THE GEOMETRIC REPRESENTATIONS OF RANK-METRIC CODES

by
ALTAN BERDAN KILIÇ

Submitted to the Graduate School of Engineering and Natural Sciences
in partial fulfilment of
the requirements for the degree of Master of Science

Sabancı University
July 2021

THE GEOMETRIC REPRESENTATIONS OF RANK-METRIC CODES

Approved by:

Date of Approval: 12/07/2021

ABSTRACT

THE GEOMETRIC REPRESENTATIONS OF RANK-METRIC CODES

ALTAN BERDAN KILIÇ

MATHEMATICS M.A. THESIS, JULY 2021

Thesis Supervisor: Prof. Dr. Michel Lavrauw

Keywords: rank-metric code, MRD code, tensor, tensor rank, complexity, MTR code, semifield, Segre variety.

In this thesis, geometric representations of rank-metric codes have been examined as well as their connection with algebraic coding theory and complexity theory. Given a vector code, we introduced an algorithm using the well-known field reduction map from projective geometry to get the corresponding rank-metric code. Following that correspondence, we revisited the codes that satisfy the analogues of the Singleton bound, called maximum rank distance(MRD) codes, and show that there is a one-to-one correspondence to finite semifields if they are additive. Given a semifield, we get a tensor associated to it. Tensor rank of various objects have been analyzed and its relation with complexity theory is explained in detail. In 1977, Kruskal proposed a lower bound on tensor rank and the codes that satisfy this bound are called minimal tensor rank(MTR) codes. We state an open problem on the existence of MTR codes deducing from the analyzed cases so far. We have solved the existence problem and proposed an attack on the characterization of all possible solutions using the algorithm Snakes and Ladders with the help of the computer algebra system GAP.

ÖZET

RANK-METRIK KODLARIN GEOMETRIK GÖSTERIMLERI

ALTAN BERDAN KILIÇ

MATEMATİK YÜKSEK LİSANS TEZİ, TEMMUZ 2021

Tez Danışmanı: Prof. Dr. Michel Lavrauw

Anahtar Kelimeler: rank-metrik kod, MRD kod, tensör, tensör rank, karmaşıklık, MTR kod, yarı cisim, Segre varyete

Bu tezde, rank-metrik kodların geometrik gösterimleri, cebirsel kodlama teorisi ve karmaşıklık teorisi ile olan ilişkileriyle beraber incelenmiştir. Bir vektör kodu verildiğinde, buna karşılık gelen rank-metrik kodunu bulmak için izdüşümsel geometride iyi bilinen cisim azaltma fonksiyonunu kullanan bir algoritma sunulmuştur. Bu ilişkiden yola çıkarak, Singleton sınırının analogunu sağlayan maksimum rank uzaklığı(MRD) kodlarının toplamsal olmaları durumunda yarı cisimler ile aralarında birebir eşleme olduğu gösterilmiştir. Bir yarı cisim verildiğinde, ona karşılık gelen tensör elde edilir. Çeşitli nesnelerin tensör rankları analiz edilip, karmaşıklık teorisi ile olan ilişkileri detaylı bir şekilde incelenmiştir. 1977'de Kruskal tensör rank için bir alt sınır sunmuş ve bu sınırı sağlayan kodlara minimal tensör rank(MTR) kodlar denilmiştir. MTR kodların varoluşu üzerine şimdiye kadar incelenen durumlar ele alınarak bir açık soru sunulmuştur. Bu açık sorunun çözümü olduğu gösterilmiş ve tüm olası çözümlerin sınıflandırılması için Yılan ve Merdivenler algoritmasını kullanarak, soruya bilgisayar cebir sistemi GAP yardımıyla hücum önerisinde bulunulmuştur.

ACKNOWLEDGEMENTS

First of all, I am thankful to my thesis advisor, Michel Lavrauw. He didn't only care for me as a student but also as a friend. Many thanks to him for all the invaluable advice along the way. Secondly, I would like to express my gratitude to Sabancı University Mathematics Program for the warm and welcoming atmosphere that I deeply enjoyed throughout my studies. I always felt very comfortable in the friendly environment they provided. I also would like to thank Cem Güneri for introducing me to the topic that I love today and Canan Kaşıkçı for her kindness and all her help. My appreciation also goes out to my parents for their unconditional love and encouragement, and to my friends for being by my side. Endless thanks to my dear brother Yalçın for his always smiling face and continuous support. Last but not least, I would like to thank TUBITAK for their scholarship.

To my family
Aileme

TABLE OF CONTENTS

LIST OF TABLES ..... **ix**

LIST OF FIGURES ..... **x**

1. INTRODUCTION AND PRELIMINARIES **1**
1.1. Rank Inequalities and Properties of Tensor Product 2

2. SINGLETON BOUND AND ITS ANALOGOUS VERSION **7**
2.1. Rank Metric Codes 14

3. 3-TENSORS REPRESENTATION OF RANK-METRIC CODES **18**
3.1. Three-Way Arrays 21
3.2. Tensor Rank 26
3.3. Vector Codes 33

4. TENSOR RANK EXTREMAL CODES **41**
4.1. Two Useful Maps 41
4.2. Extremal Triples 46
4.3. Construction of Extremal Triples using GRS Codes 53

5. CONNECTION TO COMPLEXITY THEORY **59**
5.1. Basics of Complexity Theory 59
5.2. Encoding Complexity of Rank-Metric Codes 64
5.3. Tensorial Notation for Complexity 66
5.4. Matrix Multiplication 73

6. FINITE GEOMETRIC APPROACH **76**
6.1. Semifields 76
6.2. Gabidulin Codes and the Case m=n=d 85

7. EXISTENCE OF MTR CODES **89**
7.1. The Algorithm Snakes and Ladders 94

BIBLIOGRAPHY **95**

LIST OF TABLES

Table 2.1. Repetition Codes vs. ISBN Codes 10

Table 5.1. Complexity 65
Table 5.2. 2×2 Matrix Multiplication 73

LIST OF FIGURES

Figure 2.1. Network N, Source S, and Terminals T 14
Figure 2.2. Butterfly Network 14
Figure 2.3. Error Amplification 16

Figure 3.1. Field Reduction 35
Figure 3.2. Fano Plane 36

1. INTRODUCTION AND PRELIMINARIES

Rank-metric codes have started to gain interest in the recent years due to their use against error amplification problem in Network Coding. In the first 2 sections, we give necessary information to understand the thesis and define rank-metric codes with the reason why they are effective against error amplification. In Section 3, based on the papers [15] and [23], 3-fold tensor representations of rank-metric codes are examined and tensor rank is shown as a complexity measure based on the idea provided in [16]. In Section 4, some constructions of minimal tensor rank(MTR) codes are visited. In Section 5, we show the role of tensors in complexity theory so that the correspondence between 3-tensors and rank-metric codes can be understood better. Additionally, we show the relation between complexity and rank and finish by explaining the famous matrix multiplication problem and the application of the tensor rank in that case. In Section 6, we underline the well-known connection between spreadsets and quasifields to show that semifields correspond to maximum rank distance(MRD) codes, i.e, the most popular family of rank-metric codes. We note that given a semifield, we can create a tensor associated to it, and then provide some results known in that context including the tensor rank of semifields based on the articles [16],[18], and [19]. We also provide some other geometric representations of MRD codes in Section 6, and show their relations. In Section 7, we pick up this geometric approach to attack an open problem that is related to the existence of MTR codes. By considering the points of the Segre variety as pure tensors, we deduce some well known geometric relations and try to create some new ones using algorithms with the help of [1] and [20].

1.1 Rank Inequalities and Properties of Tensor Product

Firstly, we recall some of the basic facts needed to understand this thesis. Afterwards, we will state and prove some of the rank inequalities that will be used throughout this thesis. Lastly, we will finish with some basic properties of tensor products which will come handy later.

Definition 1.1.1. *For an integer $i \geq 0$, we let $[i] := \{1, \ldots, i\}$.*

Definition 1.1.2. *Let $\mathbb{K}$ be a field. A $\mathbb{K}$-vector space is an abelian group $(V, +)$ with scalar multiplication operation $\mathbb{K} \times V \longrightarrow V$, satisfying the following properties:*

- $\lambda(v_1 + v_2) = \lambda v_1 + \lambda v_2$ *for all* $\lambda \in \mathbb{K}$, $v_1, v_2 \in V$.
- $(\lambda + \mu)v = \lambda v + \mu v$ *for all* $\lambda, \mu \in \mathbb{K}$, $v \in V$.
- $(\lambda\mu)v = \lambda(\mu v)$ *for all* $\lambda, \mu \in \mathbb{K}$, $v \in V$.
- $1v = v$ *for all* $v \in V$.

The following is known as the Grassmann's Identity.

Theorem 1.1.3. *Let $(V, +, \cdot)$ be a $\mathbb{K}$-vector space, and A and B are two finite dimensional subspaces of V. Then, we have*

$$dim(A + B) = dim(A) + dim(B) - dim(A \cap B).$$

Definition 1.1.4. *Given two vector spaces V and W, a linear transformation between them is a map $L : V \to W$ satisfying the following:*

1.1 $L(v_1 + v_2) = L(v_1) + L(v_2)$ *for all* $v_1, v_2 \in V$, *and*

1.2 $L(av) = aL(v)$ *for all* $a \in \mathbb{K}$.

Definition 1.1.5. *The group of nonsingular linear transformations of an n-dimensional vector space over $\mathbb{F}_q$ is denoted by $GL(n, q)$.*

Given a linear transformation $L : V \to W$ and a basis $B = \{v_1, \ldots, v_n\}$ for the vector space V, we can create the matrix A of that linear transformation as follows:

$$A = [L]_B = [L(v_1) \mid L(v_2) \mid \ldots \mid L(v_n)].$$

Definition 1.1.6. *Let $\mathbb{K}$ be a field and A be an $n \times m$ matrix. The space spanned by the rows of A is called the row space of A, and it is a subspace of $\mathbb{K}^m$. Similarly, the space spanned by the columns of A is called the column space of A, and it is a subspace of $\mathbb{K}^n$. The dimensions of the row space and the column space are equal, and is called the rank of A. The null space of A is*

$$null(A) = \{x \in \mathbb{K}^m : Ax = 0\}.$$

Its dimension is called the nullity of A.

Theorem 1.1.7 (**Rank Nullity Theorem**). *For any $n \times m$ matrix A,*

$$rank(A) + nullity(A) = m.$$

Definition 1.1.8. *Let $S \subseteq \mathbb{K}$. If S satisfies the field axioms with the same operations of the field $\mathbb{K}$, then S is called a subfield of $\mathbb{K}$.*

Let p be a prime. In a finite field of order p^n, there always exists a subfield of order p^m for every m dividing n. So, we can find smaller fields given a field. The following definition talks about extending one.

Definition 1.1.9. *Let $\mathbb{F}$ be a subfield of $\mathbb{K}$. Then, $\mathbb{K}$ is called an extension field of the field $\mathbb{F}$.*

The two famous examples are the complex numbers being an extension field of the real numbers, and the real numbers being an extension field of the rational numbers.

Definition 1.1.10. *Let $\mathbb{K}$ be an extension of the field $\mathbb{F}$. The extension degree (or index) is denoted $[\mathbb{K} : \mathbb{F}]$, and it is equal to the dimension of $\mathbb{K}$ when it is considered as a vector space over $\mathbb{F}$, i.e., $[\mathbb{K} : \mathbb{F}] = dim_{\mathbb{F}}\mathbb{K}$.*

Remark 1.1.11. *In general, if $\mathbb{F}$ is a field and $\mathbb{K}$ is a field extension of $\mathbb{F}$, then $\mathbb{K}$ is a $\mathbb{F}$-vector space and every $\mathbb{K}$-vector space V is also an $\mathbb{F}$-vector space. In addition, it holds that:*

$$\dim_{\mathbb{F}} V = \dim_{\mathbb{K}} V \cdot \dim_{\mathbb{F}} \mathbb{K}.$$

For example, $\dim_{\mathbb{R}} \mathbb{C}^n = \dim_{\mathbb{C}} \mathbb{C}^n \dim_{\mathbb{R}} \mathbb{C} = n \cdot 2 = 2n$.

Definition 1.1.12. *A group G acts on a set X means we have a group action $X \times G \to X : (x, g) \longmapsto x \cdot g$ such that $x \cdot e_G = x$ and $x \cdot (gg') = (x \cdot g) \cdot g'$ where $g, g' \in G$, $x \in X$ and e_G denote the identity element of the group G. For $x \in X$, we can define the orbit of x as $G(x) = \{x \cdot g \mid g \in G\}$, and the stabilizer of x as $G_x = \{g \in G \mid x \cdot g = x\}$.*

Note that, equivalent to Definition 1.1.6, we can also define the rank of a matrix as the smallest number R such that the matrix can be written as a sum of R rank 1 matrices. This way of defining the rank will be very useful when we consider tensor rank in the following parts. Now, we will show some rank inequalities.

Theorem 1.1.13. *Let A, B matrices of required size. Then, we have $rank(AB) \leq \min\{rank(A), rank(B)\}$.*

Proof. Consider the matrix AB. Note that the columns of AB can be written as a linear combination of columns of A. So, $rank(AB) \leq rank(A)$. Similarly, rows of AB can be written as a linear combination of rows of B, so $rank(AB) \leq rank(B)$. Combining these two gives us the theorem. □

Theorem 1.1.14 (**Sylvester's Rank Inequality**). *Let A, B two matrices of size $n \times n$. Then, $rank(A) + rank(B) \leq rank(AB) + n$.*

Proof. Note that left or right multiplications by invertable matrices do not change the rank of a matrix. Consider the matrix A. We can reduce it to echelon form by multiplying it with elementary matrices. Since elementary matrices are invertable, the rank of A will not change. Therefore, without loss of generality we may assume that A is in reduced row echelon form, i.e., $A = \begin{pmatrix} I_r & 0 \\ 0 & 0 \end{pmatrix}$. Then,

$$\begin{aligned} rank(A) + rank(B) &= rank(A) + rank(AB + (I - A)B) \\ &\leq rank(A) + rank(AB) + rank((I - A)B) \\ &= r + rank(AB) + rank((I - A)B) \\ &\leq r + rank(AB) + (n - r) = rank(AB) + n \end{aligned}$$

where the first inequality follows from the fact that sum of ranks can not be less than the rank of the sum, and the last inequality follows since first r rows of $I - A$ is 0, i.e., $rank(I - A)$ is at most $n - r$. □

Note that we can generalize this for non-square matrices, i.e., for matrices A and B of dimension $k \times n$ and $n \times m$, the above inequality still works. The following corollary is the combination of the above theorems.

Corollary 1.1.15. *Let A be $k \times n$ matrix and B be $n \times m$ matrix. Then,*

$$rank(A) + rank(B) - n \leq rank(AB) \leq \min\{rank(A), rank(B)\}.$$

In matrix theory, we can decompose matrices into full rank matrices. This process is called the rank factorization of matrices. Although we will define tensors more formally later, we can think of them as n-dimensional arrays, and see matrices as 2-tensors. In general, decomposition of tensors is a very important problem, and we will try to explain some parts of this huge problem in this thesis, as well.

Theorem 1.1.16. *Every finite dimensional matrix has a rank factorization.*

Proof Outline. Consider the matrix A. Let M denote its reduced row echelon form. Remove all zero rows of M and denote it by V. Similarly, remove all non-pivot columns of A and denote it by U. Then, $A = UV$ where U and V have full rank. □

For example, consider an invertible matrix A so that A has full rank. Then since the reduced row echelon form of A is the identity matrix and the matrix A itself does not have any non-pivot columns, we will have $A = AI$ by rank factorization.

Note 1.1.17. *Decomposition is a very important problem. One of the most famous matrix decompositions is the singular value decomposition(SVD) which has so many applications. However, extending this idea to higher orders, even for a well known decomposition, such as SVD, is proven to be very difficult. We will talk about this later in the thesis.*

Now, the following inequality is very important and well-known.

Theorem 1.1.18 (**Frobenius Inequality**). *Let A be $k \times n$, B be $n \times m$, and C be $m \times p$ matrices. Then, $rank(AB) + rank(BC) \leq rank(ABC) + rank(B)$.*

Proof. Assume that $rank(B) = r$. Let $B = UV$ be the full rank factorization by Theorem 1.1.16 such that U is $n \times r$ and V is $r \times m$ matrix. Note that AU is $k \times r$, and VC is $r \times p$. Then, $rank(ABC) = rank((AU)(VC))$. By applying Theorem 1.1.14 to AU and VC, we have

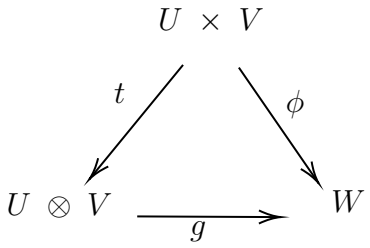
$$rank(ABC) \geq rank(AU) + rank(VC) - r \tag{1.1}$$

$$= rank(AB) + rank(BC) - rank(B) \tag{1.2}$$

where the equality follows from the fact that U and V are full rank matrices. □

Now, we will define some properties of tensor products.

Definition 1.1.19. *Let $\mathbb{K}$ be a field, and let U,V be $\mathbb{K}$-vector spaces. The **tensor product** $U \otimes V$ of this two vector spaces is itself a $\mathbb{K}$-vector space, equipped with a $\mathbb{K}$-bilinear map $t : U \times V \to U \otimes V$ satisfying:*



For all $\mathbb{K}-$space W and for all $\mathbb{K}-$bilinear map $\phi : U \times V \to W$, there exists a unique map $g : U \otimes V \to W$ such that $g \circ t = \phi$. Denote $u \otimes v = t(u,v)$.

Tensor product of 2 vectors v, w is

$$v \otimes w = vw^T = \begin{bmatrix} v_1w_1 & v_1w_2 & \dots & v_1w_m \\ v_2w_1 & v_2w_2 & \dots & v_2w_m \\ \vdots & \vdots & \ddots & \vdots \\ v_nw_1 & v_nw_2 & \dots & v_nw_m \end{bmatrix}$$

Let us also list some other well-known properties.

- $(u \otimes v)^T = (v \otimes u)^T$.
- $(v+w) \otimes u = v \otimes u + w \otimes u$.
- $u \otimes (v+w) = u \otimes v + u \otimes w$.
- $\lambda(v \otimes u) = (\lambda v) \otimes u = v \otimes (\lambda u)$.
- Given a matrix $A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$ and B, we have $A \otimes B = \begin{bmatrix} aB & bB \\ cB & dB \end{bmatrix}$

Form a new larger vector space $K = U \otimes V \otimes W$. Then, by Definition 1.1.5, consider the group $G = GL(U) \times GL(V) \times GL(W)$.

Definition 1.1.20. *The action of G on K is given as:*

$$K \times G \to K : ((u \otimes v \otimes w), (g_1, g_2, g_3) \longmapsto (u^{g_1} \otimes v^{g_2} \otimes w^{g_3}).$$

2. SINGLETON BOUND AND ITS ANALOGOUS VERSION

In this section, focusing on two examples given in [24], we will explain some basic concepts in coding theory which are needed to understand this work. After that, we will prove one of the most important bounds in coding theory and its analogue for our work. Then, we will start introducing rank-metric codes by first giving the motivation to study them. In the back cover of every book, there is an ISBN(International Standardized Book Number) code. Depending on the date of publication, it is either 10 digit or 13 digit. The books before 2007 have ISBN-10 codes. For example, $0-587-44213-1$. The last digit is called the check digit. First 9 digits contain information about the book. Consider an ISBN code $x_1 - x_2x_3x_4 - x_5x_6x_7x_8x_9 - x_{10}$ and the integer

$$R = x_1 + 2x_2 + \ldots + 9x_9.$$

If $R \equiv 10 \pmod{11}$, then we say $x_{10} = X$. Otherwise, we say $x_{10} \equiv R \pmod{11}$. In the example above, $R = 155$. Then, $R \pmod{11} \equiv 1$. Since $1 \neq 10$, we have $x_{10} = 1$, as above. If you make a mistake when typing in the check digit, then the system can catch your error by the use of above formula. So, the ISBN code can detect all single digit errors. However, it can not correct any of those detected errors. In this case, since we can simply send the message again by re-entering the correct ISBN code, it is not so crucial. In general, this will not be the case. We also say that this code is very efficient. The reason is that, we only need one non-info symbol for every nine information symbol. Now, let us examine our second example. Suppose every data is encoded as a five bit string. This time, we duplicate each data 3 times by repeating it rather than simply transmitting it. Consider 11010. It would be forwarded as 11010 11010 11010. In the case of a single error, we can correct and detect it as follows. The error has to be contained in one of the 3 blocks. That means, the other two blocks are error-free and thus will still be equal to each other. For example, the receiver can get 11010 10010 11010. In that case, we can see that the first and the third are equal, whereas the second digit of the middle is 0. Thus, we detected that error and can simply correct it by writing the second digit of the

first or the third. In general, we need more than half of the blocks agree to correct the errors. Therefore, if we want to fix m errors, we need to repeat the data $2m+1$ times. These codes are called repetition codes, and they are not efficient since we are transmitting 3 symbols for only 1 information symbol. Now, let us properly define what a code is.

Definition 2.0.1. *A subset C of Λ^n is called a code over the alphabet Λ.*

For now, consider Λ to be a finite field. However, note that codes over finite rings are also possible and very interesting. To construct a code, we need an alphabet and a desired length. Length of a code is denoted by n where $C \subseteq \Lambda^n$. C is called a linear code if it is a subspace of the vector space Λ^n, and in this case the usual notation for the dimension is $dim(C) = k$. Our main work will be about codes that are subspaces of the ring $\mathbb{F}_q^{n\times m}$. We will call them rank-metric codes.

Theorem 2.0.2. *Given a linear code C, we have $|C| = q^{dim(C)}$ where $\Lambda = \mathbb{F}_q$.*

Proof. Consider the basis B of C. Let $dim(C) = k$. So, the basis has k elements. Let $B = \{b_1, \ldots, b_k\}$. Then, any codeword can be written as $c_1b_1 + \ldots + c_kb_k$ where $c_i \in \mathbb{F}_q$. For each c_i we have q choices, so in total there are q^k elements. □

Each of these k basis elements are vectors of length n. By writing them as rows, we construct a $k \times n$ matrix. It is called a generator matrix for C. If G is a generator matrix for C, then $C = \{uG | u \in \Lambda^k\}$ since

$$uG = (u_1 \ldots u_k) \begin{pmatrix} - & r_1 & - \\ & \vdots & \\ - & r_k & - \end{pmatrix} = u_1r_1 + \ldots u_kr_k.$$

Definition 2.0.3. *The orthogonal complement $C^\perp$ of C is called the dual code of C. This makes sense since C is a subspace.*

Definition 2.0.4. *A generator matrix for the dual code $C^\perp$ is called a (parity)-check matrix H of the linear code C. It checks whether an element lies in the code or not. That is, $C = \{c \in \mathbb{F}_q^n \mid cH^T = 0\}$.*

We denote the linear codes of length n and dimension k as $[n, k]$ codes. Note that if C is an $[n, k]$ linear code, then a generator matrix is $k \times n$, and the parity-check matrix for C must be an $(n-k) \times n$ matrix. Rows of G, H are linearly independent. The following is very useful in computations.

Theorem 2.0.5. *$HG^T = 0 = GH^T$ where 0 is in appropriate size.*

Example 2.0.6. *Let* $C =< (1\ 2\ 0), (0\ 1\ 1) > \subseteq F_3^3$. *Then we have*

$$G = \begin{pmatrix} 1 & 2 & 0 \\ 0 & 1 & 1 \end{pmatrix} \text{ and } H = \begin{pmatrix} 1 & 1 & 2 \end{pmatrix}$$

See that Theorem 2.0.5 holds. Similarly, for any $c = (a\ b\ c) \in C$, *we have* $cH^T = 0$ *since* $a+b+2c = 0$ *for all* $c \in C$. *Thus,* H *checks whether* $c \in C$ *or not.*

Definition 2.0.7. *For* $x, y \in \Lambda^n$, *let* $d(x,y)$ *be the number of different entries in the same position, i.e., number of times* $x_i \neq y_i$ *is occuring. The weight of a codeword is given by* $wt(x) = d(x,0)$. *The minimum distance of a code* C *is* $d(C) = \min\{d(x,y) \mid x, y \in C,\ x \neq y\}$. *For linear codes, we have*

$$d(C) = \min\{wt(x) \mid x \in C,\ x \neq 0\}.$$

If we also know the minimum distance of a code, then we will denote it as $[n,k,d]$ code. This is of course if the field is clear from the context. We sometimes use $\mathbb{F} - [n,k,d]$ to emphasize the field which the code is defined over.

