ \mathcal{K}$, and if $f : \mathbb{R} \to \mathbb{R}$ is a convex and non-decreasing function in the range $[a, b]$, then it holds that*

$$\sum_{i=1}^{c} f(m_i) \geqslant \sum_{i=1}^{c} f(k_i) \ . \tag{6.11}$$

For convenience of notation, we define a fixed notation for the set over which we maximize in the following.

Definition 6.5. *Denote by $\mathbb{M}_{c,B}^{[a,b]} = \{\mathcal{M}, \ldots\}$ the set of all multisets $\mathcal{M} = \{\{m_1, \ldots, m_c\}\}$ of cardinality c with $b \geqslant m_1 \geqslant \ldots \geqslant m_c \geqslant a$ and $\sum_{m \in \mathcal{M}} m = B$.*

With these definitions established, we are now ready to give an upper bound on the sum over the results of a convex non-decreasing function evaluated on the elements of any multiset in $\mathbb{M}_{c,B}^{[a,b]}$.

Lemma 6.5. *Let $a, c \geqslant 1$, $b \geqslant a$, $ca \leqslant B \leqslant cb$, and $\mathbb{M}_{c,B}^{[a,b]}$ be as in Definition 6.5. For any function $f(x)$ that is convex and non-decreasing in the interval $a \leqslant x \leqslant b$, it holds that*

$$\max_{\mathcal{M} \in \mathbb{M}_{c,B}^{[a,b]}} \sum_{m \in \mathcal{M}} f(M) \leqslant \left(\frac{B - ca}{b - a} + 1 \right) (f(b) - f(a)) + c f(a) \ .$$

Proof. Let $\delta_{\mathcal{M}}^{m}$ be the multiplicity of an element $m \in \mathcal{M}$. Denote by $\widetilde{\mathcal{M}}$ the set of distinct elements in $\mathcal{M}$. By definition,

$$\sum_{M \in \mathcal{M}} M = \sum_{m \in \widetilde{\mathcal{M}}} \delta_{\mathcal{M}}^{m} \cdot m = B, \ \forall \mathcal{M} \in \mathbb{M}_{c,B}^{[a,b]}$$

and it follows that for all $\mathcal{M} \in \mathbb{M}_{c,B}^{[a,b]}$ we have

$$\delta_{\mathcal{M}}^{b} = \frac{1}{b} \left(B - \sum_{m \in \widetilde{\mathcal{M}} \setminus \{b\}} \delta_{\mathcal{M}}^{m} \cdot m \right) \leqslant \frac{B - (c - \delta_{\mathcal{M}}^{b}) a}{b} \ , \text{ and then,}$$

$$\delta_{\mathcal{M}}^{b} \leqslant \frac{B - ca}{b - a} \ .$$

Let $\mathcal{M}_{\max} = \{b, \ldots, b, a, \ldots, a\}$ be a multiset with $\delta^b_{\mathcal{M}_{\max}} = \left\lceil \frac{B-ca}{b-a} \right\rceil$ and $\delta^a_{\mathcal{M}_{\max}} = c - \delta^b_{\mathcal{M}_{\max}}$. It can readily be seen that $\mathcal{M}_{\max} \succ \mathcal{M},\ \forall\ \mathcal{M} \in \mathbb{M}^{[a,b]}_{c,B}$ (note that $\mathcal{M}_{\max} \in \mathbb{M}^{[a,b]}_{c,B}$ if $(b-a)|(B-ca)$). Since $f(x)$ is a convex non-decreasing function for $a \leqslant x \leqslant b$, it follows from Lemma 6.4 that

$$\sum_{m \in \mathcal{M}_{\max}} f(m) \geqslant \sum_{m \in \mathcal{M}} f(m)\ ,\ \forall\ \mathcal{M} \in \mathbb{M}^{[a,b]}_{c,B}\ . \tag{6.12}$$

Hence,

$$\begin{aligned} \max_{\mathcal{M} \in \mathbb{M}^{[a,b]}_{c,B}} \sum_{m \in \mathcal{M}} f(m) &\leqslant \sum_{m \in \mathcal{M}_{\max}} f(m) \\ &= \delta^b_{\mathcal{M}_{\max}} f(b) + (c - \delta^b_{\mathcal{M}_{\max}}) f(a) \\ &= \left\lceil \frac{B-ca}{b-a} \right\rceil (f(b) - f(a)) + cf(a) \end{aligned}$$

and the statement follows. □

Sum of Cardinalities of Alternant Codes

Specific subclasses of alternant codes, such as some BCH and Goppa codes, are known to have larger dimension [MS77] than the lower bound given in Lemma 6.2. However, in general it is a difficult and open problem to predict the dimension of an alternant code for arbitrary column multipliers $\boldsymbol{v}$. Nevertheless, the sum of the cardinalities of subfield subcodes over all nonzero column multipliers can be determined, given the weight enumerator of the code, as we show in the following. This approach works not only for alternant codes, but also for any linear codes with known weight enumerator.

For a linear $[n, k, d]_{q^m}$ code $\mathcal{C}$, denote by $B_{n,d,w}(\mathcal{C})$ the sum of the number of codewords of weight w in the $\mathbb{F}_q$-subfield subcodes of $\mathcal{C}$ over all nonzero column multipliers, i.e.,

$$B_{n,d,w}(\mathcal{C}) := \sum_{\boldsymbol{v} \in (\mathbb{F}^*_{q^m})^n} \left| \{\boldsymbol{c} \cdot \mathrm{diag}(\boldsymbol{v}) \mid \boldsymbol{c} \in \mathcal{C}, \mathrm{wt}_{\mathsf{H}}(\boldsymbol{c}) = w\} \cap \mathbb{F}^n_q \right| .$$

Since every linear code contains the zero codeword and there is no codeword of Hamming weight $< d$ in the $[n, k, d]_{q^m}$ code, the sum of the cardinalities of the $\mathbb{F}_q$-subfield subcodes over all nonzero column multipliers is given by

$$B_{n,d}(\mathcal{C}) := \sum_{\boldsymbol{v} \in (\mathbb{F}^*_{q^m})^n} \left| \{\boldsymbol{c} \cdot \mathrm{diag}(\boldsymbol{v}) \mid \boldsymbol{c} \in \mathcal{C}\} \cap \mathbb{F}^n_q \right| = (q^m - 1)^n + \sum_{w=d}^{n} B_{n,d,w}(\mathcal{C})\ .$$

If $\mathcal{C}$ is a $\mathsf{GRS}^d_{\boldsymbol{\alpha},\boldsymbol{v}'}$ code as defined in Definition 6.2 for some $\boldsymbol{v}' \in (\mathbb{F}^*_{q^m})^n$, then $B_{n,d,w}$ is the sum of the number of codewords of weight w in all alternant codes $\mathbb{A}^d_{\boldsymbol{\alpha}}$, and $B_{n,d}(\mathcal{C})$ is the sum of the cardinalities of all $\mathbb{A}^d_{\boldsymbol{\alpha}}$. Interestingly, while the weight enumerator and cardinality of a specific subfield subcode depend on $\boldsymbol{v}$, the sum of these values over all $\boldsymbol{v}$ only depends on the weight enumerators of $\mathcal{C}$.

Lemma 6.6. *Let $\mathcal{C}$ be an $[n, k, d]_{q^m}$ code and denote by $A^{\mathcal{C}}_w$ the w-th weight enumerator of*

$\mathcal{C}$. Then,

$$B_{n,d,w}(\mathcal{C}) = A_w^{\mathcal{C}} \cdot (q^m - 1)^{n-w}(q-1)^w \ .$$

Proof. Let $\boldsymbol{c}$ be a codeword of $\mathcal{C}$. We have $\boldsymbol{c} \cdot \mathrm{diag}(\boldsymbol{v}) \in \mathbb{F}_q^n$ if and only if $c_i v_i \in \mathbb{F}_q$ for all $i \in [n]$. If $i \in \mathrm{supp}(\boldsymbol{c})$, then there are exactly $q-1$ choices of v_i for which $c_i v_i \in \mathbb{F}_q$. Else, any of the $q^m - 1$ possible values of v_i give $c_i v_i = 0 \in \mathbb{F}_q$. Hence, we have

$$\begin{aligned} B_{n,d,w}(\mathcal{C}) &= \sum_{\boldsymbol{v} \in (\mathbb{F}_{q^m}^*)^n} \left| \{\boldsymbol{c} \cdot \mathrm{diag}(\boldsymbol{v}) \mid \boldsymbol{c} \in \mathcal{C}, \mathrm{wt}_{\mathsf{H}}(\boldsymbol{c}) = w\} \cap \mathbb{F}_q^n \right| \\ &= \sum_{\substack{\boldsymbol{c} \in \mathcal{C} \\ \mathrm{wt}_{\mathsf{H}}(\boldsymbol{c}) = w}} \left| \left\{ \boldsymbol{v} \in (\mathbb{F}_{q^m}^*)^n \mid c_i v_i \in \mathbb{F}_q, \ \forall i \in [n] \right\} \right| \\ &= A_w^{\mathcal{C}} \cdot (q^m - 1)^{n-w}(q-1)^w \ . \end{aligned}$$

□

If $\mathcal{C}$ is an MDS code, then its weight enumerators $A_w^{\mathcal{C}}$, as given in Theorem 6.2, is completely determined by the code parameters (length, dimension/distance) and independent from the specific code constructions.

Theorem 6.2 (Weight enumerators of MDS codes [MS77, Ch. 11, Theorem 6])**.** *Let $\mathcal{C}$ be an $[n,k,d]_{q^m}$ MDS code. The w-th weight enumerator A_w^{MDS} of $\mathcal{C}$ is $A_0^{\mathsf{MDS}} = 1$ and for $w \neq 0$,*

$$A_w^{\mathsf{MDS}} := |\{\boldsymbol{c} \in \mathcal{C} \mid \mathrm{wt}_{\mathsf{H}}(\boldsymbol{c}) = w\}| = \binom{n}{w} \sum_{j=0}^{w-d} (-1)^j \binom{w}{j} (q^{m(w-d+1-j)} - 1) \ .$$

Hence, for an MDS code $\mathcal{C}$, we can write the sum of the cardinalities of the $\mathbb{F}_q$-subfield subcodes of $\mathcal{C}$ without dependence on $\mathcal{C}$ as

$$B_{n,d,w}^{\mathsf{MDS}} := B_{n,d,w}(\mathcal{C}) \text{ and } B_{n,d}^{\mathsf{MDS}} := B_{n,d}(\mathcal{C}) \ . \tag{6.13}$$

Probability of a Code Containing a Random Matrix

We begin by proving a technical lemma that bounds the probability that all rows of a randomly chosen matrix with nonzero columns are in a code of a certain dimension. This is a refined version of [SSB09, Lemma 3]. Recall that $\mathbb{E}_q^{(s,n)}$ is the set of matrix without zero columns in $\mathbb{F}_q^{s \times n}$.

Lemma 6.7. *For some integers $s > 0, n \geqslant k \geqslant 0$, let $\mathcal{A}$ be an $[n,k]_q$ code and denote by $A_w^{\mathcal{A}}$ its w-th weight enumerator. Then, for a matrix $\boldsymbol{E}$ taken independently from $\mathbb{E}_q^{(s,n)}$, we have*

$$\Pr_{\boldsymbol{E}}\{\boldsymbol{e}_i \in \mathcal{A}, \ \forall i \in [s]\} \leqslant \frac{q^{ks}(q-1) - (q^s-1)(q^k - 1 - A_n^{\mathcal{A}}) - (q-1)}{(q-1)(q^s-1)^n} \ ,$$

where $\boldsymbol{e}_i$ is the i-th row of $\boldsymbol{E}$.

Proof. Let $\mathcal{L} \subset \mathbb{F}_q^{s \times n}$ the set of matrices whose rows are codewords of $\mathcal{A}$ and by $\mathcal{L}_0 \subset \mathcal{L}$ the subset of all matrices in $\mathcal{L}$ with at least one all-zero column. Denote by $\bar{\mathcal{A}} \subset \mathcal{A}$ the set of

codewords of $\mathcal{A}$ whose first nonzero entry is 1. Then $\bar{\mathcal{A}}$ has cardinality $|\bar{\mathcal{A}}| = \frac{q^k-1}{q-1}$. It can be seen that

$$\left\{ \boldsymbol{E} \;\middle|\; \boldsymbol{e}_1, \ldots, \boldsymbol{e}_s \text{ are } \mathbb{F}_q\text{-scalar multiples of } \boldsymbol{e} \in \bar{\mathcal{A}} \cup \{0\}, \mathrm{wt}_{\mathsf{H}}(\boldsymbol{e}) < n \right\} \subseteq \mathcal{L}_0 \ .$$

If $\boldsymbol{e} = \boldsymbol{0}$ there is only the zero matrix in this set. For all the nonzero $\boldsymbol{e}$ with $\mathrm{wt}_{\mathsf{H}}(\boldsymbol{e}) < n$, each row $\boldsymbol{e}_i$ of $\boldsymbol{E}$ can be an $\mathbb{F}_q$-multiple of $\boldsymbol{e}$ and all such matrices $\boldsymbol{E}$ are unique, if at least one row is not $\boldsymbol{0}$. The number of such choices is $q^s - 1$, so

$$|\mathcal{L}_0| \geqslant (q^s-1)(|\bar{\mathcal{A}}| - \underbrace{|\{\boldsymbol{c} \in \bar{\mathcal{A}} \mid \mathrm{wt}_{\mathsf{H}}(\boldsymbol{c}) = n\}|}_{=\frac{A_n^{\mathcal{A}}}{(q-1)}}) + 1 = \frac{(q^s-1)}{(q-1)}(q^k - 1 - A_n^{\mathcal{A}}) + 1 \ .$$

Recall that $\mathbb{E}_q^{(s,n)}$ does not contain any matrices with all-zero columns by definition, so $\mathcal{L}_0 \cap \mathbb{E}_q^{(s,n)} = \varnothing$. As $\mathcal{L}_0 \subset \mathcal{L}$, it follows that

$$\Pr_{\boldsymbol{E}}\{\boldsymbol{e}_i \in \mathcal{A},\ \forall i = [s]\} = \frac{|\mathcal{L} \cap \mathbb{E}_q^{(s,n)}|}{|\mathbb{E}_q^{(s,n)}|} = \frac{|\mathcal{L} \backslash \mathcal{L}_0|}{|\mathbb{E}_q^{(s,n)}|} = \frac{|\mathcal{L}| - |\mathcal{L}_0|}{|\mathbb{E}_q^{(s,n)}|} \ .$$

The statement follows from $|\mathcal{L}| = |\mathcal{A}|^s = q^{ks}$ and $|\mathbb{E}_q^{(s,n)}| = (q^s-1)^n$.

□

If $|\mathcal{L}_0|$ is large, it is worthy to deduct it from $|\mathcal{L}|$ as in Lemma 6.7. However, for some parameters, (our best lower bound on) $|\mathcal{L}_0|$ becomes negligible compared to $|\mathcal{L}|$. Therefore, we also define a simplified version of this upper bound, where we only exclude the zero matrix from $\mathcal{L}$. The difference between L.A and L.A2 in the Figs. 6.2 and 6.3 reflects the difference between Lemma 6.7 and Corollary 6.1.

Corollary 6.1. *For some integers $s > 0, n \geqslant k \geqslant 0$, let $\mathcal{A}$ be an $[n,k]_q$ code. Then, for $\boldsymbol{E}$ that is i.i.d. in $\mathbb{E}_q^{(s,n)}$, we have*

$$\Pr_{\boldsymbol{E}}\{\boldsymbol{e}_i \in \mathcal{A},\ \forall i \in [s]\} \leqslant \frac{|\mathcal{L} \backslash \{\boldsymbol{0}_{s\times n}\}|}{|\mathbb{E}_q^{(s,n)}|} = \frac{q^{ks}-1}{(q^s-1)^n} \ .$$

With all the technical tools established, we are now ready to present the bounds on the success probability of decoding interleaved alternant codes using the decoder from [FT91; SSB09] (see also Algorithm 6.1).

Recall that the success probability is given by

$$P_{\mathsf{suc}} = 1 - P_{\mathsf{fail}} - P_{\mathsf{misc}} \ ,$$

where P_{fail} and P_{misc} are the probability of a decoding failure and a miscorrection, respectively.

6.3.2 A Lower Bound on Success Probability

In order to derive a lower bound on the success probability, we first establish a connection between the multisets $\mathbb{A}_{\boldsymbol{\alpha}|\nu}^{d-t}$ as defined in (6.6) and the probability of successful decoding.

Theorem 6.3. *Let $\mathcal{IC}^{(s)}$ be an s-interleaved alternant code with $\mathcal{C} \in \mathbb{A}_{\boldsymbol{\alpha}}^{d}$, $n = |\boldsymbol{\alpha}|$ and $\mathcal{E} = \{j_1, j_2, \ldots, j_t\} \subset [n]$ be a set of $|\mathcal{E}| = t$ error positions. For a codeword $\boldsymbol{C} \in \mathcal{IC}^{(s)}$, an error matrix $\widetilde{\boldsymbol{E}} \in \mathbb{F}_q^{s\times n}$ with $\operatorname{supp}(\widetilde{\boldsymbol{E}}) = \mathcal{E}$ and $\boldsymbol{E} := \widetilde{\boldsymbol{E}}|_{\mathcal{E}}$ i.i.d. in $\mathbb{E}_q^{(s,t)}$, and a received word $\boldsymbol{R} = \boldsymbol{C} + \widetilde{\boldsymbol{E}}$, Algorithm 6.1* succeeds*, i.e., returns $\hat{\boldsymbol{C}} = \boldsymbol{C}$, with probability*

$$P_{\mathsf{suc}}(\mathcal{IC}^{(s)}, \mathcal{E}) \geqslant 1 - \sum_{w=d-t}^{t} \sum_{\substack{\mathcal{V}\subseteq[\mathcal{E}]\\|\mathcal{V}|=w}} \sum_{\mathcal{A}\in\mathbb{A}_{\boldsymbol{\alpha}|_{\mathcal{V}}}^{d-t}} \left(\delta^{\mathcal{A}}_{\mathbb{A}^{d-t}_{\boldsymbol{\alpha}|_{\mathcal{V}}}}\right)^{-1} \Pr_{\boldsymbol{E}}\{\boldsymbol{e}_i|_{\mathcal{V}} \in \mathcal{A},\ \forall i \in [s]\} \ ,$$

where $\boldsymbol{e}_i|_{\mathcal{V}}$ is the i-th row of $\boldsymbol{E}$ restricted to the entries indexed in $\mathcal{V}$ and $\delta^{\mathcal{A}}_{\mathbb{A}^{d-t}_{\boldsymbol{\alpha}|_{\mathcal{V}}}}$ is the multiplicity of $\mathcal{A}$ in the multiset $\mathbb{A}^{d-t}_{\boldsymbol{\alpha}|_{\mathcal{V}}}$.

Proof. By Lemma 6.3 the decoding of $\widetilde{\boldsymbol{E}}$ is unsuccessful if and only if

$$\exists \boldsymbol{v} \in \mathbb{F}_{q^m}^t \backslash \{\boldsymbol{0}\} \text{ such that } \boldsymbol{H} \cdot \operatorname{diag}(\boldsymbol{v}) \cdot \boldsymbol{E}^\top = \boldsymbol{0} \ ,$$

where $\boldsymbol{H} \in \mathbb{F}_{q^m}^{(d-t-1)\times t}$ is a parity-check matrix of the $[t, 2t-d+1, d-t]_{q^m}$ code $\mathsf{GRS}^{d-t}_{\boldsymbol{\alpha}|_{\mathcal{E}},\boldsymbol{1}}$.

Therefore, the probability of unsuccessful decoding is upper bounded by

$$\begin{aligned}
&1 - P_{\mathsf{suc}}(\mathcal{IC}^{(s)}, \mathcal{E}) \\
&\quad = \Pr_{\boldsymbol{E}}\{\exists \boldsymbol{v} \in \mathbb{F}_{q^m}^t \backslash \{\boldsymbol{0}\} \text{ s.t. } \boldsymbol{H} \cdot \operatorname{diag}(\boldsymbol{v}) \cdot \boldsymbol{E}^\top = \boldsymbol{0}\} \\
&\quad = \sum_{w=1}^{t} \Pr_{\boldsymbol{E}}\{\exists \boldsymbol{v} \in \mathbb{F}_{q^m}^t \text{ with } \operatorname{wt}_{\mathsf{H}}(\boldsymbol{v}) = w \text{ s.t. } \boldsymbol{H} \cdot \operatorname{diag}(\boldsymbol{v}) \cdot \boldsymbol{E}^\top = \boldsymbol{0}\} \\
&\quad = \sum_{w=d-t}^{t} \Pr_{\boldsymbol{E}}\{\exists \boldsymbol{v} \in \mathbb{F}_{q^m}^t \text{ with } \operatorname{wt}_{\mathsf{H}}(\boldsymbol{v}) = w \text{ s.t. } \boldsymbol{H} \cdot \operatorname{diag}(\boldsymbol{v}) \cdot \boldsymbol{E}^\top = \boldsymbol{0}\} \qquad (6.14) \\
&\quad = \sum_{w=d-t}^{t} \sum_{\substack{\mathcal{V}\subseteq[\mathcal{E}]\\|\mathcal{V}|=w}} \Pr_{\boldsymbol{E}}\{\exists \mathcal{A} \in \mathbb{A}^{d-t}_{\boldsymbol{\alpha}|_{\mathcal{V}}} \text{ s.t. } \boldsymbol{e}_i|_{\mathcal{V}} \in \mathcal{A},\ \forall i \in [s]\} \\
&\quad \leqslant \sum_{w=d-t}^{t} \sum_{\substack{\mathcal{V}\subseteq[\mathcal{E}]\\|\mathcal{V}|=w}} \sum_{\mathcal{A}\in\mathbb{A}_{\boldsymbol{\alpha}|_{\mathcal{V}}}^{d-t}} \left(\delta^{\mathcal{A}}_{\mathbb{A}^{d-t}_{\boldsymbol{\alpha}|_{\mathcal{V}}}}\right)^{-1} \Pr_{\boldsymbol{E}}\{\boldsymbol{e}_i|_{\mathcal{V}} \in \mathcal{A},\ \forall i \in [s]\} \ ,
\end{aligned}$$

where (6.14) holds because any $d-t-1$ columns of $\boldsymbol{H}$ are linearly independent. □

With this connection between the multisets $\mathbb{A}^{d-t}_{\boldsymbol{\alpha}|_{\mathcal{V}}}$ and the probability of successful decoding $P_{\mathsf{suc}}(\mathcal{IC}^{(s)}, \mathcal{E})$ established, we now apply the technical results of Section 6.3.1 to obtain a lower bound.

Theorem 6.4 (Lower bound on P_{suc})**.** *The probability of successful decoding $P_{\mathsf{suc}}(\mathcal{IC}^{(s)}, \mathcal{E})$ as*

in Theorem 6.3 is lower bounded by

$$P_{\mathsf{suc}}(\mathcal{IC}^{(s)},\mathcal{E}) \geqslant 1 - \sum_{w=d-t}^{t} \frac{\binom{t}{w}}{(q^m-1)(q^s-1)^w} \cdot \Bigg(\frac{(q^s-1)}{(q-1)} \Big(c_w + B^{\mathsf{MDS}}_{w,d-t,w} - B^{\mathsf{MDS}}_{w,d-t} \Big) - c_w + \left(\frac{B^{\mathsf{MDS}}_{w,d-t} - c_w a_w}{b_w - a_w} + 1 \right) (b_w^s - a_w^s) + c_w a_w^s \Bigg) ,$$

with

$$a_w = \max\{1, q^{w-(d-t-1)m}\}, \quad b_w = q^{k_q^{\mathsf{opt.}}(w,d-t)}, \quad \text{and} \quad c_w = (q^m-1)^w ,$$

where $B^{\mathsf{MDS}}_{w,d-t}$ *and* $B^{\mathsf{MDS}}_{w,d-t,w}$ *are given in* (6.13) *and* $k_q^{\mathsf{opt.}}(w,d-t)$ *is an upper bound on the dimension of a* q*-ary code of length* w *and minimum Hamming distance* $d-t$*.*

Proof. For a q-ary code $\mathcal{A}$ denote $k_{\mathcal{A}} := \dim_q(\mathcal{A})$. Starting from Theorem 6.3, we obtain

$$\begin{aligned}
1 - P_{\mathsf{suc}}(\mathcal{IC}^{(s)},\mathcal{E}) &\leqslant \sum_{w=d-t}^{t} \sum_{\substack{\mathcal{V}\subseteq[\mathcal{E}]\\|\mathcal{V}|=w}} \sum_{\mathcal{A}\in\mathbb{A}^{d-t}_{\boldsymbol{\alpha}|\mathcal{V}}} (\delta^{\mathcal{A}}_{\mathbb{A}^{d-t}_{\boldsymbol{\alpha}|\mathcal{V}}})^{-1} \Pr_{\boldsymbol{E}}\{(\boldsymbol{E}|_{\mathcal{V}})_{i,:} \in \mathcal{A} \ \forall\ i \in [s]\} \\
&\overset{(\mathsf{a})}{\leqslant} \sum_{w=d-t}^{t} \sum_{\substack{\mathcal{V}\subseteq[t]\\|\mathcal{V}|=w}} \sum_{\mathcal{A}\in\mathbb{A}^{d-t}_{\boldsymbol{\alpha}|\mathcal{V}}} (q^m-1)^{-1} \frac{(q-1)q^{sk_{\mathcal{A}}} - (q^s-1)(q^{k_{\mathcal{A}}} - 1 - A^{\mathcal{A}}_w) - (q-1)}{(q-1)(q^s-1)^w} \\
&\overset{(\mathsf{b})}{=} \sum_{w=d-t}^{t} \sum_{\substack{\mathcal{V}\subseteq[t]\\|\mathcal{V}|=w}} \frac{(q^m-1)^{-1}}{(q^s-1)^w} \left(\frac{(q^s-1)}{(q-1)} (c_w + B^{\mathsf{MDS}}_{w,d-t,w}) - c_w + \sum_{\mathcal{A}\in\mathbb{A}^{d-t}_{\boldsymbol{\alpha}|\mathcal{V}}} \left(q^{sk_{\mathcal{A}}} - \frac{(q^s-1)}{(q-1)} q^{k_{\mathcal{A}}} \right) \right) \\
&\overset{(\mathsf{c})}{\leqslant} \sum_{w=d-t}^{t} \frac{\binom{t}{w}(q^m-1)^{-1}}{(q^s-1)^w} \left(\frac{(q^s-1)}{(q-1)} (c_w + B^{\mathsf{MDS}}_{w,d-t,w}) - c_w + \max_{\substack{\mathcal{M}\in\mathbb{M}^{[a_w,b_w]}_{c_w,B^{\mathsf{MDS}}_{w,d-t}}}} \sum_{M\in\mathcal{M}} \left(M^s - \frac{(q^s-1)}{(q-1)} M \right) \right) \\
&= \sum_{w=d-t}^{t} \frac{\binom{t}{w}(q^m-1)^{-1}}{(q^s-1)^w} \left(\frac{(q^s-1)}{(q-1)} (c_w + B^{\mathsf{MDS}}_{w,d-t,w} - B^{\mathsf{MDS}}_{w,d-t}) - c_w + \max_{\mathcal{M}\in\mathbb{M}^{[a_w,b_w]}_{c_w,B_{w,d-t}}} \sum_{M\in\mathcal{M}} M^s \right)
\end{aligned}$$

where (a) holds by (6.8) and Lemma 6.7, (b) holds as $\sum_{\mathcal{A}\in\mathbb{A}^{d-t}_{\boldsymbol{\alpha}|\mathcal{V}}} A^{\mathcal{A}}_w = B^{\mathsf{MDS}}_{w,d-t,w}$ (see (6.13)) and $|\mathbb{A}^{d-t}_{\boldsymbol{\alpha}|\mathcal{V}}| = c_w$ (see (6.7)), and (c) holds as a_w and b_w are lower and upper bounds on the cardinality of all codes $\mathcal{A} \in \mathbb{A}^{d-t}_{\boldsymbol{\alpha}|\mathcal{V}}$ (see Lemma 6.2) and $\sum_{\mathcal{A}\in\mathbb{A}^{d-t}_{\boldsymbol{\alpha}|\mathcal{V}}} q^{k_{\mathcal{A}}} = B^{\mathsf{MDS}}_{w,d-t}$ by Lemma 6.6. The theorem statement follows by Lemma 6.5. □

With the use of Corollary 6.1 instead of Lemma 6.7 for the inequality at (a) in the proof we get a slightly simplified (though worse) lower bound.

Corollary 6.2 (Simplified Lower Bound on P_{suc})**.** *The probability of successful decoding* $P_{\mathsf{suc}}(\mathcal{IC}^{(s)},\mathcal{E})$ *as in Theorem 6.3 is lower bounded by*

$$P_{\mathsf{suc}}(\mathcal{IC}^{(s)},\mathcal{E}) \geqslant 1 - \sum_{w=d-t}^{t} \frac{\binom{t}{w}(q^m-1)^{-1}}{(q^s-1)^w} \cdot \left(\left(\frac{B^{\mathsf{MDS}}_{w,d-t} - c_w a_w}{b_w - a_w} + 1 \right) (b_w^s - a_w^s) + c_w (a_w^s - 1) \right) ,$$

with

$$a_w = \max\{1, q^{w-(d-t-1)m}\}, \quad b_w = q^{k_q^{\mathsf{opt.}}(w,d-t)}, \quad \textit{and} \quad c_w = (q^m-1)^w \ ,$$

where $B_{w,d-t}^{\mathsf{MDS}}$ *is given in* (6.13) *and* $k_q^{\mathsf{opt}}(w, d-t)$ *is an upper bound on the dimension of a* q*-ary code of length* w *and minimum Hamming distance* $d-t$*.*

A Lower Bound on Success Probability for Large Interleaving Order $s \geqslant t$

For large interleaving order $s \geqslant t$, the Metzner-Kapturowski generic decoder [MK90] guarantees to decode any $1 \leqslant t \leqslant d-2$ errors if $\mathrm{rank}(\boldsymbol{E}) = t$ in an s-interleaved code with *any* $[n,k,d]_q$ constituent code. The decoder has been generalized in [HV00] for the case of rank deficiency when $2t-d+2 \leqslant \mathrm{rank}(\boldsymbol{E}) < t$. However, if the structure of the constituent code is unknown, determining the error positions in a rank-deficient error matrix $\boldsymbol{E}$ where $\mathrm{rank}(\boldsymbol{E}) = \mu < t$ is equivalent to finding a subset $\mathcal{U}$ of columns of a parity-check matrix $\boldsymbol{H} \in \mathbb{F}_{q^m}^{(d-1-\mu)\times n}$ with $\mathrm{rank}(\boldsymbol{H}|_{\mathcal{U}}) = t-\mu$. This is known to be a hard problem and no polynomial-time algorithm is known if the rank deficiency $t-\mu$ becomes large [RV14]. If the code structure is given, efficient syndrome-based algorithms are proposed in [RV14] and [YL18b] to correct linearly dependent error patterns with $\mathrm{rank}(\boldsymbol{E}) \geqslant 2t-d+2$ by interleaved RS codes over $\mathbb{F}_{q^m}$. These decoders also apply to the class of alternant codes over $\mathbb{F}_q$. Consider an s-interleaved alternant code $\mathcal{IC}^{(s)}$ where $\mathcal{C} \in \mathbb{A}_{\boldsymbol{\alpha}}^d$, $n = |\boldsymbol{\alpha}|$ and any set $\mathcal{E} \subset [n]$ of $|\mathcal{E}| = t$ error positions. A lower bound on the success probability is given in [RV14, Section II.C] as

$$\begin{aligned} P_{\mathsf{suc}}(\mathcal{IC}^{(s)}, \mathcal{E}) &\geqslant 1 - \Pr\{\mathrm{rank}(\boldsymbol{E}) < 2t-d+2\} \\ &= 1 - q^{-(s+d-1-2t)(d-1-t)}(1+o(1)) \\ &= 1 - q^{-2(t-\frac{3(d-1)+s}{4})^2 + \frac{(d-1-s)^2}{8}}(1+o(1)) \ , \end{aligned} \tag{6.15}$$

where $o(1)$ is an expression that goes to 0 as $q \to \infty$.

Note that though the decoder in [RV14] can be applied to interleaved alternant codes, the above lower bound is an asymptotic result. For some applications of alternant codes that we are interested in, e.g., Goppa codes in the McEliece system, the field size q is required to be finite or rather small. Therefore, in order to be self-contained and have a general expression on the failure probability, we prove in Lemma 6.8 that Algorithm 6.1 will always succeed in decoding linearly dependent error patterns if $\mathrm{rank}(\boldsymbol{E}) \geqslant 2t-d+2$ and we then give a lower bound in Theorem 6.5 on the success probability for $s \geqslant t$.

Lemma 6.8. *Assume* $s \geqslant t$*. Let* $\mathcal{IC}^{(s)}$ *be an* s*-interleaved alternant code with* $\mathcal{C} \in \mathbb{A}_{\boldsymbol{\alpha}}^d$*,* $n = |\boldsymbol{\alpha}|$ *and* $\mathcal{E} = \{j_1, j_2, \ldots, j_t\} \subset [n]$ *be a set of* $|\mathcal{E}| = t$ *error positions. For a codeword* $\boldsymbol{C} \in \mathcal{IC}^{(s)}$*, an error matrix* $\widetilde{\boldsymbol{E}} \in \mathbb{F}_q^{s\times n}$ *with* $\mathrm{supp}(\widetilde{\boldsymbol{E}}) = \mathcal{E}$ *and* $\boldsymbol{E} := \widetilde{\boldsymbol{E}}|_{\mathcal{E}}$ *i.i.d. in* $\mathbb{E}_q^{(s,t)}$*, and a received word* $\boldsymbol{R} = \boldsymbol{C} + \widetilde{\boldsymbol{E}}$*, Algorithm 6.1* succeeds*, i.e., returns* $\hat{\boldsymbol{C}} = \boldsymbol{C}$*, if*

$$\mathrm{rank}(\boldsymbol{E}) \geqslant 2t-d+2 \ .$$

Proof. Recall from (6.14) in the proof of Theorem 6.3 that the decoding does not succeed if and only if

$$\exists \boldsymbol{v} \in \mathbb{F}_{q^m}^t \backslash \{\boldsymbol{0}\} \text{ with } \mathrm{wt}_{\mathsf{H}}(\boldsymbol{v}) \geqslant d-t \text{ s.t. } \boldsymbol{H} \cdot \mathrm{diag}(\boldsymbol{v}) \cdot \boldsymbol{E}^\top = \boldsymbol{0} \ , \tag{6.16}$$

where $\boldsymbol{H}$ is a parity-check matrix of the $[t, 2t-d+1, d-t]_{q^m}$ $\mathsf{GRS}_{\boldsymbol{\alpha}|_{\mathcal{E}},\boldsymbol{1}}^{d-t}$ code.

We show that this condition cannot be fulfilled *if* $\operatorname{rank}(\boldsymbol{E}) \geqslant 2t - d + 2$.

Assume $\operatorname{rank}(\boldsymbol{E}) \geqslant 2t - d + 2$. Denote $\operatorname{wt}_{\mathsf{H}}(\boldsymbol{v}) = w$ and $\mathcal{L} := \operatorname{supp}(\boldsymbol{v})$. Let $\bar{\boldsymbol{H}} = \boldsymbol{H}|_{\mathcal{L}}$, $\bar{\boldsymbol{v}} = \boldsymbol{v}|_{\mathcal{L}}$, and $\bar{\boldsymbol{E}} = \boldsymbol{E}|_{\mathcal{L}}$. Observe the equivalence

$$\boldsymbol{H}|_{\mathcal{E}} \cdot \operatorname{diag}(\boldsymbol{v}) \cdot \boldsymbol{E}^{\top} = \boldsymbol{0} \quad \Longleftrightarrow \quad \bar{\boldsymbol{H}} \cdot \operatorname{diag}(\bar{\boldsymbol{v}}) \cdot \bar{\boldsymbol{E}}^{\top} = \boldsymbol{0} \ . \tag{6.17}$$

Note that

$$\operatorname{rank}(\bar{\boldsymbol{E}}) \geqslant \operatorname{rank}(\boldsymbol{E}) - (t - w) \geqslant 2t - d + 2 - (t - w) = w - (d - t) + 2$$

and $\bar{\boldsymbol{H}} \cdot \operatorname{diag}(\bar{\boldsymbol{v}})$ is a parity-check matrix of the $[w, w - (d - t) + 1, d - t]_{q^m}$ $\mathsf{GRS}^{d-t}_{\boldsymbol{\alpha}|_{\mathcal{L}}, \bar{\boldsymbol{v}}}$ code.

Assume for some $\boldsymbol{v}$ (6.17) is fulfilled. Then all rows of $\bar{\boldsymbol{E}}$ are codewords of the GRS code. In other words, the code spanned by $\bar{\boldsymbol{E}}$ is a subcode of the GRS code, i.e., $\langle \bar{\boldsymbol{E}} \rangle \subseteq \mathsf{GRS}^{d-t}_{\boldsymbol{\alpha}|_{\mathcal{L}}, \bar{\boldsymbol{v}}}$. However, since $\dim(\langle \bar{\boldsymbol{E}} \rangle) = \operatorname{rank}(\bar{\boldsymbol{E}}) \geqslant w - (d - t) + 2 > w - (d - t) + 1 = \dim(\mathsf{GRS}^{d-t}_{\boldsymbol{\alpha}|_{\mathcal{L}}, \bar{\boldsymbol{v}}})$, this is a contradiction. □

Theorem 6.5 (Lower bound on P_{suc} for $s \geqslant t$)**.** *Assume $s \geqslant t$. Let $\mathcal{IC}^{(s)}$ be an s-interleaved alternant code with $\mathcal{C} \in \mathbb{A}^d_{\boldsymbol{\alpha}}$, $n = |\boldsymbol{\alpha}|$ and $\mathcal{E} = \{j_1, j_2, \ldots, j_t\} \subset [n]$ be a set of $|\mathcal{E}| = t$ error positions. For a codeword $\boldsymbol{C} \in \mathcal{IC}^{(s)}$, an error matrix $\widetilde{\boldsymbol{E}} \in \mathbb{F}_q^{s \times n}$ with $\operatorname{supp}(\widetilde{\boldsymbol{E}}) = \mathcal{E}$ and $\boldsymbol{E} := \widetilde{\boldsymbol{E}}|_{\mathcal{E}}$ i.i.d. in $\mathbb{E}_q^{(s,t)}$, and a received word $\boldsymbol{R} = \boldsymbol{C} + \widetilde{\boldsymbol{E}}$, Algorithm 6.1* succeeds*, i.e., returns $\hat{\boldsymbol{C}} = \boldsymbol{C}$, with probability*

$$P_{\mathsf{suc}}(\mathcal{IC}^{(s)}, \mathcal{E}) \geqslant \frac{\sum_{s=2t-d+2}^{t} N(s,t,s)}{(q^s - 1)^t} \ ,$$

where

$$N(s,t,s) := |\{\boldsymbol{E} \in \mathbb{E}_q^{(s,t)} \mid \operatorname{rank}(\boldsymbol{E}) = s\}| = \sum_{j=0}^{t-s} (-1)^j \binom{t}{j} \prod_{i=0}^{s-1} \frac{(q^s - q^i)(q^{t-j} - q^i)}{q^s - q^i} \ .$$

Proof. By Lemma 6.8, it can be readily seen that the success probability is bounded from below by

$$\begin{aligned} P_{\mathsf{suc}}(\mathcal{IC}^{(s)}, \mathcal{E}) &\geqslant \frac{|\{\boldsymbol{E} \in \mathbb{E}_q^{(s,t)} \mid \operatorname{rank}(\boldsymbol{E}) \geqslant 2t - d + 2\}|}{|\mathbb{E}_q^{(s,t)}|} \\ &= \frac{\sum_{s=2t-d+2}^{t} |\{\boldsymbol{E} \in \mathbb{E}_q^{(s,t)} \mid \operatorname{rank}(\boldsymbol{E}) = s\}|}{(q^s - 1)^t} \ . \end{aligned}$$

It remains to determine

$$N(s,t,s) = |\{\boldsymbol{E} \in \mathbb{E}_q^{(s,t)} \mid \operatorname{rank}(\boldsymbol{E}) = s\}| \ ,$$

the number of matrices of $\mathbb{F}_q^{s \times t}$ *without* zero column and of a given rank. The number of matrices, including those *with* zero columns, of certain rank is given in [Lan93][FA66, Theo-

rem 2]:

$$M(s,t,s) := |\{\boldsymbol{E} \in \mathbb{F}_q^{s\times t} \mid \operatorname{rank}(\boldsymbol{E}) = s\}| = \prod_{i=0}^{s-1} \frac{(q^s - q^i)(q^t - q^i)}{q^s - q^i} \ .$$

To obtain $N(s,t,s)$, we need to exclude the matrices with zero columns from $M(s,t,s)$. By the inclusion-exclusion principle, we have

$$N(s,t,s) = \sum_{j=0}^{t-s} (-1)^j \binom{t}{j} M(s,t-j,s) \ .$$

□

Comparisons between Theorem 6.5 and Theorem 6.4 for some parameters can be found in Fig. 6.3 with the labels L.T and L.A respectively.

Remark 6.3 (Upper bound on the miscorrection probability P_{misc})**.** *An upper bound on P_{misc} of decoding interleaved alternant code by Algorithm 6.1 is given in [HLN+21a, Appendix A]. We expect the bound to be a rather rough upper bound, as it does not depend on the specific alternant code, nor the dimension of the alternant code. Nevertheless, we only intend to show that the probability of unsuccessful decoding of interleaved alternant codes is dominated by the failure probability, and the bound is sufficient for this purpose, as evident from the numerical results in Figs. 6.2 and 6.3 under the label* M*.*

6.3.3 An Upper Bound on Success Probability[3]

To evaluate the performance of the lower bounds of Section 6.3.2, we derive an upper bound on the probability of a decoding success. The approach is to show that for a certain set of error matrices, the decoder given in Algorithm 6.1 is *never* successful, i.e., the condition in Lemma 6.3 never holds.

Recall from the proof of Theorem 6.3 that

$$P_{\mathsf{suc}}(\mathcal{IC}^{(s)}, \mathcal{E}) = 1 - \sum_{w=d-t}^{t} \Pr_{\boldsymbol{E}}\{\exists \boldsymbol{v} \in \mathbb{F}_{q^m}^t \text{ with } \mathrm{wt}_{\mathsf{H}}(\boldsymbol{v}) = w \text{ s.t. } \boldsymbol{H}\cdot\operatorname{diag}(\boldsymbol{v})\cdot\boldsymbol{E}^\top = \boldsymbol{0}\} \ , \qquad \text{(Recall (6.14))}$$

where $\boldsymbol{H} \in \mathbb{F}_{q^m}^{(d-t-1)\times t}$ is a parity-check matrix of an $\mathsf{RS}_{q^m}[t, 2t-d+1]$ code and any $d-t-1$ columns of $\boldsymbol{H}$ are linearly independent. It can be readily seen that

$$P_{\mathsf{suc}}(\mathcal{IC}^{(s)}, \mathcal{E}) \leqslant 1 - \Pr_{\boldsymbol{E}}\{\exists \boldsymbol{v} \in \mathbb{F}_{q^m}^t \text{ with } \mathrm{wt}_{\mathsf{H}}(\boldsymbol{v}) = d-t \text{ s.t. } \boldsymbol{H}\cdot\operatorname{diag}(\boldsymbol{v})\cdot\boldsymbol{E}^\top = \boldsymbol{0}\} \ . \qquad (6.18)$$

Lemma 6.9. *Denote by $\mathbb{E}_{\mathsf{bad}}^{d-t}$ the set of matrices $\boldsymbol{E} \in \mathbb{E}_q^{(s,t)}$ for which there exists a vector $\boldsymbol{e} \in \mathbb{F}_q^s \backslash \{\boldsymbol{0}\}$ that is collinear (i.e., a $\mathbb{F}_q$-scalar multiple) to at least $d-t$ columns of $\boldsymbol{E}$. Then,*

$$\boldsymbol{E} \in \mathbb{E}_{\mathsf{bad}}^{d-t} \iff \exists \boldsymbol{v} \in \mathbb{F}_{q^m}^t \text{ with } \mathrm{wt}_{\mathsf{H}}(\boldsymbol{v}) = d-t \text{ s.t. } \boldsymbol{H}\cdot\operatorname{diag}(\boldsymbol{v})\cdot\boldsymbol{E}^\top = \boldsymbol{0} \ .$$

[3]The main contribution of this upper bound is by the co-author L. Holzbaur of the work [HLN+21a], we include it here for completeness.

Proof. We first show the sufficiency $\Longleftarrow$. Given a matrix $\boldsymbol{E} \in \mathbb{E}_q^{(s,t)}$, let $\mathcal{L} = \operatorname{supp}(\boldsymbol{v})$ and $|\mathcal{L}| = d - t$ such that $\boldsymbol{H} \cdot \operatorname{diag}(\boldsymbol{v}) \cdot \boldsymbol{E}^\top = \boldsymbol{0}$. Let $\bar{\boldsymbol{H}} := \boldsymbol{H}|_{\mathcal{L}} \in \mathbb{F}_{q^m}^{(d-t-1)\times(d-t)}$, $\bar{\boldsymbol{v}} := \boldsymbol{v}|_{\mathcal{L}} \in \mathbb{F}_{q^m}^{d-t}$, and $\bar{\boldsymbol{E}} := \boldsymbol{E}|_{\mathcal{L}} \in \mathbb{F}_q^{s\times(d-t)}$. We have the equivalence

$$\boldsymbol{H} \cdot \operatorname{diag}(\boldsymbol{v}) \cdot \boldsymbol{E}^\top = \boldsymbol{0} \quad \Longleftrightarrow \quad \bar{\boldsymbol{H}} \cdot \operatorname{diag}(\bar{\boldsymbol{v}}) \cdot \bar{\boldsymbol{E}}^\top = \boldsymbol{0} \ . \tag{6.19}$$

As any $d-t-1$ columns of $\bar{\boldsymbol{H}} \cdot \operatorname{diag}(\bar{\boldsymbol{v}}) \in \mathbb{F}_{q^m}^{(d-t-1)\times d-t}$ are $\mathbb{F}_{q^m}$-linearly independent (therefore $\mathbb{F}_q$-linearly independent), the right $\mathbb{F}_q$-kernel of $\bar{\boldsymbol{H}} \cdot \operatorname{diag}(\bar{\boldsymbol{v}})$ is of dimension at most 1. Since (6.19) holds by assumption and there is at least one nonzero row in $\bar{\boldsymbol{E}}$, the dimension of the right $\mathbb{F}_q$-kernel of $\bar{\boldsymbol{H}} \cdot \operatorname{diag}(\bar{\boldsymbol{v}})$ is at least 1. Together with the last argument, the right $\mathbb{F}_q$-kernel of $\bar{\boldsymbol{H}} \cdot \operatorname{diag}(\bar{\boldsymbol{v}})$ is of dimension exactly 1 and generated by $\boldsymbol{a} \in \left(\mathbb{F}_q^*\right)^{d-t}$. It then follows from (6.19) that all the nonzero rows of $\bar{\boldsymbol{E}}$ are collinear to the vector $\boldsymbol{a}$. Hence, $\operatorname{rank}(\bar{\boldsymbol{E}}) = 1$ and we conclude that there exists a vector $\boldsymbol{e} \in \mathbb{F}_q^s \backslash \{\boldsymbol{0}\}$ that is collinear to all the nonzero columns of $\bar{\boldsymbol{E}}$, and $\boldsymbol{E} \in \mathbb{E}_{\mathsf{bad}}^{d-t}$ by definition.

Now we show the necessity $\Longrightarrow$. Given a matrix $\boldsymbol{E} \in \mathbb{E}_{\mathsf{bad}}^{d-t}$, let $\mathcal{L} \subset [t]$ with $|\mathcal{L}| = d - t$ be some set of columns of $\boldsymbol{E}$ that are collinear to a vector $\boldsymbol{e} \in \mathbb{F}_q^s \backslash \{\boldsymbol{0}\}$. Let $\bar{\boldsymbol{H}} := \boldsymbol{H}|_{\mathcal{L}} \in \mathbb{F}_{q^m}^{(d-t-1)\times(d-t)}$ and $\bar{\boldsymbol{E}} := \boldsymbol{E}|_{\mathcal{L}} \in \mathbb{F}_q^{s\times(d-t)}$. By assumption $\operatorname{rank}(\bar{\boldsymbol{E}}) = 1$ and at least one row $\bar{\boldsymbol{e}}_i$ of $\bar{\boldsymbol{E}}$ must be a nonzero scalar multiple of some vector $\boldsymbol{e} \in (\mathbb{F}_q^*)^{d-t}$, since $\boldsymbol{E}$ does not have any zero column. Note that any $d-t$ columns of $\boldsymbol{H}$ form a parity-check matrix of an $\mathsf{RS}_{q^m}[d-t, 1]$ code (different sets of columns correspond to different code locators) and so does $\bar{\boldsymbol{H}}$. Denote by $\mathsf{RS}_{\bar{\boldsymbol{H}}}$ the RS code defined by $\bar{\boldsymbol{H}}$. Since $\mathrm{wt}_{\mathsf{H}}(\boldsymbol{e}) = d - t = \mathrm{d}_{\mathsf{H}}(\mathsf{RS}_{\bar{\boldsymbol{H}}})$, we can always find a $\bar{\boldsymbol{v}} \in \left(\mathbb{F}_{q^m}^*\right)^{d-t}$ such that the entry-wise multiplication of $\bar{\boldsymbol{v}}$ and $\boldsymbol{e}$ is a codeword of $\mathsf{RS}_{\bar{\boldsymbol{H}}}$, i.e.,

$$\bar{\boldsymbol{H}} \cdot (\bar{v}_1 e_1, \bar{v}_2 e_2, \ldots, \bar{v}_{d-t} e_{d-t})^\top = \boldsymbol{0} \ .$$

The $\boldsymbol{v} \in \mathbb{F}_{q^m}^t$ with $\boldsymbol{v}|_{\mathcal{L}} = \bar{\boldsymbol{v}}$ and 0 entries elsewhere results in $\boldsymbol{H} \cdot \operatorname{diag}(\boldsymbol{v}) \cdot \boldsymbol{E}^\top = \boldsymbol{0}$ and hence the necessity is proven.