Now, we will analyze our two examples according to the preceeding definitions. ISBN code is a code of length 10 over $\mathbb{F}_{11}$ where X stands for $10 \in F_{11}$. Consider $a_1 = 0-587-44213-1$ and $a_2 = 2-469-65111-5$. When we try to compute $a_1 + a_2$, we see that in the fifth digit we have X. However, we can not have X in the first 9 digits by definition. So, the code is not linear. First 9 digits can take 10 different values, and the last digit is computed by the first 9 digits. So, $|C| = 10^9$. Since it is nonlinear, we can not talk about the dimension.
Repetition code is a linear code of length $5m$ over $\mathbb{F}_2$ where m is the number of blocks. Since it is linear, we can talk about the basis. Consider the case $m = 1$. The basis for the code is $\{10000, 01000, 00100, 00010, 00001\}$. So, the dimension is 5. What about the minimum distances?
Consider $I_1 = a_1 - a_2a_3a_4 - a_5a_6a_7a_8a_9 - c_1$ and $I_2 = b_1 - a_2a_3a_4 - a_5a_6a_7a_8a_9 - c_2$. Here, we assumed without loss of generality that the first digits are different and we will argue that this will force the last digits to be different as well so that the minimum distance can not be equal to 1. Note that if $a_1 = b_1$, then $c_1 = c_2$ since the last digits are determined by the first 9 digits and they are the same in this case. So, the minimum distance is 2 for ISBN code. For the repetition code, we can have $d = 1$ for length 5, so the minimum distance is m for a code of length $5m$. Now, if we combine all the knowledge we got so far, we have the following table:

Table 2.1 Repetition Codes vs. ISBN Codes

Repetition Code	ISBN Code
over $\mathbb{F}_2$	over $\mathbb{F}_{11}$
detect errors	detect errors
correct errors	-
non-efficient	efficient
linear, dimension = 5	non-linear
length = 5m	length = 10
distance = m	distance = 2

Suppose C is a linear $[n,k,d]$ code over Λ. What that really means is that each codeword in our code has k information symbols and $n-k$ non-info symbols. The more information symbols we have, the more efficient our code will be. Therefore, we want k large with respect to n so that we are not transmitting respectively large amount of non-info symbols. Besides efficiency, we mainly care about correcting as many errors as possible. The parameter d determines how many errors our code can correct. We want to say that our code C is t-error correcting for some positive integer t. To establish this, we need to make sure that given any word, there is only one codeword of distance at most t from it. Let $x \in \Lambda^n$. Geometrically, this can be described by defining the closed ball $B_t(x)$ centered at x of radius t, i.e., $B_t(x) = \{y \in \Lambda^n \mid d(x,y) \leq t\}$. In the following theorem we will prove that the code C is $\left\lfloor \frac{d-1}{2} \right\rfloor$-error correcting.

Theorem 2.0.8. *Let $d(C) = d$, and $x, y \in C$ such that $x \neq y$. Then, we have*

$$B_{\lfloor \frac{d-1}{2} \rfloor}(x) \cap B_{\lfloor \frac{d-1}{2} \rfloor}(y) = \emptyset.$$

Proof. Assume to the contrary that z lies in the intersection.
Then, $d(x,y) \leq d(x,z) + d(z,y) \leq 2\left\lfloor \frac{d-1}{2} \right\rfloor = \begin{cases} 2k = d-1 & \text{, if } d = 2k+1 \\ 2k = d-2 & \text{, if } d = 2k+2 \end{cases}$ So, $d(x,y) < d$, a contradiction. □

Since the balls can not intersect, the corrupted word will be inside a unique ball of radius $\left\lfloor \frac{d-1}{2} \right\rfloor$ by the above theorem. Then, we will understand that the correct message was the center of that particular ball. Thus, we can correct the error since we now know what was the corrupted word equal to before the noise.

Example 2.0.9. *Consider the linear binary code $C =< 001110, 110001 >$. The code has 4 elements which are $000000, 001110, 110001, 111111$. The minimum distance is 3 by definition. Thus, C is $\mathbb{F}_2 - [6,2,3]$ linear code.*

Then, $\left\lfloor \frac{d-1}{2} \right\rfloor = 1$. *Let* $x = 001110$ *and* $y = 110001$. *We have*

$$B_1(x) = \{001110, 101110, 011110, 000110, 001010, 001110, 001111\},$$

$$B_1(y) = \{110001, 010001, 100001, 111001, 110101, 110011, 110000\}.$$

See that the intersection is empty. Let say we received a codeword 110101. It is not an element of our code. However, it lies on the unique ball $B_1(y) = B_1(110001)$. Thus, we can understand that it was supposed to be 110001.
In general, we want d large with respect to n so that we will be able to correct reasonably high amount of errors. In summary, to have "good" codes, we want d, k large with respect to n. We will see that they can not be both large at the same time. Now, we move on to the family of error-correcting codes. Error-correcting codes are developed by Richard W. Hamming and they are used to transfer data in the most reliable way possible over unreliable communication channels. As the name suggest, they are transmitted in a way that the message can be recovered given unintentional errors. In almost all message transmissions, error correcting codes are being used. The reason we are calling some channels unreliable is that they have channel noise. In general, these communication channels can be physical such as wires, or most commonly not physical such as computer networks.
Repetition codes are examples of error-correcting codes. One real life example is providing clean communication with the astronauts in space using Reed-Solomon codes. Note that we will use generalized Reed-Solomon codes in Chapter 4 to construct extremal triples and they will play a crucial role for the existence of MTR codes. In coding theory, block codes are a family of error-correcting codes that encode data in blocks. When a sender wants to transmit a very long data using a block code, the sender breaks it into messages of fixed size.

Definition 2.0.10. *A block code is an injective function* $C\colon \Lambda^k \to \Lambda^n$ *where* Λ *is our finite and nonempty alphabet. A block code of length* n *is just a subset of* Λ^n.

Since Λ is injective, $k \leq n$. Note that, we do not know if C is surjective or not. Thus, the following definition is required.

Definition 2.0.11. *For* $c \in \Lambda^n$, *if* $C(m) = c$ *for some* $m \in \Lambda^k$ *then* c *is called the codeword corresponding to the element* m.

The procedure given by the block code is the following:

- Encode each message separately into a codeword which is called a block.
- Transmit all the blocks to the receiver.

Example 2.0.12. *Let* $\Lambda = \{0,1\}$. *Choose* $k = 2$ *and* $n = 6$. *We encode as*

$$\begin{aligned} 00 &\to 100000 \\ 01 &\to 010000 \\ 10 &\to 001000 \\ 11 &\to 000100 \end{aligned}$$

Consider the message $m = 011000$. *Then, this will go to the receiver as* $01 \blacktriangleright 10 \blacktriangleright 00$ *where "* $\blacktriangleright$*" represents concatenation. Then the receiver gets* $e_1 = 010000001000100000$. *Now, consider* $k = 3$ *and* $n = 6$. *Let us use the following:*

$$\begin{aligned} 000 &\to 001000 \\ 010 &\to 000100 \\ 110 &\to 000001 \\ 011 &\to 010000 \end{aligned} \qquad \begin{aligned} 100 &\to 100000 \\ 001 &\to 000010 \\ 101 &\to 110000 \\ 111 &\to 000111 \end{aligned}$$

The same message m *will go to the receiver as* $011 \blacktriangleright 000$. *Then, the receiver will get* $e_2 = 010000001000$. *See that* $e_1 = e_2 \blacktriangleright 100000$. *Thus, by increasing* k *we get a shorter code and we also increased the efficiency.*

In this example, it may seem that choosing $k = n$ is the best choice. However, we will see a bound called the Singleton bound. If $k = n$, that would mean $d = 1$. A code can detect at most $d - 1$ errors, and that means our code can not even detect any errors in the case $k = n$. As we explained before, we want both k and d large with respect to n, and this is just one of the reasons. As seen in the previous example, the transmission speed is very important, and we need a way to examine this. The following definition comes naturally.

Definition 2.0.13. ***The transmission ratio*** *of a block code is* $TR = \dfrac{k}{n}$.

Since C is injective, $k \leq n$. Therefore, $TR \leq 1$. Another way to see this is that, in practice the data can not be compressed without a loss, and thus the ratio can be at most 1. Now, we are ready to prove the singleton bound.

Definition 2.0.14. $A_q(n,d) =$ *Maximum number of codewords in a block code over* $\mathbb{F}_q$ *of length* n *and minimum distance* d.

It is easy to see that $A_q(n,n) = q$ and $A_q(n,1) = q^n$. However, there is no general formula for $A_q(n,d)$. Finding that could be one of the biggest achievements in coding theory and we will state it here as an open problem.

Open Problem 2.0.15. *Find a formula for* $A_q(n,d)$.

Theorem 2.0.16 (**Singleton Bound**).

$$A_q(n,d) \leq q^{n-d+1}$$

Proof. Let C be an arbitrary block code over $\mathbb{F}_q$ of minimum distance d. The number of words of length n is q^n. So, if $d=1$, then we are done. Now, consider the following matrix:

$$\left(\begin{array}{c|c} \underbrace{\begin{array}{c}\cdots\\ \cdots\\ \vdots\end{array}}_{d-1} & \underbrace{\begin{array}{c}u\\ v\\ \vdots\end{array}}_{n-d+1} \end{array}\right)$$

Let the rows of this matrix be all the codewords in our code. Suppose that the codeword c_1 including u is in the a-th row and the codeword c_2 including v is on the b-th row. Now, if $u=v$, then $d(c_1,c_2) \leq d-1$ since only the left part can have different letters, and there are $d-1$ columns on the left. This contradicts the fact that minimum distance is d. Therefore, all the words on the right are pairwise different. Now, assume that we deleted the first $d-1$ letters of each word, i.e., the left part of the matrix above. The newly obtain codewords have length $n-d+1$. Since C is over $\mathbb{F}_q$, there are q different options for each coordinate in such a word of length $n-d+1$. That means, we can have at most q^{n-d+1} of them. However, C was arbitrary, so this bound must hold for the largest possible code with these parameters. That is, $|C| \leq A_q(n,d) \leq q^{n-d+1}$. □

Corollary 2.0.17. *If C is a linear $[n,k,d]$ code over $\mathbb{F}_q$, then the number of codewords is equal to q^k. Theorem 2.0.16 gives $q^k \leq q^{n-d+1}$. Thus, $k \leq n-d+1$.*

Codes that meet the Singleton bound are called MDS(Maximum Distance Separable) codes. The name makes sense since when the equality holds, $d=n-k+1$ is maximum. They are very important as MDS codes have the largest error correction capacity since the amount of error correction is a non-decreasing function of d. Separable code means that any codeword can be separated, i.e, the codewords are of the form "information digits" ▶ "check digits". It can not be the case that they are mixed. For example, the ISBN code is a separable code.

Note 2.0.18. *Linear MDS codes geometrically corresponds to arcs. For more details and exposition of arcs in projective spaces, see [17].*

2.1 Rank Metric Codes

We are now ready to define our main family of codes, namely the rank-metric codes. They will be the foundation of this thesis. We will start by explaining the reason for their recently increased use by following [22]. Rank-metric was found as a solution in network coding as a tool to handle the error amplification problem. Firstly, the structure of the network is given in Figure 2.1.

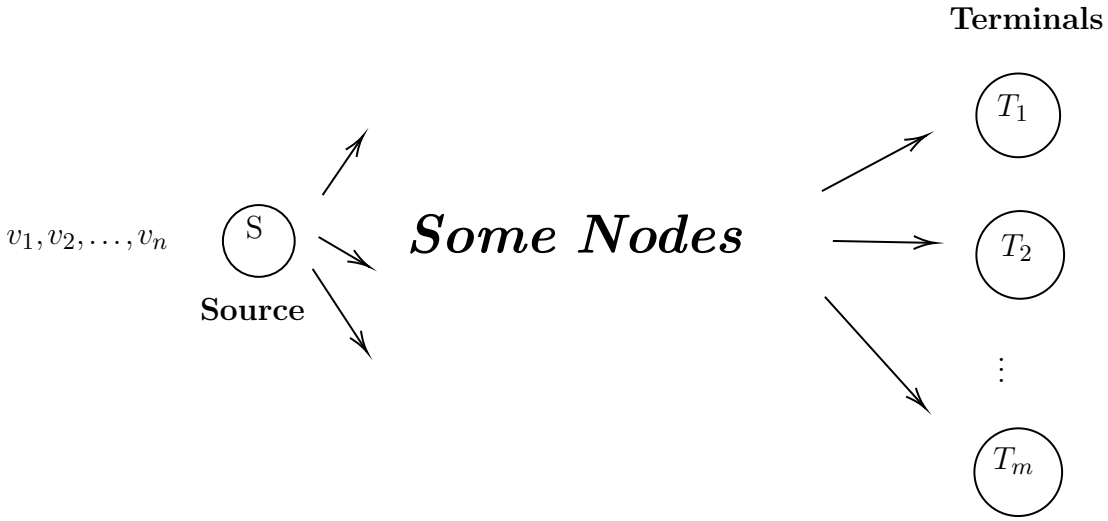


Figure 2.1 Network N, Source S, and Terminals T

Note that $v_1, v_2, \ldots, v_n$ are vectors. Terminals wants all the n messages. The goal of the network is to maximize the number of transmitted messages to all terminals per channel use, i.e., maximize the rate.

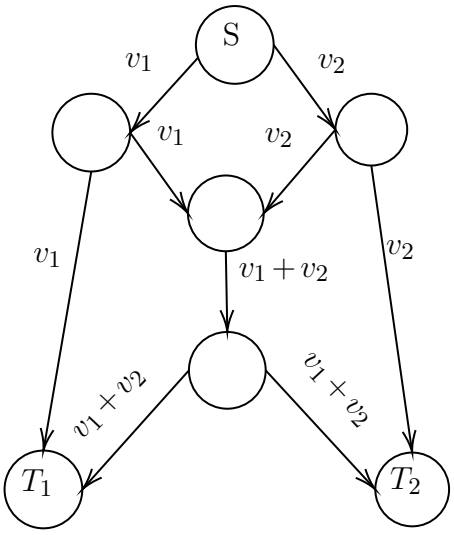


Figure 2.2 Butterfly Network

To increase the rate, there is a very common idea which is allowing the nodes to combine received vectors. Here, combining can have different meanings given the setting. In this case, we mean allowing the nodes to do linear operations. Consider the classic butterfly example of network coding given in Figure 2.2 when the node in the middle can add the received vectors. Here, T_1 does $(v_1+v_2)-v_1=v_2$, and T_2 does $(v_1+v_2)-v_2=v_1$ so that they both get all the messages. Now, we will try to calculate the rate of the butterfly network. See that the transmission between the middle nodes allow us to use the channel only once instead of twice(first sending v_1, and then v_2.) So, we are transmitting 2 messages and we are only using the channel once. So the rate is $\frac{2}{1}=2$.

Definition 2.1.1. *For any terminal T, we can associate a transfer matrix $M(T)$ such that $M(T)\cdot V$ gives us the messages that the terminal T receives where the source S sends V. Note that the rows of V are the messages S send.*

The following theorem summarizes the idea presented in [13]. Note that when we say node operations, we mean the linear operations that the nodes perform such as addition, scalar multiplication etc.

Theorem 2.1.2. *Suppose N is a linear network, i.e, the nodes can perform linear operations. Let $\mu(N)=n$ denote the minimum of minimum number of edges to be removed in the network to cut the connection of S and T_i for $i\in[m]$. Assume S sends the messages $v_1,\ldots,v_n$. We will denote the transfer matrix by $M(T)$ for a terminal T. Then,*

(i) $rate(N)=\mu(N)$.

(ii) There exists some node operations so that the transfer matrices of all the terminals are $n\times n$ invertable at the same time.

The following example is very crucial to understand the theorem above.

Example 2.1.3. *Consider the Butterfly network given in Figure 2.2. We need to remove at least 2 edges to cut the connection between S and T_1. Since the network is symmetric, the same is also true for T_2. So, $\mu(N)=\min\{2,2\}=2$. The first part of the theorem holds since $2\leq 2$. We are left to find the transfer matrices for each terminal. It is easy to see that $M(T_1)=\begin{pmatrix}1&0\\1&1\end{pmatrix}$ and $M(T_2)=\begin{pmatrix}1&1\\0&1\end{pmatrix}$ so that $M(T_1)\begin{pmatrix}v_1\\v_2\end{pmatrix}=\begin{pmatrix}v_1\\v_1+v_2\end{pmatrix}$ which says that T_1 received v_1 and v_1+v_2. Similarly, we can do the same to show that T_2 received v_2 and v_1+v_2.*

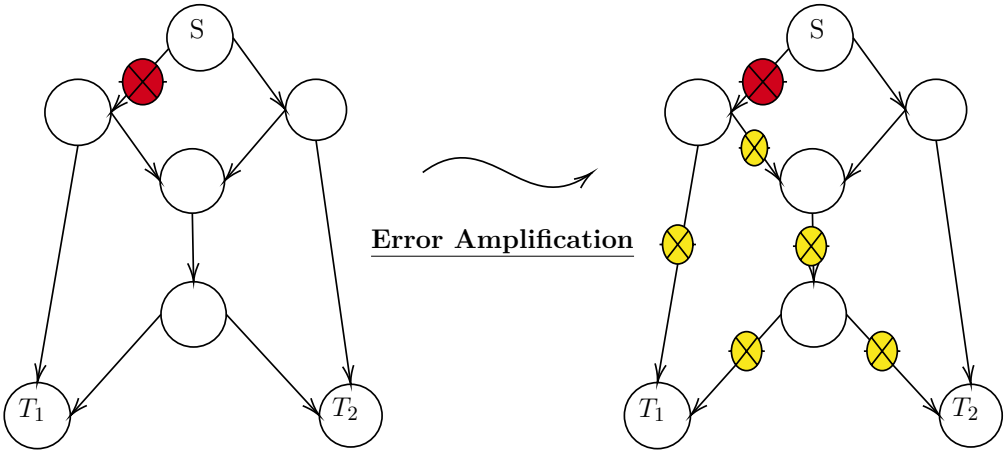
Given a transfer matrix M, the decoding is very easy by using part (ii) of the

theorem since we now have

$$\begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{pmatrix} = M^{-1}(M \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{pmatrix}).$$

It is time to talk about the amplification problem which we mentioned in the beginning. See Figure 2.3 for a visualization of the error amplification problem.

Figure 2.3 Error Amplification



As can be seen from the figure, one error leads to more errors. In general, suppose we have one adversary who can corrupt up to t edges. This number t is called the adversarial strength. The corrupted edges are known as noise in communication theory. If there is error amplification, then the number of errors in the edges, i.e., the amount of noise will be much higher than t. Fortunately for us, the rank-metric is the solution. Suppose the message X is sent. Here, X is a vector whose entries are the messages v_i. Since each message has also fixed length, X is a matrix. If we have n messages of length m, then X is of size $n \times m$. If there is no error, then the terminal receives $M(T)X$ as we showed before. However, suppose there is noise and the terminal T received some other matrix $E(T)$. It is explained in [14] that the rank-metric prevents errors from amplifying and reformulated in [22] as follows.

Theorem 2.1.4. *If the adversarial strength is t, then $rank(E(T) - M(T)X) \leq t$.*

Thanks to this theorem, if we use the rank-metric then the edges that are corrupted by the adversary can not lead to errors in the remaining part of the network. This theorem is the reason of the renewed interest in rank-metric codes. Now, we will zoom in on the theory of rank-metric codes to understand them better.

Definition 2.1.5. *Let $\mathbb{F}_q^{n\times m}$ denote the ring of $n\times m$ matrices with entries coming from $\mathbb{F}_q$. An $\mathbb{F}_q$-linear rank metric code is a subspace $C\subseteq\mathbb{F}_q^{n\times m}$. The rank distance between $X,Y\in\mathbb{F}_q^{n\times m}$ is $d(X,Y)=rank(X-Y)$. If $C\neq\{0\}$ then the minimum distance of C is the following integer*

$$d(C):=\min\{rank(X):X\in C,X\neq 0\}=\min\{d(X,Y):X,Y\in C,X\neq Y\}.$$

From this moment on, we will denote a rank-metric code C of dimension k and minimum distance d as an $\mathbb{F}_q-[n\times m,k,d]$ code.

Theorem 2.1.6 (**Rank Metric Analogue of the Singleton Bound**). *Let C be an $\mathbb{F}_q-[n\times m,k,d]$ code. Then, $k\leq\max\{n,m\}(\min\{n,m\}-d+1)$.*

Proof. Without loss of generality, let $n\leq m$. Let us consider the desired inequality in base q. So, we want to prove $q^{dim(C)}=|C|\leq q^{m(n-d+1)}$. Assume to the contrary that $|C|>q^{m(n-d+1)}$. Take any element of the code, i.e., a matrix and consider the last $n-(d-1)$ rows of the matrix $\left[\begin{array}{c}\cdots\\\hline m(n-d+1)entries\end{array}\right]$. See that there are $m(n-d+1)$ entries below. Since $|C|>q^{m(n-d+1)}$, by the Pigeonhole principle, there exists X,Y distinct in C such that they coincide in every entry of the last $n-d+1$ rows. These X and Y must exist since there are q choices for each entry but $|C|>q^{m(n-d+1)}$. So, $X-Y=\left[\begin{array}{c}A\\\hline 0\end{array}\right]$ has rank $\leq d-1$. Thus, we have $d(X,Y)\leq d-1<d$, contradicting the fact that $d(C)=d$. Therefore, $|C|\leq q^{m(n-d+1)}$, i.e., $dim(C)\leq m(n-d+1)$. □

Definition 2.1.7. *Codes that meet this bound are called MRD(Maximum Rank Distance) codes. The reason we are calling them MRD is clear since when the equality occurs, the distance is maximal.*

In [5], it is proven that MRD codes exists for all parameters m,n, and d. One famous example is the Delsarte-Gabidulin codes. It is the same paper that rank-metric codes were introduced.

Example 2.1.8. *Consider an $\mathbb{F}_q-[5\times 3,k,3]$ code C. Suppose we want to find a range for k. Rank-metric analogue of the singleton bound without considering max/min gives $k\leq 3(5-3+1)$, that is $k\leq 9$. Now, if we consider rank-metric analogue of the singleton bound properly this time, we will get $k\leq 5(3-3+1)$, that is, $k\leq 5$. So, we get a better bound. The reason we are taking the maximum/minimum of m,n is to get a better bound. In general $1-d(C)<0$. As a proof, observe that*

$$m\geq n\implies m(-d+1)\leq n(-d+1)\implies m(n-d+1)\leq n(m-d+1).$$

3. 3-TENSORS REPRESENTATION OF RANK-METRIC CODES

In this chapter we will follow the footsteps of [23] but we will interpret it in our terminology and make some additions.

Consider m vector spaces V_i over $\mathbb{F}$. Define $\mathcal{T} = V_1 \otimes \ldots \otimes V_m$. $\mathcal{T}$ is called the tensor space and it is easy to see that it is an $\mathbb{F}$-vector space. Let $dim(V_i) = d_i$. Consider the bases B_i for those m vector spaces:

$$\begin{aligned} B_1 &= \{e_{11}, \ldots, e_{1d_1}\} = \{e_{1i_1}\}_{i_1=1}^{d_1} \\ B_2 &= \{e_{21}, \ldots, e_{2d_2}\} = \{e_{2i_2}\}_{i_2=1}^{d_2} \\ &\vdots \\ B_m &= \{e_{m1}, \ldots, e_{md_m}\} = \{e_{mi_m}\}_{i_m=1}^{d_m} \end{aligned}$$

Definition 3.0.1. *Tensors of the form $v_1 \otimes \ldots \otimes v_m$ are called pure (or simple) tensors. Consider a pure tensor $v_1 \otimes \ldots \otimes v_m$ where $v_i \in V_i$. We can write each of these vectors as a linear combination of the basis elements.*

$$v_1 \otimes \ldots \otimes v_m \;\; = \;\; (\sum_{i_1=1}^{d_1} v_{1i_1} e_{1i_1}) \otimes \ldots \otimes (\sum_{i_m=1}^{d_m} v_{mi_m} e_{mi_m})$$

Arbitrary elements of $\mathcal{T}$ are expressed as sums of pure tensors. Now, let us consider 3 vector spaces. If $\{u_1, \ldots, u_k\}, \{v_1, \ldots, v_n\}, and \{w_1, \ldots, w_m\}$ are bases of $U, V, and W$ respectively, then a basis of $U \otimes V \otimes W$ is given by

$$\{u_i \otimes v_j \otimes \; w_l : 1 \leq i \leq k, 1 \leq j \leq n, 1 \leq l \leq m\}.$$

In particular, $\dim_{\mathbb{F}}(U \otimes V \otimes W) = \dim_{\mathbb{F}}(U) \dim_{\mathbb{F}}(V) \dim_{\mathbb{F}}(W)$ for a field $\mathbb{F}$. In this section, we are interested in tensor products of the form $\mathbb{F}^k \otimes \mathbb{F}^n \otimes \mathbb{F}^m$, whose elements are called 3-tensors, or 3-fold tensors. An element of this space can be represented by a 3-dimensional array. Similarly, a 2-fold tensor can be represented by a matrix, just as a 1-fold tensor can be represented by a vector. One can define

a 3-dimensional array of size $k \times n \times m$ as a function

$$X : [k] \times [n] \times [m] \to \mathbb{F}$$

which we represent as $X = (x_{ijl} : 1 \leq i \leq k, 1 \leq j \leq n, 1 \leq l \leq m)$.