□

Similar to $\mathbb{E}_{\mathsf{bad}}^{d-t}$, we defined $\mathbb{E}_{\mathsf{bad}}^{w}$ with $w \leqslant t$ to be the set of matrices $\boldsymbol{E} \in \mathbb{E}_q^{(s,t)}$ for which there exists a vector $\boldsymbol{e} \in \mathbb{F}_q^s \backslash \{\boldsymbol{0}\}$ that is a scalar multiple to at least w columns of $\boldsymbol{E}$. We give a general quantification on the cardinality of the set $\mathbb{E}_{\mathsf{bad}}^{w}$.

Lemma 6.10. *For $w \leqslant t$, the cardinality of $\mathbb{E}_{\mathsf{bad}}^{w}$ is bounded by*

$$\max_{w \leqslant \xi \leqslant t} \{Z^\xi\} \leqslant |\mathbb{E}_{\mathsf{bad}}^{w}| \leqslant (t - w + 1) \max_{w \leqslant \xi \leqslant t} \{Z^\xi\} \ , \ \textit{where}$$

$$Z^\xi = \sum_{j=1}^{\left\lfloor \frac{t}{s} \right\rfloor} (-1)^{j-1} \binom{\frac{q^s-1}{q-1}}{j} D_j^\xi \ , \ \textit{and}$$

$$D_j^\xi = \left(\prod_{z=0}^{j-1} \binom{t - z\xi}{\xi} \right) (q-1)^{j\xi} (q^s - q^j)^{t-j\xi} \ .$$

Proof. Consider the equivalence relation $\equiv_q$ on $\mathbb{F}_q^s \backslash \{\boldsymbol{0}\}$ defined by $\boldsymbol{v} \equiv_q \boldsymbol{u}$ if there exists $\lambda \in \mathbb{F}_q^*$ such that $\boldsymbol{v} = \lambda \boldsymbol{u}$. For a fixed vector $\boldsymbol{e} \in \mathbb{F}_q^s \backslash \{\boldsymbol{0}\}$ and a matrix $\boldsymbol{E} \in \mathbb{E}_q^{(s,w)}$, denote by $\delta_{\boldsymbol{E}}^{\boldsymbol{e}} := |\{i \mid \boldsymbol{E}_{[:,i]} \equiv_q \boldsymbol{e}\}|$ the number of columns of $\boldsymbol{E}$ that are equivalent to $\boldsymbol{e}$ under $\equiv_q$. For a

set of representatives $\mathcal{S} \subset \mathbb{F}_q^s \backslash \{\mathbf{0}\}$ under the given equivalence relation, we have

$$D_{|\mathcal{S}|}^{\xi} := |\{\boldsymbol{E} \in \mathbb{E}_q^{(s,w)} \mid \delta_{\boldsymbol{E}}^{\boldsymbol{e}} = \xi,\ \forall \boldsymbol{e} \in \mathcal{S}\}| = \left(\prod_{z=0}^{|\mathcal{S}|-1} \binom{t - z\xi}{\xi} \right) (q-1)^{|\mathcal{S}|\xi} (q^s - q^{|\mathcal{S}|})^{t-|\mathcal{S}|\xi} ,$$

where the first term counts the ways to position the equivalent (under $\equiv_q$) vectors to $\boldsymbol{e} \in \mathcal{S}$ into $\boldsymbol{E}$, the second term is the number of choices for the scalar coefficients of these positions, and the third term is the number of choices for the remaining columns, namely any nonzero vector that is not equivalent to any vector in $\mathcal{S}$. By the principle of inclusion-exclusion we get that the size of

$$\mathcal{Z}^{\xi} := \left\{ \boldsymbol{E} \in \mathbb{E}_q^{(s,w)} \mid \exists \boldsymbol{e} \in \mathbb{F}_q^s \backslash \{\mathbf{0}\} \text{ s.t. } \delta_{\boldsymbol{E}}^{\boldsymbol{e}} = \xi \right\}$$

is given by

$$Z^{\xi} := |\mathcal{Z}^{\xi}| = \sum_{j=1}^{\lfloor t/\xi \rfloor} (-1)^{j-1} \binom{\frac{q^s-1}{q-1}}{j} D_j^{\xi} .$$

The statement follows from the observation that

$$\mathbb{E}_{\mathsf{bad}}^{w} = \bigcup_{j=w}^{t} \mathcal{Z}^j .$$

□

Using the lower bound on the cardinality of $\mathbb{E}_{\mathsf{bad}}^{w}$, we now derive an upper bound on the probability of successful decoding.

Theorem 6.6 (Upper bound on P_{suc})**.** *Let $\mathcal{IC}^{(s)}$ be an s-interleaved alternant code with $\mathcal{C} \in \mathbb{A}_{\boldsymbol{\alpha}}^d$, $n = |\boldsymbol{\alpha}|$ and $\mathcal{E} = \{j_1, j_2, \ldots, j_t\} \subset [n]$ be a set of $|\mathcal{E}| = t \geqslant \frac{d}{2}$ error positions. For a codeword $\boldsymbol{C} \in \mathcal{IC}^{(s)}$, an error matrix $\widetilde{\boldsymbol{E}} \in \mathbb{F}_q^{s \times n}$ with $\operatorname{supp}(\widetilde{\boldsymbol{E}}) = \mathcal{E}$ and $\boldsymbol{E} := \widetilde{\boldsymbol{E}}|_{\mathcal{E}}$ i.i.d. in $\mathbb{E}_q^{(s,t)}$, and a received word $\boldsymbol{R} = \boldsymbol{C} + \widetilde{\boldsymbol{E}}$, Algorithm 6.1* succeeds*, i.e., returns $\hat{\boldsymbol{C}} = \boldsymbol{C}$, with probability*

$$P_{\mathsf{suc}}(\mathcal{IC}^{(s)}, \mathcal{E}) \leqslant 1 - \frac{|\mathbb{E}_{\mathsf{bad}}^{d-t}|}{(q^s-1)^t} \leqslant 1 - \frac{\max_{(d-t) \leqslant \xi \leqslant t} \{Z^{\xi}\}}{(q^s-1)^t} ,$$

where Z^{ξ} is given in Lemma 6.10.

Proof. The statement follows directly from (6.18), Lemma 6.9, Lemma 6.10, and $|\mathbb{E}_q^{(s,t)}| = (q^s-1)^t$.

□

6.3.4 Discussion and Numerical Results

In Sections 6.3.2 and 6.3.3 we have established lower and upper bounds on the probability

$$P_{\mathsf{suc}} = 1 - P_{\mathsf{fail}} - P_{\mathsf{misc}}$$

Table 6.1: Overview of the bounds shown in Figs. 6.2 and 6.3

Label	Defined in	Description
L.RS	Theorem 6.7	Lower bound on the probability of successful decoding for interleaved RS codes
L.A	Theorem 6.4	Lower bound on the probability of successful decoding for interleaved alternant codes where the minimum of the Singleton, Griesmer, Hamming, Plotkin, Elias, and Linear Programming bound is used for k_q^{opt}.
L.A1	Theorem 6.4	Lower bound on the probability of successful decoding for interleaved alternant codes, where the Singleton bound is used for k_q^{opt}.
L.A2	Corollary 6.2	Simplified version of Theorem 6.4. The minimum of the Singleton, Griesmer, Hamming, Plotkin, Elias, and Linear Programming bound is used for k_q^{opt}.
L.T	Theorem 6.5	Lower bound on the probability of successful decoding for interleaved alternant codes with $s \geqslant t$
M	[HLN+21a, Appendix A]	Upper bound on the probability of a miscorrection for interleaved alternant codes. We assume that the decoding radius of the interleaved decoder is $\left\lfloor \frac{s}{s+1}(d-1) \right\rfloor$, i.e., the largest number of errors for which the *RS interleaved decoder*, given in Algorithm 6.1, would succeed (see Remark 6.4).
U	Theorem 6.6	Upper bound on the probability of successful decoding for interleaved alternant codes.
SIM	Remark 6.4	Threshold number of errors such that for all numbers of errors left of the indicated line, the interleaved alternant decoder succeeds with a probability of $P_{\mathsf{suc}} > 0.9$ obtained by simulation with 100 decoding iterations per parameter set.

of successful decoding of interleaved alternant codes by the decoding algorithm from [FT91; SSB09] (see also Algorithm 6.1), assuming uniformly distributed burst errors of a given weight. In the following we present and discuss some numerical results, where we compare these upper and lower bounds[4]. In order to better emphasize the individual contributions of failures and miscorrections, we further include an upper bound on the probability of miscorrection P_{misc} from [HLPW19, Appendix A], in the plots of Figs. 6.2 and 6.3. We label, summarize, and describe the different bounds and versions thereof in Table 6.1 and, for convenience and clarity, refer to them by their respective label for the remainder of this section. Further, we fix the code length to be $n = q^m - 1$, i.e., given the base field size q and extension degree m we construct the longest possible RS/alternant codes, while excluding $\alpha_i = 0$ as a code locator

[4]For better presentation, we plot the respective bounds on the probability $1 - P_{\mathsf{suc}}$ of *unsuccessful* decoding instead of the bounds on P_{suc}.

(see Definition 6.2).

Aside from the comparison of the lower and upper bounds on the success probability, it is also interesting to see how the probability of successful decoding of an interleaved alternant code compares to that of the corresponding interleaved GRS code over $\mathbb{F}_{q^m}$. Such a bound was derived[5] and shown to be close to the probability of successful decoding obtained from simulation in [SSB09]. For the readers' convenience we restate it in Theorem 6.7 and assign it the label L.RS. Note that the decoder employed in [SSB09] is equivalent to the decoder presented in Algorithm 6.1. The difference is that the error matrix $\widetilde{\boldsymbol{E}}$ is assumed to be over $\mathbb{F}_{q^m}$ (the field of the RS code) in Theorem 6.7.

Theorem 6.7 (Probability of successful decoding for interleaved RS codes [SSB09, Theorem 7])**.** *Let $\mathcal{IC}^{(s)}$ be an s-interleaved GRS code with $\mathsf{GRS}^d_{\boldsymbol{\alpha},\boldsymbol{v}} \in \mathbb{G}^d_{\boldsymbol{\alpha}}$ as in Definition 6.2, $n = |\boldsymbol{\alpha}|$ and $\mathcal{E} = \{j_1, j_2, \ldots, j_t\} \subset [n]$ be a set of $|\mathcal{E}| = t$ error positions. For a codeword $\boldsymbol{C} \in \mathcal{IC}^{(s)}$, an error matrix $\widetilde{\boldsymbol{E}} \in \mathbb{F}_{q^m}^{s\times n}$ with $\operatorname{supp}(\widetilde{\boldsymbol{E}}) = \mathcal{E}$ and $\boldsymbol{E} = \widetilde{\boldsymbol{E}}|_{\mathcal{E}}$ i.i.d. $\mathbb{E}_{q^m}^{(s,t)}$, and a received word $\boldsymbol{R} = \boldsymbol{C} + \widetilde{\boldsymbol{E}}$, Algorithm 6.1* succeeds*, i.e., returns $\hat{\boldsymbol{C}} = \boldsymbol{C}$, with probability*

$$P_{\mathsf{suc}}(\mathcal{IC}^{(s)}, \mathcal{E}) \geqslant 1 - \left(\frac{q^{ms} - \frac{1}{q^m}}{q^{ms} - 1}\right)^t \cdot \frac{q^{-m(s+1)(t_{\max,\mathsf{RS}} - t)}}{q^m - 1} \ , \tag{6.20}$$

where $t_{\max,\mathsf{RS}} = \frac{s}{s+1}(d-1)$ as given in Theorem 6.1.

Before we discuss the numerical evaluations of the bounds, we make an important observation based on the *simulation* results.

Remark 6.4. *For most parameters, the provided lower bounds on the success probability (i.e., upper bounds on $1 - P_{\mathsf{suc}}$) of decoding interleaved alternant codes do not provide a non-trivial bound for the same decoding radius as the bounds for interleaved RS codes in [SSB09]. To determine the real decoding threshold, i.e., the smallest number of errors for which the decoder succeeds with non-negligible probability[6], we rely on simulation results. This threshold is indicated in the plots and labeled* SIM*. Notably, for all tested parameters,* the threshold for interleaved alternant codes is the same as for interleaved RS codes*, i.e., the simulation results imply that the joint decoding of interleaved alternant codes succeeds w.h.p. in presence of burst errors of weight t with*

$$t \leqslant \frac{s}{s+1}(d-1) = t_{\max,\mathsf{RS}} \ .$$

The numerical evaluations of the bounds are given in Figs. 6.2 and 6.3 for different base field size q, extension degree m, and distance d, each for varying interleaving order s:

- $\mathbf{q = 2, m = 10, d = 51}$: The parameters are chosen such that the rate of the constituent alternant code is ≈ 0.5, assuming the dimension is $k = n - (d-1)m$ (which tends to be true for most alternant codes). For wild Goppa and BCH codes the rate is ≈ 0.75 (see Page 124). The bounds for codes with these parameters are compared for different interleaving order s in Figs. 6.2a ($s = 2$), 6.2c ($s = 5$), 6.2e ($s = 10$) and 6.2g ($s = 25$). Figs. 6.3a and 6.3b are included to show the comparison between L.A and L.T.

[5] The bound in [SSB09] is presented as a bound on the probability of failure, but it is in fact a bound on the probability of unsuccessful decoding (see Remark 6.2).

[6] We arbitrarily choose this probability to be $P_{\mathsf{suc}} > 0.9$ and run 100 decoding iterations for each parameter set to determine the decoding threshold.

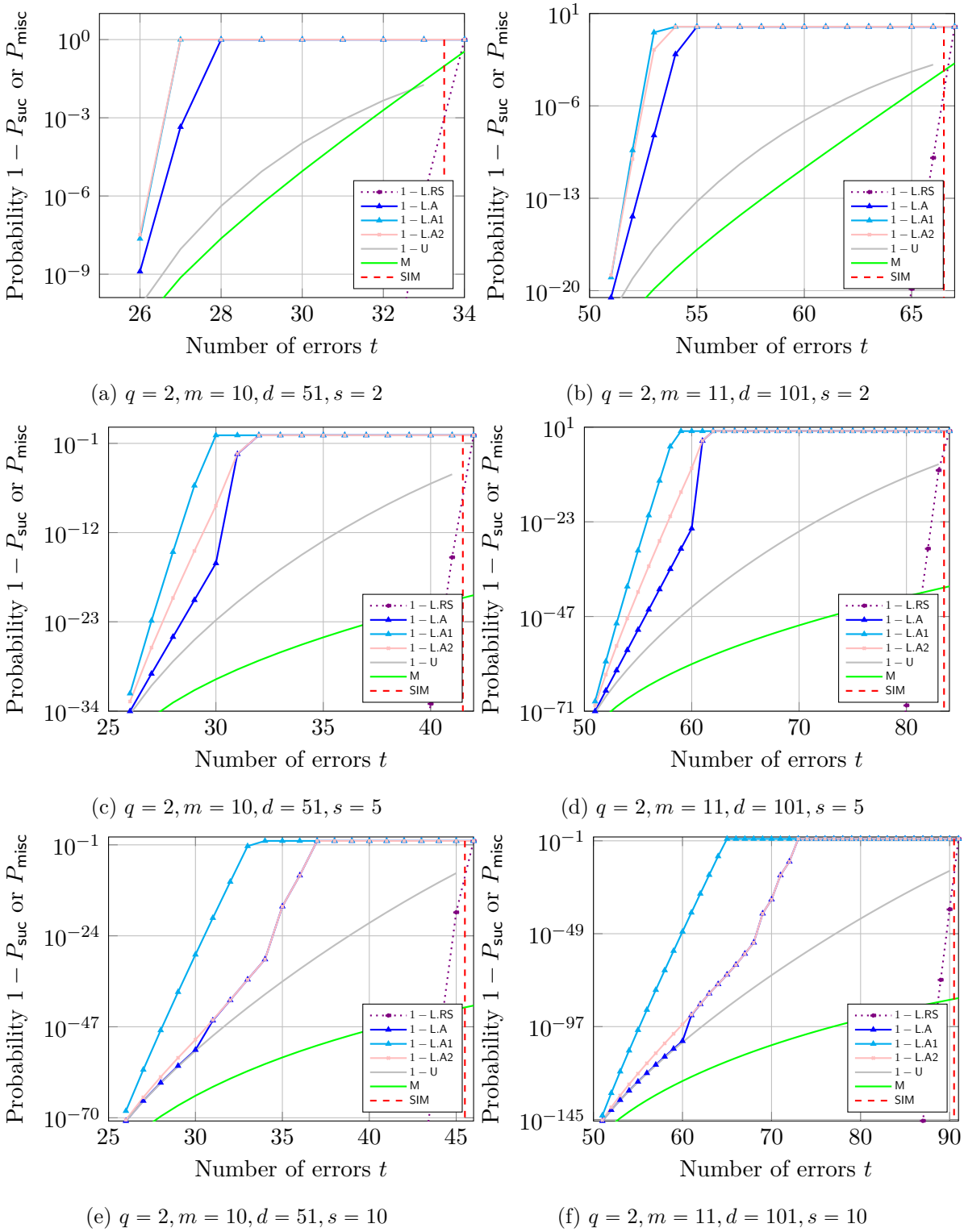
- $\mathbf{q = 2, m = 11, d = 101}$: We compare to the parameters above by changing the extension degrees m in Figs. 6.2b, 6.2d, 6.2f and 6.2h. The designed distance d changes accordingly such that the rate of the constituent alternant code is ≈ 0.5 (for wilde Goppa and BCH codes the rate is ≈ 0.75).
- $\mathbf{q = 32, m = 2, d = 51}$: To illustrate the influence of the base field size q, we show some evaluations of the bounds for $q = 32$ in Figs. 6.3c and 6.3d.

We now briefly discuss the main observations we have taken from the numerical results. As L.A1 and L.A2 are simplifications of L.A and therefore strictly worse, we leave their comparisons to a later point in the section, and begin by only comparing L.RS, L.A, L.T, M, and U. All statements on the decoding failure, miscorrection, and success probability refer to the syndrome-based joint decoder of [FT91; SSB09] given in Algorithm 6.1.

- For fixed q, m and s, the success probability for interleaved (q-ary) alternant codes is significantly smaller than that for interleaved (q^m-ary) RS codes, since even the *upper* bound U on the success probability for interleaved alternant codes is in most cases smaller than the *lower* bound L.RS on the success probability for interleaved RS codes.
- The probability of unsuccessful decoding interleaved alternant codes $1 - P_{\mathsf{suc}}$ is dominated by the probability of failure P_{fail}, as $P_{\mathsf{misc}} \ll 1 - P_{\mathsf{suc}}$, i.e., the bound on the probability of a miscorrection P_{misc}, labeled M, is multiple orders of magnitude smaller than $1 - P_{\mathsf{suc}} = P_{\mathsf{misc}} + P_{\mathsf{fail}}$ for the best bound on P_{suc} among L.A and L.T. This is consistent with the numerical results from [SSB09] for the case of decoding interleaved RS codes.
- For most parameters, L.A provides the best lower bound on the success probability P_{suc}. In particular, for higher interleaving order s ($= 10, 25$ for example) and relatively small number of errors t, it essentially matches the upper bound of Theorem 6.6 (see Figs. 6.2e, 6.2f, 6.2g, and 6.2h for example).
- For fixed q, m, and d, the relative gap between the number of errors for which the lower bounds on the success probability become non-trivial, i.e., give $P_{\mathsf{suc}} > 0$, and the simulated decoding threshold decreases for increasing interleaving order s (compare Fig. 6.2a, 6.2c, 6.2e and 6.2g or Fig. 6.2b, 6.2d, 6.2f and 6.2h).
- The lower bound L.T on the success probability for $s > t$ improves upon the bound of L.A for large number of errors that is close to the maximum decoding radius (see Figs. 6.3a and 6.3b).

Now consider the different versions of the lower bound on P_{suc} in Theorem 6.4 labeled by L.A (using the best upper bound on dimension of linear codes for k_q^{opt}), L.A1 (using the Singleton bound for k_q^{opt}), and L.A2 (Corollary 6.2, a simplified version of L.A).

- For small q, the performance of Theorem 6.4 is significantly worse when using the field-size-independent Singleton bound for k_q^{opt}, as evident from comparing L.A and L.A1 in Figs. 6.2a to 6.2h, 6.3a and 6.3b. This can be expected due to the large gap between the field-size-dependent bounds and the Singleton bound for k_q^{opt} when q is small.
- For larger interleaving order s ($\geqslant 10$ for example), the simplified lower bound L.A2 approaches the most accurate version of the bound L.A (see Figs. 6.2e to 6.2h, 6.3a and 6.3b).



(a) $q = 2, m = 10, d = 51, s = 2$

(b) $q = 2, m = 11, d = 101, s = 2$

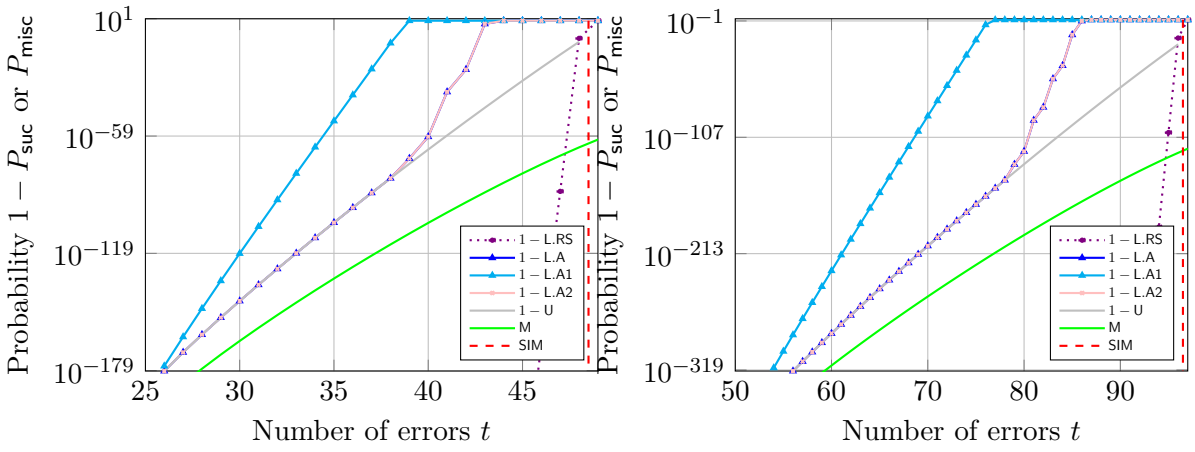
(c) $q = 2, m = 10, d = 51, s = 5$

(d) $q = 2, m = 11, d = 101, s = 5$

(e) $q = 2, m = 10, d = 51, s = 10$

(f) $q = 2, m = 11, d = 101, s = 10$

Figure 6.2: Comparison of the bounds for different interleaving order s and extension degree m. Rows are with different s while the two columns are with different m. For the bounds $\mathsf{L.RS}, \mathsf{L.A}, \mathsf{L.A1}, \mathsf{L.A2}, \mathsf{L.T}$, and U on the success probability we show the respective probabilities of unsuccessful decoding $1 - P_{\mathsf{suc}}$. The references of the bounds can be found in Table 6.1.



(g) $q = 2, m = 10, d = 51, s = 25$ (h) $q = 2, m = 11, d = 101, s = 25$

Figure 6.2: (Cont'd.) Comparison of the bounds for different s and m.

6.4 Other Results on Joint Decoding of Interleaved Codes

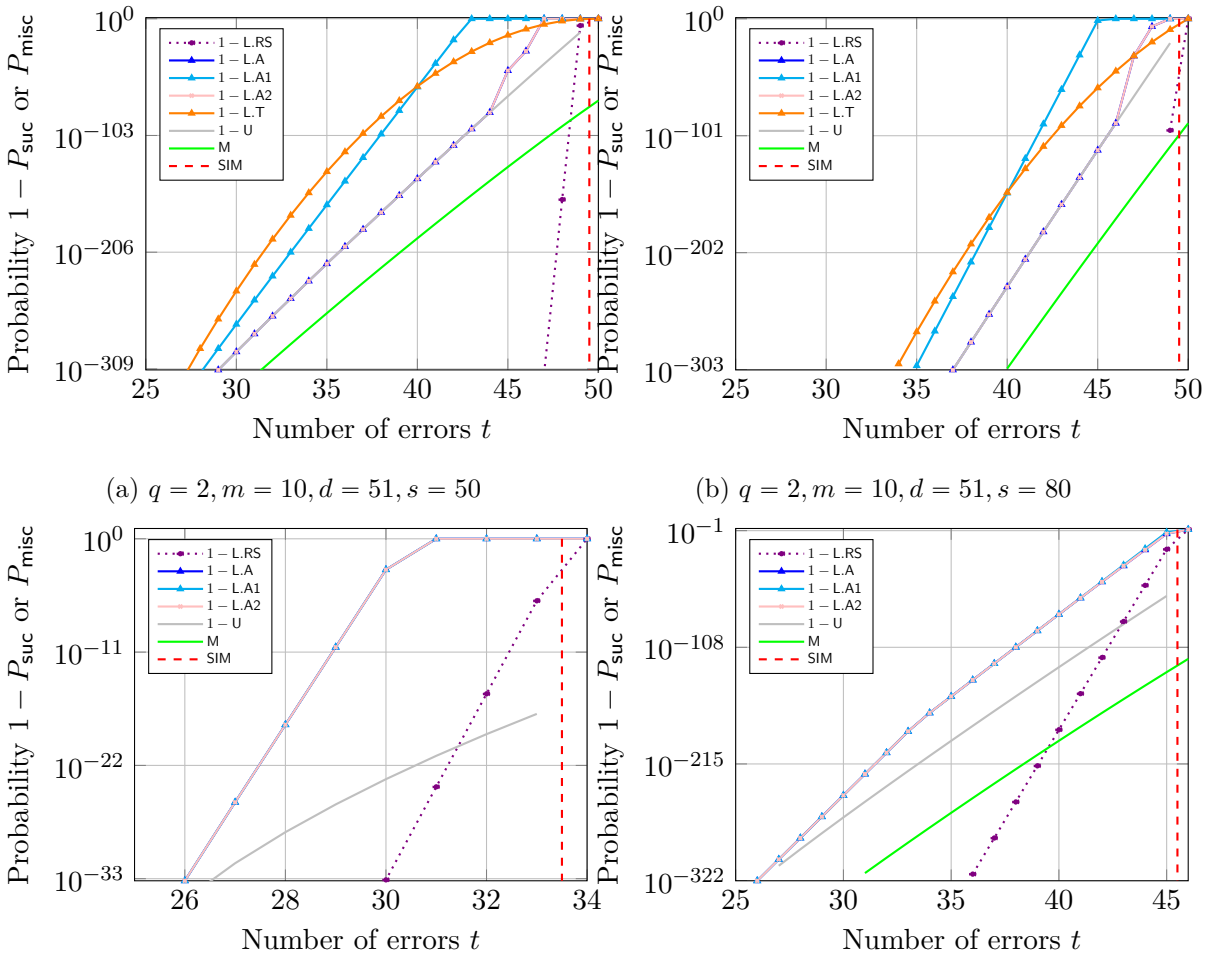
This section briefly summarizes a selection of other works on joint decoding of interleaved codes. For more details the interested reader is referred to the respective publications.