Note 3.0.2. *Given a 3-fold tensor $X = \sum_{r=1}^{R} u_r \otimes v_r \otimes w_r$, we automatically have its coordinate tensor $x_{ijl} = \sum_{r=1}^{R} u_{ir} v_{jr} w_{lr}$, where $u_r = (u_{1r}, \ldots, u_{kr}), v_r = (v_{1r}, \ldots, v_{nr})$, and $w_r = (w_{1r}, \ldots, w_{mr})$. Hence, we can represent $\mathbb{F}^k \otimes \mathbb{F}^n \otimes \mathbb{F}^m$ by $\mathbb{F}^{k \times n \times m}$.*

The natural question is how to do operations with tensors. The following maps define the multiplication of 3-fold tensors with vectors ($s = 1$) and matrices ($s > 1$):

$$m_1 : \mathbb{F}^{s \times k} \times \mathbb{F}^{k \times n \times m} \to \mathbb{F}^{s \times n \times m} : (A, X) \longmapsto m_1(A, X) = \sum_i (Au_i) \otimes v_i \otimes w_i$$

$$m_2 : \mathbb{F}^{s \times n} \times \mathbb{F}^{k \times n \times m} \to \mathbb{F}^{k \times s \times m} : (B, X) \longmapsto m_2(B, X) = \sum_i u_i \otimes (Bv_i) \otimes w_i$$

$$m_3 : \mathbb{F}^{s \times k} \times \mathbb{F}^{k \times n \times m} \to \mathbb{F}^{k \times n \times s} : (C, X) \longmapsto m_3(C, X) = \sum_i u_i \otimes v_i \otimes (Cw_i)$$

for any $X = \sum_i u_i \otimes v_i \otimes w_i \in \mathbb{F}^{k \times n \times m}$.

Definition 3.0.3. *Let $X \in \mathbb{F}^{D_1 \times D_2 \times D_3}$. For each $i \in 1, 2, 3$, we define the $i-th$ contraction space of X to be $cs_i(X) = \langle m_i(e_1, X), \ldots, m_i(e_{D_i}, X) \rangle$, where $e_i \in \mathbb{F}^{1 \times k}$ such that $i-th$ entry is 1 and other entries are 0.*

Clearly, contraction spaces are $\mathbb{F}$-vector spaces, so we can talk about their dimensions. The following definition will be used all around in the thesis.

Definition 3.0.4. *We denote the dimension of $cs_i(X)$ by $dim_i(X)$. X is called i-concise if $dim_i(X) = D_i$. X is called concise if it is i-concise for all i.*

Let $X = \sum_{r=1}^{R} u_r \otimes v_r \otimes w_r \in \mathbb{F}^{k \times n \times m}$. Now, we will try to calculate $m_1(e_j, X)$ where $1 \leq j \leq k$. By the map defined above, $m_1(e_j, X) = \sum_{r=1}^{R} (e_j u_r) \otimes v_r \otimes w_r$. Note that $(e_j u_r) = u_{jr}$, and $u_{jr} \otimes v_r = 1 \otimes u_{jr} v_r = u_{jr} v_r$. Thus, we have the following.

Note 3.0.5. $m_1(e_j, X) = \sum_{r=1}^{R} u_{jr} v_r \otimes w_r$.

So, the first contraction space $cs_1(X) = \langle \sum_{r=1}^{R} u_{jr} v_r \otimes w_r : 1 \leq j \leq k \rangle$ is a span of k matrices of rank at most R.

Example 3.0.6. *Let $X = e_1 \otimes (e_1 \otimes e_1 + e_2 \otimes e_3 + e_2 \otimes e_4 + e_3 \otimes e_2 + e_3 \otimes e_3)$ $+ e_2 \otimes (e_1 \otimes e_4 + e_2 \otimes e_2 + e_2 \otimes e_4 + e_3 \otimes e_1)$ in $\mathbb{F}^2 \otimes \mathbb{F}^3 \otimes \mathbb{F}^4$. Find $cs_i(X)$ for $i \in 1, 2, 3$. For $X \in \mathbb{F}^k \otimes \mathbb{F}^n \otimes \mathbb{F}^m$, $cs_1(X)$ is the span of k matrices of size $n \times m$, $cs_2(X)$ is the span of n matrices of size $k \times m$, and $cs_3(X)$ is the span of m matrices of size*

$k \times n$. *Basically, to find* $cs_1(X)$, *we write* $X = e_1 \otimes (\ldots) + e_2 \otimes (\ldots) + \ldots + e_k \otimes (\ldots)$. *Similarly, to find* $cs_3(X)$, *we write* $X = (\ldots) \otimes e_1 + \ldots + (\ldots) \otimes e_m$. *The same reasoning works for* $cs_2(X)$, *where we look for* $e_1, \ldots, e_n$ *as the middle elements of the pure tensors whose sum is* X.
$X = \underline{e_1} \otimes (e_1 \otimes e_1 + e_2 \otimes e_3 + e_2 \otimes e_4 + e_3 \otimes e_2 + e_3 \otimes e_3) + \underline{e_2} \otimes (e_1 \otimes e_4 + e_2 \otimes e_2 + e_2 \otimes e_4 + e_3 \otimes e_1)$. *By definition,* $cs_1(X)$ *is the span of two* 3×4 *matrices. Thus,*

$$cs_1(X) = \left\langle \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix} \right\rangle$$

where $e_1 \otimes (e_1 \otimes e_1 + e_2 \otimes e_3 + e_2 \otimes e_4 + e_3 \otimes e_2 + e_3 \otimes e_3)$ *corresponds to*

$$A = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$

such that $e_i \otimes e_j = a_{ij}$.
By the same process, we get the following results:
$X = (e_1 \otimes \underline{e_1} \otimes e_1 + e_2 \otimes \underline{e_1} \otimes e_4) + (e_1 \otimes \underline{e_2} \otimes e_3 + e_1 \otimes \underline{e_2} \otimes e_4 + e_2 \otimes \underline{e_2} \otimes e_2 + e_2 \otimes \underline{e_2} \otimes e_4) + (e_1 \otimes \underline{e_3} \otimes e_2 + e_1 \otimes \underline{e_3} \otimes e_3 + e_2 \otimes \underline{e_3} \otimes e_1)$, *which implies*

$$cs_2(X) = \left\langle \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \right\rangle.$$

Also, $X = (e_1 \otimes e_1 + e_2 \otimes e_3) \otimes \underline{e_1} + (e_1 \otimes e_3 + e_2 \otimes e_2) \otimes \underline{e_2} + (e_1 \otimes e_2 + e_1 \otimes e_3) \otimes \underline{e_3} + (e_1 \otimes e_2 + e_2 \otimes e_1 + e_2 \otimes e_2) \otimes \underline{e_4}$, *which implies that*

$$cs_3(X) = \left\langle \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 1 & 0 \\ 1 & 1 & 0 \end{pmatrix} \right\rangle.$$

Now, let us define one of the main complexity measures we are going to use throughout the thesis.

Definition 3.0.7. *Let* $X \in \mathbb{F}^{k \times n \times m}$. *The tensor rank of* X *is the minimum integer* R *such that* $X = \sum_{r=1}^{R} u_r \otimes v_r \otimes w_r$ *for* $u_1, \ldots, u_R \in \mathbb{F}^k$, $v_1, \ldots, v_R \in \mathbb{F}^n$, *and* $w_1, \ldots, w_R \in \mathbb{F}^m$. *We write* $trk(X)$ *to denote the tensor rank of* X.

We will also see that this has very interesting geometric meanings and try to create codes whose tensor ranks are minimal. In the next subsection, we will state and prove a lower bound on the tensor rank that will be one of the fundamental theorems which we are going to use in this thesis.

3.1 Three-Way Arrays

The following theorem stated in [15] provides a lower bound for the tensor rank.

Theorem 3.1.1. *Let $X \in \mathbb{F}^{k\times n\times m}$ be 1-concise. Then,*

$$trk(X) \geq dim_1(X) + \min\{trk(m_1(u,X)) : u \in \mathbb{F}^k \setminus \{0\}\} - 1.$$

In our language, X being 1-concise means $dim_1(X) = k$. Recall that a rank metric code is a subspace $C \subseteq \mathbb{F}_q^{n\times m}$. If $C \neq \{0\}$, then the minimum distance of the code is $d(C) := \min\{rank(M) : M \in C, M \neq 0\} = \min\{d(M,N) : M,N \in C, M \neq N\}$, as stated in Definition 2.1.5. Since $cs_1(X) = \langle m_1(e_1,X),\ldots,m_1(e_k,X)\rangle$ is the span of $n\times m$ matrices, and the e_i's are nonzero, we have $cs_1(X) \subseteq \mathbb{F}_q^{n\times m}$, and $cs_1(X) \neq \{0\}$, for $X \neq 0$. Thus, $d(cs_1(X)) = \min\{trk(m_1(u,X)) : u \in \mathbb{F}^k \setminus \{0\}\}$. That is, in our terminology, Kruskal's Theorem is equivalent to say that $trk(X) \geq k + d(cs_1(X)) - 1$.

Theorem 3.1.2 (**Kruskal's Theorem**)**.** *Let $X \in \mathbb{F}^{k\times n\times m}$ be 1-concise. Then, $trk(X) \geq k + d(cs_1(X)) - 1$.*

We will prove Kruskal's Theorem using three-way arrays. Note that, in this subsection, we will also construct correspondences with the previous parts and show the analogues versions in three-way arrays language, just as we did above with rewriting the Kruskal theorem. A three-way array (or 3D matrix) is an array of numbers x_{ijk} for $i \in [I]$, $j \in [J]$, and $k \in [K]$. We say X is an $I \times J \times K$ array, x_{ijk}.

Definition 3.1.3. *A v-slice of X is a matrix formed by fixing the v-th index, for $v = 1,2,3$. We will use X_i to indicate the i-th slice of X, which is a $J \times K$ matrix.*

It is clear that there are I such slices since i runs from 1 to I. We let $dim_i(X)$ to be the dimension of the space consisting of all linear combinations of X_i's. This is actually the same as we defined them in Definition 3.0.3 and 3.0.4. Let us show the correspondence with our original definitions by showing $dim_1(X) = dim(cs_1(X))$. First observation is that, v-slice corresponds to m_v map. We will only examine the case $v = 1$. The other cases $v = 2$ and $v = 3$ are similar. We can see this by recalling the m_1 map:

$$m_1 : \mathbb{F}^{S\times I} \times \mathbb{F}^{I\times J\times K} \to \mathbb{F}^{S\times J\times K}.$$

Since we are fixing the first index, we have $S = 1$ in the map. Thus, we have a map

$\mathbb{F}^{1\times I} \times \mathbb{F}^{I\times J\times K} \to \mathbb{F}^{1\times J\times K}(J \times K \ matrices)$, that is, m_1 takes I different inputs that fixes the first index just as there are I many 1-slices. So,

$$cs_1(X) = \langle m_1(e_1, X), \ldots, m_1(e_I, X)\rangle = \langle X_i \mid i = 1, \ldots, I\rangle.$$

Thus, $dim_1(X) = dim(cs_1(X))$, as desired. We shall call X to be 1-concise if $dim_1(X)$ is the number of slices X_i, i.e., $dim_1(X) = I$.
The natural continuation is the representation of X in this context. Let A be $I \times R$ matrix of elements (a_{ir}), and similarly B, and C are $J \times R$ and $K \times R$ matrices.

Definition 3.1.4. *The triple product $[A, B, C]$ of three matrices is a three-way array whose $(i, j, k) - th$ element is $x_{ijk} = \sum_{r=1}^{R} a_{ir} b_{jr} c_{kr}$.*

Obviously, a triple product can be taken only when all three matrices have the same number of columns. At this point, we would like to continue our analogy by realizing that the definition above is the same as the definition of a coordinate tensor given in Note 3.0.2. Suppose $X = [A, B, C]$ and the decomposition involves R columns. This corresponds to $X = \sum_{i=1}^{R} A_i \otimes B_i \otimes C_i$, where

$$A = \left(\begin{array}{c|c|c} A_1 & \ldots & A_R \end{array} \right), B = \left(\begin{array}{c|c|c} B_1 & \ldots & B_R \end{array} \right), C = \left(\begin{array}{c|c|c} C_1 & \ldots & C_R \end{array} \right).$$

Let V be an n-dimensional vector space. Let $a \otimes b \otimes c \in V \otimes V \otimes V$, where $a = (a_1, \ldots, a_n)$, $b = (b_1, \ldots, b_n)$, and $c = (c_1, \ldots, c_n)$. Also, let $\{e_1, \ldots, e_n\}$ be a basis for V. Then, $a = \sum_{i=1}^{n} a_i e_i,\ b = \sum_{j=1}^{n} b_j e_j,\ c = \sum_{k=1}^{n} c_k e_k$. Therefore, we have

$$\begin{aligned} a \otimes b \otimes c &= (\sum_{i=1}^{n} a_i e_i) \otimes (\sum_{j=1}^{n} b_j e_j) \otimes (\sum_{k=1}^{n} c_k e_k) \\ &= \sum_{i,j,k}^{n} [a_i b_j c_k (e_i \otimes e_j \otimes e_k)] \\ &= \sum_{i,j,k}^{n} [x_{ijk} (e_i \otimes e_j \otimes e_k)] \end{aligned}$$

This connection shows how we defined the 3-tensors in the first place. Again just as in the beginning of the chapter, it is time to introduce multiplication of an array by a matrix to continue our analogy. The product in general is an array, as well. We are only interested in left multiplication, the m_1 map, which we write as

$$UX = U[A, B, C] = [UA, B, C]$$

This multiplication is associative just as matrix multiplication.

Definition 3.1.5. *Let $X = [A, B, C]$. We say that X is a representation with R*

columns if the number of columns in each of the matrices A, B, and C is R. Then, we say that rank of this array is R.

Note that, if X is 1-concise, then A has full rank. The reason is that, A is an $I \times R$ matrix by definition and X being 1-concise implies $dim_1(X) = I$. Since we have $dim_1(X) \leq rank(A)$, we conclude that A has full rank. Here, let us also note the following inequality which follows from the definition:

$$rank(X) \geq dim_1(X).$$

The following lemma shows the importance of being 1-concise.

Lemma 3.1.6. *Let X be 1-concise. If $u \neq 0$, then $uX \neq 0$.*

Proof. Here u is a vector, so that uX is a matrix formed by taking a linear combination of the 1-slices of X. However, we are also given that X is 1-concise. Thus, all 1-slices of X are linearly independent. That means, if $u \neq 0$, then $uX \neq 0$. □

Recall that our goal is to prove Kruskal's Theorem 3.1.2. The following lemma and theorem are needed to proceed that are stated in [15].

Lemma 3.1.7. *Let $X = [A, B, C]$ be a representation with R columns. Then, $rank(X) \leq R -$ number of zero columns of A.*

Proof. Suppose A has p zero columns. Without loss of generality, call them $A_1, ..., A_p$ and represent A as $A = \left(\begin{array}{c|c|c|c} A_1 & \dots & A_p & \dots & A_R \end{array} \right)$. Then, we have $x_{ijk} = \sum_{r=1}^{R} A_{ir}B_{jr}C_{kr} = \sum_{r=p+1}^{R} A_{ir}B_{jr}C_{kr}$. So, $rank(X) \leq R - p$, as desired. □

Theorem 3.1.8. *Suppose X is 1-concise, and $\Upsilon = \{u | u \neq 0\}$, where u is a vector. Then, $rank(X) \geq \min_{u \in \Upsilon} rank(uX) + dim_1(X) - 1$.*

We claim that this theorem is equivalent with Kruskal's Theorem. Observe that, in both cases $u \neq 0$. In both statements, $I - 1$ are the same since X being 1-concise in Theorem 3.1.8 implies that $dim_1(X) = I$. Note that tensor rank equals to rank when we have 2-tensors. Multiplication is given by the m_1 map in tensors. So, we are multiplying u and X and taking the minimum of the rank in both cases. Thus, $\min\{trk(m_1(u, X)) : u \in \mathbb{F}^k \setminus \{0\}\}$ and $\min_{u \in \Upsilon} rank(uX)$ where $\Upsilon = \{u | u \neq 0\}$ are equivalent. In addition to that, X is 1-concise in both cases, so we have two equivalent statements. We will prove this theorem, and that will mean proving Kruskal's Theorem 3.1.2. Here is the proof of Theorem 3.1.8.

Proof. Since X is 1-concise, by Lemma 3.1.6, $u \neq 0$ implies $uX \neq 0$. So, $\min_{u\in\Upsilon} rank(uX) > 0$. Observe that we can have $rank(uX) = 1$ by choosing u equals to 1 in a single entry and 0 elsewhere. So, $\min_{u\in\Upsilon} rank(uX) = 1$. Then, we are left to prove that $rank(X) \geq 1 + dim_1(X) - 1 = dim_1(X)$. Of course, $rank(X) \geq dim_1(X)$ as we noted before. So, we are done. □

Kruskal proves it in a different setting by first proving the following theorem.

Theorem 3.1.9. *[Theorem 2 in [15]] If $X = [A, B, C]$ is any representation, then we have $R \geq \min_{T\in\Upsilon} rank(TX) + max_{S\in\delta}$(number of zero columns in SA), where Υ is any set of matrices and $\delta \subseteq \Upsilon$.*

Proof. Let R be the number of columns in A, B, C. For any $S \in \delta$, we have $\min_{T\in\Upsilon} rank(TX) \leq rank(SX)$ since $\delta \subset \Upsilon$. We know that $X = [A, B, C]$, so $SX = [SA, B, C]$ and $rank(SX) = rank([SA, B, C])$.

$$\begin{aligned} \min_{T\in\Upsilon} rank(TX) &\leq rank(SX) = rank([SA, B, C]) \\ &\leq R - \text{number of zero columns in SA} \end{aligned}$$

where the second inequlity follows from Lemma 3.1.7. Since this is true for any S, we can take the maximum over all $S \in \delta$. Thus, we have

$$min_{T\in\Upsilon} rank(TX) \leq R - max_{S\in\delta}(\text{number of zero columns in SA}).$$

□

Kruskal considered Theorem 3.1.8 as a corollary of this theorem and proves it differently but in an elegant manner. We give the proof here, as well.

Proof. Let $X = [A, B, C]$ and R be the number of columns in X, i.e., by definition $rank(X) = R$. Define $\delta = \{u | uA \neq 0\}$. X being 1-concise with the Lemma 3.1.6 implies $\Upsilon = \{u | u \neq 0\} = \{u | uX \neq 0\}$. We claim that $\Upsilon = \delta$. We will show both inclusions. $uX = [uA, B, C] \neq 0$ means $uA \neq 0$, so $\Upsilon \subset \delta$. However, if $uA \neq 0$, then $u \neq 0$, so $\delta \subset \{u | u \neq 0\} = \Upsilon$. Thus, we have $\Upsilon = \delta$. Now, consider

$$\begin{aligned} &max_{u\in\delta}(\text{number of zero columns in uA}) \\ &= max_{u\in\delta}(\text{number of columns of A which are orthogonal to u}) \geq rank(A) - 1 = I - 1. \end{aligned}$$

First of all, since X is 1-concise, $rank(A) = I$. Last inequality follows from picking I independent columns of A, and select u orthogonal to $I - 1$ of them so that we

have $u \neq 0$ and $uA \neq 0$. Thus, we get

$$max_{u \in \delta}(\text{number of zero columns in uA}) \geq I - 1.$$

Now, we will use Theorem 3.1.9 which states that if $\delta \subseteq \Upsilon$, then we have

$$R \geq \min_{u \in \Upsilon} rank(uX) + max_{u \in \delta}(\text{number of zero columns in uA}).$$

Here, $\delta = \Upsilon$, so we can use the theorem.

$$\begin{aligned} R &\geq \min_{u \in \Upsilon} rank(uX) + max_{u \in \delta}(\text{number of zero columns in uA}) \\ &\geq \min_{u \in \Upsilon} rank(uX) + dim_1(X) - 1 \end{aligned}$$

Since $rank(X) = R$ and $rank(A) = dim_1(X)$, we are done. □

Now, we have a lower bound on the tensor rank. We noted in Definition 3.0.7 that the rank of a tensor equals to the minimal number of pure tensors whose sum give us the tensor back. This minimal decomposition of a given tensor into its pure tensor components is a very important problem and has many applications from machine learning to quantum information as well as algebraic geometry. In the last chapter, we will pick up the algebraic geometric approach and see that the points of the Segre variety corresponds to pure tensors. Afterwards, we will argue that we can construct some minimal tensor rank codes by this method. To explain the difficulty of this problem, it is important to note that there are no known algorithms like Gauss-Jordan elimination for computing the tensor rank.

Open Problem 3.1.10. *Given a tensor, find an algorithm to compute its rank.*

We will start examining the tensor rank in the next subsection and finally define what we mean by the tensor rank of a code.

3.2 Tensor Rank

For a given $\mathbb{F}_q - [n \times m, k]$ code C, encoding is done via an $\mathbb{F}_q$-monomorphism(i.e., an injective homomorphism)

$$E : \mathbb{F}_q^k \to \mathbb{F}_q^{n \times m}.$$

Here $\mathbb{F}_q^k$ is called the information space and $\mathbb{F}_q^{n \times m}$ is called an ambient space. Roughly speaking, ambient space is a space surrounding an object along with the object itself. The space of all such encoding maps is a subset of set of all $\mathbb{F}_q$-module homomorphisms $\mathrm{Hom}_{\mathbb{F}_q}(\mathbb{F}_q^k, \mathbb{F}_q^{n \times m}) = H$.

Theorem 3.2.1. *If R is a commutative ring and A, B are two R-modules, then $Hom_R(A, B)$ is an R-module.*

By the above theorem, we have H is also an $\mathbb{F}_q$-module since $\mathbb{F}_q$ is commutative. We also know that $\mathbb{F}_q$ is a field, so H is a $\mathbb{F}_q$-vector space of dimension $k \times n \times m$. Therefore, $\mathrm{Hom}_{\mathbb{F}_q}(\mathbb{F}_q^k, \mathbb{F}_q^{n \times m}) \cong \mathbb{F}_q^{k \times n \times m}$. The isomorphism is the following:

$$\begin{aligned} \varphi : \mathbb{F}_q^{k \times n \times m} &\longrightarrow H \\ \mathbf{X} &\longmapsto E_{\mathbf{X}} \\ \mathbb{F}_q^k &\longmapsto \mathbb{F}_q^{n \times m} : g \longrightarrow m_1(g, X) \end{aligned}$$

Remark 3.2.2. *One might argue that $m_1(g, X)$ was defined to be an element of $\mathbb{F}_q^{1 \times n \times m}$, and so it should not lie in $\mathbb{F}_q^{n \times m}$. Here we are seeing $m_1(g, X) \in \mathbb{F}_q^{n \times m}$ by considering the isomorphism $\mathbb{F}_q^{1 \times n \times m} \cong \mathbb{F}_q^{n \times m}$ since the first component is a scalar.*

So, we can see 3-tensors as encoders. Thus, as an analogy of the generator matrix of a linear code, we define a generator tensor of a rank-metric code.

Definition 3.2.3. *A generator tensor for an $\mathbb{F}_q - [n \times m, k]$ code C is an element $X \in \mathbb{F}^{k \times n \times m}$ such that $cs_1(X) = C$.*

Lemma 3.2.4. *If $X \in \mathbb{F}^{k \times n \times m}$ is a generator tensor of the code C, then any codeword in C is of the form $m_1(a, X)$ where $a \in F^k$.*

Proof. By definition, $C = cs_1(X) = \langle m_1(e_1, X), \ldots, m_1(e_k, X) \rangle$. Thus, any codeword is of the desired form. □

Now, recall that X is 1-concise if $dim(cs_1(X)) = k$. This means, if X is a generator tensor of C, then X is 1-concise. This is very useful but we need a clever way to

make use of this definition. The important question is how to realize a code C as a contraction space of a tensor X. It depends on the tensor rank! We want to somehow realize a code as a contraction space of a tensor X. We want to express the generating tensors as minimal sums of pure tensors. However, given a tensor, finding a tensor rank is NP-complete as proven in [9]. Here, the importance of generator tensors will appear via help of the following theorem.

Theorem 3.2.5. *[Theorem 14.45 in [2]] Consider $T \in \mathbb{F}^{k\times n\times m}$, and $R > 0$. Let $X_1, \ldots, X_k, Y_1, \ldots, Y_n, Z_1, \ldots, Z_m$ be indeterminates over $\mathbb{F}$. TFAE:*

(1) $trk(T) \leq R$.

(2) $\exists A_1, \ldots, A_R \in \mathbb{F}^{n\times m}$ of rank 1 matrices such that $cs_1(T) \subseteq span\{A_1, ..., A_R\}$.

(3) $\exists D_1, \ldots, D_k \in \mathbb{F}^{R\times R}$ and $P \in \mathbb{F}^{n\times R}$, $Q \in \mathbb{F}^{m\times R}$ such that
$cs_1(T) = P\langle D\rangle Q^T = P\langle D_1, \ldots, D_k\rangle Q^T = \langle PD_1Q^T, \ldots, PD_kQ^T\rangle$.