6.4.1 Joint Decoding of Generalized Goppa Codes

This abstract summarizes the results of the work [LPZW21] published in the proceeding of 2021 IEEE International Symposium on Information Theory (ISIT).

Generalized Goppa codes (GGCs) are an extension of Goppa codes, which are defined by a set of *code locator polynomials* and a *Goppa polynomial* [SM81; BS97]. In [NB20a], a code-based cryptosystem using binary GGCs with code locator polynomials of degree 1 and 2 is proposed. A special class of binary GGCs which is perfect in the weighted Hamming metric was introduced in [BS13] and cyclic GGCs were investigated in [Bez14; Bez15].

In this work, basic properties, decoding and potential cryptographic applications of binary GGCs are investigated. First, we derive a parity-check matrix for GGCs with code locators of any degree (an instance for GGCs with code locator polynomials of degree 2 was presented in [NB20b],[NB20a]). We provide a formal proof for the lower bound on the minimum Hamming distance of binary GGCs, which was stated in [NB20b],[NB20a], and we show that the lower bound for GGCs with even-degree code locator polynomials is improved compared to the general lower bound. Then, a quadratic-time decoding algorithm that can decode errors up to half of the minimum Hamming distance is presented. We further consider GGCs as the constituent code of interleaved codes. An explicit decoding algorithm based on Algorithm 6.1 and extended Euclidean algorithm is presented, and new maximum decoding radius for interleaved GGCs are derived. Finally, we list some code parameters of GGCs and discuss their applicability to the McEliece cryptosystem. By comparing the public key sizes for several code parameters, it can be observed that, under the same security level, the GGCs with degree-2 code locator polynomials provides smaller public key size than the binary Goppa codes.



(a) $q = 2, m = 10, d = 51, s = 50$

(b) $q = 2, m = 10, d = 51, s = 80$

(c) $q = 32, m = 2, d = 51, s = 2$

(d) $q = 32, m = 2, d = 51, s = 10$

Figure 6.3: Comparison of the bounds for large interleaving order $s \geqslant t$ (a and b) and different base field size q (c and d). For the bounds L.RS, L.A, L.A1, L.A2, L.T, and U on the success probability we show the respective probabilities of unsuccessful decoding $1 - P_{\text{suc}}$. The references of the bounds can be found in Table 6.1.

6.4.2 List Decoding of 2-Interleaved Binary Alternant Codes

This abstract summarizes the results of the work [HLH+22] published in the proceeding of 2022 IEEE International Symposium on Information Theory (ISIT).

Parvaresh [Par07] combined list and interleaved decoding by adapting the Guruswami-Sudan algorithm to the decoding of 2-interleaved GRS codes. Trivariate polynomials are used to set up the interpolation constraints and *resultants* of polynomials are used to recover the codeword. By combining the approaches of interleaved decoding and the Guruswami-Sudan algorithm, this decoder achieves a larger decoding radius than the Guruswami-Sudan algorithm, however, at the cost of a small probability of failure.

In this work, a list decoding algorithm for 2-interleaved binary alternant codes is proposed. The new algorithm combines the approach in [ABC11] that applies the Koetter-Vardy list decoding algorithm [KV03] to alternant codes, with the Parvaresh's algorithm for interleaved

GRS codes [Par07]. Similar to Parvareh's algorithm, it is difficult to make a precise statement on the decoding radius of this code. Instead, we present an upper bound on the decoding radius, along with simulation results showing that the decoding radius of the algorithm exceeds the decoding radii of all other algorithms known in literature for the chosen parameters. The drawback of the presented algorithm is that decoding is not guaranteed to succeed (similar to [Par07]). However, the simulation results indicate that this probability of failure is small, if the parameters of the algorithm are chosen suitably.

6.5 Summary and Outlooks

In this chapter, we investigated joint decoding of interleaved evaluation codes, in particular, of interleaved alternant codes by the decoder from [FT91; SSB09] in occurrence of burst errors. We first recapped the syndrome-based joint decoding algorithm for interleaved Reed-Solomon from [FT91; SSB09] and showed that a sufficient condition on decoding success given in [SSB09] for interleaved Reed-Solomon codes is also a necessary condition. After adapting the condition to the crux of jointly decoding interleaved alternant codes, we provided a framework for characterizing the probability of decoding success. Within this framework, we derived a lower bound and an upper bound on the success probability that holds for any interleaving order. Inspired by a generic decoding method from [MK90; RV14], we derived another lower bound that works for the interleaving orders that are larger than the number of error positions. Moreover, we numerically evaluated the obtained bounds for different code parameters, which show that one of the new lower bounds is tight for some parameters, as it matches the corresponding newly derived upper bound. Finally, other two works related to joint decoding interleaved evaluation codes are summarized.

It can be seen from the plots of the bounds that there is a gap between the number of errors t, where the upper bound $1-\mathsf{L.A}$ on $1-P_{\mathsf{suc}}$ provides a non-trivial value, and the $t_{\max}$, where simulated decoding succeeds with high probability. For future research, the most apparent open problem is to improve the general lower bound $\mathsf{L.A}$, in particular for smaller interleaving order, to close this gap. On the other hand, the current upper bound U on decoding success is derived by considering one out of $2t-d+1$ cases that the decoding never succeeds, where d is the designed distance of the alternant code. Therefore improvements upon the upper bound should be further investigated. Another closely related question, out of purely theoretical interest, is to determine the distribution of the dimensions of all alternant codes for a given set of code locators. For specific applications, such as code-based cryptography, improvements of the bounds for specific error distributions, e.g., full-rank errors, could be of practical relevance.

Although we only briefly summarized the recent results on list decoding of interleaved alternant codes, many open problems are left to be investigated. For example, the bottle neck of extending the algorithm to larger interleaving order $s \geqslant 3$ is the recovery step (root-finding) step. Efficient algorithms in eliminating variables in $\mathbb{F}_q[x][y_1, \ldots, y_s]$ are needed. Moreover, the upper bound given in [HLH+22] on the list decoding radius is quite far above the simulated number of decodable errors. Tighter upper bound or estimation by simulating on more code parameters should be further studied.

7
Concluding Remarks

This dissertations concerns new code constructions with properties that are desired in quantum error-correction, distributed storage system and network coding based on non-conventional classes of polynomials, and joint decoding on evaluation codes to decode beyond half the minimum distance.

Chapter 3 is devoted to constructing Euclidean and σ-Hermitian dual-containing (θ, δ)-polycyclic codes over finite commutative Frobenius rings from skew polynomials. We have developed an algorithm to find all dual-containing codes by transforming the problem of searching for dual-containing codes into a system of polynomial equations and using Gröbner bases to solve it. By applying this algorithm to several rings of order 4, the results show that there are dual-containing (θ, δ)-codes that can only be constructed from skew polynomials with non-trivial endomorphisms θ (not automorphisms) or nonzero derivations δ. Moreover, we have presented another algorithm with the usage of Gröbner basis to test whether the dual code is also a (θ, δ)-polycyclic code. Applying this algorithm to the resulting dual-containing codes found by the previous algorithm, we find some of those dual-containing codes whose dual is not a (θ, δ)-code.

Chapter 4 focuses on the condition of constructing support-constrained codes from evaluation codes based on skew polynomials (i.e., the linearlized Reed-Solomon (LRS) codes) and network coding. We have derived a necessary and sufficient condition on the existence of an LRS code fulfilling given support constraints and give an upper bound on the field size to construct such a code. With the help of the condition, we have proposed a scheme to design *distributed LRS codes* for multi-source unicast networks via solving an integer linear programming problem. For a class of multicast networks, the generalized combination networks, we have derived upper and lower bounds on the gap between the minimum required alphabet size of scalar solutions and vector solutions. The asymptotic behavior of the newly derived upper and lower bounds on the gap show that the number of bits that scalar solutions overpay is increasing sub-linearly with the size of the network.

Chapter 5 contains two new results on locally recoverable codes from evaluation codes based on multivariate polynomials. The first is on the $[q^2, 1 - \Theta\left((q/r)^{-0.2284}\right)]_q$ quadratic lifted Reed-Solomon codes (QLRS), where each codeword symbol has q^2 local recovery sets and within each local set r erasures can be corrected locally. We have compared its local recovery performance with the $[q^2, 1-\Theta\left((q/r)^{-0.4150}\right)]_q$ lifted Reed-Solomon codes which has q local recovery sets for each codeword symbol. Simulation results showed that, for a fixed dimension, QLRS codes are more likely to locally recover an erasure at a certain position in the presence of other erasures, if the erasure probability is $\leqslant 0.7$. The second result is on the

almost affinely disjoint (AAD) subspace family motivated by batch codes. We have given a construction based on Reed-Solomon codes of such family for $k = 1, 2$. The newly derived upper bound on the asymptotic growth of the cardinality of the family shows the optimality of this construction.

Chapter 6 is dedicated to joint decoding of interleaved evaluation codes. We have derived a necessary and sufficient condition on decoding success by the Schimidt-Sidorenko-Bossert joint decoding algorithm for interleaved alternant codes and lower and upper bounds on the probability of decoding success based on this condition. Numerical evaluations show that one of the provided lower bounds is tight for some parameters, as it matches the corresponding newly derived upper bound. The short summary on utilizing list-decoding for interleaved codes has shown the potential of this approach to further increase the decoding radius of evaluation codes in the presence of burst errors.

Various future research directions are presented in the outlooks at the end of each chapter.

Appendix

A.1 Derivation of $\deg_{\beta_{l,t}} P_{\boldsymbol{T}}$

This proof is an extension of the analysis on linearized polynomials for Gabidulin codes in [YH20, Section II.F] to skew polynomials.

From (4.8), the entry $T_{i,j}$ in $\boldsymbol{T}$ is the coefficient of X^{j-1} in $f_i(X)$. Note that we have set $T_{i,k} = 1$ in order to have the other entries in $\boldsymbol{T}$ uniquely determined given the roots of f_i's. For $h \in [k-1]$, $T_{i,h}$ is a commutative multivariate polynomial in R_n (see (4.12)) and

$$\deg_{\beta_{l,t}} T_{i,h} \leqslant \deg_{\beta_{l,t}} f_i(X) \ .$$

For any $l \in [\ell]$ and $t \in [n_l]$, to find $\deg_{\beta_{l,t}} f_i(X)$, consider the definition of $f_i(X)$ in (4.11). Suppose that $j = \varphi(l,t) \in Z_i$, otherwise $\deg_{\beta_{l,t}} f_i(X) = 0$. Recall that $\alpha_j = a_l \beta_{l,t}^{q-1}$ and $\mathcal{Z}_i$'s, $i \in [k]$ are as defined in (4.10). Let $f_i' \in \mathbb{F}_{q^m}[X;\sigma]$ be the minimal polynomial of $\mathcal{Z}_i' := \mathcal{Z}_i \backslash \{\alpha_j\}$, i.e.,

$$f_i'(X) = \operatorname*{lclm}_{\alpha \in \mathcal{Z}_i'} \{X - \alpha\} \ ,$$

whose X-degree is $\deg_X f_i'(X) = |\mathcal{Z}_i'| = k-2$. Since $j = \varphi(l,t) \notin Z_i'$, the coefficients of $f_i'(X)$ are independent of $\beta_{l,t}$, i.e., $\deg_{\beta_{l,t}} f_i'(X) = 0$.

By the remainder evaluation of skew polynomials in (2.4),

$$f_i'(a_l \beta_{l,t}^{q-1}) = \sum_{h=1}^{k-1} f_{i,h}' N_{h-1}(a_l \beta_{l,t}^{q-1}) \ ,$$

and

$$\begin{aligned} \deg_{\beta_{l,t}} f_i'(a_l \beta_{l,t}^{q-1}) &= \deg_{\beta_{l,t}} N_{k-2}(a_l \beta_{l,t}^{q-1}) \\ &= \deg_{\beta_{l,t}} (a_l \beta_{l,t}^{q-1})^{(q^{k-2}-1)/(q-1)} \\ &= q^{k-2} - 1 \ . \end{aligned}$$

By the Newton interpolation in (2.7), we can write

$$\begin{aligned} f_i(X) &= \left(X - \sigma(f_i'(a_l \beta_{l,t}^{q-1})) \cdot a_l \beta_{l,t}^{q-1} \cdot \left(f_i'(a_l \beta_{l,t}^{q-1}) \right)^{-1} \right) \cdot f_i'(X) \\ &= \left(X - \left(f_i'(a_l \beta_{l,t}^{q-1}) \right)^{q-1} \cdot a_l \beta_{l,t}^{q-1} \right) \cdot f_i'(X) \\ &= X \cdot f_i'(X) - \left(f_i'(a_l \beta_{l,t}^{q-1}) \right)^{q-1} \cdot a_l \beta_{l,t}^{q-1} \cdot f_i'(X) \ . \end{aligned}$$

Since $\deg_{\beta_{l,t}} f_i'(X) = 0$ and so is $\deg_{\beta_{l,t}}(X \cdot f_i'(X))$, we have

$$\begin{aligned}\deg_{\beta_{l,t}} f_i(X) &= (q-1) \cdot \deg_{\beta_{l,t}} f_i'(a_l \beta_{l,t}^{q-1}) + \deg_{\beta_{l,t}}(a_l \beta_{l,t}^{q-1}) \\ &= (q-1) \cdot (q^{k-2} - 1) + (q-1) \\ &= (q-1) \cdot q^{k-2} \ ,\end{aligned}$$

for all l, t such that $\varphi(l,t) \in Z_i$. Hence, $\deg_{\beta_{l,t}} T_{i,h} \leqslant \deg_{\beta_{l,t}} f_i(X) = (q-1)q^{k-2}, \forall h \in [k-1]$. Then,

$$\begin{aligned}\deg_{\beta_{l,t}} P_{\boldsymbol{T}} &= \deg_{\beta_{l,t}} \det \boldsymbol{T} \\ &\leqslant \max_{\pi \in \xi_k} \sum_{h=1}^{k} \deg_{\beta_{l,t}} T_{\pi(h),h} \\ &\leqslant (k-1)(q-1) \cdot q^{k-2} \ , \end{aligned} \tag{A.1}$$

where ξ_k denotes the set of permutations of $[k]$ and the $(k-1)$ in (A.1) is because $T_{ik} = 1$ and hence $\deg_{\beta_{l,t}} T_{ik} = 0$.

A.2 Proofs of Properties of Skew Polynomials over R_n

P1: $R_n[X;\sigma]$ is a ring without zero divisors.

Proof. The ring properties of $R_n[X;\sigma]$ are trivial, we only need to show that it has no zero divisors.

Note that for any $a, b \in R_n$, $\sigma(a+b) = \sigma(a) + \sigma(b)$. It can be seen from (4.18) that if $f, g \neq 0$, then $f \cdot g \neq 0$ since the leading coefficients of f_{d_f}, g_{d_g} are nonzero and therefore $f_{d_f} \sigma^{d_f}(g_{d_g})$ is nonzero. Hence, $R_n[X;\sigma]$ does not have zero divisors. □

P2: For any sets $\mathcal{Z}_1, \mathcal{Z}_2 \subseteq R_n$ s.t. $\mathcal{Z}_1 \cup \mathcal{Z}_2$ is P-independent, $\operatorname{gcrd}(f_{\mathcal{Z}_1}, f_{\mathcal{Z}_2}) = f_{\mathcal{Z}_1 \cap \mathcal{Z}_2}$. In particular, $\mathcal{Z}_1 \cap \mathcal{Z}_2 = \varnothing \iff \operatorname{gcrd}(f_{\mathcal{Z}_1}, f_{\mathcal{Z}_2}) = 1$.

Proof. This property has been proven as a part of [LMK17, Theorem 7] (cf.. Theorem 2.10) for $\mathbb{F}_{q^m}[X;\sigma]$. For completeness, we also include our proof here. We can write the minimal polynomial of the set $\mathcal{Z}_1 \cap \mathcal{Z}_2$ by the least common left multiplier as in (2.8), i.e., $f_{\mathcal{Z}_1 \cap \mathcal{Z}_2} = \operatorname*{lclm}_{\alpha \in \mathcal{Z}_1 \cap \mathcal{Z}_2} \{X - \alpha\}$. Then we can write $f_{\mathcal{Z}_1} = g_1 \cdot \operatorname*{lclm}_{\alpha \in \mathcal{Z}_1 \cap \mathcal{Z}_2} \{X - \alpha\}$ and $f_{\mathcal{Z}_2} = g_2 \cdot \operatorname*{lclm}_{\alpha \in \mathcal{Z}_1 \cap \mathcal{Z}_2} \{X - \alpha\}$, for some $g_1, g_2 \in R_n[X;\sigma]$. Therefore, it is clear that $f_{\mathcal{Z}_1 \cap \mathcal{Z}_2} \mid \operatorname{gcrd}(f_{\mathcal{Z}_1}, f_{\mathcal{Z}_2})$.

Now we only need to show that $\deg f_{\mathcal{Z}_1 \cap \mathcal{Z}_2} = \deg \operatorname{gcrd}(f_{\mathcal{Z}_1}, f_{\mathcal{Z}_2})$. Since $\mathcal{Z}_1 \cup \mathcal{Z}_2$ is P-independent, $\mathcal{Z}_1, \mathcal{Z}_2$ and $\mathcal{Z}_1 \cap \mathcal{Z}_2$ are also P-independent. Then $\deg f_{\mathcal{Z}_1 \cup \mathcal{Z}_2} = |\mathcal{Z}_1 \cup \mathcal{Z}_2|$, $\deg f_{\mathcal{Z}_1} = |\mathcal{Z}_1|$, $\deg_{\mathcal{Z}_2} = |\mathcal{Z}_2|$ and $\deg f_{\mathcal{Z}_1 \cap \mathcal{Z}_2} = |\mathcal{Z}_1 \cap \mathcal{Z}_2|$. It follows from [Glu21, Proposition 5.12] that the minimal polynomial of $\mathcal{Z}_1 \cup \mathcal{Z}_2$ is

$$f_{\mathcal{Z}_1 \cup \mathcal{Z}_2} = \operatorname{lclm}(f_{\mathcal{Z}_1}, f_{\mathcal{Z}_2})$$

and from [Ore33, Eq.(24)] that

$$\begin{aligned}\deg \operatorname{gcrd}(f_{\mathcal{Z}_1}, f_{\mathcal{Z}_2}) &= \deg f_1 + \deg f_2 - \deg \operatorname{lclm}(f_{\mathcal{Z}_1}, f_{\mathcal{Z}_2}) \\ &= |\mathcal{Z}_1| + |\mathcal{Z}_2| - |\mathcal{Z}_1 \cup \mathcal{Z}_2| \\ &= |\mathcal{Z}_1 \cap \mathcal{Z}_2| = \deg f_{\mathcal{Z}_1 \cap \mathcal{Z}_2} \ .\end{aligned}$$

Together with $f_{\mathcal{Z}_1 \cap \mathcal{Z}_2} \mid \operatorname{gcrd}(f_{\mathcal{Z}_1}, f_{\mathcal{Z}_2})$, the property is proven. □

P3: For $t \in \mathbb{N}$ and any $f \in R_n[X;\sigma]$, $X^t|_l f \iff X^t|_r f$. In this case, we write $X^t|f$.

Proof. If $X^t|_l f$, then with some $g \in R_n[X;\sigma]$ we can write $f = X^t \cdot g = \sigma^t(g) \cdot X^t$, where $\sigma^t(g) = \sum_i \sigma^t(g_i) X^i$. Then it is obvious that $X^t|_r f$. Similarly, if $X^t|_r f$, we can write $f = g \cdot X^t = X^t \cdot \sigma^{-t}(g)$ and it is obvious that $X^t|_l f$. This property has been also shown in [Ore33, Theorem 7]. □

P4: For $t \in \mathbb{N}$ and any $f_1, f_2 \in R_n[X;\sigma]$ such that $X \nmid f_2$, then $X^t|(f_1 \cdot f_2) \iff X^t \mid f_1$.

Proof. We first show $X^t \mid (f_1 \cdot f_2) \impliedby X^t \mid f_1$. Suppose $X^t \mid f_1$, then we can write $f_1 = X^t \cdot f_1'$ with some $f_1' \in R_n[X;\sigma]$. Then $f_1 \cdot f_2 = X^t \cdot f_1' \cdot f_2$ and it can be seen that $X^t \mid_l (f_1 \cdot f_2)$. By **P3**, we have $X^t \mid (f_1 \cdot f_2)$.

For the other direction, we first show that $X \mid (f_1 \cdot f_2) \implies X \mid f_1$ by contradiction. Assume $X \nmid f_1$, then we can write $f_1 = f_1' + a$ with some $f_1' \in R_n[X;\sigma]$ such that $X \mid f_1'$ and $a \in R_n \backslash \{0\}$. Since $X \nmid f_2$, we can write $f_2 = f_2' + b$, with some $f_2' \in R_n[X;\sigma]$ such that $X \mid f_2'$ and $b \in R_n \backslash \{0\}$. Then,

$$\begin{aligned}f_1 \cdot f_2 &= (f_1' + a)(f_2' + b) \\ &= f_1' \cdot f_2' + a \cdot f_2' + f_1' \cdot b + a \cdot b\end{aligned}$$

where the first three summands are all divisible by X but $a \cdot b \neq 0$ (since R_n is a ring without zero divisor) and $X \nmid a \cdot b$. This implies $X \nmid (f_1 \cdot f_2)$, which is a contradiction. Note that $X^2 \mid (f_1 \cdot f_2) \implies X \mid (f_1 \cdot f_2) \implies X \mid f_1$. Write $f_1 = X \cdot g$ with some $g \in R_n[X;\sigma]$, then

$$\begin{aligned}X^2 \mid (f_1 \cdot f_2) &\implies X \mid (g \cdot f_2) \underset{X \nmid f_2}{\implies} X \mid g \\ &\implies (X \cdot X) \mid (X \cdot g) \implies X^2 \mid f_1 \ .\end{aligned}$$

We can extend steps above t times and the property is proven. □

For any $Z_i \subseteq [n], i \in [k], l \in [\ell]$, we denote $Z_i^{(l)} \coloneqq \{t \mid \varphi(l,t) \in Z_i\}$ and $\mathcal{Z}_i^{(l)} = \{a_l \beta_{l,t}^{q-1} \mid t \in Z_i^{(l)}\}$, where $\varphi(l,t)$ is defined in (4.7). We need the following results on the set of roots of skew polynomials in order to prove **P5**. It follows from Lemma 2.3 that $f_{\mathcal{Z}_i}$ only vanishes on $\mathcal{Z}_i$ while evaluating on $\mathcal{L}$. The following lemma gives the structure of the roots of $f_{\mathcal{Z}_i}$ while evaluating on R_n.

Lemma A.1 ([LMK15, Theorem 4])**.** *For $l = 1, \ldots, \ell$, let $f_i^{(l)}$ be the minimal polynomial of*

$\mathcal{Z}_i^{(l)}$ *and* $\overline{\mathcal{Z}_i^{(l)}} \coloneqq \{\alpha \in R_n \mid f_i^{(l)}(\alpha) = 0\}$. *Then, for all* $l = 1, \ldots, \ell$,

$$\overline{\mathcal{Z}_i^{(l)}} = \{a_l \beta^{q-1} \mid \beta \in \langle \beta_{l,t} \rangle_{t \in Z_i^{(l)}} \setminus \{0\}\} \subseteq C_\sigma(a_l) \tag{A.2}$$

$$|\overline{\mathcal{Z}_i^{(l)}}| = q^{|Z_i^{(l)}|} - 1 \tag{A.3}$$

where $C_\sigma(a_l)$ *is the* σ*-conjugacy class of* a_l *as defined in* (2.6).