(4) There exists linear forms $f_s \in X$, $g_s \in Y$, and $h_s \in Z$ for $s \in [R]$ such that

$$\sum_{i,j,l} t_{ijl}X_iY_jZ_l = \sum_{s=1}^{R} f_s(X)g_s(Y)h_s(Z).$$

Proof. We will denote $u_r = (u_{jr} : 1 \leq j \leq k) \in \mathbb{F}^k$ and $e_i \in \mathbb{F}^{1\times k}$.
"1 $\implies$ 2" Let $trk(T) \leq R$. Then, $T = \sum_{r=1}^{R} u_r \otimes v_r \otimes w_r$ for some $u_r \in \mathbb{F}^k$, $v_r \in \mathbb{F}^n$ and $w_r \in \mathbb{F}^m$. Recall that $m_1(e_j, T) = \sum_{r=1}^{R} u_{jr}v_r \otimes w_r$ by Note 3.0.5. Thus,

$$\begin{aligned} cs_1(T) &= \langle m_1(e_1, T), \ldots, m_1(e_k, T)\rangle \\ &= \langle \sum_{r=1}^{R} u_{jr}v_r \otimes w_r : 1 \leq j \leq k\rangle \\ &\subseteq \langle v_r \otimes w_r : 1 \leq r \leq R\rangle = span\{A_1, \ldots, A_R\}. \end{aligned}$$

"2 $\implies$ 1" Assume the hypothesis $cs_1(T) \subseteq span\{A_1, ..., A_R\}$. Let $A_r = v_r \otimes w_r$ be rank 1 matrices. Then, for all $1 \leq j \leq k$, there exists $u_{jr} \in \mathbb{F}$ by Note 3.0.2 such that

$$m_1(e_j, T) = \sum_{r=1}^{R} u_{jr}A_r = \sum_{r=1}^{R} u_{jr}v_r \otimes w_r.$$

Thus, $T = \sum_{r=1}^{R} u_r \otimes v_r \otimes w_r$ and $trk(T) \leq R$.
"1 $\iff$ 4" Apply Note 3.0.2 to Definition 3.0.7, and then set

$$f_s(X) = \sum_{i=1}^{k} u_{ir}X_i,\ g_s(X) = \sum_{j=1}^{n} v_{jr}Y_j,\ h_s(X) = \sum_{l=1}^{m} w_{lr}Z_l.$$

"$1 \implies 3$" Let D_j be $R \times R$ matrix such that the diagonal elements of D_j are u_{jr} for $1 \leq r \leq R$. Now, construct two matrices $P = (v_{jr} : 1 \leq j \leq n, 1 \leq r \leq R)$ and $Q = (w_{jr} : 1 \leq j \leq m, 1 \leq r \leq R)$. Then,

$$PD_jQ^T = P \begin{pmatrix} u_{j1} & & 0 \\ & \ddots & \\ 0 & & u_{jR} \end{pmatrix} Q^T = \begin{pmatrix} | & & | \\ v_1 u_{j1} & \dots & v_R u_{jR} \\ | & & | \end{pmatrix} \begin{pmatrix} - & w_1 & - \\ & \vdots & \\ - & w_R & - \end{pmatrix} = \sum_{r=1}^{R} u_{jr} v_r \otimes w_r.$$

"$3 \implies 1$" Construct $T = \sum_{r=1}^{R} u_r \otimes v_r \otimes w_r$ such that v_r is the r-th column of P, w_r is the r-th column of Q and u_{jr} is the r-th element of the main diagonal of D_j for each j. Then, by construction the tensor rank is at most R, as desired. □

We will write an example to understand what is going on.

Example 3.2.6. *Let* $X = e_1 \otimes (e_1 \otimes e_4 + e_2 \otimes e_2 + e_3 \otimes e_2 + e_3 \otimes e_4) + e_2 \otimes (e_1 \otimes e_3 - e_3 \otimes e_3)$ *in* $\mathbb{F}^2 \otimes \mathbb{F}^3 \otimes \mathbb{F}^4$. *Firstly, we will find the first contraction space just as in the Example 3.0.6. We know that* $cs_1(X)$ *is the span of two* 3×4 *matrices. Thus,*

$$cs_1(X) = \left\langle \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \end{pmatrix} \right\rangle = \langle A, B \rangle$$

Now, the critical part here is to write X *as sums of rank 1 tensors. As mentioned before, this is a very hard problem in general. In this case, considering the small dimension of the example, we were able to find the following pure tensors* $X_1 = e_2 \otimes (e_1 - e_3) \otimes (e_2 + e_3 + e_4)$, $X_2 = (e_1 - e_2) \otimes (e_1 + e_3) \otimes (e_2 + e_4)$, $X_3 = 2e_2 \otimes e_3 \otimes (e_2 + e_4)$, *and* $X_4 = e_1 \otimes (-e_1 + e_2) \otimes e_2$. *Observe that*

$$X = X_1 + X_2 + X_3 + X_4.$$

Now, we choose P, Q *and* D_j *exactly in the proof of 3.2.5 in the part "*$1 \implies 3$.*" So,*

$$P = \begin{pmatrix} 1 & 1 & 0 & -1 \\ 0 & 0 & 0 & 1 \\ -1 & 1 & 1 & 0 \end{pmatrix}, Q = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 \end{pmatrix}, D_1 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, D_2 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

since $u_1 = e_2, u_2 = (e_1 - e_2), u_3 = 2e_2, u_4 = e_1, v_1 = (e_1 - e_3), v_2 = (e_1 + e_3), v_3 = e_3,$ $v_4 = (-e_1 + e_2), w_1 = (e_2 + e_3 + e_4), w_2 = (e_2 + e_4), w_3 = (e_2 + e_4), w_4 = e_2.$
Now, also see that we get the desired $A = PD_1Q^T$, $B = PD_2Q^T$.

Remark 3.2.7. *Note that in Example 3.2.6, X was written as a sum of 6 pure tensors in the beginning. That means, $trk(X) \leq 6$. However, we found 4 rank 1 tensors whose sum is X. Thus, we actually showed that $trk(X) \leq 4$.*

Theorem 3.2.5 also has a very useful corollary which will be used a lot.

Corollary 3.2.8. *Let C be an $\mathbb{F}_q - [n \times m, k, d]$ code. If X_1 and X_2 are different generator tensors for C, then*

$$trk(X_1) = trk(X_2).$$

Proof. By Theorem 3.2.5 and the fact that $cs_1(X_1) = C$ for a generator tensor X_1 of the code C, we have $trk(X_1) = \min\{R : C \subseteq span\{A_1, ..., A_R\}\}$. Similarly, by definition $cs_1(X_2) = C$. We know from basic linear algebra that the number of basis vectors is unique although the choice of basis vectors for a given vector space is not unique. Therefore, we have the desired result

$$trk(X_1) = \min\{R : C \subseteq span\{A_1, ..., A_R\}\} = trk(X_2).$$

□

As a conclusion, we have the following definition.

Definition 3.2.9. $trk(C) = trk(\text{any generating tensor of } C)$.

Now, naturally we wonder how small or how large can tensor rank be.

Theorem 3.2.10. $trk(C) \geq k + d - 1$.

Proof. Let X be a generating tensor for C. Then, $trk(C) = trk(X)$ as we observed above. Since any generating tensor is 1-concise, i.e., $dim_1(X) = k$, we can apply the Kruskal's Theorem 3.1.2. Thus,

$$\begin{aligned} trk(C) &= trk(X) \\ &\geq k + d(cs_1(X)) - 1 = k + d(C) - 1 = k + d - 1. \end{aligned}$$

□

The natural question comes to mind is that whether the tensor rank is invariant under code equivalence or not. To determine this, we first define what does it mean for codes to be equivalent.

Definition 3.2.11. *A bijective map $f : \mathbb{F}_q^{n\times m} \to \mathbb{F}_q^{n\times m}$ is called an isometry if it preserves rank distance, i.e,*

$$d(M,N) = d(f(M), f(N))$$

for all $M, N \in F_q^{n\times m}$. Two codes $C_1, C_2 \subseteq \mathbb{F}_q^{n\times m}$ are equivalent if and only if there exist an $\mathbb{F}_q$-linear isometry f such that $f(C_1) = C_2$.

The following theorem is very important because it shows that tensor rank is invariant under code equivalence. Note that, there are other types of equivalences but there we restrict ourselves to Definition 3.2.11.

Theorem 3.2.12. *Let C_1 and C_2 be two equivalent codes, then*

$$trk(C_1) = trk(C_2).$$

Definition 3.2.13. *Let C be an $\mathbb{F}_q - [n \times m, k]$ code with $trk(C) = R$. A set $A = \{A_1, \ldots, A_R\} \subseteq \mathbb{F}_q^{n\times m}$ of rank 1 matrices such that $C \subseteq \langle A \rangle$ is called an R-basis for the code C.*

Note that if C_1 and C_2 are a pair of codes satisfying $C_2 = f(C_1)$ for an isometry f, then any R-basis A for C_1 gives an R-basis $f(A)$ for C_2. Since f is a bijection, we have $dim(\langle A \rangle) = dim(\langle f(A) \rangle)$. Tensor rank of a code equals to the tensor rank of any of its generator tensors. Then, by Corollary 3.2.8, we get Theorem 3.2.12, as desired.

Definition 3.2.14. *A code C is called MTR(Minimal Tensor Rank) if the equality holds in Theorem 3.2.10, i.e, $trk(C) = k + d - 1$.*

In algebraic complexity theory, it is known that existence of MTR codes implies the existence of MDS codes. We can think of it as follows. Suppose $trk(C) = R$. We will see in Theorem 4.1.2 that there can be constructed a $[R, k, d']$ code where $d' \geq d$. Since, C is MTR, we have $R = k + d - 1$. Thus, by the Singleton bound, $d' = d$. So, we get a $[k + d - 1, k, d]$ code, i.e, an MDS code. What about the converse?

Open Problem 3.2.15. *Given an MDS code, find an MTR code.*

Note 3.2.16. *Existence of MTR codes for any given variable is still an open question, as well. The problem is adressed in [23] and they constructed some classes of MTR codes which we will explain later in the thesis.*

In coding theory, finding efficient ways to realize equivalence or inequivalence of codes is a big problem. We are going to define generalized tensor ranks of a code

C that can be used for that manner. Equivalent codes have the same generalized tensor ranks. However, note that generalized tensor ranks fail in terms of duality, i.e., there exists two codes with same generalized tensor rank but their duals have different generalized tensor ranks. For examples of these, see [23].

Definition 3.2.17. *Let $\mathcal{S}$ be the set of all subspaces of $\mathbb{F}_q^{n\times m}$ which are generated by rank 1 matrices. Let $r \in \mathbb{Z}$ such that $1 \leq r \leq k$ where k is the dimension of an $\mathbb{F}_q - [n \times m, k, d]$ code C. The r-th generalized tensor rank of C is*

$$d_r(C) = \min\{dim(S) \mid S \in \mathcal{S},\ dim(C \cap S) \geq r\}.$$

Let us explain this definition. We are looking for subspaces of rank 1 matrices in $\mathbb{F}_q^{n\times m}$ such that the intersection of those with the code is at least a space of dimension r. Then, among those subspaces, we take the one that has the minimum dimension. Basically, we are constructing an inclusion type of relation with subspaces of rank 1 matrices and the code C, that is, we are trying to find the tensor rank!

Note 3.2.18. *Observe that r-th generalized tensor rank is invariant under the group of the Segre variety that we will see in Chapter 7.*

Given a code C as in the above definition, we have the following theorem which shows the relation between the k-th generalized tensor rank and the tensor rank, and that is the reason we are looking at generalized tensor ranks in this thesis.

Theorem 3.2.19. *Let $1 \leq r \leq k$. Then, we have*

(1) $d = d_1(C)$.

(2) $trk(C) = d_k(C)$.

(3) If $1 \leq r \leq mn-1$, then $d_r(C)+1 \leq d_{r+1}(C)$.

(4) $trk(C) - k + r \geq d_r(C) \geq d + r - 1$.

Proof. (1) We are given that $d(C) = d$. Thus, consider a matrix $A \in C$ such that $rank(A) = d$. That means, we can represent A as a sum of rank 1 matrices A_i such that

$$A = A_1 + \ldots + A_d.$$

Clearly, the subspace $S = \langle A_1, \ldots, A_d \rangle$ gives us the desired subspace to calculate $d_1(C)$. Since $dim(S) = d$, we have $d_1(C) = d$, as desired.
(2)

$$trk(C) = \min\{R : C \subseteq span\{A_1, ..., A_R\}\}.$$

Let $A = span\{A_1, ..., A_R\}$. Note that $A \in \mathcal{S}$. Since $C \subseteq A$, we have $C \cap A = C$. By definition, A is the smallest dimensional set in $\mathcal{S}$ such that it contains C. Thus, $d_k(C) = dim(A) = R = trk(C)$, as desired.
(3) Note that if $r \geq mn$, then $d_{r+1}(C) = \emptyset$. Let $d_{r+1}(C) = dim(S)$. Then, we have $dim(C \cap S) \geq r+1$. Now, consider an hyperplane H of S. By definition, $H \subseteq S$ and $dim(H) = dim(S) - 1$. To finish, we will show that $d_r(C) \leq dim(H)$. Since $dim(H) = d_{r+1}(C) - 1$, that would imply that $d_r(C) + 1 \leq d_{r+1}(C)$, as desired. However, $d_r(C) \leq dim(H)$ is obvious since every hyperplane H of S meet C in dimension at least r due to the assumption $dim(C \cap S) \geq r+1$.
(4) Note that $d_{r+1}(C) \geq d_r(C) + 1 \geq d_{r-1}(C) + 2 \geq \ldots \geq d_1(C) + r = d + r$ where the first inequlity follows from (3) and the last inequality follows from (1). Thus, $d_r(C) \geq d + r - 1$. For the other part, we will use (2), i.e., $trk(C) = d_k(C)$. Since $1 \leq r \leq k$, we have $trk(C) = d_k(C) \geq d_{k-1}(C) + 1 \geq \ldots \geq d_r(C) + k - r$. □

We will close this section by noting that the theorem above provides another proof for Theorem 3.2.10. Here is the proof:

$$\begin{aligned} trk(C) = d_k(C) &\geq d_r(C) + k - r \\ &\geq d_{r-1}(C) + k - r + 1 \geq \ldots \\ &\geq d_1(C) + k - r + (r-1) = d + k - 1. \end{aligned}$$

We will close this subsection by showing one of the reasons why evaluating the tensor rank is very difficult. That is, it depends on field we are working on.

Example 3.2.20. *Let* $X \in \mathbb{F}^2 \otimes \mathbb{F}^2 \otimes \mathbb{F}^2$ *such that*

$$X = e_1 \otimes e_1 \otimes e_1 - e_1 \otimes e_2 \otimes e_2 - e_2 \otimes e_2 \otimes e_1 + e_2 \otimes e_1 \otimes e_2.$$

We will show that it has different ranks over $\mathbb{R}$ and $\mathbb{C}$.

$$X = (-e_1 - e_2) \otimes e_2 \otimes e_2 + (e_1 - e_2) \otimes e_1 \otimes e_1 + e_2 \otimes (e_1 + e_2) \otimes (e_1 + e_2).$$

So, its rank is 3 over $\mathbb{R}$.

$$X = (\frac{1}{2}e_1 + \frac{1}{2i}e_2) \otimes (e_1 + ie_2) \otimes (e_1 + ie_2) + (\frac{1}{2}e_1 - \frac{1}{2i}e_2) \otimes (e_1 - ie_2) \otimes (e_1 - ie_2).$$

So, its rank is 2 over $\mathbb{C}$.

3.3 Vector Codes

In this subsection we will provide some projective geometry background and show some constructions of rank-metric codes from the vector codes.

Definition 3.3.1. *Given a vector space V over a field $\mathbb{K}$, the projective space $PG(V)$ is the geometry obtained from the nontrivial subspaces of V, i.e., it is the set of equivalence classes of $V \setminus \{0\}$ with the equivalence relation $\sim$ given by $x \sim y$ if they are scalar multiple of each other.*

Let $V = \mathbb{K}^{n+1}$. Then, the subspaces of V of dimension 1, 2, 3, 4, and n are called

$$\text{points, lines, planes, solids, and hyperplanes of } PG(V) = PG(n, K).$$

Their projective dimensions are 0, 1, 2, 3, and $n-1$ respectively. Note that if $\mathbb{K} = \mathbb{F}_q$, then we denote $PG(n, \mathbb{K}) = PG(n, q)$.

Definition 3.3.2. *A vector code is an $\mathbb{F}_{q^m}$-subspace $C \subseteq \mathbb{F}_{q^m}^n$.*

We have two main questions in this subsection.

Question 3.3.3. *How do we obtain rank-metric codes from vector codes?*

Question 3.3.4. *How do we define the tensor rank of any vector code?*

We will start with the first question and then it will provide us a clue about the second one. Consider a basis $B = \{b_1, \ldots, b_m\}$ of $\mathbb{F}_{q^m}$ over $\mathbb{F}_q$. Let $v \in \mathbb{F}_{q^m}^n$. Define the matrix $B(v) \in F_q^{n \times m}$ whose (i, j)-th entry is the j-th coordinate of v_i with respect to B. Then, we define the corresponding rank-metric code as

$$B(C) = \{B(v) : v \in C\}.$$

In addition to that, we have $dim_{\mathbb{F}_q}(B(C)) = m \cdot dim_{\mathbb{F}_{q^m}}(C)$ and the minimum distance of the vector code is the minimum distance of the code $B(C)$ for any basis B. Note that, when $m \geq n$ the code C is MRD if and only if $B(C)$ is MRD if and only if $dim_{\mathbb{F}_q}(B(C)) = m(n-d+1)$ if and only if $m \cdot dim_{\mathbb{F}_{q^m}}(C) = m(n-d+1)$ if and only if $dim_{\mathbb{F}_{q^m}}(C) = n-d+1$. Thus, we have the following theorem.

Theorem 3.3.5. *Let $m \geq n$. A vector code $C \subseteq \mathbb{F}_{q^m}^n$ is MRD if and only if*

$$d(C) = n - dim_{\mathbb{F}_{q^m}}(C) + 1.$$

The following example explains the concepts defined above.

Example 3.3.6. *Consider* $C =< (1,\alpha) > \subseteq \mathbb{F}_8^2$. *Let* $\mathbb{F}_8 = \mathbb{F}_2[\alpha]/(\alpha^3+\alpha+1)$. *Consider the standard basis* $B = \{1, \alpha\ , \alpha^2\}$. *Note that*

$$1\cdot(1,\alpha) = (1,\alpha),\ \alpha\cdot(1,\alpha) = (\alpha,\alpha^2),\ \alpha^2(1,\alpha) = (\alpha^2,\alpha+1).$$

So, we have the matrices

$$B(1,\alpha) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}, B(\alpha,\alpha^2) = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, B(\alpha^2,\alpha+1) = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix}.$$

Thus, the corresponding rank-metric code is

$$B(C) = \langle \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix} \rangle = \langle B_1, B_2, B_3 \rangle.$$

Now, consider the normal basis $N = \{\alpha, \alpha^2, \alpha^2+\alpha+1\}$.*Note that*

$$(\alpha^2+\alpha+1)\cdot(1,\alpha) = (\alpha^2+\alpha+1, \alpha^2+1),\ \alpha\cdot(1,\alpha) = (\alpha,\alpha^2),\ \alpha^2(1,\alpha) = (\alpha^2,\alpha+1).$$

So, we have the matrices

$$N(\alpha^2+\alpha+1,\alpha^2+1) = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix}, N(\alpha,\alpha^2) = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, N(\alpha^2,\alpha+1) = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix}.$$

Thus, the corresponding rank-metric code is

$$N(C) = \langle \begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix} \rangle = \langle N_1, N_2, N_3 \rangle.$$

See that $B_2 = N_2$ *and* $B_3 = N_3$. *We also have the relations* $N_1 = B_1 + B_2 + B_3$ *and* $B_1 = N_1 + N_2 + N_3$. *So,* $N(C) = B(C)$.

Thus, we answered Question 3.3.3. This example also motivates the following remark and that will help us answering Question 3.3.4.

Remark 3.3.7. *Let* $C \subseteq \mathbb{F}_{q^m}^n$ *a vector code, and* B, N *be bases of* $\mathbb{F}_{q^m}/\mathbb{F}_q$. *Then, the rank-metric codes* $B(C)$ *and* $N(C)$ *are equivalent.*

The following corollary answers Question 3.3.4.

Corollary 3.3.8. *Let* $C \subseteq \mathbb{F}_{q^m}^n$ *a vector code, and* B, N *be bases of* $\mathbb{F}_{q^m}/\mathbb{F}_q$. *Then,*

$$trk(B(C)) = trk(N(C)).$$

Proof. By Remark 3.3.7, $B(C)$ and $N(C)$ are equivalent rank-metric codes. Then, by Theorem 3.2.12, we get the desired result. □

So, tensor rank of a vector code C is the tensor rank of any of its rank-metric representation. Of course, we already answered the Question 3.3.3 but there is a beautiful geometric answer to it, as well. Now, we will provide that here.
A point v of $PG(n-1,q^m)$ is a one dimensional subspace of $\mathbb{F}_{q^m}^n$ and consists of the set $S_v = \{\lambda v \mid \lambda \in \mathbb{F}_{q^m}\}$. To generalize this, consider a $(k-1)$-dimensional subspace Π of $PG(n-1,q^m)$. In the above, $k=1$. Note that $k=0$ corresponds to the empty set. Let $\Pi = PG(U)$ where $U = \langle u_1, \ldots, u_k \rangle$. Similarly, we get

$$S_U = \{a_1u_1, \ldots, a_ku_k | a_i \in \mathbb{F}_{q^m}\}.$$

Define the km-dimensional subspace $F(\Pi)$ as the set spanned by the elements of S_U. Now, the big idea is to use the field reduction map which we define next.

$$\begin{aligned} F_{n,m,q} : PG(n-1,q^m) &\longrightarrow PG(nm-1,q) \\ \Pi &\longmapsto F(\Pi) \end{aligned}$$

Consider Example 3.3.6 again. Here, $q=2$, $m=3$, and $n=2$.

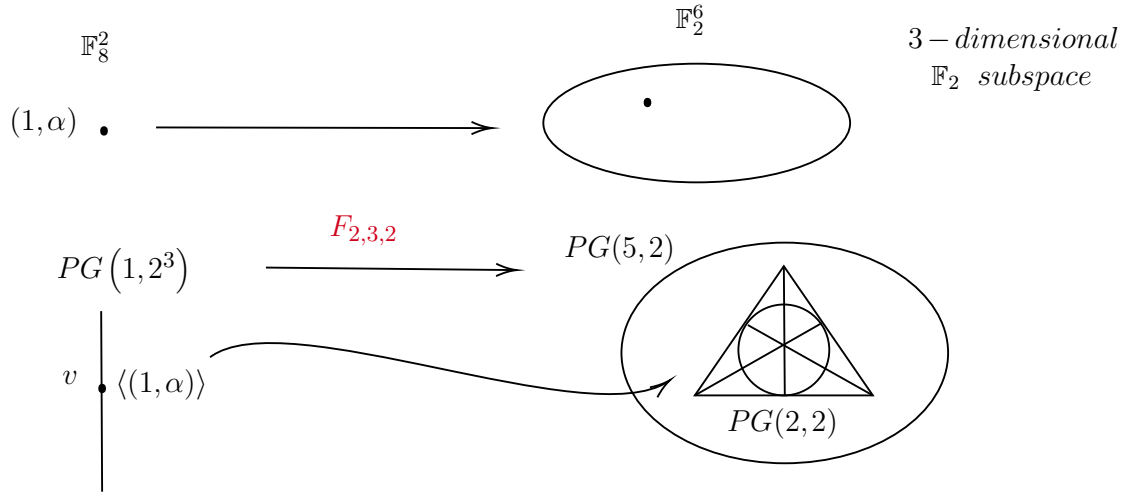


Figure 3.1 Field Reduction

Now consider $U = \langle(1,\alpha)\rangle$ and $\mathbb{F}_8 = \{0,1,\alpha,\alpha+1,\alpha^2,\alpha^2+1,\alpha^2+\alpha,\alpha^2+\alpha+1\}$. Then, we have $S_U = \{(0,0),(1,\alpha),(\alpha,\alpha^2),(\alpha^2,\alpha+1),(\alpha+1,\alpha^2+\alpha),(\alpha^2+1,1),(\alpha^2+\alpha,\alpha^2+\alpha+1),(\alpha^2+\alpha+1,\alpha^2+1)\}$. Let $\Pi = PG(U)$. Then, $F_{2,3,2}(\Pi) = F(\Pi)$ is actually the image of S_U in $PG(5,2)$ since $q=2$. Thus, we have $F(\Pi) = \{(1,0,0,0,1,0),(0,1,0,0,0,1),(0,0,1,1,1,0),(1,1,0,0,1,1),(1,0,1,1,0,0),(0,1,1,1,1,1),(1,1,1,1,0,1)\} = \{p_1,p_2,p_3,p_4,p_5,p_6,p_7\}$. Note that we removed $(0,0,0,0,0,0)$ since it is not a point of the projective space by definition. Write

these points as rows of a matrix, then reduce it into echelon form and remove the zero rows to get the generator matrix G of the code. So, we get

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 \end{pmatrix} \rightarrow \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 \end{pmatrix} = G.$$

However, our code must be consisting of matrices of size 2×3. So, transform each row of G into 2×3 matrices. Then we get the desired code

$$C' = \langle \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix} \rangle.$$

See that we get the same result as in Example 3.3.6.

Remark 3.3.9. *Note that the coordinates for the 7 points we just gave are consistent with the homogeneous coordinates of the points of $PG(2,2)$,i.e., the Fano Plane which is drawed in the Figure 3.1. Fano plane is the smallest projective plane since it is a projective plane of order 2. By definition, a projective plane of order 2 has 7 points and 7 lines(The circle in the figure is a line). Also, each line consist of 3 points, and there are 3 lines passing from each point. See that $p_1+p_2=p_4$, $p_1+p_3=p_5$, $p_1+p_6=p_7$, $p_2+p_3=p_6$, $p_2+p_5=p_7$, $p_3+p_4=p_7$, and $p_4+p_5=p_6$. Thus, our 7 lines are $L_1=\{1,2,4\},\ldots,L_7=\{3,4,7\}$. So, we can put the points as follows:*

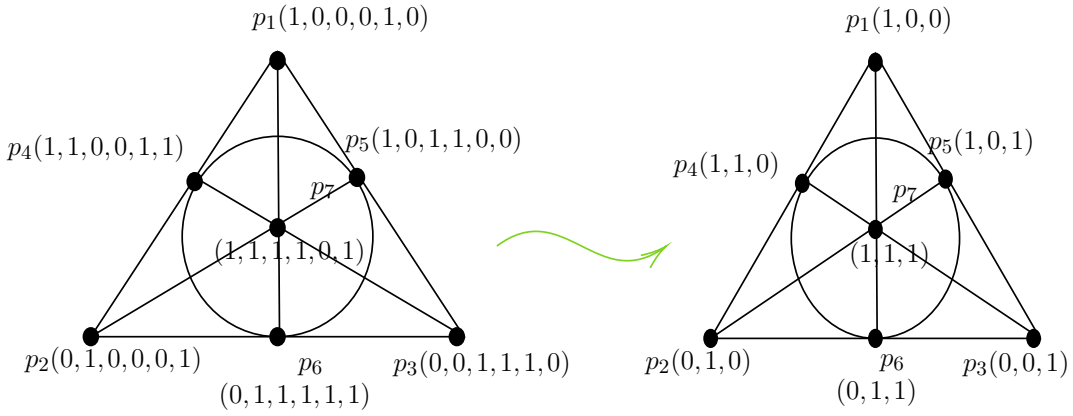


Figure 3.2 Fano Plane

Observe that if we send the first 3 coordinates of the points to points in $PG(2,2)$, then we get one of the coordinatization of the Fano plane. So, it is consistent in that aspect, as well. That would still work if we do the same for the last 3 coordinates.

In summary, the geometric way of creating a rank-metric code from a given vector code is as follows:

- We are given a code C in $\mathbb{F}_{q^m}^n$ of dimension k. We will transform it into a code in $\mathbb{F}_q^{n\times m}$ of dimension km.
- Consider it in $PG(n-1,q^m)$ as $\Pi = PG(C)$ where $C = \langle c_1, \ldots, c_k \rangle$ is of dimension k.
- Construct $S_C = \{a_1c_1 + \ldots + a_kc_k | a_i \in \mathbb{F}_{q^m}\}$ and the set spanned by the vectors in S_C as $F(\Pi)$. Note that $F(\Pi)$ corresponds to the image of Π under the field reduction map $F_{m,n,q}$.
- Write all the elements of $F(\Pi)$ as rows of a matrix A. Then, take the nonzero rows of $rref(A)$ to form the generator matrix G.
- Transform rows of G into $n \times m$ matrices and take their span as our newly produced rank-metric code.

Lastly, let us analyze if the rank-metric code

$$C' = \langle \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix} \rangle$$

corresponding to the vector code $C = \langle (1, \alpha) \rangle \subseteq \mathbb{F}_8^2$ is MRD or not. We will use the criteria Theorem 3.3.5 provides. See that $m = 3$, $n = 2$ and $dim_{\mathbb{F}_{q^m}}(C) = 1$. So, we will check whether $d(C) = 2$ or not. We know that $d(C) = d(C')$. After writing all the 8 elements of C' and checking that all the nonzero matrices have rank 2, we conclude $d(C') = d(C) = 2$, and thus the vector code C is MRD.

Remark 3.3.10. *In general, when we talk about codes, we also talk about their duals. Dual of a vector code H is defined as*

$$H^\perp := \{x \in \mathbb{F}_{q^m}^n \mid \langle x, y \rangle = 0 \, for \, all \, y \in C\}.$$

Naturally, we can ask the same question about the dual of a rank-metric code C.

Definition 3.3.11. *The dual of $\mathbb{F}_q - [n \times m, k]$ code C is*

$$C^\perp := \{X \in \mathbb{F}_q^{n\times m} \mid \langle X, Y \rangle = 0 \, for \, all \, Y \in C\}.$$

Note that $C^\perp$ is an $\mathbb{F}_q - [n \times m, nm - k]$ code. Of course, we need to define this product given in the definition.

Definition 3.3.12. *Trace Product of* $X,Y \in \mathbb{F}_q^{n\times m}$ *is* $\langle X,Y\rangle := Tr(XY^T)$.

We note that the map $(X,Y) \to Tr(XY^T)$ defines a symmetric bilinear form on $\mathbb{F}_q^{n\times m}$. In general, a dual code of C is the annihilator of C with respect to the bilinear form. Observe that

$$\begin{aligned}\langle aX+Y,Z\rangle &= Tr(Z^T(aX+Y))\\ &= Tr(aZ^TX+Z^TY) = aTr(Z^TX)+Tr(Z^TY)\\ &= a\langle Z,X\rangle+\langle Z,Y\rangle\end{aligned}$$

The other part can also be shown similarly. Let us do an example on finding the dual of a given rank-metric code.

Example 3.3.13. *Find the dual of the rank-metric code*

$$C = \langle \begin{pmatrix}1&0&0\\0&1&0\end{pmatrix}, \begin{pmatrix}0&1&0\\0&0&1\end{pmatrix}, \begin{pmatrix}0&0&1\\1&1&0\end{pmatrix}\rangle \subseteq \mathbb{F}_2^{2\times 3}.$$

Consider $M = \begin{pmatrix}a&b&c\\x&y&z\end{pmatrix} \in C^\perp$. *Then,*

$$Tr(\begin{pmatrix}1&0&0\\0&1&0\end{pmatrix}\begin{pmatrix}a&x\\b&y\\c&z\end{pmatrix}) = Tr(\begin{pmatrix}a&x\\b&y\end{pmatrix}) = a+y = 0.$$

$$Tr(\begin{pmatrix}0&1&0\\0&0&1\end{pmatrix}\begin{pmatrix}a&x\\b&y\\c&z\end{pmatrix}) = Tr(\begin{pmatrix}b&y\\c&z\end{pmatrix}) = b+z = 0.$$

$Tr(\begin{pmatrix}0&0&1\\1&1&0\end{pmatrix}\begin{pmatrix}a&x\\b&y\\c&z\end{pmatrix}) = Tr(\begin{pmatrix}c&z\\a+b&x+y\end{pmatrix}) = x+y+c = 0$. *After solving the three equations together, we get that* $M = \begin{pmatrix}a&b&c\\a-c&-a&-b\end{pmatrix}$. *So, we can write all possible 8 elements of the dual code. We know that it is of dimension* $nm-k$, *i.e.,* 3. *So, by inspection we can find 3 matrices which will form the basis. After doing so, we get*

$$C^\perp = \langle \begin{pmatrix}1&0&0\\1&1&0\end{pmatrix}, \begin{pmatrix}1&1&0\\1&1&1\end{pmatrix}, \begin{pmatrix}0&0&1\\1&0&0\end{pmatrix}\rangle.$$

Another occasion where the trace product being used is the double-dot product.

Definition 3.3.14. *Let p,k,n,m be natural numbers and define 3-tensors*

$$A=\sum_i u_i\otimes v_i\otimes w_i\in\mathbb{F}^{k\times n\times m},\ B=\sum_j u'_j\otimes v'_j\otimes w'_j\in\mathbb{F}^{p\times n\times m}.$$

The double-dot product $A:B$ is a 2-tensor lying in $\mathbb{F}^{k\times p}$ with the formula

$$A:B=\sum_{i,j}(v_i\cdot v'_j)(w_i\cdot w'_j)u_i\otimes u'_j.$$

If we consider their 3-way array representations as in [15] and as mentioned in the beginning of this section, we get $A=a_{ijl}$ and $B=b_{sjl}$. Then, we define the double product as

$$(A:B)_{is}=\sum_{j,l}a_{ijl}b_{sjl}\ \textit{where}\ 1\leq i\leq k,\ 1\leq s\leq p.$$

As a remark, note that we can apply this definition to 2-tensors. It is even okay if only one of them is a 2-tensor, as well. The trick is to see $\mathbb{F}^{n\times m}$ as $\mathbb{F}^{1\times n\times m}$. For two matrices $X,Y\in\mathbb{F}^{n\times m}$, we have

$$X:Y=Tr(XY^T).$$

The following theorem is handy in computations.

Theorem 3.3.15. *Let $X\in\mathbb{F}^{s\times k}$, $Y\in\mathbb{F}^{s\times p}$, $A\in\mathbb{F}^{k\times n\times m}$, and $B\in\mathbb{F}^{p\times n\times m}$. Then,*

$$m_1(X,A):B=X(A:B),\ A:m_1(Y,B)=(A:B)Y^T.$$

Proof. In the first chapter, we noted the tensor product of two vectors as $a\otimes b=ab^T$. By using this definition, we have

$$(Xu_i)\otimes u'_j=(Xu_i){u'_j}^T=X(u_i{u'_j}^T)=X(u_i\otimes u'_j),$$

$$u_i\otimes(Yu'_j)=u_i(Yu'_j)^T=u_i({u'_j}^TY^T)=(u_i{u'_j}^T)Y^T=(u_i\otimes u'_j)Y^T.$$

We have $m_1(X,A)=\sum_i(Xu_i)\otimes v_i\otimes w_i$, by definition. Thus,

$$\mathrm{m}_1(X,A):B=\textstyle\sum_{i,j}(v_i\cdot v'_j)(w_i\cdot w'_j)(Xu_i)\otimes u'_j=\sum_{i,j}(v_i\cdot v'_j)(w_i\cdot w'_j)X(u_i\otimes u'_j)=X(A:B).$$

This proves the first equality. Proving the second one is again in a similar fashion. We have $m_1(Y,B)=\sum_j(Yu'_j)\otimes v'_j\otimes w'_j$, by definition. Thus,

$$\mathrm{A{:}m}_1(Y,B)=\textstyle\sum_{i,j}(v_i\cdot v'_j)(w_i\cdot w'_j)(u_i)\otimes(Yu'_j)=\sum_{i,j}(v_i\cdot v'_j)(w_i\cdot w'_j)(u_i\otimes u'_j)Y^T=(A:B)Y^T.$$

□

Now, using the double-dot product, as an analogy to parity-check matrix of a vector code, we define a parity-check tensor for a rank-metric code.

Definition 3.3.16. *Let C be an $\mathbb{F}_q - [n \times m, k]$ code, and let $Y \in \mathbb{F}_q^{(mn-k)\times n\times m}$. The tensor Y is called a parity-check tensor for C if $C = \{M \in \mathbb{F}^{n\times m} \mid Y : M = 0\}$.*

Choose a matrix $M \in \mathbb{F}^{n\times m}$. We can represent it as a vector code of length nm by

$$M \to (M_{11} \dots M_{1m} | \dots | M_{n1} \dots M_{nm}).$$

Then we can construct the generator matrix $G \in \mathbb{F}^{k\times nm}$ by

$$G_{it} = x_{ijl}, \text{ where } t = (j-1)m + l \text{ for } 1 \le j \le n, 1 \le l \le m.$$

Let $X \in \mathbb{F}^{k\times n\times m}$ be a generator tensor for C and Y be a parity check tensor for C. Form the generator matrix $G \in \mathbb{F}^{k\times nm}$ as above. Let $H \in \mathbb{F}^{(nm-k)\times nm}$ be a parity-check matrix for the vector code. So, $H_{st} = Y_{sjl}$ where $1 \le s \le nm - k$. Recall that $GH^T = 0$ by Theorem 2.0.5. Then we get

$$0 = (GH^T)_{is} = \sum_{t=1}^{nm} G_{it} H_{ts} = \sum_{j,l} X_{ijl} Y_{sjl} = (X : Y)_{is}.$$

Therefore, we get the desired analogy which says

$$X : Y = 0 \iff GH^T = 0.$$

Thus, given a generator tensor X of a rank-metric code C, we say Y is a parity-check tensor for C if and only if $X : Y = 0$. The following proposition is only natural.

Proposition 3.3.17. *Let $Y \in \mathbb{F}^{(nm-k)\times n\times m}$ and let C be an $\mathbb{F} - [n \times m, k]$ code. Then, Y is a generator tensor for $C^\perp$ if and only if Y is a parity check tensor for C.*

Proof. Y is a parity check tensor for C if and only if $Y : M = 0 \in \mathbb{F}^{(nm-k)\times 1}$ for all $M \in C$. Let $A \in \mathbb{F}^{1\times(nm-k)}$. Then, $A(Y : M) = 0$ for all $M \in C$. By Theorem 3.3.15, this is equivalent of saying $0 = m_1(A, Y) : M$. This is true for all $A \in \mathbb{F}^{1\times(nm-k)}$ and for all $M \in C$. However, that means we have

$$C^\perp = \{m_1(A, Y) | A \in \mathbb{F}^{nm-k}\}.$$

So, by Lemma 3.2.4, Y is a generator tensor for $C^\perp$, as desired. □

4. TENSOR RANK EXTREMAL CODES

In this section, our main goal is to explain tensor rank extremal and MTR codes.

4.1 Two Useful Maps

We will introduce two maps which will be very useful for the upcoming parts of the thesis.

Definition 4.1.1. *Let k, d be positive integers.*

$$N_q(k,d) = \min\{N \in \mathbb{N} |\ \textit{There exists an } \mathbb{F}_q - [N,k,d]\ \textit{code}\}.$$

It is clear from the definition that $N_q(k,d') \geq N_q(k,d)$ where $d' \geq d$.

Consider the linearly independent set of rank 1 matrices $A = \{A_1, \ldots, A_R\} \subseteq \mathbb{F}_q^{n \times m}$. Then, we have an $\mathbb{F}_q$-linear isomorphism between two vector spaces:

$$\psi_A :\ < A > \longrightarrow \mathbb{F}_q^R :\ \sum_{i=1}^{R} \mu_i A_i \longmapsto \sum_{i=1}^{R} \mu_i e_i.$$

There is a connection with linear block codes. Consider an R-basis A for the code C. Define the linear block code as the image of C under the map ψ_A, that is,

$$\psi_A(C) = C_A.$$

By Lemma 3.2.4, we know that any element M of the code C is of the form $M = m_1(a, X) = \sum_{i=1}^{R} (a \cdot u_r)(v_r \otimes w_r)$ for $a \in \mathbb{F}_q^k$. Let $A_r = v_r \otimes w_r$ for $1 \leq r \leq R$. Then, $\psi_A(M) = (a \cdot u_r : 1 \leq r \leq R)$. This means, C_A is an $\mathbb{F}_q - [R,k]$ code with the

generator matrix $\begin{pmatrix} | & & | \\ u_1 & \dots & u_R \\ | & & | \end{pmatrix}$. The following theorem explains more about this.

Theorem 4.1.2 (Theorem 4.11 in [23]). *Let C be an $\mathbb{F}_q - [n \times m, k, d]$ code with tensor rank R. Let A be an R-basis for C. Then,*

(1) For all $M \in C$, we have $rank(M) \leq wt(\psi_A(M))$.

(2) C_A is an $\mathbb{F}_q - [R, k, \geq d]$ code.

(3) $trk(C) \geq N_q(k, d)$.

Proof. Let $r = rank(M)$. That is, $M = \sum_{i=1}^{R} \mu_i A_i$ for some $\mu_i \in \mathbb{F}_q$ with at least r of its coordinates are nonzero. Thus, $\psi_A(M)$ has at least r nonzero entries, i.e, $r \leq wt(\psi_A(M))$ as desired. This proves the first one. For the second one, we only need to prove that the linear block code C_A has distance $\geq d$ since we already showed above that C_A is an $\mathbb{F}_q - [R, k]$ code.

$$d(C) = d = \min\{rank(M) : M \in C\} \text{ and } d(C_A) = \min\{wt(\psi_A(M)) : M \in C\}.$$

By the first part, result follows. Lastly, Suppose C_A is $\mathbb{F}_q - [R, k, d']$ code with $d' \geq d$. Then, $R \geq N_q(k, d')$. We are done since $N_q(k, d') \geq N_q(k, d)$, as explained before. □

Definition 4.1.3. *Let C be an $\mathbb{F}_q - [n \times m, k, d]$ code. C is called tensor rank extremal if $trk(C) = N_q(k, d)$.*

The following remark gives us our second useful map $\phi_{V,W}$ that provides a new way to represent the multiplication map using Note 3.0.5.

Remark 4.1.4. *Consider two full-rank matrices $V \in \mathbb{F}_q^{n \times R}$ and $W \in \mathbb{F}_q^{m \times R}$. Define the $\mathbb{F}_q$ linear map*

$$\phi_{V,W} : \mathbb{F}_q^R \longrightarrow \mathbb{F}_q^{n \times m} : x \longmapsto V diag(x) W^T.$$

Let v_r and w_r denote the r-th columns of the matrices V and W. Let $A_r = v_r \otimes w_r$ as before. Let $U \in \mathbb{F}_q^{k \times R}$ and $a \in \mathbb{F}_q^k$. Then we have $aU \in \mathbb{F}_q^R$. Let u_r represent the r-th column of U. Then, we have

$$aU = \begin{pmatrix} a_1 u_{11} + \ldots + a_k u_{k1} \\ \vdots \\ a_1 u_{1R} + \ldots + a_k u_{kR} \end{pmatrix} = \begin{pmatrix} a \cdot u_1 \\ \vdots \\ a \cdot u_R \end{pmatrix}.$$

Note that $diag(aU)$ is an $R \times R$ matrix whose diagonal entries are coming from aU and all the other entries are zero. Let $X = \sum_{r=1}^{R} u_r \otimes v_r \otimes w_r$. Then, we have

$$\begin{aligned}
Vdiag(aU)W^T &= \begin{pmatrix} v_{11}(a\cdot u_1) & v_{12}(a\cdot u_2) & \dots & v_{1R}(a\cdot u_R) \\ \vdots & \vdots & \vdots & \vdots \\ v_{n1}(a\cdot u_1) & v_{n2}(a\cdot u_2) & \dots & v_{nR}(a\cdot u_R) \end{pmatrix} \begin{pmatrix} w_{11} & \dots & w_{m1} \\ \vdots & \vdots & \vdots \\ w_{1R} & \dots & w_{mR} \end{pmatrix} \\
&= \begin{pmatrix} | & & | \\ (a\cdot u_1)v_1 & \dots & (a\cdot u_R)v_R \\ | & & | \end{pmatrix} \begin{pmatrix} - & w_1 & - \\ & \vdots & \\ - & w_R & - \end{pmatrix} \\
&= (a\cdot u_1)v_1 w_1^T + \ldots + (a\cdot u_R)v_R w_R^T \\
&= \sum_{r=1}^{R} (a\cdot u_r)v_r \otimes w_r \\
&= m_1(a, X).
\end{aligned}$$

To proceed, we need the concept of nondegenerate codes.

Definition 4.1.5. *Let C be a rank-metric code. The column support and the row support of C are $\mathbb{F}_q$-subspaces of $\mathbb{F}_q^n$ and $\mathbb{F}_q^m$ respectively, given by*

$$csupp(C) = \langle colsp(M)\rangle_{M\in C},\ rsupp(C) = \langle rowsp(M)\rangle_{M\in C}.$$

The code C is called nondegenerate if $csupp(C) = \mathbb{F}_q^n$, and $rsupp(C) = \mathbb{F}_q^m$.

Let $X \in \mathbb{F}_q^{k\times m\times n}$ be any generator tensor for the rank-metric code C. Then, we have

- $dim_1(X) = k$,
- $dim_2(X) = dim(csupp(C))$,
- $dim_3(X) = dim(rsupp(C))$.

Thus, by only knowing the generator tensor, we can say if the code is nondegenerate or not. We will see why that is useful in the following lemma. Note that, what we mean by $C = V\langle D\rangle W^T$ is explained in part (3) of Theorem 3.2.5.

Lemma 4.1.6. *Let C be nondegenerate $\mathbb{F}_q - [n\times m, k]$ code. Suppose $C = V\langle D\rangle W^T$ for $D = \{D_1, \ldots, D_k\}$ a set of $R\times R$ diagonal matrices and $V \in \mathbb{F}_q^{n\times R}$, $W \in \mathbb{F}_q^{m\times R}$ are full rank matrices(rank n and rank m). Define $A = \{A_r : 1 \le r \le R\}$ where $A_r = v_r \otimes w_r$. Then, $\phi_{V,W}(\psi_A(C)) = C$.*

Proof. Consider $Vdiag(p)W^T$, an arbitrary element in C. Then, we have $Vdiag(p)W^T = \sum_{i=1}^{R} p_r v_r \otimes w_r = \sum_{i=1}^{R} p_r A_r \in \langle A\rangle$. So, $C \subseteq \langle A\rangle$. This means $\psi_A(C)$

makes sense. Let $M \in \langle A \rangle$ such that $M = \sum_{i=1}^{R} \mu_r A_r$. Then, $\psi_A(M) = \mu \in \mathbb{F}_q^R$. Now, observe that

$$\phi_{V,W}(\mu) = V diag(\mu) W^T = \sum_{r=1}^{R} \mu_r v_r \otimes w_r = \sum_{r=1}^{R} \mu_r A_r = M.$$

Thus, we have $\phi_{V,W}(\psi_A(M)) = M$ for all $M \in \langle A \rangle$ and $C \subseteq \langle A \rangle$. Result follows directly. □

One might ask if the converse is true or not. The answer is no. We will give a counterexample here.

Example 4.1.7. *Let* $V = \begin{pmatrix} 1 & 1 \\ 1 & 0 \\ 0 & 1 \end{pmatrix}$ *and* $W = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$.
Observe that V *and* W *are full-rank matrices in* $\mathbb{F}_2$. *We can compute* A_1 *and* A_2 *directly from the tensor product of their columns.*

$$A_1 = \begin{pmatrix} 1 & 1 \\ 1 & 1 \\ 0 & 0 \end{pmatrix} \text{ and } A_2 = \begin{pmatrix} 1 & 0 \\ 0 & 0 \\ 1 & 0 \end{pmatrix}.$$

Also, let $D = \{ \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \}$. *Then, the code* $C = V\langle D \rangle W^T$ *is an* $\mathbb{F}_2 - [3 \times 2, 2]$ *code. We can compute* $\langle D \rangle$ *to find the 4 codewords of* C. *We get*

$$C = \{ \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}, \begin{pmatrix} 1 & 1 \\ 1 & 1 \\ 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 1 \\ 1 & 1 \\ 1 & 0 \end{pmatrix}, \begin{pmatrix} 1 & 0 \\ 0 & 0 \\ 1 & 0 \end{pmatrix} \} = \{c_1, c_2, c_3, c_4\}.$$

We have $c_1 = 0$, $c_2 = A_1$, $c_3 = A_1 + A_2$, *and* $c_4 = A_2$. *This means,* $\psi_A(c_1) = (0,0)$, $\psi_A(c_2) = (1,0)$, $\psi_A(c_3) = (1,1)$, *and* $\psi_A(c_4) = (0,1)$. *It can easily be checked that* $\phi_{V,W}(\psi_A(C)) = C$. *Now, we will show that* C *is not nondegenerate to finish up the example. Although* $rsupp(C) = \mathbb{F}_q^2$, *we have* $csupp(C) = \left\{ \begin{matrix} 0 \\ 0 \\ 0 \end{matrix}, \begin{matrix} 1 \\ 0 \\ 1 \end{matrix}, \begin{matrix} 1 \\ 1 \\ 0 \end{matrix}, \begin{matrix} 0 \\ 1 \\ 1 \end{matrix} \right\} \neq \mathbb{F}_q^3$.

Now, we are ready to state the big corollary of this section. Note the following.

Note 4.1.8.

$$\phi_{V,W}^{-1}(C) = \{x \in \mathbb{F}_q^R : V diag(x) W^T \in C\}.$$

We simply rewrite Theorem 4.1.2 and get our crucial result.

Corollary 4.1.9 (Corollary 4.14 in [23]). *Let C be an $\mathbb{F}_q - [n \times m, k, d]$ code with $trk(C) = R$. Let $D = \{D_1, \ldots, D_k\}$ be a set of $R \times R$ diagonal matrices such that $C = V\langle D\rangle W^T$ where V and W are full rank matrices of rank n and m, respectively. Then,*

(i) For all $M \in C$, we have $rank(M) \leq wt(\phi_{V,W}^{-1}(M))$,

(ii) $\phi_{V,W}^{-1}(C)$ is an $\mathbb{F}_q - [R, k, \geq d]$ code,

(iii) If C is a tensor rank extremal code, then $\phi_{V,W}^{-1}(C)$ is an $\mathbb{F}_q - [R, k, d]$ code of length $N_q(k, d)$. In particular, if C is MTR, then $\phi_{V,W}^{-1}(C)$ is an MDS code.

Proof. (i) By Lemma 4.1.6, $\phi_{V,W}^{-1}(M) = \psi_A(M)$. By Theorem 4.1.2,

$$rank(M) \leq wt(\psi_A(M)) = wt(\phi_{V,W}^{-1}(M)).$$

(ii) By Lemma 4.1.6, we have $\phi_{V,W}^{-1}(C) = \psi_A(C)$. By Theorem 4.1.2, $\psi_A(C)$ is an $\mathbb{F}_q - [R, k, \geq d]$ code. Thus, $\phi_{V,W}^{-1}(C)$ is an $\mathbb{F}_q - [R, k, \geq d]$ code.
(iii) Let C be a tensor rank extremal code. That means, $trk(C) = N_q(k, d)$. Since C is of the form $V\langle D\rangle W^T$, and $d(C) = d$, there exists x_0 such that

$$rank(V diag(x_0) W^T) = d.$$

Since V is a full rank matrix, $d = rank(V diag(x_0) W^T) = rank(diag(x_0) W^T)$. Similarly, since W is a full rank matrix, $d = rank(diag(x_0) W^T) = rank(diag(x_0))$. Thus, we get $wt(x_0) = d$. Therefore, we have $d(\phi_{V,W}^{-1}(C)) = d$. Since $R = N_q(k, d)$, we have $\phi_{V,W}^{-1}(C)$ is an $\mathbb{F}_q - [R = N_q(k, d), k, d]$ code. Now, in particular we consider the case $R = k + d - 1$. Then, $[R = N_q(k, d), k, d] \implies [k + d - 1, k, d]$. That is, it becomes MDS, as desired. □

This gives us the third proof of Theorem 3.2.10. We know that $trk(C) = R$. By Corollary 4.1.9, we know that $\phi_{V,W}^{-1}(C)$ is an $[R, k, d']$ code where $d' \geq d$. Just using the Singleton bound 2.0.16. we get

$$trk(C) = R \geq k + d' - 1 \geq k + d - 1.$$

This is not the only use of Corollary 4.1.9 as we will constantly go back to this in the following parts of the thesis.