Theorem A.1. *Let* f_i *be the minimal polynomial of* $\mathcal{Z}_i$. *Denote the set of roots of* f_i *while evaluating on* R_n *by* $\overline{\mathcal{Z}_i} \coloneqq \{\alpha \in R_n \mid f_i(\alpha) = 0\}$. *Then*

$$\overline{\mathcal{Z}_i} = \bigcup_{l=1}^{\ell} \overline{\mathcal{Z}_i^{(l)}}, \text{ where } \overline{\mathcal{Z}_i^{(l)}} \text{ is as in (A.2)} \tag{A.4}$$

$$|\overline{\mathcal{Z}_i}| = \sum_{l=1}^{\ell} |\overline{\mathcal{Z}_i^{(l)}}| = \sum_{l=1}^{\ell} q^{|Z_i^{(l)}|} - \ell \ . \tag{A.5}$$

Proof. Note that for all $l \in [\ell]$, $\mathcal{Z}_i^{(l)}$ are P-independent and they are from different conjugacy classes. It follows from [LL04, Corollary 4.4] that for such sets, $\overline{\bigcup_{l=1}^{\ell} \mathcal{Z}_i^{(l)}} = \bigcup_{l=1}^{\ell} \overline{\mathcal{Z}_i^{(l)}}$. □

It is clear that the α's in (4.19) are P-independent. It follows from Definition 2.14 that $\deg \mathsf{f}(Z, \tau) = |Z| + \tau$. By Theorem A.1, the set of roots of $\mathsf{f}(Z, \tau)$ is

$$\{0\}^\tau \cup \bigcup_{l=1}^{\ell} \{a_l \beta^{q-1} \mid \beta \in \langle \beta_{l,t} \rangle_{t \in Z^{(l)}} \setminus \{0\}\} \tag{A.6}$$

where $Z^{(l)} = \{t \mid \varphi(l,t) \in Z\}$. The notation $\{0\}^\tau$ is to imply that $X^\tau \mid \mathsf{f}(Z,\tau)$ and $X^{\tau+1} \nmid \mathsf{f}(Z,\tau)$.

P5: For any $f_1 = \mathsf{f}(Z_1, \tau_1), f_2 = \mathsf{f}(Z_2, \tau_2) \in \mathcal{S}_{n,k}$, we have

$$\operatorname{gcrd}(f_1, f_2) = \mathsf{f}(Z_1 \cap Z_2, \min\{\tau_1, \tau_2\}) \in \mathcal{S}_{n,k} \ .$$

Proof. We prove the property by showing that the skew polynomials on both side have the same set of roots. Denote by $\overline{\mathcal{Z}_1}, \overline{\mathcal{Z}_2}, \overline{\mathcal{Z}_{1,2}} \subseteq R_n$ the set of all roots in R_n of $f_1, f_2, \mathsf{f}(Z_1 \cap Z_2, \min\{\tau_1, \tau_2\})$, respectively. By the structure of roots of $\mathsf{f}(Z,t)$ given in (A.6),

$$\overline{\mathcal{Z}_i} = \{0\}^{\tau_i} \cup \bigcup_{l=1}^{\ell} \Big\{ a_l \beta^{q-1} \mid \beta \in \langle \beta_{l,t} \rangle_{t \in Z_i^{(l)}} \setminus \{0\} \Big\}, \quad i = 1, 2$$

$$\overline{\mathcal{Z}_{1,2}} = \{0\}^{\min\{\tau_1,\tau_2\}} \cup \bigcup_{l=1}^{\ell} \Big\{ a_l \beta^{q-1} \mid \beta \in \langle \beta_{l,t} \rangle_{t \in Z_{1,2}^{(l)}} \setminus \{0\} \Big\}$$

where $Z_i^{(l)} \coloneqq \{t \mid \varphi(l,t) \in Z_i\}$ and $Z_{1,2}^{(l)} \coloneqq \{t \mid \varphi(l,t) \in Z_1 \cap Z_2\}$. The set of roots of $\operatorname{gcrd}(f_1, f_2)$ is

$$\overline{\mathcal{Z}_1} \cap \overline{\mathcal{Z}_2} = \{0\}^{\min\{\tau_1,\tau_2\}} \cup \bigcup_{l=1}^{\ell} \Big\{ a_l \beta^{q-1} \mid \beta \in \langle \beta_{l,t} \rangle_{t \in Z_1^{(l)} \cap Z_2^{(l)}} \setminus \{0\} \Big\} \ .$$

It can be seen that $Z_1^{(l)} \cap Z_2^{(l)} = Z_{1,2}^{(l)}, \forall l \in [\ell]$. Hence, $\overline{\mathcal{Z}_1} \cap \overline{\mathcal{Z}_2} = \overline{\mathcal{Z}_{1,2}}$. □

P6: Let $f = \mathsf{f}(Z,\tau) \in \mathcal{S}_{n,k}$ and let $f' = f|_{\beta_{\ell,n_\ell}=0} \in R_{n-1}[X;\sigma]$ (namely, we substitute $\beta_{\ell,n_\ell} = 0$ in each coefficient of f). Then $f' \in \mathcal{S}_{n-1,k}$ and

$$f' = \begin{cases} \mathsf{f}(Z,\tau) & n \notin Z \\ \mathsf{f}(Z\backslash\{n\},\tau+1) & n \in Z \end{cases}.$$

Proof. Denote by $\mathcal{Z}$ the subset of $\mathcal{L}$ corresponding to Z as in (4.10). It is trivial that $f' \in \mathcal{S}_{n-1,k}$ and $f' = \mathsf{f}(Z,\tau)$ when $n \notin Z$. Suppose $n \in Z$, then $a_\ell \beta_{\ell,n_\ell}^{q-1} \in \mathcal{Z}$. Let $g = \operatorname*{lclm}_{\alpha \in \mathcal{Z}\backslash\{a_\ell\beta_{\ell,n_\ell}\}} \{X-\alpha\}$, then

$$\begin{aligned} f' &= X^\tau \cdot \left(\operatorname*{lclm}_{\alpha\in\mathcal{Z}}\{X-\alpha\}\right)\Big|_{\beta_{\ell,n_\ell}=0} \\ &= X^\tau \cdot \left(\left(X - (a_\ell\beta_{\ell,n_\ell}^{q-1})^{g(a_\ell\beta_{\ell,n_\ell}^{q-1})}\right)\cdot g\right)\Big|_{\beta_{\ell,n_\ell}=0} \\ &= X^\tau \cdot X \cdot g \\ &= X^{\tau+1} \cdot g \\ &= X^{\tau+1} \cdot \left(\operatorname*{lclm}_{\alpha\in\mathcal{Z}\backslash\{a_\ell\beta_{\ell,n_\ell}^{q-1}\}}\{X-\alpha\}\right) \\ &= \mathsf{f}(Z\backslash\{n\},\tau+1) \in \mathcal{S}_{n-1,k}\ , \end{aligned}$$

where the second line holds by the Newton interpolation in (2.7). □

A.3 Induction Proof of Theorem 4.5

In the part we elaborate the induction proof of Theorem 4.5 for all the cases on page 63.
Case 1 For $s \geqslant 3$ and $n \geqslant 2$,
Case 1a $\forall i \in [s]$, $\tau_i \geqslant 1$ (i.e., $|Z_i| \leqslant k-2$).

Proof. For convenience we denote $k' = k-1$. For all $i \in [s]$, we can write $f_i = X \cdot f_i'$, where $f_i' = \mathsf{f}(Z_i, \tau_i - 1) \in \mathcal{S}_{n,k-1} = \mathcal{S}_{n,k'}$. Note that since $\min_{i\in[s]} \tau_i \geqslant 1$, we have $\deg f_\Omega \geqslant 1$ for any $\Omega \subseteq [s]$. For $\Omega = [s]$, (ii) implies $k - 1 \geqslant k - \deg f_{[s]} \geqslant \sum_{i\in[s]}(k - \deg f_i) \geqslant s$.

(Step 1) (ii) holds for $(f_1', \ldots, f_s')$ because for any nonempty $\Omega \subseteq [s]$,

$$\begin{aligned} k' - \deg f_\Omega' &= k - \deg f_\Omega \\ &\geqslant \sum_{i\in\Omega}(k - \deg f_i) \\ &= \sum_{i\in\Omega}(k' - \deg f_i') \end{aligned} \tag{A.7}$$

where (A.7) holds because (ii) holds for $(f_1, \ldots, f_s)$ by **H2**. By **H1**, (i) then holds for $(f_1', \ldots, f_s') \in \mathcal{S}_{n,k'}$. Note here that we used the induction hypothesis by reducing k to k'.

(Step 2) We then show that (i) also holds for $(f_1, \ldots, f_s)$. Suppose that for $g_1, \ldots, g_s \in R_n[X;\sigma]$ with $\deg(g_i \cdot f_i) \leqslant k-1$, we have $\sum_{i=1}^{s} g_i \cdot f_i = 0 = \sum_{i=1}^{s} g_i \cdot (X \cdot f_i') \overset{\mathbf{P1},\mathbf{P3}}{\Longrightarrow} \sum_{i=1}^{s} g_i f_i' = 0$, which implies that $g_1 = \cdots = g_s = 0$ since (i) holds for $(f_1', \ldots, f_s') \in \mathcal{S}_{n,k'}^{s}$. □

Case 1b $\exists$ a unique $i \in [s]$ such that $\tau_i = 0$.

Proof. Suppose w.l.o.g. $\tau_s = 0$ and write $f_s' = f_s \in \mathcal{S}_{n,k}$. For $i \in [s-1], \tau_i \geqslant 1$, then we can write $f_i = X \cdot f_i'$, where $f_i' = \mathsf{f}(Z_i, \tau_i - 1) \in \mathcal{S}_{n,k-1}$. Note that $f_s' = f_s \in \mathcal{S}_{n,k-1}$ if and only if $\deg f_s \leqslant k-2$, in which case for $\Omega = [s]$, **H2** implies

$$k \geqslant k - \deg f_\Omega \geqslant \sum_{i \in \Omega} (k - \deg f_i) \geqslant s + 1.$$

(Step 1) We show that (ii) holds for $(f_1', \ldots, f_s')$ when k is replaced by $k' = k-1$. First consider the case of $\Omega \subseteq [s-1]$. Since $\forall i \in [s-1], \tau_i \geqslant 1$, the claim follows similarly to ***Case 1a***. Additionally, by the induction hypothesis for $(k' = k-1, s-1, n)$ we get that (i) is true for $(f_1', \ldots, f_{s-1}')$. Then consider the case of Ω such that $s \in \Omega$. Since $f_s = \operatorname{lclm}_{\alpha \in \{a_l \beta_{l,t}^{q-1} \mid \varphi(l,t) \in Z_s\}} \{(X - \alpha)\}$ has no factor X, we have $\operatorname{gcrd}\{f_s, f_i\} = \operatorname{gcrd}\{f_s', f_i'\}, \forall i \in [s-1]$, hence $f_\Omega = f_\Omega'$ where we define $f_\Omega' = \operatorname{gcrd}_{i \in \Omega}\{f_i'\}$. Then

$$\begin{aligned} k - 1 - \deg f_\Omega' &= -1 + k - \deg f_\Omega \\ &\geqslant -1 + \sum_{i \in \Omega} \deg(k - \deg f_i) \qquad \text{(A.8)} \\ &= k - 1 - \deg f_s + \sum_{i \in \Omega \setminus \{s\}} (k - \deg f_i) \\ &= k - 1 - \deg f_s' + \sum_{i \in \Omega \setminus \{s\}} (k - 1 - \deg f_i') \\ &= \sum_{i \in \Omega} (k - 1 - \deg f_i') \ , \end{aligned}$$

where (A.8) holds from **H2**. By **H1**, (ii) $\implies$ (i) is true for $(f_1', \ldots, f_s')$ with parameters $(k' = k-1, s, n)$ if $\deg f_s' \leqslant k-2.$, which implies $k \geqslant s+1$.

(Step 2) Suppose that for some $g_1, \ldots, g_s \in R_n[X;\sigma]$ with $\deg(g_i \cdot f_i) \leqslant k-1$ we have $\sum_{i=1}^{s} g_i \cdot f_i = 0$. Then $0 = \sum_{i=1}^{s} g_i \cdot f_i = g_s \cdot f_s + \sum_{i=1}^{s-1} g_i \cdot (X \cdot f_i')$, which implies $X \mid (g_s \cdot f_s)$. However, since $X \nmid f_s$, by **P4**, $X \mid g_s$. Then we can write $g_s = g_s' \cdot X$ for some $g_s' \in R_n[X;\sigma]$ with $\deg g_s' = \deg g_s - 1$.

If $\deg f_s = k-1$, then $\deg g_s' = -1$, implying $g_s = 0$. Since (i) holds for $(f_1', \ldots, f_{s-1}') \in \mathcal{S}_{n,k-1}^{s-1}$ with the parameter tuple $(k-1, s-1, n)$, $g_1, \ldots, g_{s-1}$ are also zero. Note that here we used the induction hypothesis by reducing k to $k-1$ and s to $s-1$.

If $\deg f_s \leqslant k-2$, we have

$$\begin{aligned} 0 &= \sum_{i=1}^{s} g_i \cdot f_i \\ &= (g_s' \cdot X) \cdot f_s + \sum_{i=1}^{s-1} g_i \cdot (X \cdot f_i') \end{aligned}$$

$$= (g'_s \cdot X) \cdot f'_s + \sum_{i=1}^{s-1} (g_i \cdot X) \cdot f'_i \ .$$

Then $g_1 = \cdots = g_{s-1} = g'_s = 0$ since (i) holds for $(f'_1, \ldots, f'_s) \in \mathcal{S}^s_{n,k-1}$ with the parameter tuple $(k-1, s, n)$. Hence, all $g_1 = \cdots = g_s = 0$. Note that here we used the induction hypothesis by reducing k to $k-1$. □

Case 1c $\exists\ \Omega \subset [s]$ with $2 \leqslant |\Omega| \leqslant s-1$ such that (4.21) holds with equality.

Proof. W.l.o.g., assume that (4.21) holds with equality for $\Omega' = \{1, \ldots, \nu\}$, $1 < \nu < s$, i.e.,

$$k - \deg f_0 = \sum_{i \in \Omega'} (k - \deg f_i) \ , \tag{A.9}$$

where $f_0 = f_{\Omega'} = \operatorname{gcrd}_{i \in \Omega'} f_i$. Since $f_0 \mid_r f_i, \forall i \in \Omega'$, there exists $f'_i \in R_n[X;\sigma]$ such that $f_i = f'_i \cdot f_0$. Since $\nu < s$ and $s - \nu + 1 < s$, we split $(f_1, \ldots, f_s) \in \mathcal{S}^s_{n,k}$ into two smaller problems $(f_1, \ldots, f_\nu) \in \mathcal{S}^\nu_{n,k}$ with the parameter tuple $(k, \nu < s, n)$ and $(f_0, f_{\nu+1}, \ldots, f_s) \in \mathcal{S}^{s-\nu+1}_{n,k}$ with the tuple $(k, s - \nu + 1 < s, n)$.

(Step 1) Note that by **H2**, (ii) is true for $(f_1, \ldots, f_\nu)$ and for $(f_0, f_{\nu+1}, \ldots, f_s)$ when $0 \notin \Omega'' \subseteq \{0, \nu+1, \ldots, s\}$. We show in the following that (ii) is also true for $(f_0, f_{\nu+1}, \ldots, f_s)$ with $0 \in \Omega''$:

$$\begin{aligned}
k - \deg f_{\Omega''} &= k - \deg \operatorname{gcrd}\{f_0, f_{\Omega''\setminus\{0\}}\} \\
&= k - \deg \operatorname{gcrd}\{f_{\Omega'}, f_{\Omega''\setminus\{0\}}\} \\
&= k - \deg \operatorname{gcrd}_{i \in \Omega' \cup \Omega''\setminus\{0\}} f_i \\
&\geqslant \sum_{i \in \Omega' \cup \Omega''\setminus\{0\}} (k - \deg f_i) \qquad (A.10) \\
&= \sum_{i \in \Omega'} (k - \deg f_i) + \sum_{i \in \Omega''\setminus\{0\}} (k - \deg f_i) \\
&= k - \deg f_0 + \sum_{i \in \Omega''\setminus\{0\}} (k - \deg f_i) \qquad (A.11) \\
&= \sum_{i \in \Omega''} (k - \deg f_i)
\end{aligned}$$

Note that $\Omega' \cup \Omega''\setminus\{0\}$ is a subset of Ω. Therefore, the inequality in (A.10) follows from **H2**. The equality (A.11) follows from (A.9). Now we can conclude that (ii) is true for $(f_1, \ldots, f_\nu)$ and for $(f_0, f_{\nu+1}, \ldots, f_s)$.

By **H1**, (i) is true for both smaller problems $(f_1, \ldots, f_\nu) \in \mathcal{S}^\nu_{n,k}$ and $(f_0, f_{\nu+1}, \ldots, f_s) \in \mathcal{S}^{s-\nu+1}_{n,k}$.

(Step 2) Then we show (i) is also true for $(f_1, \ldots, f_s)$. Suppose that for some $g_1, \ldots, g_s \in R_n[X;\sigma]$ with $\deg g_i \cdot f_i \leqslant k-1, \forall i \in [s]$, we have

$$\sum_{i=1}^{s} g_i \cdot f_i = 0 \ . \tag{A.12}$$

Since $f_0 \mid_r f_i$ for all $i \in \Omega' = [\nu]$, f_0 is a right factor $\sum_{i=1}^{\nu} g_i \cdot f_i$ and we can then write

$\sum_{i=1}^{\nu} g_i \cdot f_i = g_0 \cdot f_0$, for some $g_0 \in R_n[X;\sigma]$. Then

$$\begin{aligned} 0 &= \sum_{i=1}^{s} g_i \cdot f_i \\ &= \sum_{i=1}^{\nu} g_i \cdot f_i + \sum_{i=\nu+1}^{s} g_i \cdot f_i \\ &= g_0 \cdot f_0 + \sum_{i=\nu+1}^{s} g_i \cdot f_i \end{aligned} \tag{A.13}$$

From the conclusion that (i) is true for $(f_0, f_{\nu+1}, \ldots, f_s)$, (A.13) holds only if $g_0 = g_{\nu+1} = \cdots = g_s = 0$. Similarly, since (i) is true for $(f_1, \ldots, f_\nu)$, $0 = g_0 \cdot f_0 = \sum_{i=1}^{\nu} g_i \cdot f_i$ only if $g_1 = \cdots = g_\nu = 0$. Therefore, (A.12) holds only if $g_1 = \cdots = g_s = 0$ and (i) is proven for $(f_1, \ldots, f_s) \in \mathcal{S}^s_{n,k}$ with the parameter tuple (k, s, n). □

Case 1d $\forall\ \Omega \subset [s]$ with $2 \leqslant |\Omega| \leqslant s-1$, (4.21) holds strictly and $\exists$ at least two $i \in [s]$ such that $\tau_i = 0$.

Proof. Assume w.l.o.g. that $\tau_{s-1} = \tau_s = 0$. Then for $i = s-1, s$, $\deg f_i = |Z_i|$. If $Z_{s-1} = Z_s$, then for $\Omega = \{s-1, s\}$, (ii) implies

$$\begin{aligned} k - \deg f_s &= k - \deg f_{s-1} \\ &= k - \deg \operatorname{gcrd}\{f_{s-1}, f_s\} \\ &\geqslant k - \deg f_{s-1} + k - \deg f_s \end{aligned}$$

which contradicts with $\deg f_i \leqslant k-1$ for any $i \in [s]$. Hence, $Z_{s-1} \neq [n]$ or $Z_s \neq [n]$. W.l.o.g., assume $Z_s \neq [n]$ and $n \notin Z_s$.

Note that $n = \varphi(\ell, n_\ell)$. We will substitute the variable $\beta_{\ell,n_\ell} = 0$. For all $i \in [s]$, let $f'_i := f_i|_{\beta_{\ell,n_\ell}=0}$. Since $n \notin Z_s$, we have $f'_s = f_s \in \mathcal{S}_{n-1,k}$. For other $i \in [s-1]$, by **P6**, $f'_i \in \mathcal{S}_{n-1,k}$ and

$$f'_i = \begin{cases} \mathtt{f}(Z_i, \tau_i) & n \notin Z_i \\ \mathtt{f}(Z_i \backslash \{n\}, \tau_i + 1) & n \in Z_i \end{cases}. \tag{A.14}$$

In the first case of (A.14) we denote $Z'_i = Z_i$ and $\tau'_i = \tau_i$, whereas in the second we denote $Z'_i = Z_i \backslash \{n\}$ and $\tau'_i = \tau_i + 1$. Additionally, we define $f'_\Omega = \operatorname{gcrd}_{i\in\Omega} f'_i$.

(Step 1) We will first show that $(f'_1, \ldots, f'_s)$ satisfies (ii). That is, we show that $\forall \varnothing \neq \Omega' \subseteq [s]$, $k - \deg f'_{\Omega'} \geqslant \sum_{i\in\Omega'}(k - \deg f'_i)$.

For $|\Omega'| = 1$, it is trivial.

For $2 \leqslant |\Omega'| \leqslant s-1$,

$$\begin{aligned} k - \deg f'_{\Omega'} &= k - |\bigcap_{i\in\Omega'} Z'_i| - \min_{i\in\Omega'} \tau'_i \\ &\geqslant k - |\bigcap_{i\in\Omega'} Z_i| - \min_{i\in\Omega'} \tau_i - 1 \\ &= k - \deg f_{\Omega'} - 1 \end{aligned} \tag{A.15}$$

$$\geqslant \sum_{i\in\Omega'} (k - \deg f_i) \tag{A.16}$$

$$= \sum_{i\in\Omega'} (k - \deg f'_i)\ . \tag{A.17}$$

The inequality (A.15) is because $|\bigcap_{i\in\Omega} Z'_i| \leqslant |\bigcap_{i\in\Omega} Z_i|$ and $\min_{i\in\Omega} \tau'_i \leqslant \min_{i\in\Omega} \tau_i + 1$. The inequality (A.16) is because we assume the inequality (4.21) in (ii) holds strictly for all $2 \leqslant |\Omega| \leqslant s-1$. The equality (A.17) holds because $\deg f'_i = \deg f_i, \forall i \in [s]$ by observing (A.14).

For $|\Omega'| = s$, (4.21) is not necessarily strict. However, since

$$n \notin Z_s \implies n \notin \bigcap_{i\in[s]} Z_i$$

$$\implies \left|\bigcap_{i\in[s]} Z'_i\right| = \left|\bigcap_{i\in[s]} Z_i\right| \implies f'_{[s]} = f_{[s]},$$

we have

$$\begin{aligned} k - \deg f'_{[s]} &= k - \deg f_{[s]} \\ &\geqslant \sum_{i\in[s]} (k - \deg f_i) \\ &= \sum_{i\in[s]} (k - \deg f'_i)\ . \end{aligned}$$

Hence, (ii) holds for $(f'_1, \ldots, f'_s) \in \mathcal{S}^s_{n-1,k}$. By **H1**, (i) holds for $(f'_1, \ldots, f'_s) \in \mathcal{S}^s_{n-1,k}$ with the parameter tuple $(k \geqslant s \geqslant 3, n-1)$ where $n \geqslant 2$. Note that here we used the induction hypothesis by reducing n to $n-1$.

(Step 2) Suppose that for some $g_1, \ldots, g_s \in R_n[X;\sigma]$, not all zero, with $\deg(g_i \cdot f_i) \leqslant k-1$, we have $\sum_{i=1}^s g_i \cdot f_i = 0$. Let $g'_i = g_i|_{\beta_{\ell,n_\ell}=0} \in R_{n-1}[X;\sigma]$. Further assume that at least one coefficient of some g_i is not divisible by β_{ℓ,n_ℓ} (otherwise, divide them by β_{ℓ,n_ℓ}). Then g'_i are not all zero. We can write

$$\sum_{i=1}^s g'_i \cdot f'_i = \left(\sum_{i=1}^s g_i \cdot f_i\right)\Bigg|_{\beta_{\ell,n_\ell}=0} = 0|_{\beta_{\ell,n_\ell}=0} = 0\ .$$

However, this contradicts (i) being true for $(f'_1, \ldots, f'_s)$ with the parameter tuple $(k, s, n-1)$. Therefore, $g_1, \ldots, g_s \in R_n[X;\sigma]$ must be all zero to have $\sum_{i=1}^s g_i \cdot f_i = 0$. □

Case 2 For $s = 2$ and $n \geqslant 2$,

Case 2a $\forall i \in \{1,2\}$, $\tau_i \geqslant 1$ (i.e., $|Z_i| \leqslant k-2$).

The proof for this case is the same as for ***Case 1a***. We use the induction hypothesis by reducing k.

Case 2b $\exists$ a unique $i \in \{1,2\}$ such that $\tau_i = 0$.

The proof for this case is the same as for ***Case 1b***. We use the induction hypothesis by reducing k. We may need to reduce s, too.

Case 2c $\forall i \in \{1,2\}$, $\tau_i = 0$.