4.2 Extremal Triples

In this section we start by proposing the two main questions in the paper [23].

Question 4.2.1. *Let $R = N_q(k,d)$. Decide for which values of n and m, we can find an $\mathbb{F}_q - [n \times m, k, d]$ tensor rank extremal code $\phi_{V,W}(C)$,i.e., with $trk(\phi_{V,W}(C)) = R$ such that $V \in \mathbb{F}_q^{n \times R}$, $W \in \mathbb{F}_q^{m \times R}$ and C is an $\mathbb{F}_q - [R,k,d]$ code.*

Definition 4.2.2. *The triple (C,V,W) is called an extremal triple if it is a solution to Question 4.2.1. Note that this happens if and only if*

$$rank(Vdiag(x)W^T) \geq d \text{ for all } x \in C - \{0\}.$$

We will use this characterization when we try to show that some triple is extremal.

They also propose a special case of Question 4.2.1 as follows:

Question 4.2.3. *Let $R = k + d - 1$. Decide for which values of n and m, we can find an $\mathbb{F}_q - [n \times m, k, d]$ MTR code $\phi_{V,W}(C)$ such that $V \in \mathbb{F}_q^{n \times R}$, $W \in \mathbb{F}_q^{m \times R}$ and C is an $\mathbb{F}_q - [R,k,d]$ MDS code.*

Note that we are trying to work with full rank matrices V and W. However, in the statements of both of the questions we are not given that. Now, we will prove a lemma which will provide us to assume without loss of generality that V and W are full-rank matrices. So, this is a pretty big lemma for the analysis of Question 4.2.1. Before we state the lemma, we need the following definition.

Definition 4.2.4. *For an arbitrary matrix $M \in \mathbb{F}_q^{k \times R}$ and a vector $x \in \mathbb{F}_q^R$, define*

$$C_M = rowsp(M) \text{ and } C_{M_x} = rowsp(Mdiag(x)).$$

Lemma 4.2.5. *Let C be an $\mathbb{F}_q - [R = N_q(k,d), k, d]$ code. Let matrices $V \in \mathbb{F}_q^{n \times R}$ and $W \in \mathbb{F}_q^{m \times R}$ such that (C,V,W) is an extremal triple. Then, for all integers $n' \geq n$, $m' \geq m$ and for all matrices $V' \in \mathbb{F}_q^{n' \times R}$, $W' \in \mathbb{F}_q^{m' \times R}$ with $rowsp(V) \subseteq rowsp(V')$ and $rowsp(W) \subseteq rowsp(W')$, we have (C,V',W') is an extremal triple.*

This lemma means, we can increase n and m until we get full rank matrices V and W, as desired. We will prove it here.

Proof. Let (C,V,W) be an extremal triple. Also let $V' \in \mathbb{F}_q^{n' \times R}$, and $W' \in \mathbb{F}_q^{m' \times R}$ such that $rowsp(V) \subseteq rowsp(V')$ and $rowsp(W) \subseteq rowsp(W')$. Then, there exists matrices $A \in \mathbb{F}_q^{n' \times n'}$ and $B \in F_q^{m' \times m'}$ such that

$$AV' = \begin{pmatrix} V \\ \widetilde{V} \end{pmatrix} \text{ and } BW' = \begin{pmatrix} W \\ \widetilde{W} \end{pmatrix}.$$

Now, let $x \in C - \{0\}$. Let $D = diag(x) \in \mathbb{F}_q^{R \times R}$. To show that (C,V',W') is an extremal triple, we need to show that $rank(V'D(W')^T) \geq d$. Multiplying a matrix from left or right can not increase its rank. Thus, we have

$$\begin{aligned} rank(V'D(W')^T) &\geq rank(AV'D(W')^TB^T) \\ &= rank(\begin{pmatrix} V \\ \widetilde{V} \end{pmatrix} D \begin{pmatrix} W^T & \widetilde{W}^T \end{pmatrix}) \\ &= rank(\begin{pmatrix} VD \\ \widetilde{V}D \end{pmatrix} \begin{pmatrix} W^T & \widetilde{W}^T \end{pmatrix}) \\ &= rank(\begin{pmatrix} VDW^T & VD\widetilde{W}^T \\ \widetilde{V}DW^T & \widetilde{V}D\widetilde{W}^T \end{pmatrix}) \\ &\geq rank(VDW^T) \\ &\geq d. \end{aligned}$$

The last inequality follows from the fact that (C,V,W) is an extremal triple by the criteria given in Definition 4.2.2. □

In general showing that (C,V,W) is an extremal triple is a difficult job. To handle that problem, we will state some equivalent forms of it. To prove that theorem, we will need the following lemma.

Lemma 4.2.6. *Let $A \in \mathbb{F}_q^{n \times R}$ and $B \in \mathbb{F}_q^{m \times R}$. Then,*

$$rank(AB^T) = rk(A) - dim(C_A \cap C_B^{\perp}) = rank(B) - dim(C_B \cap C_A^{\perp}).$$

Proof. We will consider this in two cases where the matrices are full rank and not full rank. First, suppose $rank(A) = n$ and $rank(B) = m$. Note that rank of AB^T is the rank of the bilinear map

$$\varphi : \mathbb{F}_q^n \times \mathbb{F}_q^m \longrightarrow \mathbb{F}_q : (x,y) \longmapsto xAB^Ty^T.$$

By definition and using the rank-nullity theorem 1.1.7, we have

$$(\star) \qquad rank(\varphi) = n - dimker_L(\varphi) = m - dimker_R(\varphi).$$

Since A and B are full rank matrices, $C_A \cong \mathbb{F}_q^n$ and $C_B \cong \mathbb{F}_q^m$. Now, by calculating left and right kernel, we will show the desired result.

$$\begin{aligned} ker_L(\varphi) &= \{x \in \mathbb{F}_q^n | xAB^T y^T = 0 \ \forall y \in \mathbb{F}_q^m\} \\ &\cong \{v \in C_A | vB^T = 0\} \\ &= C_A \cap C_B^{\perp}. \end{aligned}$$

Similarly,

$$\begin{aligned} ker_R(\varphi) &= \{y \in \mathbb{F}_q^m | xAB^T y^T = 0 \ \forall x \in \mathbb{F}_q^n\} \\ &\cong \{w \in C_B | Aw^T = 0\} \\ &= C_B \cap C_A^{\perp}. \end{aligned}$$

By the Equation $(\star)$, result follows. Now suppose $rank(A) = s \leq n$ and $rank(B) \leq m$ excluding the case we analyzed above. By Theorem 1.1.16, we can use full-rank factorization. That is, there exists matrices $M \in \mathbb{F}_q^{s \times n}$ and $N \in \mathbb{F}_q^{t \times m}$ such that $MA \in \mathbb{F}_q^{s \times m}$ and $NB \in \mathbb{F}_q^{t \times m}$ are full-rank matrices with

$$(\star\star) \qquad C_A = C_{MA} \text{ and } C_B = C_{NB}.$$

Since we can not increase the rank by multiplying with other matrices, we have $rank(AB^T) \geq rank(MAB^T N^T)$. We will show that $rank(MAB^T N^T) \geq rank(AB^T)$ so that they are equal. By the Frobenius rank inequality 1.1.18, we have

$$rank(MA) + rank(AB^T N^T) \leq rank(MAB^T N^T) + rank(A).$$

This implies $rank(AB^T N^T) \leq rank(MAB^T N^T)$ since $rank(A) = rank(XA)$ by Equation $(\star\star)$. Again, Frobenius rank inequality implies

$$rank(AB^T) + rank(B^T N^T) \leq rank(AB^T N^T) + rank(B^T).$$

This means $rank(AB^T) \leq rank(AB^T N^T$ since $rank(B^T N^T) = rank(B^T)$ by Equation $(\star\star)$. Thus, we have

$\text{rank(AB}^T) \leq rank(AB^T N^T \leq rank(MAB^T N^T) \implies rank(AB^T) \leq rank(MAB^T N^T)$. Therefore, $rank(AB^T) = rank(MAB^T N^T) = rank((MA)(B^T N^T))$ where MA and

NB are full rank matrices. Then, we have by the first part

$$rank(AB^T) = rank(MA) - dim(C_{MA} \cap C_{YB}^{\perp}).$$

By Equation ($\star\star$) result follows. □

One might wonder about how to guarantee the equation ($\star\star$). We provide an example here to make it absolutely clear.

Example 4.2.7. *$A = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 4 & 6 & 8 \end{pmatrix}$, and $B = \begin{pmatrix} 2 & 0 & 1 & 1 \\ -1 & 1 & 3 & -2 \\ 1 & 1 & 4 & -1 \end{pmatrix}$. See that $n = 2$, $m = 3$, $R = 4$, $s = 1$, $t = 2$ and we have*

$$C_A = \langle \begin{pmatrix} 1 & 2 & 3 & 4 \end{pmatrix} \rangle \text{ and } C_B = \langle \begin{pmatrix} 2 & 0 & 1 & 1 \end{pmatrix}, \begin{pmatrix} -1 & 1 & 3 & -2 \end{pmatrix} \rangle.$$

So, M should be a 1×2, and N should be a 2×3 matrix. In general, if M is $p \times k$ matrix, then choose its rows as $e_i's$ where $i-th$ row of A contributes to the rank. That is, $M = \begin{pmatrix} 1 & 0 \end{pmatrix}$ and $N = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$ so that MA and YB are full rank matrices such that $C_A = C_{MA}$ and $C_B = C_{YB}$ as desired.

Now, we have the resources to prove an important theorem which gives 9 equivalent statements of being an extremal triple.

Theorem 4.2.8. *Let C be an $\mathbb{F}_q - [R = N_q(k,d), k, d]$ code. Let $n, m \in \mathbb{N}$ such that $d \leq n, m < R$ and $V \in \mathbb{F}_q^{n \times R}$, $W \in \mathbb{F}_q^{m \times R}$. The followings are equivalent:*

1. *(C, V, W) is an extremal triple.*
2. *$\forall c \in C - \{0\}$, $dim(C_V \cap C_{W_c}^{\perp}) \leq rank(V) - d$.*
3. *$\forall c \in C - \{0\}$, $dim(C_W \cap C_{V_c}^{\perp}) \leq rank(W) - d$.*
4. *$\forall c \in C - \{0\}$, $dim(C_{V_c} \cap C_W^{\perp}) \leq dim(C_{V_c}) - d$.*
5. *$\forall c \in C - \{0\}$, $dim(C_{W_c} \cap C_V^{\perp}) \leq dim(C_{W_c}) - d$.*
6. *$\forall c \in C - \{0\}$, $dim(C_{W_c} + C_V^{\perp}) \geq R - rank(V) + d$.*
7. *$\forall c \in C - \{0\}$, $dim(C_{V_c} + C_W^{\perp}) \geq R - rank(W) + d$.*
8. *$\forall c \in C - \{0\}$, $dim(C_V + C_{W_c}^{\perp}) \geq R - dim(C_{W_c}) + d$.*
9. *$\forall c \in C - \{0\}$, $dim(C_W + C_{V_c}^{\perp}) \geq R - dim(C_{V_c}) + d$.*

Proof. We will use Definition 4.2.2, Lemma 4.2.6 and Grassmann's Identity 1.1.3.
"1 $\iff$ 2"

$$\begin{aligned} rank(V diag(c) W^T) &= rank(V(W diag(c))^T) \\ &= rank(V) - dim(C_V \cap C_{W_c}^{\perp}) \geq d. \end{aligned}$$

So, we get $rank(V) - d \geq dim(C_V \cap C_{W_c}^{\perp})$.
"1 $\iff$ 3"

$$\begin{aligned} rank(V diag(c) W^T) &= rank(W diag(c)^T V^T) = rank(W(V diag(c))^T) \\ &= rank(W) - dim(C_W \cap C_{V_c}^{\perp}) \geq d. \end{aligned}$$

So, we get $rank(W) - d \geq dim(C_W \cap C_{V_c}^{\perp})$.
"1 $\iff$ 4"

$$\begin{aligned} rank(V diag(c) W^T) &= rank(V diag(c)) - dim(C_{V_c} \cap C_W^{\perp}) \\ &= dim(C_{V_c}) - dim(C_{V_c} \cap C_W^{\perp}) \geq d. \end{aligned}$$

So, we get $dim(C_{V_c}) - d \geq dim(C_{V_c} \cap C_W^{\perp})$.
"1 $\iff$ 5"

$$\begin{aligned} rank(V diag(c) W^T) &= rank(W diag(c) V^T) \\ &= rank(W diag(c)) - dim(C_{W_c} \cap C_V^{\perp}) \\ &= dim(C_{W_c}) - dim(C_{W_c} \cap C_V^{\perp}) \geq d. \end{aligned}$$

So, we get $dim(C_{W_c}) - d \geq dim(C_{W_c} \cap C_V^{\perp})$.
"5 $\iff$ 6"

$$\begin{aligned} dim(C_{W_c} + C_V^{\perp}) &= dim(C_{W_c}) + dim(C_V^{\perp}) - dim(C_{W_c} \cap C_V^{\perp}) \\ &\geq dim(C_{W_c}) + dim(C_V^{\perp}) - dim(C_{W_c}) + d \\ &= dim(C_V^{\perp}) + d \\ &= R - dim(C_V) + d = R - rank(V) + d. \end{aligned}$$

"4 $\iff$ 7"

$$\begin{aligned} dim(C_{V_c} + C_W^{\perp}) &= dim(C_{V_c}) + dim(C_W^{\perp}) - dim(C_{V_c} \cap C_W^{\perp}) \\ &\geq dim(C_{V_c}) + dim(C_W^{\perp}) - dim(C_{V_c}) + d \\ &= dim(C_W^{\perp}) + d \\ &= R - dim(C_W) + d = R - rank(W) + d. \end{aligned}$$

"2 $\iff$ 8"

$$\begin{aligned} dim(C_V + C_{W_c}^{\perp}) &= dim(C_V) + dim(C_{W_c}^{\perp}) - dim(C_V \cap C_{W_c}^{\perp}) \\ &\geq dim(C_V) + dim(C_{W_c}^{\perp}) - rank(V) + d \\ &= dim(C_V) + dim(C_{W_c}^{\perp}) - dim(C_V) + d \\ &= dim(C_{W_c}^{\perp}) + d \\ &= R - dim(C_{W_c}) + d. \end{aligned}$$

"3 $\iff$ 9"

$$\begin{aligned} dim(C_W + C_{V_c}^{\perp}) &= dim(C_W) + dim(C_{V_c}^{\perp}) - dim(C_W \cap C_{V_c}^{\perp}) \\ &\geq dim(C_W) + dim(C_{V_c}^{\perp}) - rank(W) + d \\ &= dim(C_W) + dim(C_{V_c}^{\perp}) - dim(C_W) + d \\ &= dim(C_{V_c}^{\perp}) + d \\ &= R - dim(C_{V_c}) + d. \end{aligned}$$

□

Next, we are going to examine how we can get (C, V, W) extremal triple in the case of C_V and C_W being MDS codes. Given that $V \in \mathbb{F}_q^{n \times R}$, the code C_V is generated by the rows of V, so V is a generator matrix of C_V. Thus, C_V is an $[R, n, d]$ code. Since it is MDS, $d = R - n + 1$. So, C_V is an $\mathbb{F}_q - [R, n, R - n + 1]$ code.

Lemma 4.2.9. *G is a generator matrix of an MDS code C of length n and dimension k if and only if any subset of k columns of G are linearly independent.*

Proof. Suppose every subset of k columns of G are linearly independent. Then any $k \times k$ submatrix of G are full rank. Recall that $C = \{uG | u \in \mathbb{F}_q^k\}$. So, any codeword of C is of the form uG. Since any $k \times k$ submatrix are full rank, that means any codeword $c = uG \in C$ has at most $k - 1$ zero coordinates. That is, $d \geq n - (k - 1)$, i.e, $d \geq n - k + 1$. By Singleton Bound, $d = n - k + 1$, i.e, C is MDS. Conversely, suppose C is MDS. Then, $d \geq n - k + 1$ so that with the Singleton bound, we have an MDS code C. The fact $d \geq n - (k - 1)$ means, there can not be any codeword with at least k zeros. Since any codeword is of the form uG, any $k \times k$ submatrix is full rank, and thus any k columns of G are linearly independent, as desired. □

Remark 4.2.10. *This is a very powerful lemma since it removes the necessity to mention the code. In addition to that, this lemma is basis-free, i.e, it works for any basis which is super nice.*

Also, note that Lemma 4.2.9 is the correspondence between linear MDS codes and arcs given in Note 2.0.18. By Lemma 4.2.9, any n columns of V are linearly independent since C_V is an MDS code of dimension n. We have the following proposition.

Proposition 4.2.11. *Let C be an $\mathbb{F}_q - [R = N_q(k,d), k, d]$ code. Let $n, m \in \mathbb{N}$ such that $d \leq n, m < R$ and $V \in \mathbb{F}_q^{n \times R}$, $W \in \mathbb{F}_q^{m \times R}$ are matrices such that C_V and C_W are MDS codes of dimension n and m. Then, we have*

$$n + m \geq R + d \implies (C, V, W) \text{ is an extremal triple.}$$

Before the proof, observe that C_V and C_W being MDS codes of dimension n and m implies that V and W are full-rank matrices since we have $dim(C_V) = rank(V) = n$ and $dim(C_W) = rank(W) = m$ by the fact $n, m < R$ in the hypothesis. It is good to note that, if we do not have the hypothesis $n, m < R$, then we do not necessarily have full rank matrices V and W. In that case, Lemma 4.2.5 will be used again.

Proof. Let $c \in C - \{0\}$ such that $wt(c) = d' \geq d$. Note that since C has length R, we have $R \geq d'$. Suppose $rank(Vdiag(c)W^T) = r$. We want to show that $r \geq d$. Note that, $rank(Vdiag(c)) \leq \min\{n, d'\}$. Since C_V is MDS, we have $rank(Vdiag(c)) = \min\{n, d'\}$ by Lemma 4.2.9. Similarly, since C_W is MDS, we have $rank(diag(c)W^T) = \min\{d', m\}$. By Frobenius inequality 1.1.18, we get

$$\begin{aligned} rank(Vdiag(c)W^T) &\geq rank(Vdiag(c)) + rank(diag(c)W^T) - rank(diag(c)) \\ &= \min\{n, d'\} + \min\{m, d'\} - d'. \end{aligned}$$

So, we have 4 cases to check.

- $r \geq n + m - d' \geq R + d - d' = (R - d') + d \geq d$.
- $r \geq n + d' - d' = n \geq d$ by definition.
- $r \geq d' + m - d' = m \geq d$ by definition.
- $r \geq d' + d' - d' = d' \geq d$.

See that $r \geq d$ in all of the cases. Thus, we are done. □

4.3 Construction of Extremal Triples using GRS Codes

In this subsection, construction of extremal triples using Generalized Reed Solomon codes will be explained. We will also present a GAP code which can give extremal triples for suitable parameters. Another nice thing is that it will actually be an MTR code, as well.

Definition 4.3.1. *For each $k \in \mathbb{N}$, let $\mathbb{F}_q[x,y]_{k-1}$ denote the set of homogeneous polynomials of degree $k-1$.*

For any homogenous polynomial $f(x,y) = \sum_{i=0}^{k-1} f_i x^i y^{k-1-i} \in \mathbb{F}_q[x,y]_{k-1}$, define the map f as

$$\mathbb{F}_q \cup \{\infty\} \longrightarrow \mathbb{F}_q \,:\, p \longmapsto f(p) = \begin{cases} f(p,1) & \text{if } \ p \in \mathbb{F}_q, \\ f(1,0) & \text{if } \ p = \infty. \end{cases}$$

Let $N \in \mathbb{N}$. For $P = (p_1, \ldots, p_N) \in (\mathbb{F}_q \cup \{\infty\})^N$, define the evaluation map

$$\begin{aligned} ev_P : \mathbb{F}_q[x,y]_{k-1} &\longrightarrow \mathbb{F}_q^N \\ f(x,y) &\longmapsto (f(p_1), \ldots, f(p_N)) \end{aligned}$$

Let $1 \leq k \leq N-1$ and $B = (b_1, \ldots, b_N) \in \mathbb{F}_q^N$. Also, we suppose $p_1, \ldots, p_N$ are pairwise distinct in $\mathbb{F}_q \cup \{\infty\}$. We can now define what a generalized Reed-Solomon(GRS) code is.

Definition 4.3.2. *The Generalized Reed-Solomon code $GRS(P,k,B)$ is the set*

$$GRS(P,k,B) = \{(b_1 f(p_1), \ldots, b_N f(p_N)) \ : \ f \in \mathbb{F}_q[x,y]_{k-1}\}.$$

By using the evaluation map we define above, we can also represent the GRS code by using componentwise multiplication, which we denote by $*$.

$$GRS(P,k,B) = \{B * ev_P(f) \ : \ f \in \mathbb{F}_q[x,y]_{k-1}\}.$$

Theorem 4.3.3. *Let $0 < d < k < R$ positive integers such that $R = k+d-1$. Let the vector $P = (p_1, \ldots, p_R) \in (\mathbb{F}_q \cup \{\infty\})^R$ such that $p_i's$ are pairwise distinct. Let $g(x,y)$ be an irreducible polynomial in $\mathbb{F}_q[x,y]_k$. Define $C = GRS(P,k,1)$. Let $V \in \mathbb{F}_q^{k \times R}$ be a parity-check matrix of $GRS(P, R-k, ev_P(g))$ and let $W \in \mathbb{F}_q^{d \times R}$ be a generator matrix of $GRS(P,d,1)$. Then, (C,V,W) is an extremal triple.*

Here we stated the big theorem of this subsection. However, we will prove it at the

end after giving a detailed example to see how it works. Now, we will give a detailed analysis of the example provided in [23].

Example 4.3.4. *Let $q=8$, $R=7$, $k=5$ and $d=R-k+1=3$. Let p be the generator of $\mathbb{F}_8^*$ and let $P=(1,p,\ldots,p^6)$. Consider $g(x)=x^5+x^2+1$ an irreducible polynomial in $\mathbb{F}_8[x]$. By Theorem 4.3.3, the code C is the $\mathbb{F}_q-[7,5,3]$ code $GRS(P,5,1)$. After considering $\mathbb{F}_8=\mathbb{F}_2[p]/(p^3+p+1)$, we can find the evaluation as*

$$ev_P(g)=(1,p,p^2,p^4,p^4,p^2,p).$$

Note that $R-k=2$. Then, we have

$$GRS(P,2,ev_P(g))=\{(f(1),pf(p),p^2f(p^2),p^4f(p^3),p^4f(p^4),p^2f(p^5),pf(p^6)) \ : \ f\in\mathbb{F}_q[x,y]_1\}.$$

Since f is of degree 1, f is either x or y. So, consider

$$f_1=x \text{ and } f_2=y.$$

For f_1, we get the vector $(1,p^2,p^4,1,p,1,1)$. For f_2, using the definition we have $f_2(P)=f_2(P,1)=1$. So, we get $(1,p,p^2,p^4,p^4,p^2,p)$. Then, the generator matrix for the code is the following

$$\begin{pmatrix} 1 & p & p^2 & p^4 & p^4 & p^2 & p \\ 1 & p^2 & p^4 & 1 & p & 1 & 1 \end{pmatrix}.$$

Note that, to apply the theorem, we need to find the parity check matrix of this code. To do so, we will transform the generator matrix into the standard form $\begin{pmatrix} I & | & M \end{pmatrix}$. Then, the parity check matrix V equals to $\begin{pmatrix} -M^T & | & I \end{pmatrix}$. We have

$$\begin{pmatrix} 1 & p & p^2 & p^4 & p^4 & p^2 & p \\ 1 & p^2 & p^4 & 1 & p & 1 & 1 \end{pmatrix} \longrightarrow \begin{pmatrix} 1 & 0 & p^3 & p & p^3 & p^5 & p^3 \\ 0 & 1 & p^4 & p & p^5 & p^2 & p^6 \end{pmatrix}.$$

Then, V becomes

$$V=\begin{pmatrix} p^3 & p^4 & 1 & 0 & 0 & 0 & 0 \\ p & p & 0 & 1 & 0 & 0 & 0 \\ p^3 & p^5 & 0 & 0 & 1 & 0 & 0 \\ p^5 & p^2 & 0 & 0 & 0 & 1 & 0 \\ p^3 & p^6 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

Now, we need to find the generator matrix W of $GRS(P,3,1)$. Note that

$$GRS(P,3,1)=\{(f(1),f(p),f(p^2),f(p^3),f(p^4),f(p^5),f(p^6)) \ : \ f\in\mathbb{F}_q[x,y]_2\}.$$

Since f is of degree 2, we could have $f_1 = x^2$, $f_2 = xy$ and $f_3 = y^2$. First of all, for f_3 we have $(1,1,1,1,1,1,1)$. For f_2 we have $(1,p,p^2,p^3,p^4,p^5,p^6)$. Similarly, for f_1 we get $(1,p^2,p^4,p^6,p,p^3,p^5)$. Thus, the generator matrix is

$$W = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & p & p^2 & p^3 & p^4 & p^5 & p^6 \\ 1 & p^2 & p^4 & p^6 & p & p^3 & p^5 \end{pmatrix}.$$

Thus, by the Theorem 4.3.3, we conclude that (C,V,W) is an extremal triple.