Proof. In this case we have $\Omega = \{1,2\}$ and $\tau_1 = \tau_2 = 0$. Similar to ***Case 1d***, $Z_1 \neq [n]$ or

$Z_2 \neq [n]$. W.l.o.g., assume $Z_2 \neq [n]$ and $n \notin Z_2$. Note that $n = \varphi(\ell, n_\ell)$. We substitute the variable $\beta_{\ell,n_\ell} = 0$. For $i = 1, 2$, let $f_i' \coloneqq f_i|_{\beta_{\ell,n_\ell}=0}$ and $f_\Omega' \coloneqq \operatorname{gcrd}\{f_1', f_2'\}$. Since $n \notin Z_2$, $f_2' = f_2$. By **P6**, $f_1' \in \mathcal{S}_{n-1,k}$ and

$$f_1' = \begin{cases} \mathsf{f}(Z_1, 0) & n \notin Z_1 \\ \mathsf{f}(Z_1 \backslash \{n\}, 1) & n \in Z_1 \end{cases} .$$

(Step 1) We first show that $(f_1', f_2') \in \mathcal{S}_{n-1,k}^2$ satisfies (ii). That is, we show that $\forall \varnothing \neq \Omega' \subseteq \Omega$, $k - \deg f_{\Omega'}' \geqslant \sum_{i\in\Omega'}(k - \deg f_i')$.

For $|\Omega'| = 1$, it is trivial.

For $\Omega' = \{1, 2\}$, since

$$\begin{aligned} n \notin Z_2 \implies n \notin Z_1 \cap Z_2 &\implies |Z_1' \cap Z_2'| = |Z_1 \cap Z_2| \\ &\implies \deg f_{\Omega'}' = \deg f_{\Omega'} , \end{aligned}$$

we have

$$\begin{aligned} k - \deg f_{\Omega'}' &= k - \deg f_{\Omega'} \\ &\geqslant \sum_{i\in\Omega'} (k - \deg f_i) \\ &= \sum_{i\in\Omega'} (k - \deg f_i') . \end{aligned}$$

Hence, (ii) holds for $(f_1', f_2') \in \mathcal{S}_{n-1,k}^2$. By **H1**, (i) holds for $(f_1', f_2') \in \mathcal{S}_{n-1,k}^2$ with parameter tuple $(k \geqslant s = 2, n-1)$ where $n \geqslant 2$. Here we used the induction hypothesis by reducing n to $n - 1$.

(Step 2) This step can be shown in the same manner as in ***Case 1d***. □

Case 3 For $s \geqslant 2$ and $n = 1$,

Case 3a $\forall i \in [s]$, $\tau_i \geqslant 1$ (i.e., $|Z_i| \leqslant k - 2$).

The proof for this case is the same as for ***Case 1a***. We use the induction hypothesis by reducing k.

Case 3b $\exists$ a unique $i \in \{1, 2\}$ such that $\tau_i = 0$.

The proof for this case is the same as for ***Case 1b***. We use the induction hypothesis by reducing k. We may need to reduce s, too.

Case 3c $\exists$ at least two $i \in [s]$, $\tau_i = 0$.

Proof. W.l.o.g., assume that $\tau_1 = \tau_2 = 0$. Since $n = 1$, $|Z_i| \leqslant n = 1, \forall i \in [s]$. By the definition of $f_i \in \mathcal{S}_{1,k}$ in (4.20), $\deg f_i \leqslant n = 1$. Assume (ii) is true for this case, then for $\Omega = \{1, 2\}$, we have

$$k - \deg f_\Omega \geqslant k - \deg f_1 + k - \deg f_2 . \tag{A.18}$$

If $Z_1 = Z_2 = \varnothing$ or $\{1\}$, then $\deg f_\Omega = \deg f_1 = \deg f_2 \leqslant n = 1$ and (A.18) implies $\deg f_1 \geqslant k$, which contradicts $k \geqslant s \geqslant 2$. Otherwise, w.l.o.g., assume $Z_1 = \varnothing$ and $Z_2 = \{1\}$, then $\deg f_\Omega = 0$ and (A.18) implies $1 = \deg f_2 \geqslant k$, which contradicts $k \geqslant s \geqslant 2$. Therefore, if (ii) is true for $(k \geqslant s \geqslant 2, n = 1)$, this case cannot happen. □

Bibliography

References

[AS10] T. Abualrub and P. Seneviratne. "Skew codes over rings". In: *Proc. IMECS, Hong Kong, II* (2010).

[AL22] W. W. Adams and P. Loustaunau. *An introduction to Gröbner bases*. Vol. 3. American Mathematical Society, 2022.

[AAD+22] C. Aguilar-Melchor, N. Aragon, V. Dyseryn, P. Gaborit, and G. Zémor. "LRPC codes with multiple syndromes: near ideal-size KEMs without ideals". In: *International Conference on Post-Quantum Cryptography*. Springer. 2022, pp. 45–68.

[ACLY00] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung. "Network information flow". In: *IEEE Transactions on information theory* 46.4 (2000), pp. 1204–1216.

[AIP+21] A. Alahmadi, H. Islam, O. Prakash, P. Solé, A. Alkenani, N. Muthana, and R. Hijazi. "New quantum codes from constacyclic codes over a non-chain ring". In: *Quantum Information Processing* 20 (2021), pp. 1–17.

[ALNW22] G. Alfarano, F. J. Lobillo, A. Neri, and A. Wachter-Zeh. "Sum-rank product codes and bounds on the minimum distance". In: *Finite Fields and Their Applications* 80 (2022), p. 102013.

[AMN20] P. Almeida, U. Martínez-Peñas, and D. Napp. "Systematic maximum sum rank codes". In: *Finite Fields and Their Applications* 65 (2020), p. 101677.

[Alo99] N. Alon. "Combinatorial Nullstellensatz". In: *Combinatorics, Probability and Computing* 8.1-2 (1999), pp. 7–29.

[AKK+05] N. Alon, T. Kaufman, M. Krivelevich, S. Litsyn, and D. Ron. "Testing reed-muller codes". In: *IEEE Transactions on Information Theory* 51.11 (2005), pp. 4032–4039.

[AS08] N. Alon and J. H. Spencer. *The Probabilistic Method*. 3rd. Wiley Publishing, 2008. ISBN: 978-0-470-17020-5.

[AKS07] S. A. Aly, A. Klappenecker, and P. K. Sarvepalli. "On quantum and classical BCH codes". In: *IEEE Transactions on Information Theory* 53.3 (2007), pp. 1183–1188.

[ABD+23] N. Aragon, P. Briaud, V. Dyseryn, P. Gaborit, and A. Vinçotte. *The Blockwise Rank Syndrome Learning problem and its applications to cryptography*. Cryptology ePrint Archive, Paper 2023/1875. 2023. URL: `https://eprint.iacr.org/2023/1875`.

[ADG+23] N. Aragon, V. Dyseryn, P. Gaborit, P. Loidreau, J. Renner, and A. Wachter-Zeh. "LowMS: a new rank metric code-based KEM without ideal structure". In: *Designs, Codes and Cryptography* (2023), pp. 1–19.

[AGHT18] N. Aragon, P. Gaborit, A. Hauteville, and J.-P. Tillich. "A new algorithm for solving the rank syndrome decoding problem". In: *2018 IEEE International Symposium on Information Theory (ISIT)*. IEEE. 2018, pp. 2421–2425.

[Ari08] E. Arikan. "A performance comparison of polar codes and Reed-Muller codes". In: *IEEE Communications Letters* 12.6 (2008), pp. 447–449.

[AS97] S. Arora and M. Sudan. "Improved low-degree testing and its applications". In: *Proceedings of the twenty-ninth annual ACM symposium on Theory of computing*. 1997, pp. 485–495.

[AM63] E. Assmus Jr and H. Mattson. "Error-correcting codes: An axiomatic approach". In: *Information and Control* 6.4 (1963), pp. 315–330.

[ABC11] D. Augot, M. Barbier, and A. Couvreur. "List-decoding of binary Goppa codes up to the binary Johnson bound". In: *2011 IEEE Information Theory Workshop*. IEEE. 2011, pp. 229–233.

[AA16] N. Aydin and T. Abualrub. "Optimal quantum codes from additive skew cyclic codes". In: *Discrete Mathematics, Algorithms and Applications* 8.03 (2016), p. 1650037.

[BFLS91] L. Babai, L. Fortnow, L. A. Levin, and M. Szegedy. "Checking computations in polylogarithmic time". In: *Proceedings of the twenty-third annual ACM symposium on Theory of computing*. 1991, pp. 21–32.

[BDUY19] T. Bag, H. Q. Dinh, A. K. Upadhyay, and W. Yamaka. "New non-binary quantum codes from cyclic codes over product rings". In: *IEEE Communications Letters* 24.3 (2019), pp. 486–490.

[Bal20] S. Ball. "Maximum Distance Separable Codes". In: *A Course in Algebraic Error-Correcting Codes*. Cham: Springer International Publishing, 2020, pp. 83–103. ISBN: 978-3-030-41153-4. DOI: 10.1007/978-3-030-41153-4_6.

[BBB+20] M. Bardet, P. Briaud, M. Bros, P. Gaborit, V. Neiger, O. Ruatta, and J.-P. Tillich. "An algebraic attack on rank metric code-based cryptosystems". In: *Annual International Conference on the Theory and Applications of Cryptographic Techniques*. Springer. 2020, pp. 64–93.

[BJPR21] H. Bartz, T. Jerkovits, S. Puchinger, and J. Rosenkilde. "Fast decoding of codes in the rank, subspace, and sum-rank metric". In: *IEEE Transactions on Information Theory* 67.8 (2021), pp. 5026–5050.

[BJR22] H. Bartz, T. Jerkovits, and J. Rosenkilde. "Fast Kötter-Nielsen-Høholdt Interpolation over Skew Polynomial Rings and its Application in Coding Theory". In: *arXiv preprint arXiv:2207.01319* (2022).

[BP22] H. Bartz and S. Puchinger. "Fast Decoding of Interleaved Linearized Reed-Solomon Codes and Variants". In: *IEEE Transactions on Information Theory* (2022).

[Bau16] T. Baumbaugh. "Results on Common Left/Right Divisors of Skew Polynomials". PhD thesis. Clemson University, 2016.

[Bea19] B. Beach. *Backblaze Vaults: Zettabyte-Scale Cloud Storage Architecture*. `https://www.backblaze.com/blog/vault-cloud-storage-architecture/`. Last Published: 2019-06-18. 2019.

[Bec91] J. Beck. "An Algorithmic Approach to the Lovász Local Lemma. I". In: *Random Structures and Algorithms* 2.4 (Jan. 1991), pp. 343–365. ISSN: 1098-2418. DOI: 10.1002/rsa.3240020402.

[BGK+13] E. Ben-Sasson, A. Gabizon, Y. Kaplan, S. Kopparty, and S. Saraf. "A New Family of Locally Correctable Codes Based on Degree-Lifted Algebraic Geometry Codes". In: STOC '13. Palo Alto, California, USA: Assoc. Comput. Machinery, 2013, 833–842.

[Ber73] E. R. Berlekamp. "Goppa codes". In: *IEEE Transactions on Information Theory* 19.5 (1973), pp. 590–592.

[Ber74] E. R. Berlekamp. *Key Papers in the Development of Coding Theory*. IEEE Press selected reprint series. IEEE, 1974. ISBN: 9780879420321.

[BLP11] D. J. Bernstein, T. Lange, and C. Peters. "Wild McEliece". In: *Selected Areas in Cryptography: 17th International Workshop, SAC 2010, Waterloo, Ontario, Canada, August 12-13, 2010, Revised Selected Papers*. Vol. 6544. Springer. 2011, p. 143.

[Bez14] S. Bezzateev. "One subclass of cyclic generalized (L, G) codes with separable Goppa polynomial". In: *2014 XIV International Symposium on Problems of Redundancy in Information and Control Systems*. IEEE. 2014, pp. 30–32.

[Bez15] S. Bezzateev. "Cyclic Generalized Separable (L, G) Codes". In: *Coding Theory and Applications: 4th International Castle Meeting, Palmela Castle, Portugal, September 15-18, 2014*. Springer. 2015, pp. 53–60.

[BS97] S. Bezzateev and N. Shekhunova. "One generalization of Goppa codes". In: *Proceedings of IEEE International Symposium on Information Theory*. IEEE. 1997, p. 299.

[BS13] S. Bezzateev and N. Shekhunova. "Class of Binary Generalized Goppa Codes Perfect in Weighted Hamming Metric". In: *Designs Codes and Cryptography* 66 (Aug. 2013), pp. 391–399. DOI: 10.1007/s10623-012-9739-6.

[BKS+10] A. Bhattacharyya, S. Kopparty, G. Schoenebeck, M. Sudan, and D. Zuckerman. "Optimal testing of Reed-Muller codes". In: *2010 IEEE 51st Annual Symposium on Foundations of Computer Science*. IEEE. 2010, pp. 488–497.

[Bla72] I. F. Blake. "Codes over certain rings". In: *Information and Control* 20.4 (1972), pp. 396–404.

[Bla75] I. F. Blake. "Codes over integer residue rings". In: *Information and Control* 29.4 (1975), pp. 295–300.

[BKY03] D. Bleichenbacher, A. Kiayias, and M. Yung. "Decoding of Interleaved Reed Solomon Codes Over Noisy Data". In: *International Colloquium on Automata, Languages, and Programming*. Springer. 2003, pp. 97–108.

[BR60] R. C. Bose and D. Ray-Chaudhuri. "On a class of error correcting binary group codes". In: *Information and control* 3.1 (1960), pp. 68–79.

[BCP97] W. Bosma, J. Cannon, and C. Playoust. "The Magma algebra system. I. The user language". In: *J. Symbolic Comput.* 24.3-4 (1997). Computational algebra and number theory (London, 1993), pp. 235–265. ISSN: 0747-7171. DOI: 10.1006/jsco.1996.0125.

[Bou20] D. Boucher. "An algorithm for decoding skew Reed–Solomon codes with respect to the skew metric". In: *Designs, Codes and Cryptography* 88.9 (2020), pp. 1991–2005.

[BGG+10] D. Boucher, P. Gaborit, W. Geiselmann, O. Ruatta, and F. Ulmer. "Key exchange and encryption schemes based on non-commutative skew polynomials". In: *International Workshop on Post-Quantum Cryptography*. Springer. 2010, pp. 126–141.

[BGU07] D. Boucher, W. Geiselmann, and F. Ulmer. "Skew-cyclic codes". In: *Applicable Algebra in Engineering, Communication and Computing* 18.4 (2007), pp. 379–389.

[BSU08] D. Boucher, P. Solé, and F. Ulmer. "Skew constacyclic codes over Galois rings". In: *Advances in mathematics of communications* 2.3 (2008), pp. 273–292.

[BU09a] D. Boucher and F. Ulmer. "Codes as modules over skew polynomial rings". In: *12th IMA International Conference, Cryptography and Coding 2009, Cirencester, UK, December 15-17, 2009. Proceedings 12*. Springer. 2009, pp. 38–55.

[BU09b] D. Boucher and F. Ulmer. "Coding with skew polynomial rings". In: *Journal of Symbolic Computation* 44.12 (2009), pp. 1644–1656.

[BU11] D. Boucher and F. Ulmer. "A note on the dual codes of module skew codes". In: *IMA International Conference on Cryptography and Coding*. Springer. 2011, pp. 230–243.

[BU14b] D. Boucher and F. Ulmer. "Self-dual skew codes and factorization of skew polynomials". In: *Journal of Symbolic Computation* 60 (2014), pp. 47–61.

[BU14a] D. Boucher and F. Ulmer. "Linear codes using skew polynomials with automorphisms and derivations". In: *Designs, codes and cryptography* 70 (2014), pp. 405–431.

[BD18] M. Boulagouaz and A. Deajim. "Characterizations and properties of principal (f, σ, δ)-codes over rings". In: *arXiv preprint arXiv:1809.10409* (2018).

[BL13] M. Boulagouaz and A. Leroy. "(σ, δ)-codes". In: *Advances in Mathematics of Communications* 7.4 (2013), pp. 463–474.

[BCG+23] S. Bravyi, A. W. Cross, J. M. Gambetta, D. Maslov, P. Rall, and T. J. Yoder. "High-threshold and low-overhead fault-tolerant quantum memory". In: *arXiv preprint arXiv:2308.07915* (2023).

[BMS04] A. Brown, L. Minder, and A. Shokrollahi. "Probabilistic Decoding of Interleaved RS-Codes on the q-Ary Symmetric Channel". In: *IEEE International Symposium on Information Theory (ISIT)*. 2004, pp. 326–326.

[BMS05] A. Brown, L. Minder, and A. Shokrollahi. "Improved Decoding of Interleaved AG Codes". In: *IMA International Conference on Cryptography and Coding*. Springer. 2005, pp. 37–46.

[Buc65] B. Buchberger. “An algorithm for finding a basis for the residue class ring of a zero-dimensional polynomial ideal (in German)”. PhD thesis. Ph. D. thesis, University of Innsbruck, Austria, 1965.

[Buc83] B. Buchberger. “A note on the complexity of constructing Gröbner-bases”. In: *Computer Algebra: EUROCAL'83, European Computer Algebra Conference London, England, March 28–30, 1983 Proceedings*. Springer. 1983, pp. 137–145.

[BGR21] E. Byrne, H. Gluesing-Luerssen, and A. Ravagnani. “Fundamental Properties of Sum-Rank-Metric Codes”. In: *IEEE Transactions on Information Theory* 67.10 (2021), pp. 6456–6475. DOI: 10.1109/TIT.2021.3074190.

[CCE+20] H. Cai, J. Chrisnata, T. Etzion, M. Schwartz, and A. Wachter-Zeh. “Network-coding solutions for minimal combination networks and their sub-networks”. In: *IEEE Transactions on Information Theory* 66.11 (2020), pp. 6786–6798.

[CMST21] H. Cai, Y. Miao, M. Schwartz, and X. Tang. “A construction of maximally recoverable codes with order-optimal field size”. In: *IEEE Transactions on Information Theory* 68.1 (2021), pp. 204–212.

[CHK+93] A. R. Calderbank, A. R. Hammons, P. V. Kumar, N. J. Sloane, and P. Solé. “A linear construction for certain Kerdock and Preparata codes”. In: *Bulletin of the American Mathematical Society* 29.2 (1993), pp. 218–222.

[CRSS98] A. R. Calderbank, E. M. Rains, P. M. Shor, and N. J. Sloane. “Quantum error correction via codes over GF(4)”. In: *IEEE Transactions on Information Theory* 44.4 (1998), pp. 1369–1387.

[CGL+22] E. Camps-Moreno, E. Gorla, C. Landolina, E. L. García, U. Martínez-Peñas, and F. Salizzoni. “Optimal Anticodes, MSRD Codes, and Generalized Weights in the Sum-Rank Metric”. In: *IEEE Transactions on Information Theory* 68.6 (2022), pp. 3806–3822.

[CB12] X. Caruso and J. L. Borgne. “Some algorithms for skew polynomials over finite fields”. In: *arXiv preprint arXiv:1212.3582* (2012).

[CD18] X. Caruso and A. Durand. “Reed-Solomon-Gabidulin Codes”. In: *arXiv preprint arXiv:1812.09147* (2018).

[CD22] X. Caruso and A. Durand. “Duals of linearized Reed–Solomon codes”. In: *Designs, Codes and Cryptography* (2022), pp. 1–31.

[CLB17] X. Caruso and J. Le Borgne. “Fast multiplication for skew polynomials”. In: *Proceedings of the 2017 ACM on International Symposium on Symbolic and Algebraic Computation*. 2017, pp. 77–84.

[CS96] F. Chabaud and J. Stern. “The cryptographic security of the syndrome decoding problem for rank distance codes”. In: *Advances in Cryptology—ASIACRYPT'96: International Conference on the Theory and Applications of Cryptology and Information Security Kyongju, Korea, November 3–7, 1996 Proceedings*. Springer. 1996, pp. 368–381.

[CLZ15] B. Chen, S. Ling, and G. Zhang. “Application of constacyclic codes to quantum MDS codes”. In: *IEEE Transactions on Information Theory* 61.3 (2015), pp. 1474–1484.

[CYK+22] E. H. Chen, T. J. Yoder, Y. Kim, N. Sundaresan, S. Srinivasan, M. Li, A. D. Córcoles, A. W. Cross, and M. Takita. "Calibrated decoders for experimental quantum error correction". In: *Physical Review Letters* 128.11 (2022), p. 110504.

[CLX05] H. Chen, S. Ling, and C. Xing. "Quantum codes from concatenated algebraic-geometric codes". In: *IEEE transactions on information theory* 51.8 (2005), pp. 2915–2920.

[CO92] J Chen and P Owsley. "A burst-error-correcting algorithm for Reed-Solomon codes". In: *IEEE transactions on information theory* 38.6 (1992), pp. 1807–1812.

[CZ22] T. Chen and X. Zhang. "Sparse and Balanced MDS Codes Over Small Fields". In: *IEEE Transactions on Information Theory* 68.8 (2022), pp. 5112–5125. DOI: 10.1109/TIT.2022.3162524.

[CP88] K.-M. Cheung and F Pollara. "Phobos lander coding system: Software and analysis". In: *The Telecommunications and Data Acquisition Report* (1988).

[CLS09] H. Choi, W. Liu, and W. Sung. "VLSI implementation of BCH error correction for multilevel cell NAND flash memory". In: *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 18.5 (2009), pp. 843–847.

[CLSZ95] I. L. Chuang, R. Laflamme, P. W. Shor, and W. H. Zurek. "Quantum computers, factoring, and decoherence". In: *Science* 270.5242 (1995), pp. 1633–1635.

[CH13] H. Cohn and N. Heninger. "Approximate Common Divisors via Lattices". In: *The Open Book Series* 1.1 (2013), pp. 271–293.

[CS03] D. Coppersmith and M. Sudan. "Reconstructing Curves in Three (and Higher) Dimensional Space from Noisy Data". In: *ACM Symposium on the Theory of Computing*. 2003.

[CW90] D. Coppersmith and S. Winograd. "Matrix multiplication via arithmetic progressions". In: *Journal of Symbolic Computation* 9.3 (1990), pp. 251–280.

[CHH01] R. S. Coulter, G. Havas, and M. Henderson. "Giesbrecht's algorithm, the HFE cryptosystem and Ore's ps-polynomials". In: *Computer Mathematics*. World Scientific, 2001, pp. 36–45.

[CMP17] A. Couvreur, I. Márquez-Corbella, and R. Pellikaan. "Cryptanalysis of McEliece Cryptosystem Based on Algebraic Geometry Codes and their Subcodes". In: *IEEE Transactions on Information Theory* 63.8 (2017), pp. 5404–5418.

[CP20] A. Couvreur and I. Panaccione. "Power error locating pairs". In: *Designs, Codes and Cryptography* 88 (2020), pp. 1561–1593.

[CR20] A. Couvreur and H. Randriambololona. "Algebraic Geometry Codes and some Applications". In: *arXiv preprint arXiv:2009.01281* (2020).

[D'A22] G. D'Alconzo. "Code Equivalence in the Sum-Rank Metric: Hardness and Completeness". In: *Cryptology ePrint Archive* (2022).

[DSDY13] S. H. Dau, W. Song, Z. Dong, and C. Yuen. "Balanced sparsest generator matrices for MDS codes". In: *2013 IEEE International Symposium on Information Theory*. IEEE. 2013, pp. 1889–1893.

[DSY14b] S. H. Dau, W. Song, and C. Yuen. "On the existence of MDS codes over small fields with constrained generator matrices". In: *2014 IEEE International Symposium on Information Theory*. IEEE. 2014, pp. 1787–1791.

[DSY14a] S. H. Dau, W. Song, and C. Yuen. “On simple multiple access networks”. In: *IEEE Journal on Selected Areas in Communications* 33.2 (2014), pp. 236–249.

[Del78] P. Delsarte. “Bilinear forms over a finite field, with applications to coding theory”. In: *Journal of combinatorial theory, Series A* 25.3 (1978), pp. 226–241.

[Del75] P. Delsarte. “On subfield subcodes of modified Reed-Solomon codes (Corresp.)” In: *IEEE Transactions on Information Theory* 21.5 (1975), pp. 575–576.

[DHJ+11] T. K. Dikaliotis, T. Ho, S. Jaggi, S. Vyetrenko, H. Yao, M. Effros, J. Kliewer, and E. Erez. “Multiple-access network information-flow and correction codes”. In: *IEEE transactions on information theory* 57.2 (2011), pp. 1067–1079.

[DBU+21] H. Q. Dinh, T. Bag, A. K. Upadhyay, R. Bandi, and R. Tansuchat. “A class of skew cyclic codes and application in quantum codes construction”. In: *Discrete Mathematics* 344.2 (2021), p. 13.

[DS01] S. T. Dougherty and K. Shiromoto. “Maximum distance codes over rings of order 4”. In: *IEEE Transactions on Information Theory* 47.1 (2001), pp. 400–404.

[EB99] Y. Edel and J. Bierbrauer. “Recursive constructions for large caps”. In: *Bulletin of the Belgian Mathematical Society-Simon Stevin* 6.2 (1999), pp. 249–258.

[EGH03] H. El Gamal and A. R. Hammons. “On the design of algebraic space-time codes for MIMO block-fading channels”. In: *IEEE Transactions on Information Theory* 49.1 (2003), pp. 151–163.

[EWZ18] M. Elleuch, A. Wachter-Zeh, and A. Zeh. *A Public-Key Cryptosystem from Interleaved Goppa Codes*. 2018. arXiv: 1809.03024 [cs.IT].

[EKOO20] T. Etzion, S. Kurz, K. Otal, and F. Özbudak. “Subspace Packings: Constructions and Bounds”. en. In: *Designs, Codes and Cryptography* (Feb. 2020). ISSN: 1573-7586. DOI: 10.1007/s10623-020-00732-z. (Visited on 05/13/2020).

[EW18] T. Etzion and A. Wachter-Zeh. “Vector Network Coding Based on Subspace Codes Outperforms Scalar Linear Network Coding”. In: *IEEE Transactions on Information Theory* 64.4 (Apr. 2018), pp. 2460–2473. ISSN: 0018-9448. DOI: 10.1109/TIT.2018.2797183.

[EZ19] T. Etzion and H. Zhang. “Grassmannian Codes with New Distance Measures for Network Coding”. In: *IEEE Transactions on Information Theory* 65.7 (July 2019), pp. 4131–4142. ISSN: 0018-9448. DOI: 10.1109/TIT.2019.2899748.

[FT91] G.-L. Feng and K. K. Tzeng. “A generalization of the Berlekamp-Massey algorithm for multisequence shift-register synthesis with applications to decoding cyclic codes”. In: *IEEE Transactions on Information Theory* 37.5 (1991), pp. 1274–1287.

[FA66] S. Fisher and M. Alexander. “Matrices over a finite field”. In: *The American Mathematical Monthly* 73.6 (1966), pp. 639–641.

[For65] G. Forney. “On decoding BCH codes”. In: *IEEE Transactions on information theory* 11.4 (1965), pp. 549–557.

[Fou17] A. S. Foundation. *APACHE hadoop: HDFS Erasure Coding*. https://hadoop.apache.org/docs/stable/hadoop-project-dist/hadoop-hdfs/HDFSErasureCoding.html. Last Published: 2023-06-18. 2017.

[FND+20] B. Foxen, C. Neill, A. Dunsworth, P. Roushan, B. Chiaro, A. Megrant, J. Kelly, Z. Chen, K. Satzinger, R. Barends, et al. "Demonstrating a continuous set of two-qubit gates for near-term quantum algorithms". In: *Physical Review Letters* 125.12 (2020), p. 120504.

[Gab85] E. M. Gabidulin. "Theory of codes with maximum rank distance". In: *Problemy peredachi informatsii* 21.1 (1985), pp. 3–16.

[GRS15] P. Gaborit, O. Ruatta, and J. Schrek. "On the complexity of the rank syndrome decoding problem". In: *IEEE Transactions on Information Theory* 62.2 (2015), pp. 1006–1019.

[GY06] M. Gadouleau and Z. Yan. "GENp1-1: Properties of Codes with the Rank Metric". In: *IEEE Globecom 2006*. IEEE. 2006, pp. 1–5.

[GW18] J. Gao and Y. Wang. "u-Constacyclic codes over $\mathbb{F}_p + u\mathbb{F}_p$ and their applications of constructing new non-binary quantum codes". In: *Quantum Information Processing* 17 (2018), pp. 1–9.

[GU19] W. Geiselmann and F. Ulmer. *Skew Reed Muller codes*. 2019.

[Gie98] M. Giesbrecht. "Factoring in skew-polynomial rings over finite fields". In: *Journal of Symbolic Computation* 26.4 (1998), pp. 463–486.

[Gil52] E. N. Gilbert. "A comparison of signalling alphabets". In: *The Bell system technical journal* 31.3 (1952), pp. 504–522.

[Glu21] H. Gluesing-Luerssen. "Introduction to Skew-Polynomial Rings and Skew-Cyclic Codes". In: *Concise Encyclopedia of Coding Theory*. Chapman and Hall/CRC, 2021, pp. 151–180.

[GRS00] O. Goldreich, R. Rubinfeld, and M. Sudan. "Learning polynomials with queries: The highly noisy case". In: *SIAM Journal on Discrete Mathematics* 13.4 (2000), pp. 535–570.

[GSRM19] D. Gonçalves, S. Signorello, F. M. Ramos, and M. Médard. "Random Linear Network Coding on Programmable Switches". In: *2019 ACM/IEEE Symposium on Architectures for Networking and Communications Systems (ANCS)* (2019). DOI: 10.1109/ancs.2019.8901883.

[Goo23] A. Google Quantum. "Suppressing quantum errors by scaling a surface code logical qubit". In: *Nature* 614.7949 (2023), pp. 676–681.

[GG22] S. Gopi and V. Guruswami. "Improved maximally recoverable LRCs using skew polynomials". In: *IEEE Transactions on Information Theory* (2022).

[Gop70] V. D. Goppa. "A new class of linear correcting codes". In: *Problemy Peredachi Informatsii* 6.3 (1970), pp. 24–30.

[Gop71] V. D. Goppa. "A rational representation of codes and (L, G)-codes". In: *Problemy Peredachi Informatsii* 7.3 (1971), pp. 41–49.

[GS19] G. Greaves and J. Syatriadi. "Reed–Solomon codes over small fields with constrained generator matrices". In: *IEEE Transactions on Information Theory* 65.8 (2019), pp. 4764–4770.

[GG14] K. Guenda and T. A. Gulliver. "Quantum codes over rings". In: *International Journal of Quantum Information* 12.04 (2014), p. 1450020.

[Guo16] A. Guo. “High-Rate Locally Correctable Codes via Lifting”. In: *IEEE Trans. Inf. Theory* 62.12 (2016), pp. 6672–6682.

[GK16a] A. Guo and S. Kopparty. “List-decoding algorithms for lifted codes”. In: *IEEE Transactions on Information Theory* 62.5 (2016), pp. 2719–2725.

[GKS13] A. Guo, S. Kopparty, and M. Sudan. “New affine-invariant codes from lifting”. In: *Proc. the 4th Conf. Innov. Theor. Comput. Sc.* 2013, pp. 529–540.

[GOS17] F. Gursoy, E. S. Oztas, and I. Siap. “Reversible DNA codes using skew polynomial rings”. In: *Applicable Algebra in Engineering, Communication and Computing* 28.4 (2017), pp. 311–320.

[GK16b] V. Guruswami and S. Kopparty. “Explicit subspace designs”. In: *Combinatorica* 36.2 (2016), pp. 161–185.

[GX13] V. Guruswami and C. Xing. “List decoding Reed-Solomon, Algebraic-Geometric, and Gabidulin subcodes up to the Singleton bound”. In: *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*. 2013, pp. 843–852.

[GXY18] V. Guruswami, C. Xing, and C. Yuan. “Subspace designs based on algebraic function fields”. In: *Transactions of the American Mathematical Society* 370.12 (2018), pp. 8757–8775.

[HHD14] W. Halbawi, T. Ho, and I. Duursma. “Distributed Gabidulin codes for multiple-source network error correction”. In: *2014 International Symposium on Network Coding (NetCod)*. IEEE. 2014, pp. 1–6.

[HHYD14] W. Halbawi, T. Ho, H. Yao, and I. Duursma. “Distributed Reed-Solomon codes for simple multiple access networks”. In: *2014 IEEE International Symposium on Information Theory*. IEEE. 2014, pp. 651–655.

[HLD+18] W. Halbawi, Z. Liu, I. M. Duursma, H. Dau, and B. Hassibi. “Sparse and balanced Reed-Solomon and Tamo-Barg codes”. In: *IEEE Transactions on Information Theory* 65.1 (2018), pp. 118–130.

[HLH16] W. Halbawi, Z. Liu, and B. Hassibi. “Balanced Reed-Solomon codes for all parameters”. In: *2016 IEEE Information Theory Workshop (ITW)*. IEEE. 2016, pp. 409–413.

[HLMV24] M. Hamburg, E. Linstadt, D. Moore, and T. Vogelsang. “Unraveling codes: fast, robust, beyond-bound error correction for DRAM”. In: *arXiv preprint arXiv:2401.10688* (2024).

[Ham50] R. Hamming. “Error-Detecting and Error-Correcting Codes”. In: *The Bell Systems Technical Journal* 29.2 (Apr. 1950), pp. 147–160.

[HKC+94] A. R. Hammons, P. V. Kumar, A. R. Calderbank, N. J. Sloane, and P. Solé. “The $\mathbb{Z}_4$-linearity of Kerdock, Preparata, Goethals, and related codes”. In: *IEEE Transactions on Information Theory* 40.2 (1994), pp. 301–319.

[HV00] C Haslach and A. Vinck. “Efficient Decoding of Interleaved Linear Block Codes”. In: *IEEE International Symposium on Information Theory (ISIT)*. IEEE. 2000, p. 149.

[HV99] C. Haslach and A. H. Vinck. “A Decoding Algorithm With Restrictions for Array Codes”. In: *IEEE Transactions on Information Theory* 45.7 (1999), pp. 2339–2344.

[HKLW21] J. Hastings, A. Kanne, R. Li, and M. Wootters. “Wedge-lifted codes”. In: *2021 IEEE International Symposium on Information Theory (ISIT)*. IEEE. 2021, pp. 2990–2995.

[HS17] A. Heidarzadeh and A. Sprintson. “An algebraic-combinatorial proof technique for the GM-MDS conjecture”. In: *2017 IEEE International Symposium on Information Theory (ISIT)*. IEEE. 2017, pp. 11–15.

[HOW15] B. Hemenway, R. Ostrovsky, and M. Wootters. “Local correctability of expander codes”. In: *Information and Computation* 243 (2015), pp. 178–190.

[HW18] B. Hemenway and M. Wootters. “Linear-time list recovery of high-rate expander codes”. In: *Information and Computation* 261 (2018), pp. 202–218.

[Hil93] D. Hilbert. “Über die vollen Invariantensysteme (On the full invariant systems)”. In: *Mathematische Annalen*. 1893, pp. 313–373.

[Hir64] H. Hironaka. “Resolution of singularities of an algebraic variety over a field of characteristic zero”. In: *Annals of Mathematics* (1964), I:109–203, II:205–326.

[HS01] J. W. Hirschfeld and L. Storme. “The packing problem in statistics, coding theory and finite projective spaces: update 2001”. In: *Finite geometries*. Springer, 2001, pp. 201–246.

[Hoc59] A. Hocquenghem. “Codes Correcteurs d’Erreurs”. In: *Chiffres (Paris)* 2 (Sept. 1959), pp. 147–156.

[HPPV20] L. Holzbaur, R. Polyanskaya, N. Polyanskii, and I. Vorobyev. “Lifted reed-solomon codes with application to batch codes”. In: *2020 IEEE International Symposium on Information Theory (ISIT)*. IEEE. 2020, pp. 634–639.

[HPP+21] L. Holzbaur, R. Polyanskaya, N. Polyanskii, I. Vorobyev, and E. Yaakobi. “On lifted multiplicity codes”. In: *2020 IEEE Information Theory Workshop (ITW)*. IEEE. 2021, pp. 1–5.

[HP21] L. Holzbaur and N. Polyanskii. “Decoding of Lifted Affine-Invariant Codes”. In: *2020 IEEE Information Theory Workshop (ITW)*. IEEE. 2021, pp. 1–5.

[HPW19] L. Holzbaur, S. Puchinger, and A. Wachter-Zeh. “On error decoding of locally repairable and partial MDS codes”. In: *2019 IEEE Information Theory Workshop (ITW)*. IEEE. 2019, pp. 1–5.

[HB22] F. Hörmann and H. Bartz. “Efficient Decoding of Folded Linearized Reed-Solomon Codes in the Sum-Rank Metric”. In: *WCC 2022: The Twelfth International Workshop on Coding and Cryptography*. 2022. URL: `https://elib.dlr.de/146410/`.

[HBH22] F. Hörmann, H. Bartz, and A.-L. Horlemann. “Security Considerations for McEliece-like Cryptosystems Based on Linearized Reed-Solomon Codes in the Sum-Rank Metric”. In: *CBCrypto 2022: International Workshop on Code-Based Cryptography, May 29-30, 2022, Trondheim, Norway*. 2022.

[HBP22] F. Hörmann, H. Bartz, and S. Puchinger. “Error-Erasure Decoding of Linearized Reed-Solomon Codes in the Sum-Rank Metric”. In: (2022). DOI: `10.48550/ARXIV.2202.06758`. URL: `https://arxiv.org/abs/2202.06758`.

[Huf05] W. C. Huffman. “On the classification and enumeration of self-dual codes”. In: *Finite Fields and Their Applications* 11.3 (2005), pp. 451–490.

[IKOS04] Y. Ishai, E. Kushilevitz, R. Ostrovsky, and A. Sahai. "Batch codes and their applications". In: *Proceedings of the thirty-sixth annual ACM symposium on Theory of computing*. 2004, pp. 262–271.

[JHB22] T. Jerkovits, F. Hörmann, and H. Bartz. "Universal Decoding of Interleaved Linearized Reed–Solomon Codes in the Sum-Rank Metric". In: *Coding Theory and Cryptography: A conference in honor of Joachim Rosenthal's 60th birthday*. 2022.

[JTH04] J. Justesen, C. Thommesen, and T. Høholdt. "Decoding of Concatenated Codes with Interleaved Outer Codes". In: *IEEE International Symposium on Information Theory (ISIT)*. 2004, pp. 328–328.

[KDLM05] Z. Kadelburg, D. Dukic, M. Lukic, and I. Matic. "Inequalities of Karamata, Schur and Muirhead, and some applications". In: *The Teaching of Mathematics* 8.1 (2005), pp. 31–45.

[KZ11] X. Kai and S. Zhu. "Quaternary construction of quantum codes from cyclic codes over". In: *International Journal of Quantum Information* 9.02 (2011), pp. 689–700.

[Kam14] S. Kampf. "Bounds on Collaborative Decoding of Interleaved Hermitian Codes and Virtual Extension". In: *Designs, Codes and Cryptography* 70.1-2 (2014), pp. 9–25.

[KK16] J. Y. Kim and S. Kopparty. "Decoding Reed-Muller Codes Over Product Sets". In: *31st Conference on Computational Complexity (CCC 2016)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik. 2016.

[Kit03] A. Y. Kitaev. "Fault-tolerant quantum computation by anyons". In: *Annals of physics* 303.1 (2003), pp. 2–30.

[KK08] R. Koetter and F. R. Kschischang. "Coding for errors and erasures in random network coding". In: *IEEE Transactions on Information theory* 54.8 (2008), pp. 3579–3591.

[KV03] R. Koetter and A. Vardy. "Algebraic soft-decision decoding of Reed-Solomon codes". In: *IEEE Transactions on Information Theory* 49.11 (2003), pp. 2809–2825.

[Kop15] S. Kopparty. "List-decoding multiplicity codes". In: *Theory of Computing* 11.1 (2015), pp. 149–182.

[KRR+20] S. Kopparty, N. Resch, N. Ron-Zewi, S. Saraf, and S. Silas. "On list recovery of high-rate tensor codes". In: *IEEE Transactions on Information Theory* 67.1 (2020), pp. 296–316.

[KSY14] S. Kopparty, S. Saraf, and S. Yekhanin. "High-rate codes with sublinear-time decoding". In: *J. ACM* 61.5 (2014), pp. 1–20.

[KL97] V. Y. Krachkovsky and Y. X. Lee. "Decoding for Iterative Reed–Solomon Coding Schemes". In: *IEEE Transactions on Magnetics* 33.5 (1997), pp. 2740–2742.

[KL98] V. Y. Krachkovsky and Y. X. Lee. "Decoding of Parallel Reed–Solomon Codes with Applications to Product and Concatenated Codes". In: *IEEE International Symposium on Information Theory*. 1998, p. 55.

[KM96] K. Kühnle and E. W. Mayr. “Exponential space computation of Gröbner bases”. In: *Proceedings of the 1996 international symposium on Symbolic and algebraic computation*. 1996, pp. 63–71.

[LJY+15] C. Lai, S. Jiang, L. Yang, S. Lin, G. Sun, Z. Hou, C. Cui, and J. Cong. “Atlas: Baidu’s key-value storage system for cloud data”. In: *2015 31st Symposium on Mass Storage Systems and Technologies (MSST)*. IEEE. 2015, pp. 1–14.

[Lam86] T.-Y. Lam. “A general theory of Vandermonde Matrices”. In: *Expositions Mathematicae* 4 (1986), pp. 193–215.

[LL88a] T.-Y. Lam and A. Leroy. “Algebraic conjugacy classes and skew polynomial rings”. In: *Perspectives in ring theory*. Springer, 1988, pp. 153–203.

[LL88b] T.-Y. Lam and A. Leroy. “Vandermonde and Wronskian matrices over division rings”. In: *Journal of Algebra* 119.2 (1988), pp. 308–336.

[LL04] T. Y. Lam and A. Leroy. “Wedderburn polynomials over division rings, I”. In: *Journal of Pure and Applied Algebra* 186.1 (2004), pp. 43–76.

[LLO08] T. Lam, A Leroy, and A Ozturk. “Wedderburn polynomials over division rings, II”. In: *Contemporary Mathematics* 456 (2008), pp. 73–98.

[Lan93] G. Landsberg. “Über eine Anzahlbestimmung und eine damit zusammenhängende Reihe.” In: *Journal für die reine und angewandte Mathematik* 111 (1893), pp. 87–88.

[LS09] M. Langberg and A. Sprintson. “Recent results on the algorithmic complexity of network coding”. In: *Proc. of the 5th Workshop on Network Coding, Theory, and Applications*. 2009.

[LSB06] M. Langberg, A. Sprintson, and J. Bruck. “The encoding complexity of network coding”. In: *IEEE Transactions on Information Theory* 52.6 (2006), pp. 2386–2397.

[LN20] J. Lavauzelle and J. Nardi. “Weighted Lifted Codes: Local Correctabilities and Application to Robust Private Information Retrieval”. In: *IEEE Trans. Inf. Theory* 67.1 (2020), pp. 111–123.

[LG14] F. Le Gall. “Powers of tensors and fast matrix multiplication”. In: *Proceedings of the 39th international symposium on symbolic and algebraic computation*. 2014, pp. 296–303.

[Ler95] A. Leroy. “Pseudo linear transformations and evaluation in Ore extensions.” In: *Bulletin of the Belgian Mathematical Society-Simon Stevin* 2.3 (1995), pp. 321–347.

[Ler12] A. Leroy. “Noncommutative polynomial maps”. In: *Journal of Algebra and its Applications* 11.04 (2012), p. 1250076.

[LW19] R. Li and M. Wootters. “Lifted Multiplicity Codes and the Disjoint Repair Property”. In: *Leibniz Int. Proc. Inform.* (2019).

[LG17] S. Li and M. Gastpar. “Cooperative data exchange based on MDS codes”. In: *2017 IEEE International Symposium on Information Theory (ISIT)*. IEEE. 2017, pp. 1411–1415.

[LXW08] Z. Li, L.-J. Xing, and X.-M. Wang. "Quantum generalized Reed-Solomon codes: Unified framework for quantum maximum-distance-separable codes". In: *Physical Review A* 77.1 (2008), p. 012308.

[LN97] R. Lidl and H. Niederreiter. *Finite fields*. 20. Cambridge university press, 1997.

[LS01] S. Ling and P. Solé. "Type II Codes Over $\mathbb{F}_4 + u\mathbb{F}_4$". In: *European Journal of Combinatorics* 22.7 (2001), pp. 983–997. ISSN: 0195-6698.

[Lip90] R. J. Lipton. "Efficient checking of computations". In: *Annual Symposium on Theoretical Aspects of Computer Science*. Springer. 1990, pp. 207–215.

[LMK14] S. Liu, F. Manganiello, and F. R. Kschischang. "Kötter interpolation in skew polynomial rings". In: *Designs, codes and cryptography* 72.3 (2014), pp. 593–608.

[LMK15] S. Liu, F. Manganiello, and F. R. Kschischang. "Construction and decoding of generalized skew-evaluation codes". In: *2015 IEEE 14th Canadian Workshop on Information Theory (CWIT)*. IEEE. 2015, pp. 9–13.

[LMK17] S. Liu, F. Manganiello, and F. R. Kschischang. "Matroidal structure of skew polynomial rings with application to network coding". In: *Finite Fields and Their Applications* 46 (2017), pp. 326–346.

[LRS06] W. Liu, J. Rho, and W. Sung. "Low-power high-throughput BCH error correction VLSI design for multi-level cell NAND flash memories". In: *2006 IEEE Workshop on Signal Processing Systems Design and Implementation*. IEEE. 2006, pp. 303–308.

[LMM+21] H. H. López, B. Malmskog, G. L. Matthews, F. Piñero-González, and M. Wootters. "Hermitian-lifted codes". In: *Des. Codes Cryptogr.* 89.3 (2021), pp. 497–515.

[Lov18] S. Lovett. "MDS matrices over small fields: A proof of the GM-MDS conjecture". In: *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE. 2018, pp. 194–199.

[Lu06] H.-F. Lu. "On constructions of algebraic space-time codes with AM-PSK constellations satisfying rate-diversity tradeoff". In: *IEEE transactions on information theory* 52.7 (2006), pp. 3198–3209.

[LK05] H.-f. Lu and P. V. Kumar. "A unified construction of space-time codes with optimal rate-diversity tradeoff". In: *IEEE Transactions on Information Theory* 51.5 (2005), pp. 1709–1730.

[Luc78] E. Lucas. "Théorie des Fonctions Numériques Simplement Périodiques". In: *Amer. J. Math.* 1.2 (1878), pp. 184–196. ISSN: 00029327, 10806377.

[MGF22] F. Ma, J. Gao, and F.-W. Fu. "$(x^n - (a + bw), \xi, \eta)$-skew constacyclic codes over $\mathbb{F}_q + w\mathbb{F}_q$ and their applications in quantum codes". In: *Quantum Information Processing* 21.10 (2022), p. 348.

[MGLF21] F. Ma, J. Gao, J. Li, and F.-W. Fu. "(σ, δ)-Skew quasi-cyclic codes over the ring $\mathbb{Z}_4 + u\mathbb{Z}_4$". In: *Cryptography and Communications* 13 (2021), pp. 307–320.

[Mac61] F. J. MacWilliams. "Error-Correcting Codes for Multiple-Level Transmission". In: *Bell System Technical Journal* 40.1 (1961), pp. 281–308.

[Mac62] F. J. MacWilliams. “Combinatorial problems of elementary abelian groups”. PhD thesis. 1962.

[MS77] F. J. MacWilliams and N. J. A. Sloane. *The Theory of Error Correcting Codes*. Vol. 16. Elsevier, 1977.

[MBK16] R. Mahmood, A. Badr, and A. Khisti. “Convolutional codes with maximum column sum rank for network streaming”. In: *IEEE Transactions on Information Theory* 62.6 (2016), pp. 3039–3052.

[Mar18] U. Martínez-Peñas. “Skew and linearized Reed–Solomon codes and maximum sum rank distance codes over any division ring”. In: *Journal of Algebra* 504 (2018), pp. 587–612.

[Mar19] U. Martínez-Peñas. “Theory of supports for linear codes endowed with the sum-rank metric”. In: *Designs, Codes and Cryptography* 87.10 (2019), pp. 2295–2320.

[Mar20] U. Martínez-Peñas. “Hamming and simplex codes for the sum-rank metric”. In: *Designs, Codes and Cryptography* 88.8 (2020), pp. 1521–1539.

[Mar21] U. Martínez-Peñas. “Sum-rank BCH codes and cyclic-skew-cyclic codes”. In: *IEEE Transactions on Information Theory* 67.8 (2021), pp. 5149–5167.

[Mar22] U. Martínez-Peñas. “A general family of MSRD codes and PMDS codes with smaller field sizes from extended Moore matrices”. In: *SIAM Journal on Discrete Mathematics* 36.3 (2022), pp. 1868–1886.

[MK19b] U. Martínez-Peñas and F. R. Kschischang. “Reliable and Secure Multishot Network Coding using Linearized Reed-Solomon Codes”. In: *IEEE Transactions on Information Theory* (2019).

[MK19c] U. Martínez-Peñas and F. R. Kschischang. “Universal and dynamic locally repairable codes with maximal recoverability via sum-rank codes”. In: *IEEE Transactions on Information Theory* (2019).

[MK19a] U. Martínez-Peñas and F. R. Kschischang. “Evaluation and interpolation over multivariate skew polynomial rings”. In: *Journal of algebra* 525 (2019), pp. 111–139.

[MN20] U. Martínez-Peñas and D. Napp. “Locally repairable convolutional codes with sliding window repair”. In: *IEEE Transactions on Information Theory* 66.8 (2020), pp. 4935–4947.

[MSK+22] U. Martínez-Peñas, M. Shehadeh, F. R. Kschischang, et al. “Codes in the sum-rank metric: Fundamentals and applications”. In: *Foundations and Trends® in Communications and Information Theory* 19.5 (2022), pp. 814–1031.

[Mas92] J. L. Massey. “Deep-space communications and coding: A marriage made in heaven”. In: *Advanced Methods for Satellite and Deep Space Communications: Proceedings of an International Seminar Organized by Deutsche Forschungsanstalt für Luft-und Raumfahrt (DLR) Bonn, Germany, September 1992*. Springer. 1992, pp. 1–17.

[MMS21] G. L. Matthews, A. W. Murphy, and W. Santos. “Fractional decoding of codes from Hermitian curves”. In: *2021 IEEE International Symposium on Information Theory (ISIT)*. IEEE. 2021, pp. 515–520.

[McE78] R. J. McEliece. “A public-key cryptosystem based on algebraic”. In: *Coding Thv* 4244 (1978), pp. 114–116.

[MK90] J. J. Metzner and E. J. Kapturowski. “A General Decoding Technique Applicable to Replicated File Disagreement Location and Concatenated Code Decoding”. In: *IEEE Transactions on Information Theory* 36.4 (1990), pp. 911–917.

[MME+13] R. Micheloni, A. Marelli, K. Eshghi, A Marelli, and R Micheloni. “BCH for solid-state-drives”. In: *Inside Solid State Drives (SSDs)* (2013), pp. 259–292.