So, we get an extremal triple. That means, $\phi_{V,W}(C)$ is a tensor rank extremal code with $trk(\phi_{V,W}(C)) = 7$. Let us find a basis for this code. Note that

$$C = GRS(P,5,1) = \{(f(1), f(p), f(p^2), f(p^3), f(p^4), f(p^5), f(p^6)) \ : \ f \in \mathbb{F}_q[x,y]_4\}.$$

Thus, we need to consider

$$f_1 = x^4, \quad f_2 = x^3y, \quad f_3 = x^2y^2, \quad f_4 = xy^3, \quad f_5 = y^4.$$

We get the codewords $(1,p^4,p,p^5,p^2,p^6,p^3)$, $(1,p^3,p^6,p^2,p^5,p,p^4)$, $(1,p^2,p^4,p^6,p,p^3,p^5)$, $(1,p,p^2,p^3,p^4,p^5,p^6)$, and $(1,1,1,1,1,1,1)$. Let us denote them by c_1, c_2, c_3, c_4, c_5 respectively. Recall that we would like to find the basis elements of the $\mathbb{F}_8 - [5 \times 3, 5, 3]$ code $\phi_{V,W}(C)$ with tensor rank of it equals to 7. If the basis is $B = \{b_1, b_2, b_3, b_4, b_5\}$, then we can find those basis elements by

$$b_i = V diag(c_i) W^T \text{ for } 1 \leq i \leq 5.$$

After the computations, we get

$$B = \left\{ \begin{pmatrix} p^3 & p^2 & p^5 \\ p & p^6 & p^6 \\ p^3 & p^6 & p^4 \\ p^5 & 0 & 1 \\ p^3 & 1 & p^4 \end{pmatrix}, \begin{pmatrix} p^5 & p^3 & p^2 \\ 0 & p & p^6 \\ p^4 & p^3 & p^6 \\ p & p^5 & 0 \\ 1 & p^3 & 1 \end{pmatrix}, \begin{pmatrix} 0 & p^5 & p^3 \\ p^2 & 0 & p \\ 0 & p^4 & p^3 \\ p & p & p^5 \\ p^4 & 1 & p^3 \end{pmatrix}, \begin{pmatrix} 0 & 0 & p^5 \\ p^6 & p^2 & 0 \\ 0 & 0 & p^4 \\ p^3 & p & p \\ p^5 & p^4 & 1 \end{pmatrix}, \begin{pmatrix} p^2 & 0 & 0 \\ 1 & p^6 & p^2 \\ p^6 & 0 & 0 \\ p & p^3 & p \\ p^5 & p^5 & p^4 \end{pmatrix} \right\}.$$

We know that tensor rank equals to 7. Thus, for a generator tensor X of the code, we can represent it as

$$X = \sum_{r=1}^{7} u_r \otimes v_r \otimes w_r$$

where v_r and w_r are the $r-th$ columns of the matrices V and W. Remember that,

we can define $A_r = v_r \otimes w_r$ such that the set

$$A = \{A_1, A_2, A_3, A_4, A_5, A_6, A_7\}$$

is an R-basis for the code $\phi_{V,W}(C)$. Now, we will find that R-basis as well. Since in this case we know the exact tensor rank, we can say that

$$cs_1(X) = \langle A \rangle$$

by the equivalence given Theorem 3.2.5. Applying $A_r = v_r \otimes w_r$, we get

$$\langle A \rangle = \left\{ \begin{pmatrix} p^3 & p^3 & p^3 \\ p & p & p \\ p^3 & p^3 & p^3 \\ p^5 & p^5 & p^5 \\ p^3 & p^3 & p^3 \end{pmatrix}, \begin{pmatrix} p^4 & p^5 & p^6 \\ p & p^2 & p^3 \\ p^5 & p^6 & 1 \\ p^2 & p^3 & p^4 \\ p^6 & 1 & p \end{pmatrix}, \begin{pmatrix} 1 & p^2 & p^4 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 0 \\ 1 & p^3 & p^6 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & p^4 & p \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & p^5 & p^3 \\ 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & p^6 & p^5 \end{pmatrix} \right\}.$$

Remark 4.3.5. *Let us summarize here the importance of the above example. We found that (C,V,W) is an extremal triple. That means $\phi_{V,W}(C)$ is an tensor rank extremal code. So, we get a solution to Question 4.2.1. Note that $\phi_{V,W}(C)$ is an $\mathbb{F}_8 - [5 \times 3, 5, 3]$ code with tensor rank equals to 7. Observe that $7 = 5 + 3 - 1$. That means, $\phi_{V,W}(C)$ is an MTR code, as well. By Corollary 4.1.9, we have $\phi^{-1}_{V,W}(\phi_{V,W}(C)) = C$ is an MDS code. Therefore, we get an answer for Question 4.2.3, too. Additionally, this is another way to see that the GRS codes are MDS when they are of the form $GRS(P,k,1)$. In general, $GRS(P,k,B)$ is MDS if $B \in (\mathbb{F}_q^*)^N$.*

We present an GAP algorithm to find MTR codes given the condition that $0 < d < k < R$ such that $R = k + d - 1$. The reason we are not given the pseudo code is that it is basically given in the statement of Theorem 4.3.3. The algorithm will take R and k as inputs. Then by using $R = k + d - 1$, we will know d. Similarly, we will also take q as an input to clarify the field we are working on. Lastly, an irreducible polynomial on $GF(q)$ will be taken as an input. Note that it is beneficial to add an irreducibility check function inside the code just to be sure. We will consider x as our indeterminate over the field. We used the GUAVA package [3] to write the following algorithm.

Algorithm 1: MTR(R,q,k,g)

Result: This algorithm will create an $\mathbb{F}_q - [k \times d, k, d]$ MTR code.

```
K := GF(q), z := Z(q), F := PolynomialRing(K,"x");
P := List([0..(R-1)],i → z^i);
eval := List(P,i → Value(g,i));
B := List([1..R], i → 1);
C := GeneralizedReedSolomonCode(P,k,F,B);
V := CheckMat(GeneralizedReedSolomonCode(P,R-k,F,eval));
W := GeneratorMat(GeneralizedReedSolomonCode(P,R-k+1,F,B));
Phi := [ ];
for c in C do
    Add(Phi, V*DiagonalMat(c)*TransposedMat(W));
end
return Phi
```

It is of very good use to create a record and return it instead of returning the elements of the code since we would like to see the the general characteristics of the code. However, we leave it like this so that it can be called inside other functions. We will get Example 4.3.4 if we use

$$MTR(7, 8, 5, x^5 + x^2 + 1).$$

Remark 4.3.6. *Note that in GAP, $Z(q)$ returns a generator of $\mathbb{F}_q^*$. The parameters P, eval, B are exactly the same as they defined in this subsection.*

Last but not least, we are going to prove Theorem 4.3.3 to end this section.

Proof of Theorem 4.3.3. We will use Theorem 4.2.8 and show that

$$\forall c \in C - \{0\}, \quad dim(C_{W_c} \cap C_V^{\perp}) \leq dim(C_{W_c}) - d.$$

W is a generator matrix of $GRS(P, d, 1)$. So, we have

$$C_W = GRS(P, d, 1) \text{ and } C_{W_c} = GRS(P, d, c).$$

Thus, $dim(C_{W_c}) = d$. So, we reduced our goal to showing $dim(C_{W_c} \cap C_V^{\perp}) = 0$. That means, we want to show that

$$C_{W_c} \cap C_V^{\perp} = \{0\}.$$

Recall that $C = GRS(P,k,1) = \{1 * ev_P(g) \ \mid \ g \in \mathbb{F}_q[x,y]_{k-1}\}$. Thus, any nonzero codeword c is of the form $c = ev_P(g)$ for some nonzero $g(x,y) \in \mathbb{F}_q[x,y]_{k-1}$. Thus, we have $C_{W_c} = GRS(P,d,ev_P(g))$. Now, let $b \in C_V^{\perp} \cap C_{W_c}$. We want to show that $b = 0$. Since V is a parity check matrix of $GRS(P,R-k,ev_P(f))$, this means

$$C_V^{\perp} = GRS(P,R-k,ev_P(f)).$$

Note that we have

$$C_{W_c} = \{ev_P(g) * ev_P(h) \ \mid \ h \in \mathbb{F}_q[x,y]_{d-1}\},$$

$$C_V^{\perp} = \{ev_P(f) * ev_P(s) \ \mid \ s \in \mathbb{F}_q[x,y]_{R-k-1}\}.$$

Since b lies on both of them, we get the equation

$$b = ev_P(f) * ev_P(s) = ev_P(g) * ev_P(h).$$

This means, $b_i = f(p_i)s(p_i) = g(p_i)h(p_i)$ for $i \in [R]$. Keep in mind that $deg(f) = k$, $deg(g) = k-1$, $deg(h) = d-1$, and $deg(s) = R-k-1$. Then,

$$deg(fs) = R-1 < R \text{ and } deg(gh) = R-2 < R.$$

So, we have two polynomials of degree strictly less than R and they agree on R inputs. In one variable polynomails this means they are equal. However, here we have polynomials in $\mathbb{F}_q[x,y]$. Remember our way of evaluation

$$\mathbb{F}_q \cup \{\infty\} \longrightarrow \mathbb{F}_q \, : \, p \longmapsto f(p) = \begin{cases} f(p,1) & \text{if } \ p \in \mathbb{F}_q, \\ f(1,0) & \text{if } \ p = \infty. \end{cases}$$

So, we are actually computing in one variable, and thus $fs = gh$. Since $\mathbb{F}_q[x,y]$ is a unique factorization domain, f is being irreducible implies that f is prime. That is,

$$f|g \text{ or } f|h.$$

Note that $deg(g) = k-1 < k = deg(f)$ and $deg(h) = d < k = deg(f)$ because of the assumption $d < k$. That means, $h = 0$ since g is nonzero by assumption. Then, we get $b = 0$, as desired. $\square$

5. CONNECTION TO COMPLEXITY THEORY

In this section, we will mention some complexity results and look into the theory of tensors more closely. Throughout this chapter, we will follow the book [2].

5.1 Basics of Complexity Theory

Definition 5.1.1. *Algebraic complexity theory is the study of understanding the required computational power to solve algorithmic problems using algebraic tools and models.*

The aim is to find the best possible model and to prove its optimality. Let us give a simple example to understand this concept.

Example 5.1.2. *Suppose we are only allowed to do multiplication. Consider a ring R, a positive natural number n. We want to compute r^n given any $r \in R$.*

The input r can be modeled by an indeterminate X since it is not known. We transform the question into computing X^n given X. Some notation is required here. We say that X is the 0th intermediate result, X^2 is the 1st intermediate result, and so on. We understand the computation is over when X^n is one of the intermediate results. The model we are going to use for this example is "Multiplication Chain of Length r for n". Consider a sequence $(a_0 = X, \ldots, a_r = X^n)$ such that $a_k = a_i \times a_j$ for some $0 \leq i, j \leq k$. Define

$$u(n) = \text{minimum number of multiplications needed to compute } X^n.$$

Then, we have $u(n) =$ shortest length of an multiplication chain for n. An obvious upper bound is $u(n) \leq n-1$ since we can just consider the sequence $(X, X^2, \ldots, X^n)$.

However, it can be done much faster. Consider

$$X^{13} = X \cdot (((X \cdot X^2)^2)^2).$$

Definition 5.1.3. *$w_2(n)$ is the Hamming weight of the binary expansion of n.*

Suppose n is odd. Then, $n = 2m+1$ for some integer m. To compute X^n, we first compute X^{n-1} and then multiply it with X. So, $u(2m+1) \leq u(2m)+1$. Similarly, if n is even, then $n = 2m$. We first compute X^m and then take its square. So, $u(2m) \leq u(m)+1$.

Lemma 5.1.4. *$w_2(2m+1) = w_2(2m)+1$ and $w_2(2m) = w_2(m)$.*

Proof. Let $m = (---A--)_2$. Then, we have $2m = (---A---0)_2$, and $2m+1 = (---A---1)_2$ where $()_2$ represents the binary number system. It can be easily seen that the result follows. □

Example 5.1.5. *$10 = (1010)_2$, $20 = (10100)_2$ and $21 = (10101)_2$.*

Theorem 5.1.6. *$u(n) \leq \lfloor log(n) \rfloor + w_2(n) - 1$.*

Proof. We will induct on n. Let $n = 2$. $u(2) = 1 \leq 1+1-1 = \lfloor log(2) \rfloor + w_2(2) - 1$. Assume $u(2m) \leq \lfloor log(2m) \rfloor + w_2(2m) - 1$. Then, $u(2m)+1 \leq \lfloor log(2m) \rfloor + w_2(2m)$. We want to show that

$$u(2m+1) \leq \lfloor log(2m+1) \rfloor + w_2(2m+1) - 1 = \lfloor log(2m+1) \rfloor + w_2(2m),$$

where the last equality follows from the above lemma. As we explained above $u(2m+1) \leq u(2m)+1$. Using this, we conclude the proof as

$$u(2m+1) < u(2m)+1 \leq \lfloor log(2m) \rfloor + w_2(2m) \leq \lfloor log(2m+1) \rfloor + w_2(2m).$$

□

Example 5.1.7. *Let $n = 100$. Then, $100 = (1100100)_2$ and thus $w_2(100) = 3$. Also, note that $\lfloor log(100) \rfloor = 6$. So, $u(100) \leq 8$. The trivial upper bound we mentioned before gives $u(100) \leq 100 - 1 = 99$. Thus, this is much better.*

Now, let us try to put a lower bound. Observe that the 0th intermediate result have degree 1. Maximum of the degrees obtained can be at most doubled in each step. That is, h-th intermediate step can have degree at most 2^h. So, if we want

to compute X^n from X, then n is at most 2^h. So, $\lceil log(n) \rceil \leq u(n)$. We combine our bounds in the following theorem.

Theorem 5.1.8. $\lceil log(n) \rceil \leq u(n) \leq \lfloor log(n) \rfloor + w_2(n) - 1$.

Putting nontrivial bounds is one of the main study areas of algebraic complexity theory. The example above shows such a process. Now, suppose allowing division as well as multiplication between intermediate results. Let $l(n)$ denotes minimum number of operations. Clearly $l(n) \leq u(n)$. Note that, we also have $\lceil log(n) \rceil \leq l(n)$.

Example 5.1.9. *Show that* $l(31) = 6 < 7 = u(31)$.

We will start with $u(31)$. Recall that we are only allowed to do multiplication using intermediate results. $X \to X^2 \to X^4 \to X^8 \to X^{16} \to X^{24} \to X^{28} \to X^{30} \to X^{31}$. This process takes 8 steps. We need a better approach. Consider the following two steps first : $X \to X^2$ and $X \cdot X^2 \to X^3$. So, we are left with 5 more steps. Consider $X^3 \to X^6 \to X^{12} \to X^{24} \to X^{30} \to X^{31}$, as desired. So, $u(31) \leq 7$. Note that $\lceil log(n) \rceil \leq u(n),\ l(n)$. So, $5 \leq l(31)$ and $5 \leq u(31) \leq 7$. Let us try to calculate $l(31)$. Consider $X \to X^2 \to X^4 \to X^8 \to X^{16} \to X^{32} \to X^{31}$ where in the last step we use division as it is allowed for $l(n)$. Thus, $5 \leq l(n) \leq 6$. If we show that $l(31) \neq 5$ then it will imply that $u(31) \neq 5$. Thus, we are left to show

- $l(31) \neq 5$
- $u(31) \neq 6$

$X \to X^2$ is a forced move. Then, even we square in every step, in the fifth move we will arrive at X^{32}. So, $l(31) \neq 5$. Since we can not divide when calculating $u(n)$, the move $X^{32} \to X^{31}$ is not possible. Using X^3 requires 7 step as we explained above. Therefore, $u(31) = 6$ is not possible, as desired.

Remark 5.1.10. *In the previous example, we see that computing X^n from X only makes sense when the algebraic operations admitted for an algorithmic solution and the cost of each operation have been agreed. The complexity of a problem depends on these agreements and in particular on the selected cost function.*

The following is known as the Scholz-Brauer conjecture.

Open Problem 5.1.11. *Prove or disprove that given a positive integer n, we have $u(2^n - 1) \leq n - 1 + u(n)$.*

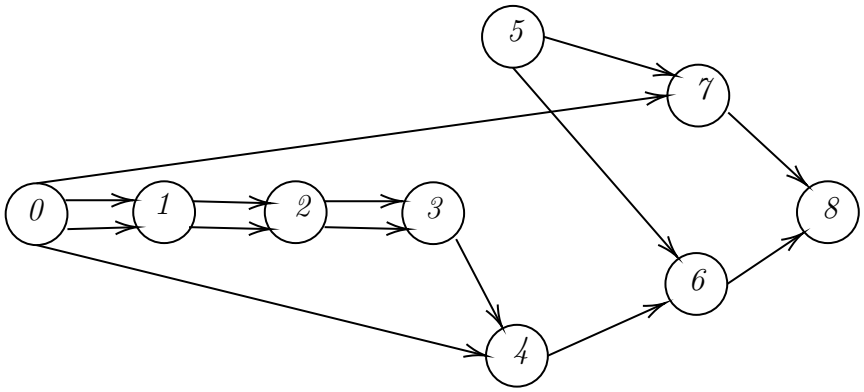
Now, we need to explain a little bit about the straight-line programming to build up our argument. We will do this by an example. Let $\lambda \in \mathbb{F}$. We denote λ^c as the operation of taking the constant λ. Denote the entity of all these operations by $\mathbb{F}^c$.

Here, c denotes the cost function

$$c : \sigma \to \mathbb{N}$$

where $\sigma = \mathbb{F} \cup \mathbb{F}^c \cup \{\cdot, /, +, -\}$. Note that $c(\lambda^c) = 0$, i.e., we can use constants freely.

Example 5.1.12. *Suppose we want to compute $\frac{X^9-1}{X-1}$. We first compute X^9 as follows. Let $X_0 := X$. Then, define $X_1 := X_0 \cdot X_0$, $X_2 := X_1 \cdot X_1$, $X_3 := X_2 \cdot X_2$, and $X_4 := X_0 \cdot X_3$. Now, we need to substract 1. This can be done by using two steps. Let $X_5 := 1$. Then, we can define $X_6 := X_4 - X_5$ and $X_7 := X_0 - X_5$. Now, we can divide and get $X_8 := \frac{X_6}{X_7}$. We can create a straight-line programming $S = (S_1, \ldots, S_8)$ such that each S_i gives us an instruction as follows: $S_1 = (\cdot; 0, 0)$, $S_2 = (\cdot; 1, 1)$, $S_3 = (\cdot; 2, 2)$, $S_4 = (\cdot; 0, 3)$, $S_5 = (1^c)$, $S_6 = (-; 4, 5)$, $S_7 = (-; 0, 5)$ and $S_8 = (/; 6, 7)$. Any straight-line programming can be represented by a directed acyclic multigraph. For this example, we have*



The length of the longest directed path in this graph is called the depth D_S of the straight-line program S. In this example, we have $D_S = 6$.

Definition 5.1.13. *Let o_i denote the operation symbol used in S_i. Then, define the c-length of the program S as*

$$cl(S) = \sum_i c(o_i).$$

Now, we can precisely define what does complexity mean with the help of the following definition of an algebra. We will also talk about algebras in Section 6, specifically semifields, and show a very beautiful connection with MRD codes.

Definition 5.1.14. *An algebra is a vector space U over a field $\mathbb{F}$ with an additional binary operation $\cdot$ called multiplication such that given $x, y, z \in U$ and $a, b \in \mathbb{F}$, the following holds*

$$(x+y) \cdot z = x \cdot z + y \cdot z, \quad z \cdot (x+y) = z \cdot x + z \cdot y, \quad (ax) \cdot (by) = (ab)(x \cdot y)$$

Note 5.1.15. *The 3 properties in the above definition can also be seen as the existence of an $\mathbb{F}-$bilinear map $f : U \times U \to U$ which satisfies $f(ax, by) = (ab)f(x, y)$, $f(x+y, z) = f(x, z) + f(y, z)$, and $f(z, x+y) = f(z, x) + f(z, y)$.*

Definition 5.1.16. *Let A be an $\mathbb{F}$-algebra. The complexity $L_A^c(G|I) \in \mathbb{N} \cup \{\infty\}$ of the finite subset $G \subseteq A$ with respect to the input set $I \subseteq A$ is the minimal c-length of a straight line programming which computes G. If what we mean by $L_A^c(G|I)$ is clear from the context, then we just denote it as $L_A(G)$, or even simpler $L(G)$.*

Remark 5.1.17. *In the above example, A is $\mathbb{F}[X]$, $G = \frac{X^9-1}{X-1}$ and $I = \{X\}$. Assuming that all the operations have cost 1, we get $L(G) = 7$ since $c(1^c) = 0$.*

We will finish with a very useful theorem.

Theorem 5.1.18. *Let $|\mathbb{F}| \geq n+1$. Then the product of two polynomials a and b in an $\mathbb{F}$-algebra $A[X]$ with $n = deg(ab)$ can be computed with $n+1$ nonscalar operations. This is true if $|\mathbb{F}| = n$ provided that $deg(ab) = deg(a) + deg(b)$.*

Proof. Let a and b have degree α and β. Since $|\mathbb{F}| \geq n+1$, take $n+1$ distinct elements $p_0, \ldots, p_n$ in $\mathbb{F}$. Assuming $c = ab$, for all $0 \leq r \leq n$, we have

$$\sum_{k=0}^{n} c_k p_r^k = (\sum_{i=0}^{\alpha} a_i p_r^i)(\sum_{j=0}^{\beta} b_j p_r^j) = u_r \cdot v_r = g_r$$

where u_r is the evaluation of polynomial a at the point p_r, and v_r is the evaluation of b at the point p_r. As both u_r and v_r are linear combinations of the input coefficients, the computation of u_r and v_r are free of charge in the nonscalar model, i.e., in a model such that addition, substraction and scalar multiplication has no cost. Thus, each g_r can be computed with cost 1, which is coming from the multiplication $u_r \cdot v_r$. Hence, $g_0, \ldots, g_n$ can be computed with $n+1$ operations. Now, note that

$$(g_0 \ldots g_n)^T = (p_r^k) \cdot (c_0 \ldots c_n)^T.$$

Since p_r's are pairwise distinct, the Vandermonde matrix (p_r^k) of size $(n+1) \times (n+1)$ is invertable. Hence, each of c_k is a linear combination of $g_0, \ldots, g_n$, and thus can be computed with no additional nonscalar cost. □

Lastly, note that we can also represent Theorem 5.1.18 by

$$L_F(Cf(ab) \mid Cf(a) \cup Cf(b)) = n+1$$

where $Cf(a)$ is the set of coefficients of the polynomial a.

5.2 Encoding Complexity of Rank-Metric Codes

Now, we have the necessary background to talk about the encoding complexity of rank-metric codes. Consider an $\mathbb{F}_q - [n \times m, k, d]$ code C. Let $X = \sum_{r=1}^{R} u_r \otimes v_r \otimes w_r$ be its generating tensor. Given a vector $a \in \mathbb{F}_q^k$, we encode it as

$$m_1(a, X) = \sum_{r=1}^{R} (a \cdot u_r) v_r \otimes w_r.$$

In Remark 4.1.4, it is shown that $m_1(a, X) = V diag(aU) W^T$. Recall that a matrix M is said to be in standard form if $M = [I|M']$ for some other matrix M' of suitable size. Similarly, we can define the concept of standard form for generator tensors.

Definition 5.2.1. *Consider X as in Definition 3.0.7. We say that X is in standard form if the matrices U, V, and W are all in standard form where u_r,v_r and w_r are columns of $U \in \mathbb{F}^{k \times R}$, $V \in \mathbb{F}^{n \times R}$, and $W \in \mathbb{F}^{m \times R}$ respectively.*

Thus, X has storage complexity $kR + nR + mR = R(k + n + m)$ for the general case and $n(R-n) + m(R-m) + k(R-k) = R(k+n+m) - (n^2 + m^2 + k^2)$ if it is in standard form. We noted before that any element of the code is of the form $m_1(a, X)$ for some vector $a \in \mathbb{F}_q^k$. So, to represent a codeword, we just need to compute aU since we already know V and W. See that computing aU requires k multiplication for each column and $(k-1)$ addition per column, and thus in total of kR multiplications and $(k-1)R$ additions. So, the encoding complexity becomes $2kR - R$. If U is in standard form, then the encoding complexity becomes $k(R-k) + (k-1)(R-k) = 2kR - R - (k^2 - k)$. Secondly, we can use generator matrix to handle the encoding. Since elements of C are matrices of size $n \times m$, we first see them as vectors of length nm as we did before. Then, the generator matrix G is of size $k \times nm$ and we encode as $a \to aG$. Clearly, the storage complexity is knm for the general case and $k(nm-k)$ if G is in standard form. Now, consider evaluating the multiplication aG. We need to do k multiplication for each column and $(k-1)$ additions per column to decide on the final entry. So, the encoding complexity is $knm + (k-1)nm = 2knm - nm$. If G is in standard form, then the encoding complexity becomes $k(nm-k) + (k-1)(nm-k) = 2knm - nm - (2k^2 - k)$. We summarize what we wrote above in the following table. In the following table, GM is short for generator matrix and GMS is short for generator matrix in standard form. Similarly, GT and GTS is used. Since $R \leq nm$, it might seem that using generator tensors will always give lower complexity. Although this is true most of the time, it is not always true as we will show in the next example.