[MZ23] D. Minzer and K. Z. Zheng. “Optimal testing of generalized reed-muller codes in fewer queries”. In: *2023 IEEE 64th Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE. 2023, pp. 206–233.

[MHU14] M. Mondelli, S. H. Hassani, and R. L. Urbanke. “From polar to Reed-Muller codes: A technique to improve the finite-length performance”. In: *IEEE Transactions on Communications* 62.9 (2014), pp. 3084–3091.

[Muk78] A. Mukhopadhyay. “Lower bounds on $mt(r, s)$”. In: *Journal of Combinatorial Theory, Series A* 25.1 (1978), pp. 1–13.

[Mul54] D. E. Muller. “Application of Boolean algebra to switching circuit design and to error detection”. In: *Transactions of the IRE professional group on electronic computers* 3 (1954), pp. 6–12.

[MLR+14] S. Muralidhar, W. Lloyd, S. Roy, C. Hill, E. Lin, W. Liu, S. Pan, S. Shankar, V. Sivakumar, L. Tang, et al. “f4: Facebook’s warm BLOB storage system”. In: *11th USENIX Symposium on Operating Systems Design and Implementation (OSDI 14)*. 2014, pp. 383–398.

[Nak10] T. Nakashima. “AG codes from vector bundles”. In: *Designs, Codes and Cryptography* 57 (2010), pp. 107–115.

[Ner22] A. Neri. “Twisted linearized Reed-Solomon codes: A skew polynomial framework”. In: *Journal of Algebra* 609 (2022), pp. 792–839.

[Nie13] J. S. Nielsen. “Generalised Multi-Sequence Shift-Register Synthesis Using Module Minimisation”. In: *IEEE International Symposium on Information Theory (ISIT)*. 2013, pp. 882–886.

[NU09] R. W. Nóbrega and B. F. Uchôa-Filho. “Multishot codes for network coding: Bounds and a multilevel construction”. In: *2009 IEEE International Symposium on Information Theory*. IEEE. 2009, pp. 428–432.

[NB20b] I. K. Noskov and S. Bezzateev. “One Realization of the Generalized (L, G)-Codes”. In: *2020 Wave Electronics and its Application in Information and Telecommunication Systems (WECONF)*. IEEE. 2020, pp. 1–5.

[NB20a] I. K. Noskov and S. Bezzateev. “Effective implementation of modern McEliece cryptosystem on generalized (L, G)-codes”. In: *Journal Scientific and Technical Of Information Technologies, Mechanics and Optics* 128.4 (2020), pp. 539–544.

[Ore33] Ø. Ore. “Theory of non-commutative polynomials”. In: *Annals of mathematics* (1933), pp. 480–508.

[OPB21] C. Ott, S. Puchinger, and M. Bossert. "Bounds and genericity of sum-rank-metric codes". In: *2021 XVII International Symposium" Problems of Redundancy in Information and Control Systems"(REDUNDANCY)*. IEEE. 2021, pp. 119–124.

[OJ02] A. V. Ourivski and T. Johansson. "New technique for decoding codes in the rank metric and its cryptography applications". In: *Problems of Information Transmission* 38 (2002), pp. 237–246.

[Ows88] P. A. Owsley. *Burst error correction extensions for Reed-Solomon codes*. University of Idaho, 1988.

[Par07] F. Parvaresh. "Algebraic List-Decoding of Error-Correcting Codes". PhD thesis. University of California, San Diego, 2007.

[PV04] F. Parvaresh and A. Vardy. "Multivariate Interpolation Decoding Beyond the Guruswami–Sudan Radius". In: *Allerton Conference on Communication, Control and Computing*. 2004.

[PIP22] S. Patel, H. Islam, and O. Prakash. "(f, σ, δ)-skew Polycyclic Codes and Their Applications to Quantum Codes". In: *International Journal of Theoretical Physics* 61.2 (2022), p. 47.

[PP22] S. Patel and O. Prakash. "(θ, δ_θ)-Cyclic codes over $\mathbb{F}_q[u, v]/ < u^2-u, v^2-v, uv-vu >$". In: *Designs, Codes and Cryptography* 90.11 (2022), pp. 2763–2781.

[Pet60] W. Peterson. "Encoding and error-correction procedures for the Bose-Chaudhuri codes". In: *IRE Transactions on information theory* 6.4 (1960), pp. 459–470.

[PW72] W. Peterson and E. J. Weldon. *Error-correcting codes*. MIT press, 1972.

[PV19a] N. Polyanskii and I. Vorobyev. "Constructions of batch codes via finite geometry". In: *2019 IEEE International Symposium on Information Theory (ISIT)*. IEEE. 2019, pp. 360–364.

[PV19b] N. Polyanskii and I. Vorobyev. "Trivariate lifted codes with disjoint repair groups". In: *2019 XVI International Symposium" Problems of Redundancy in Information and Control Systems"(REDUNDANCY)*. IEEE. 2019, pp. 64–68.

[PVS23] O. Prakash, R. K. Verma, and A. Singh. "Quantum and LCD codes from skew constacyclic codes over a finite non-chain ring". In: *Quantum Information Processing* 22.200 (2023), p. 25.

[Pra57] E. Prange. "Cyclic error-correcting codes in two symbols". In: *AFCRC-TN-57-103* (1957).

[Pra85] E. Prange. "Some cyclic error-correcting codes with simple decoding algorithms". In: *AFCRC-TN-58-156* (1985).

[PRR22] S. Puchinger, J. Renner, and J. Rosenkilde. "Generic decoding in the sum-rank metric". In: *IEEE Transactions on Information Theory* 68.8 (2022), pp. 5075–5097.

[PR17] S. Puchinger and J. Rosenkilde. "Decoding of Interleaved Reed–Solomon Codes Using Improved Power Decoding". In: *IEEE International Symposium on Information Theory (ISIT)*. 2017.

[PR21] S. Puchinger and J. Rosenkilde. “Bounds on List Decoding of Linearized Reed-Solomon Codes”. In: *2021 IEEE International Symposium on Information Theory (ISIT)*. IEEE. 2021, pp. 154–159.

[PRB19] S. Puchinger, J. Rosenkilde, and I. Bouw. “Improved Power Decoding of Interleaved One-Point Hermitian Codes”. In: *Designs, Codes and Cryptography* 87.2-3 (2019), pp. 589–607.

[PRS21] S. Puchinger, J. Rosenkilde, and G. Solomatov. “Improved power decoding of algebraic geometry codes”. In: *2021 IEEE International Symposium on Information Theory (ISIT)*. IEEE. 2021, pp. 509–514.

[PW18] S. Puchinger and A. Wachter-Zeh. “Fast operations on linearized polynomials and their applications in coding theory”. In: *Journal of Symbolic Computation* 89 (2018), pp. 194–215.

[QMG09] J. Qian, W. Ma, and W. Guo. “Quantum codes from cyclic codes over finite ring”. In: *International Journal of Quantum Information* 7.06 (2009), pp. 1277–1283.

[RK18] A. Ravagnani and F. R. Kschischang. “Adversarial network coding”. In: *IEEE Transactions on Information Theory* 65.1 (2018), pp. 198–219.

[Ree54] I. Reed. “A Class of Multiple-Error-Correcting Codes and the Decoding Scheme”. In: *IRE Trans. Inf. Theory* 4.4 (Sept. 1954), pp. 38–49. ISSN: 2168-2690.

[RS60] I. Reed and G. Solomon. “Polynomial codes over certain finite fields”. In: *Journal of the society for industrial and applied mathematics* 8.2 (1960), pp. 300–304.

[RPW21] J. Renner, S. Puchinger, and A. Wachter-Zeh. “LIGA: a cryptosystem based on the hardness of rank-metric list and interleaved decoding”. In: *Designs, Codes and Cryptography* 89 (2021), pp. 1279–1319.

[RS12] N. Ron-Zewi and M. Sudan. “A new upper bound on the query complexity for testing generalized reed-muller codes”. In: *International Workshop on Approximation Algorithms for Combinatorial Optimization*. Springer. 2012, pp. 639–650.

[Ros18] J. Rosenkilde. “Power Decoding Reed–Solomon Codes up to the Johnson Radius”. In: *Advances in Mathematics of Communications* 12.1 (2018), pp. 81–106.

[RS21] J. Rosenkilde and A. Storjohann. “Algorithms for simultaneous Hermite–Padé approximations”. In: *Journal of Symbolic Computation* 102 (2021), pp. 279–303.

[Rot06] R. Roth. *Introduction to Coding Theory*. Cambridge University Press, 2006.

[Rot91] R. M. Roth. “Maximum-rank array codes and their application to crisscross error correction”. In: *IEEE transactions on Information Theory* 37.2 (1991), pp. 328–336.

[RV14] R. M. Roth and P. O. Vontobel. “Coding for Combined Block–Symbol Error Correction”. In: *IEEE Transactions on Information Theory* 60.5 (2014), pp. 2697–2713.

[RS96] R. Rubinfeld and M. Sudan. "Robust characterizations of polynomials with applications to program testing". In: *SIAM Journal on Computing* 25.2 (1996), pp. 252–271.

[SSB07] G. Schmidt, V. Sidorenko, and M. Bossert. "Enhancing the Correcting Radius of Interleaved Reed–Solomon Decoding Using Syndrome Extension Techniques". In: *IEEE International Symposium on Information Theory (ISIT)*. 2007, pp. 1341–1345.

[SSB05] G. Schmidt, V. R. Sidorenko, and M. Bossert. "Interleaved Reed–Solomon Codes in Concatenated Code Designs". In: *IEEE Information Theory Workshop*. 2005, 5–pp.

[SSB09] G. Schmidt, V. R. Sidorenko, and M. Bossert. "Collaborative Decoding of Interleaved Reed-Solomon Codes and Concatenated Code Designs". In: *IEEE Transactions on Information Theory* 55.7 (2009), pp. 2991–3012.

[SSB10] G. Schmidt, V. R. Sidorenko, and M. Bossert. "Syndrome Decoding of Reed–Solomon Codes Beyond Half the Minimum Distance Based on Shift-Register Synthesis". In: *IEEE Transactions on Information Theory* 56.10 (2010), pp. 5245–5252.

[Seg55] B. Segre. "Ovals in a finite projective plane". In: *Canadian Journal of Mathematics* 7 (1955), pp. 414–416.

[Sen11] N. Sendrier. "Decoding one out of many". In: *International Workshop on Post-Quantum Cryptography*. Springer. 2011, pp. 51–67.

[Sha79] P. Shankar. "On BCH codes over arbitrary integer rings". In: *IEEE Transactions on Information Theory* 25.4 (1979), pp. 480–483.

[Sha48] C. E. Shannon. "A mathematical theory of communication". In: *Bell System Technical Journal* 27 (1948), pp. 379–423, 623–656.

[SB18] A. Sharma and M. Bhaintwal. "A class of skew-cyclic codes over $\mathbb{Z}_4 + u\mathbb{Z}_4$ with derivation". In: *Advances in Mathematics of Communications* 12.4 (2018).

[She20] J. Sheekey. "New semifields and new MRD codes from skew polynomial rings". In: *Journal of the London Mathematical Society* 101.1 (2020), pp. 432–456.

[SK21] M. Shehadeh and F. R. Kschischang. "Space–time codes from sum-rank codes". In: *IEEE Transactions on Information Theory* 68.3 (2021), pp. 1614–1637.

[SM81] N. A. Shekhunova and E. T. Mironchikov. "Cyclic (L, g)-Codes". In: *Problemy Peredachi Informatsii* 17.2 (1981), pp. 3–9.

[SLGK20] V. Sidorenko, W. Li, O. Günlü, and G. Kramer. "Skew convolutional codes". In: *Entropy* 22.12 (2020), p. 1364.

[SS11] V. R. Sidorenko and G. Schmidt. "A linear algebraic approach to multisequence shift-register synthesis". In: *Problems of Information Transmission* 47.2 (2011), pp. 149–165.

[SKK08] D. Silva, F. R. Kschischang, and R. Koetter. "A rank-metric approach to error control in random network coding". In: *IEEE transactions on information theory* 54.9 (2008), pp. 3951–3967.

[Sin64] R. Singleton. "Maximum distance q^n-array codes". In: *IEEE Transactions on Information Theory* 10.2 (1964), pp. 116–118.

[Sle74] D. Slepian. *Key Papers in the Development of Information Theory*. IEEE Press selected reprint series. IEEE Press, 1974. ISBN: 9780879420277.

[SC18] W. Song and K. Cai. "Generalized reed-solomon codes with sparsest and balanced generator matrices". In: *2018 IEEE International Symposium on Information Theory (ISIT)*. IEEE. 2018, pp. 1–5.

[SWY+12] W. Song, X. Wang, C. Yuen, T. J. Li, and R. Feng. "Error correction for cooperative data exchange". In: *IEEE communications letters* 16.11 (2012), pp. 1856–1859.

[SZHW23] Y. Song, J. Zhang, X. Huang, and W. Wu. "Blockwise rank decoding problem and LRPC codes: cryptosystems with smaller sizes". In: *International Conference on the Theory and Application of Cryptology and Information Security*. Springer. 2023, pp. 284–316.

[Spi77] E. Spiegel. "Codes over $\mathbb{Z}_m$". In: *Information and control* 35.1 (1977), pp. 48–51.

[Spi78] E. Spiegel. "Codes over $\mathbb{Z}_m$, revisited". In: *Information and Control* 37.1 (1978), pp. 100–104.

[Ste99] A. M. Steane. "Quantum reed-muller codes". In: *IEEE Transactions on Information Theory* 45.5 (1999), pp. 1701–1703.

[SHN19] A. M. Subramaniam, A. Heidarzadeh, and K. R. Narayanan. "Collaborative decoding of polynomial codes for distributed computation". In: *2019 IEEE Information Theory Workshop (ITW)*. IEEE. 2019, pp. 1–5.

[STV99] M. Sudan, L. Trevisan, and S. Vadhan. "Pseudorandom generators without the XOR lemma". In: *Proceedings of the thirty-first annual ACM symposium on Theory of computing*. 1999, pp. 537–546.

[SKHN76] Y. Sugiyama, M. Kasahara, S. Hirasawa, and T. Namekawa. "Further results on Goppa codes and their applications to constructing efficient binary codes". In: *IEEE Transactions on Information Theory* 22.5 (1976), pp. 518–526.

[SRZ06] F. Sun, K. Rose, and T. Zhang. "On the use of strong BCH codes for improving multilevel NAND flash memory storage capacity". In: *IEEE Workshop on Signal Processing Systems (SiPS): Design and Implementation*. Vol. 5. 2006.

[SYK+23] N. Sundaresan, T. J. Yoder, Y. Kim, M. Li, E. H. Chen, G. Harper, T. Thorbeck, A. W. Cross, A. D. Córcoles, and M. Takita. "Demonstrating multi-round subsystem quantum error correction using matching and maximum likelihood decoders". In: *Nature Communications* 14.1 (2023), p. 2852.

[Sup23] D. Suprijanto. "Linear codes and cyclic codes over finite rings and their generalizations: a survey". In: *Electronic Journal of Graph Theory and Applications (EJGTA)* 11.2 (2023), pp. 467–490.

[ST21] D. Suprijanto and H. C. Tang. "Skew cyclic codes over $\mathbb{Z}_4+v\mathbb{Z}_4$ with derivation". In: *arXiv preprint arXiv:2110.01580* (2021).

[SW17] S. Szabo and J. A. Wood. "Properties of dual codes defined by nondegenerate forms". In: *Journal of Algebra Combinatorics Discrete Structures and Applications* (2017), pp. 105–113.

[TZKD16] Y. Tang, S. Zhu, X. Kai, and J. Ding. “New quantum codes from dual-containing cyclic codes over finite rings”. In: *Quantum Information Processing* 15 (2016), pp. 4489–4500.

[The22] The Sage Developers. *SageMath, the Sage Mathematics Software System (Version 9.5)*. https://www.sagemath.org. 2022.

[Var57] R. R. Varshamov. “Estimate of the number of signals in error correcting codes”. In: *Docklady Akad. Nauk, SSSR* 117 (1957), pp. 739–741.

[VPIS22] R. K. Verma, O. Prakash, H. Islam, and A. Singh. “New non-binary quantum codes from skew constacyclic and additive skew constacyclic codes”. In: *The European Physical Journal Plus* 137.213 (2022), p. 13.

[Vid10] M. Viderman. “A Note on high-rate Locally Testable Codes with sublinear query complexity.” In: *Electron. Colloquium Comput. Complex.* Vol. 17. 2010, p. 171.

[VG13] J. Von Zur Gathen and J. Gerhard. *Modern computer algebra*. Cambridge university press, 2013.

[WZB14] A. Wachter-Zeh, A. Zeh, and M. Bossert. “Decoding Interleaved Reed–Solomon Codes Beyond Their Joint Error-Correcting Capability”. In: *Designs, Codes and Cryptography* 71.2 (2014), pp. 261–281.

[WLLG20] J. Wang, R. Li, Y. Liu, and G. Guo. “Some negacyclic BCH codes and quantum codes”. In: *Quantum Information Processing* 19.2 (2020), p. 74.

[WDPZ11] X. Wang, G. Dong, L. Pan, and R. Zhou. “Error Correction Codes and Signal Processing in Flash Memory”. In: *Flash Memories*. IntechOpen, 2011, pp. 57–82. ISBN: 978-953-307-272-2.

[Wir88] M. Wirtz. “On the parameters of Goppa codes”. In: *IEEE transactions on information theory* 34.5 (1988), pp. 1341–1343.

[Woo99] J. A. Wood. “Duality for modules over finite rings and applications to coding theory”. In: *American journal of Mathematics* (1999), pp. 555–575.

[Woo08] J. A. Wood. “Code equivalence characterizes finite Frobenius rings”. In: *Proceedings of the American Mathematical Society* 136.2 (2008), pp. 699–706.

[Woo09] J. A. Wood. “Foundations of linear codes defined over finite modules: the extension theorem and the MacWilliams identities”. In: *Codes over rings*. World Scientific, 2009, pp. 124–190.

[Wu15] L. Wu. “Revisiting the multiplicity codes: A new class of high-rate locally correctable codes”. In: *2015 53rd Annual Allerton Conference on Communication, Control, and Computing (Allerton)*. IEEE. 2015, pp. 509–513.

[WBC+21] Y. Wu, W.-S. Bao, S. Cao, F. Chen, M.-C. Chen, X. Chen, T.-H. Chung, H. Deng, Y. Du, D. Fan, et al. “Strong quantum computational advantage using a superconducting quantum processor”. In: *Physical review letters* 127.18 (2021), p. 180501.

[YS11] M. Yan and A. Sprintson. “Weakly secure network coding for wireless cooperative data exchange”. In: *2011 IEEE Global Telecommunications Conference-GLOBECOM 2011*. IEEE. 2011, pp. 1–5.

[YSZ14] M. Yan, A. Sprintson, and I. Zelenko. "Weakly secure data exchange with generalized Reed-Solomon codes". In: *2014 IEEE International Symposium on Information Theory*. IEEE. 2014, pp. 1366–1370.

[YEC11] C. Yang, Y. Emre, and C. Chakrabarti. "Product code schemes for error correction in MLC NAND flash memories". In: *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 20.12 (2011), pp. 2302–2314.

[YD04] S. Yekhanin and I. Dumer. "Long nonbinary codes exceeding the Gilbert-Varshamov bound for any fixed distance". In: *IEEE transactions on information theory* 50.10 (2004), pp. 2357–2362.

[YH18a] H. Yildiz and B. Hassibi. "Further progress on the GM-MDS conjecture for Reed-Solomon codes". In: *2018 IEEE International Symposium on Information Theory (ISIT)*. IEEE. 2018, pp. 16–20.

[YH18b] H. Yildiz and B. Hassibi. "Optimum linear codes with support constraints over small fields". In: *2018 IEEE Information Theory Workshop (ITW)*. IEEE. 2018, pp. 1–5.

[YH20] H. Yildiz and B. Hassibi. "Gabidulin codes with support constrained generator matrices". In: *IEEE Transactions on Information Theory* 66.6 (2020), pp. 3638–3649.

[YL18a] J.-H. Yu and H.-A. Loeliger. "Simultaneous Partial Inverses and Decoding Interleaved Reed–Solomon Codes". In: *IEEE Transactions on Information Theory* 64.12 (2018), pp. 7511–7528.

[YL18b] J.-H. Yu and H.-A. Loeliger. "Simultaneous Partial Inverses and Decoding Interleaved Reed–Solomon Codes". In: *IEEE Transactions on Information Theory* 64.12 (2018), pp. 7511–7528.

[Zha10] Y. Zhang. "A secret sharing scheme via skew polynomials". In: *2010 International Conference on Computational Science and Its Applications*. IEEE. 2010, pp. 33–38.

Publications Containing Parts of this Dissertation

[HLN+21b] L. Holzbaur, H. Liu, A. Neri, S. Puchinger, J. Rosenkilde, V. Sidorenko, and A. Wachter-Zeh. "Success probability of decoding interleaved alternant codes". In: *2020 IEEE Information Theory Workshop (ITW)*. IEEE. 2021, pp. 1–5.

[HLN+21a] L. Holzbaur, H. Liu, A. Neri, S. Puchinger, J. Rosenkilde, V. Sidorenko, and A. Wachter-Zeh. "Decoding of interleaved alternant codes". In: *IEEE Transactions on Information Theory* 67.12 (2021), pp. 8016–8033.

[HLH+22] C.-C. Huang, H. Liu, L. Holzbaur, S. Puchinger, and A. Wachter-Zeh. "List decoding of 2-interleaved binary alternant codes". In: *2022 IEEE International Symposium on Information Theory (ISIT)*. IEEE. 2022, pp. 2338–2343.

[LHP+22] H. Liu, L. Holzbaur, N. Polianskii, S. Puchinger, and A. Wachter-Zeh. "Quadratic Curve-Lifted Reed-Solomon Codes". In: *12th International Workshop on Coding and Cryptography (WCC)*. 2022.

[LOU23] H. Liu, C. Ott, and F. Ulmer. "A Gröbner Approach to Dual-Containing Cyclic Left Module (θ, δ)-Codes $Rg/Rf \subset R/Rf$ over Finite Commutative Frobenius Rings". In: *arXiv preprint arXiv:2308.13395* (2023).

[LPZW21] H. Liu, S. Pircher, A. Zeh, and A. Wachter-Zeh. "Decoding of (Interleaved) Generalized Goppa Codes". In: *2021 IEEE International Symposium on Information Theory (ISIT)*. IEEE. 2021, pp. 664–669.

[LPVW21] H. Liu, N. Polianskii, I. Vorobyev, and A. Wachter-Zeh. "Almost affinely disjoint subspaces". In: *Finite Fields and Their Applications* 75 (2021), p. 101879.

[LWP+20] H. Liu, H. Wei, S. Puchinger, A. Wachter-Zeh, and M. Schwartz. "On the Gap between Scalar and Vector Solutions of Generalized Combination Networks". In: *2020 IEEE International Symposium on Information Theory (ISIT)*. IEEE. 2020, pp. 1646–1651.

[LWP+21] H. Liu, H. Wei, S. Puchinger, A. Wachter-Zeh, and M. Schwartz. "On the gap between scalar and vector solutions of generalized combination networks". In: *IEEE Transactions on Information Theory* 67.8 (2021), pp. 5580–5591.

[LWWS23] H. Liu, H. Wei, A. Wachter-Zeh, and M. Schwartz. "Linearized Reed-Solomon Codes with Support-Constrained Generator Matrix". In: *2023 IEEE Information Theory Workshop (ITW)*. IEEE. 2023, pp. 7–12.

Other Joint Publications

[BHL+22] H. Bartz, L. Holzbaur, H. Liu, S. Puchinger, J. Renner, A. Wachter-Zeh, et al. "Rank-metric codes and their applications". In: *Foundations and Trends® in Communications and Information Theory* 19.3 (2022), pp. 390–546.

[HLPW19] L. Holzbaur, H. Liu, S. Puchinger, and A. Wachter-Zeh. "On decoding and applications of interleaved Goppa codes". In: *2019 IEEE International Symposium on Information Theory (ISIT)*. IEEE. 2019, pp. 1887–1891.

[MXP+22] G. Maringer, M. Xhemrishi, S. Puchinger, K. Garb, H. Liu, T. Jerkovits, L. Kürzinger, M. Hiller, and A. Wachter-Zeh. "Analysis of Communication Channels Related to Physical Unclonable Functions". In: *12th International Workshop on Coding and Cryptography (WCC)*. 2022.

[OLW22] C. Ott, H. Liu, and A. Wachter-Zeh. "Covering properties of sum-rank metric codes". In: *2022 58th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*. IEEE. 2022, pp. 1–7.

[OLW23] C. Ott, H. Liu, and A. Wachter-Zeh. "Geometrical Properties of Balls in Sum-Rank Metric". In: *WSA & SCC 2023; 26th International ITG Workshop on Smart Antennas and 13th Conference on Systems, Communications, and Coding*. VDE. 2023, pp. 1–6.

[PHL+22] A. Porwal, L. Holzbaur, H. Liu, J. Renner, A. Wachter-Zeh, and V. Weger. "Interleaved Prange: a new generic decoder for interleaved codes". In: *International Conference on Post-Quantum Cryptography*. Springer. 2022, pp. 69–88.

[SL23] H. Sauerbier Couvée and H. Liu. "Notes on the Sum-Rank Weight of a Matrix with Restricted Rank". In: *arXiv preprint arXiv:2311.10159* (2023).