Table 5.1 Complexity

	GM	GMS	GT	GTS
STORAGE	knm	$k(nm-k)$	$R(k+n+m)$	$R(k+n+m)-(n^2+m^2+k^2)$
ENCODING	$2knm-nm$	$2knm-nm-(2k^2-k)$	$2kR-R$	$2kR-R-(k^2-k)$

Example 5.2.2. *$X = e_1\otimes e_1\otimes e_1 + e_1\otimes e_2\otimes e_2 + e_2\otimes e_3\otimes e_1 + e_2\otimes e_4\otimes e_2 + e_3\otimes e_1\otimes e_3 + e_3\otimes e_2\otimes e_4 + e_4\otimes e_3\otimes e_3 + e_4\otimes e_4\otimes e_4 \in \mathbb{F}^4\otimes F^4\otimes F^4$. In Theorem 5.4.5, we will prove that $trk(X) = 7$. Thus, we have $R = 7$ and $k = n = m = 4$. Assuming X as the generator tensor, the first contraction space is the code C by definition. Thus, we can also form the generator matrix as explained above.*

$$G = \begin{pmatrix} 1&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&1&0&0&0&0&1&0&0\\ 0&0&1&0&0&0&0&1&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&1 \end{pmatrix}.$$

Using the proof of Theorem 5.4.5, we can form the matrices U, V and W.

$$U=\begin{pmatrix} 1&0&1&0&1&-1&0\\ 0&1&0&0&0&1&0\\ 0&0&0&0&1&0&1\\ 1&1&0&1&0&0&-1 \end{pmatrix},\ V=\begin{pmatrix} 1&1&0&-1&0&1&0\\ 0&0&0&1&0&0&1\\ 0&0&1&0&0&1&0\\ 1&0&-1&0&1&0&1 \end{pmatrix}, W=\begin{pmatrix} 1&0&0&1&-1&0&1\\ 0&1&0&0&0&0&0\\ 0&0&1&1&1&0&0\\ 1&-1&1&0&0&1&0 \end{pmatrix}.$$

Calculations explained before give us lower complexity when encoding via G.

Remark 5.2.3. *In practice, we need to find a way to choose between these two approaches. Suppose that X and G are in standard form. Just a small calculation shows that using generator tensors is preferable if and only if $R < \frac{knm+n^2+m^2}{k+n+m}$. This inequality comes from comparing the storage complexities found in the Table 5.1 as $R(k+n+m)-(n^2+m^2+k^2) < k(nm-k)$. If we apply the same idea for the encoding complexities, it can be seen that we need $R < \frac{2knm-nm-k^2}{2k-1}$.*
Thus, we can guarantee that complexity of the generator tensor in standard form is lower if $R < \min\{\frac{knm+n^2+m^2}{k+n+m}, \frac{2knm-nm-k^2}{2k-1}\}$. Similar calculation gives a formula when they are not in standard form as $R < \min\{\frac{knm}{k+n+m}, \frac{2knm-nm}{2k-1}\}$. By combining these 2 formulas, checking the following will guarantee lower complexity if the generator tensor approach is used:

$$R < \min\{\frac{knm}{k+n+m}, \frac{2knm-nm-k^2}{2k-1}\}.$$

See that in the above example 7 is not less than $\min\{\frac{64}{12}, \frac{96}{7}\}$, comfirming the result.

5.3 Tensorial Notation for Complexity

In this subsection, we will give a tensorial approach to complexity theory and see some more correspondences with our main topic. In Definition 3.0.1, we defined pure tensors and noted that a general element T in our tensor space $\mathcal{T}$ can be written as sums of pure tensors. Thus, T is of the form

$$T = \sum_{i_1,\ldots,i_m} t_{i_1\ldots i_m}(e_{1i_1} \otimes \ldots \otimes e_{mi_m}).$$

Example 5.3.1. *Consider $m = 2$. Then, $T \in U \otimes V$ for vector spaces U and V. Let $\{u_i\}$ and $\{v_j\}$ be bases for U and V respectively. Then, a general element of $U \otimes V$ is of the form $T = \sum_{i,j} t_{ij}(u_i \otimes v_j)$. Consider the following tensor:*

$$T = 2u_1 \otimes v_2 - u_2 \otimes v_1 + u_2 \otimes v_2 + 3u_3 \otimes v_3.$$

Then, $t_{12} = 2$, $t_{21} = -1$, $t_{22} = 1$, and $t_{33} = 3$. We can see this as a matrix of size 3×3 as

$$A_T = \begin{pmatrix} 0 & 2 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 3 \end{pmatrix}.$$

Naturally, we can consider a linear map $L_T : U \to V$ such that

$$L_T(u) = L_T(\sum_i x_i u_i) = \sum_i x_i L_T(u_i) = \sum_i x_i \sum_j t_{ij} v_j.$$

In our example, we have

$$L_T(u) = L_T(a\ b\ c) = (a\ b\ c)A_T = (-b\ \ 2a+b\ \ 3c).$$

The following note is clear from the previous example.

Note 5.3.2. *Linear maps corresponds to 2-fold tensors.*

We will provide an alternative way to define the linear map corresponding to the tensor T. We will use the dual vector space.

Definition 5.3.3. *The dual of the vector space U is the set of all linear functionals on U. That is,*

$$U^D = \{u^D : U \to \mathbb{F}\}.$$

Choose a basis $\{u_i^D\}$. Then we have

$$u_i^D(u_j) = \begin{cases} 1 & \text{, if } i = j \\ 0 & \text{, if } i \neq j \end{cases}$$

Now define $L_T : U^D \to V$ such that $L_T(u_k^D) = \sum_j t_{kj} v_j$. Using the definition above, we have $t_{kj} = \sum_i t_{ij} u_k^D(u_i)$. Thus, we get

$$L_T(u_k^D) = \sum_{i,j} t_{ij} u_k^D(u_i) v_j.$$

The following theorem also shows the correspondence between 2-fold tensors and homomorphisms from U^D to V. In the following theorem, choose $A = U^D$ and $B = V$. Then, we have $U \otimes V \cong Hom(U^D, V)$, as desired.

Theorem 5.3.4. *Given two vector spaces A and B, we have $A^D \otimes B \cong Hom(A, B)$.*

Proof. Let $u^D \in U^D$ and $a \in U$. Define $f : U^D \otimes V \longrightarrow Hom(U, V)$ with the rule $f(u^D \otimes v)(a) = u^D(a)v$. Suppose $\{e_i\}$ is a basis of U and $\{e_i^D\}$ is a basis for U^D. Define

$$g : Hom(U, V) \longrightarrow U^D \otimes V \; : \; H \to \sum_i e_i^D \otimes H(e_i)$$

where $H : U \to V$. We claim that f and g are inverses. Consider

$$f(g(H))(a) = \sum_i e_i^D(a) H(e_i) = H(\sum_i e_i^D(a) e_i) = H(a).$$

Similarly,

$$g(f(u^D \otimes V)) = \sum_i e_i^D \otimes f(u^D \otimes v) e_i = \sum_i e_i^D \otimes u^D(e_i) v = \sum_i u^D(e_i) e_i^D \otimes v = u^D \otimes v.$$

The linearity of f and g can be easily checked. So, we have two linear maps that are inverses of each other. That means, we have the desired isomorphism. □

Let us apply this new approach to the same example.

$$T = 2u_1 \otimes v_2 - u_2 \otimes v_1 + u_2 \otimes v_2 + 3u_3 \otimes v_3.$$

Suppose $f : U \otimes V \to Hom(U^D, V)$ is the isomorphism. We denote the image of $T \in U \otimes V$ under f by T^f. So, we have

$$\begin{aligned} T^f(u_1^D) &= (2u_1 \otimes v_2)^f(u_1^D) - (u_2 \otimes v_1)^f(u_1^D) + (u_2 \otimes v_2)^f(u_1^D) + (3u_3 \otimes v_3)^f(u_1^D) \\ &= (u_1^D)(2u_1)v_2 - (u_1^D)(u_2)v_1 + (u_1^D)(u_2)v_2 + (u_1^D)(3u_3)v_3 \\ &= 2v_2 = \sum_j t_{1j}v_j. \end{aligned}$$

So, $t_{12} = 2$.

$$\begin{aligned} T^f(u_2^D) &= (2u_1 \otimes v_2)^f(u_2^D) - (u_2 \otimes v_1)^f(u_2^D) + (u_2 \otimes v_2)^f(u_2^D) + (3u_3 \otimes v_3)^f(u_2^D) \\ &= (u_2^D)(2u_1)v_2 - (u_2^D)(u_2)v_1 + (u_2^D)(u_2)v_2 + (u_2^D)(3u_3)v_3 \\ &= -v_1 + v_2 = \sum_j t_{2j}v_j. \end{aligned}$$

So, $t_{21} = -1$ and $t_{22} = 1$.

$$\begin{aligned} T^f(u_3^D) &= (2u_1 \otimes v_2)^f(u_3^D) - (u_2 \otimes v_1)^f(u_3^D) + (u_2 \otimes v_2)^f(u_3^D) + (3u_3 \otimes v_3)^f(u_3^D) \\ &= (u_3^D)(2u_1)v_2 - (u_3^D)(u_2)v_1 + (u_3^D)(u_2)v_2 + (u_3^D)(3u_3)v_3 \\ &= 3v_3 = \sum_j t_{3j}v_j. \end{aligned}$$

Lastly, we have $t_{33} = 3$, as we found before.

Remark 5.3.5. *We started by showing the relation between 2-fold tensors and matrices. Using that, we found an alternative way to represent the space of 2-fold tensors by $Hom(U^D,V)$. Now, we will extend it to create a relation for 3-fold tensors.*

$$U \otimes V \cong Hom(U^D,V)$$

$$\downarrow \text{ extend}$$

$$U \otimes V \otimes W \cong Bil(U^D \times V^D,W).$$

Then, we will have $(u \otimes v \otimes w)^f(u^D,v^D) = u^D(u)v^D(v)w$ supposing that the isomorphism is given by f. So, a pure tensor $T = u \otimes v \otimes w$ defines a bilinear map

$$T : U^D \times V^D \to W : (u^D,v^D) \longmapsto u^D(u)v^D(v)w.$$

Thus, we get a new correspondence, that is, bilinear maps corresponds to 3-fold tensors. Observing that $(v_1,\ldots,v_m) \longmapsto v_1 \otimes \ldots \otimes v_m$ is multilinear, we can conclude that multilinear maps correnspponds to m-fold tensors.

Example 5.3.6. *Consider an algebra A. Given our Definition 5.1.14, A is a vector space U with a multiplication $\phi : U \times U \to U$. A corresponds to a 3-fold tensor T_A in $U^D \otimes U^D \otimes U$ such that $T_A^f = \phi$. Let us give a precise correspondence to understand what is going on here. If $T_A = \sum_{i=1}^r u_i^D \otimes v_i^D \otimes w_i$, then $\phi(u,v) = \sum_{i=1}^r u_i^D(u) v_i^D(v) w_i$.*

What is so special about T_A is that the tensor rank of the algebra A corresponds to the rank of T_A. Recall that tensor rank of $T \in \mathcal{T}$ is the minimum positive integer R such that there exists a decomposition of T into R pure tensors. Similarly, rank of a subspace U of $\mathcal{T}$ is the minimal number of pure tensors needed to span a subspace containing U. Thus, tensor rank of an algebra gives a measure of complexity of multiplication. We can also see this by following a very similar approach as follows. We call a map $\phi : V \to W$ quadratic if there exists bases $\{v_i\}_{i \leq n}$ and $\{w_j\}_{j \leq p}$ of V and W respectively, and quadratic forms $g_1, \ldots, g_p \in \mathbb{F}[x_1, \ldots, x_n]$ such that

$$\phi(\sum_{i=1}^{n} b_i v_i) = \sum_{j=1}^{p} g_j(b) w_j \text{ for all } b = (b_1, \ldots, b_n) \in \mathbb{F}^n.$$

Definition 5.3.7. *The quadratic forms $g_1, \ldots, g_p$ are called coordinate functions of ϕ with respect to bases $\{v_i\}$ and $\{w_j\}$.*

Let $u_1^D, \ldots, u_r^D, v_1^D, \ldots, v_r^D \in V^D$ and $w_1, \ldots, w_r \in W$ such that for all $v \in V$ we have

$$\phi(v) = \sum_{i=1}^{r} u_i^D(v) v_i^D(v) w_i.$$

Then, $(u_1^D, v_1^D, w_1; \ldots; u_r^D, v_r^D, w_r)$ is called a quadratic computation algorithm for ϕ of length r.

Definition 5.3.8. *We have $L(\phi) = L_{\mathbb{F}[X_1,\ldots,X_n]}(\{g_1, \ldots, g_p\})$. This is called the multiplicative complexity of ϕ. It is equivalent to the shortest quadratic computation algorithm for ϕ.*

Similarly, we can define the bilinear complexity by making the quadratic map into a bilinear map. That is, we have $\phi : U \times V \to W$ such that

$$\phi(u,v) = \sum_{i=1}^{r} u_i^D(v) v_i^D(v) w_i$$

where $u_i^D \in U^D$, $v_i^D \in V^D$ and $w_i \in W$.

Definition 5.3.9. *$(u_1^D, v_1^D, w_1; \ldots; u_r^D, v_r^D, w_r)$ is called a bilinear computation for ϕ of length r. The length of a shortest bilinear computation for ϕ is called the bilinear complexity, or the most common name for it, the rank of ϕ. It is denoted by $Rk(\phi)$.*

So, we finally arrived at the concept of rank. The big theorem combining the relation between multiplicative complexity and the rank is the following.

Theorem 5.3.10. $L(\phi) \leq Rk(\phi) \leq 2L(\phi)$.

Proof. By definition, $L(\phi) \leq Rk(\phi)$. To see the other inequality, consider the bilinear map $\phi : U \times V \longrightarrow W$. Denote $L(\phi) = L$. Let $a_i, b_i \in (U \times V)^D$ and $w_i \in W$. So, given $(u, v) \in U \times V$, using the definition of bilinearity, we have the following

$$\begin{aligned}\phi(u,v) &= \sum_{i=1}^{L} a_i(u,v)b_i(u,v)w_i \\ &= \sum_{i=1}^{L} (a_i(u,0) + a_i(0,v))(b_i(u,0) + b_i(0,v))w_i \\ &= \sum_{i=1}^{L} a_i(u,0)b_i(0,v)w_i + \sum_{i=1}^{L} b_i(u,0)a_i(0,v))w_i.\end{aligned}$$

where in the last equality we used the fact that $\sum_{i=1}^{L} a_i(u,0)b_i(u,0)w_i = 0$ and $\sum_{i=1}^{L} a_i(0,v)b_i(0,v))w_i = 0$. So, we get the upper bound $2L$, as desired.

□

The following example shows one of the advantages of using rank instead of the multiplicative complexity.

Example 5.3.11 (Remark 14.22 in [2])**.** *Consider a permutation $\pi \in S_3$. It induces an isomorphism*

$$U_1 \otimes U_2 \otimes U_3 \to U_{\pi^{-1}(1)} \otimes U_{\pi^{-1}(2)} \otimes U_{\pi^{-1}(3)}$$

that send tensor T to πT. $Rk(T) = Rk(\pi T)$, however $L(\pi) \neq L(\pi T)$.

Given bilinear maps $\phi : U \times V \to W$ and $\phi' : U' \times V' \to W'$, consider

$$\phi \oplus \phi' : (U \oplus U') \times (V \oplus V') \to W \oplus W' \text{ and } \phi \otimes \phi' : (U \otimes U') \times (V \otimes V') \to W \otimes W'$$

that send $(u \oplus u', v \oplus v')$ to $\phi(u,v) \oplus \phi(u',v')$ and $\phi(u,v) \otimes \phi(u',v')$.

Theorem 5.3.12 (Prop 14.23 in [2])**.** *Let ϕ_1 and ϕ_2 be two bilinear maps.*

- $Rk(\phi_1 \oplus \phi_2) \leq Rk(\phi_1) + Rk(\phi_2)$.
- $Rk(\phi_1 \otimes \phi_2) \leq Rk(\phi_1)Rk(\phi_2)$.

One of the other advantages of rank comparing to the complexity is that, we do not know if $L(\phi_1 \otimes \phi_2) \leq L(\phi_1)L(\phi_2)$ or not. Additivity conjecture states the rank of the direct sum is equal to the sum of the ranks. However, common belief is that the conjecture is wrong. So, the following open problem is natural.

Open Problem 5.3.13. *Find an example with* $Rk(\phi_1 \oplus \phi_2) < Rk(\phi_1) + Rk(\phi_2)$.

We can do some comments on the rank of the polynomial multiplication which will give us a great theory on tensor rank. Let k be a field. Consider the bilinear map

$$\phi_k^{m,n} : k[X]_{<m} \times k[X]_{<n} \to k[X]_{<m+n-1}.$$

Theorem 5.3.14. $R(\phi_k^{m,n}) \geq m+n-1$. *We have equality if* $|k| \geq m+n-2$.

See that the equality is just the special case of Theorem 5.1.18 by putting the degrees $m-1$ and $n-1$. Since bilinear maps corresponds to 3-tensor, the bilinear map $\phi_k^{m,n}$ can be denoted by a tensor $T_{m,n,k} \in \mathbb{F}_q^{m \times n \times k}$. Thus, we get the following theorem.

Theorem 5.3.15. $trk(T_{m,n,k}) \geq m+n-1$. *We have equality if* $q \geq m+n-2$.

Let us denote $\phi_k^{n,n}$ by ϕ_q^n when $k = \mathbb{F}_q$. From the above theorem, $R(\phi_k^{n,n}) \geq 2n-1$. We will give some known results and open problems here.

Theorem 5.3.16 ([10])**.** $R(\phi_q^n) \geq \max\{N_q(r, 2n-r) \mid n \leq r < 2n\}$.

Theorem 5.3.17 ([2])**.** *Let* $r = \phi_q^n$. *Then, for any* $1 \leq x \leq n$, *there exists and* $[r-n+x, x, n]_q$ *code.*

Open Problem 5.3.18. *Determine* $R(\mathbb{F}_{q^n})$ *for any given* q *and* n.

Open Problem 5.3.19. *Determine* $R(\phi_k^{m,n})$ *for all* m, n *and finite field* k.

Open Problem 5.3.20. *Determine for which values of* q *and* n, *the rank* $Rk(\mathbb{F}_q^{n \times n})$ *is minimal.*

Consider a bilinear map $\phi : U \times V \to W$. The following definition is the correspondence of Definition 3.0.3.

Definition 5.3.21. ϕ *is 1-concise if* $\{u \in U \mid \phi(u, V) = 0\} = \{0\}$.

As we noted, ϕ gives us a 3-tensor T. If $T = \sum_{i=r}^{R} u_r \otimes v_r \otimes w_r \in U \otimes V \otimes W$ and T is 1-concise, then we have $< u_1, \ldots, u_R >= U$. Thus, $Rk(T) \geq dim(U)$. This is very crucial since it help us characterize n-dimensional $\mathbb{F}$-algebras of rank n.

Theorem 5.3.22 (Proposition 14.39 in [2])**.** *Let* A *be an n-dimensional* $\mathbb{F}$*-algebra. Then* $Rk(A) = n$ *if and only if* $A \cong \mathbb{F}^n$.

Example 5.3.23. *Consider the $\mathbb{R}$-algebra $\mathbb{C}$. Note that $\mathbb{C}$ is 2-dimensional over $\mathbb{R}$ with the basis $\{1,i\}$. We need to define a bilinear map. We multiply two complex numbers as $(x+yi)(c+di) = (xc-dy)+i(xd+yc)$. By Definition 5.1.14 and Theorem 5.3.22, we have the multiplication map*

$$\phi : \mathbb{R}^2 \times \mathbb{R}^2 \to \mathbb{R}^2 : ((a,b),(a',b') \longmapsto (aa'-bb', a'b+ab').$$

In $\mathbb{R}^2$, we have $1=(1,0)$ and $i=(0,1)$. Let e_1 and e_2 be standard basis for $\mathbb{R}^2$. Then,

$$\phi((1,0),(1,0)) = (1,0) = \sum_{k=1}^{2} T_{11k}e_k = T_{111}e_1 + T_{112}e_2 \implies T_{111}=1,\ T_{112}=0.$$

$$\phi((1,0),(0,1)) = (0,1) = \sum_{k=1}^{2} T_{12k}e_k = T_{121}e_1 + T_{122}e_2 \implies T_{121}=0,\ T_{122}=1.$$

$$\phi((0,1),(1,0)) = (0,1) = \sum_{k=1}^{2} T_{21k}e_k = T_{211}e_1 + T_{212}e_2 \implies T_{211}=0,\ T_{212}=1.$$

$$\phi((0,1),(0,1)) = (-1,0) = \sum_{k=1}^{2} T_{22k}e_k = T_{221}e_1 + T_{222}e_2 \implies T_{221}=-1,\ T_{222}=0.$$

Thus, we have $T = e_1\otimes e_1\otimes e_1 + e_1\otimes e_2\otimes e_2 + e_2\otimes e_1\otimes e_2 - e_2\otimes e_2\otimes e_1$. Equivalently, $T = e_1\otimes(e_1\otimes e_1 + e_2\otimes e_2) + e_2\otimes(e_1\otimes e_2 - e_2\otimes e_1)$. So, we have

$$cs_1(T) = \left\langle \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \right\rangle = \langle T_1, T_2 \rangle .$$

See that T is 1-concise since the dimension of the first contraction space is 2. Similarly, this can also be seen by checking Definition 5.3.21.

Recall that in Chapter 3, we noted that $X \in \mathbb{F}^{k\times n\times m}$ is 1-concise if and only if all 1-slices of X are linearly independent. For example, see Lemma 3.1.6. We end this subsection by rewriting it in the following theorem that explains this concept in bilinear maps language.

Theorem 5.3.24. *Let $\phi : \mathbb{F}^k \times \mathbb{F}^n \longrightarrow \mathbb{F}^m$ be a bilinear map such that the first contraction space of the corresponding tensor is the span of $T_1,\ldots,T_k$. Then,*

$$\phi \text{ is 1-concise } \iff \{T_1,\ldots,T_k\} \text{ is linearly independent.}$$

5.4 Matrix Multiplication

One of the leading problems in algebraic complexity theory is matrix multiplication. Multiplying two $n \times n$ matrices C and D using

$$(c_{ij})(d_{jk}) = \sum_j c_{ij} d_{jk}$$

uses a number of operations which is proportional to n^3. We can see this by considering the first entry c_{11}. It will be multiplied with n entries of D. So, in total we will need n^3 multiplications since C has n^2 entries. There will be $(n-1)$ additions for each entry. This will give us $n^2(n-1) = n^3 - n^2$ additions. So, we will have $2n^3 - n^2$ operations in total.

Example 5.4.1. *Consider multiplication of 2×2 matrices. Let $\mathbb{F}$ be a field. Consider 2 different bases $A = \{A_0, A_1, A_2, A_3\}$ and $B = \{B_0, B_1, B_2, B_3\}$ where*

$$A_0 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix},\ A_1 = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix},\ A_2 = \begin{pmatrix} 0 & 0 \\ -1 & 1 \end{pmatrix},\ A_3 = \begin{pmatrix} 0 & 1 \\ 0 & 1 \end{pmatrix},$$

$$B_0 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix},\ B_1 = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix},\ B_2 = \begin{pmatrix} 1 & -1 \\ 0 & 0 \end{pmatrix},\ B_3 = \begin{pmatrix} 1 & 0 \\ 1 & 0 \end{pmatrix}.$$

Note that this is a very smart choice since for all $0 \leq i, j \leq 3$, we have the property that $A_i B_j \in \{A_i, B_j, 0^{2\times 2}\}$. This is checked in Table 5.2.

Table 5.2 2×2 Matrix Multiplication

	B_0	B_1	B_2	B_3
A_0	A_0	B_1	B_2	B_3
A_1	A_1	0	B_2	A_1
A_2	A_2	B_1	A_2	0
A_3	A_3	A_3	0	B_3

Consider $n = 2m$. Then, $X, Y \in (\mathbb{F}^{n\times n}) = (\mathbb{F}^{m\times m})^{2\times 2}$. In Section 1.2, we mentioned that given $C \in \mathbb{K}^{r\times r}$ and $D \in \mathbb{K}^{s\times s}$, we have

$$C \otimes D = \begin{bmatrix} c_{11}D & c_{12}D & \dots \\ c_{21}D & c_{22}D & \dots \\ \vdots & \vdots & \end{bmatrix} \in (\mathbb{F}^{s\times s})^{r\times r}.$$

Now, let us write X by using the basis A and Y by using the basis B. We have

$$X = \sum_{0 \le i \le 3} A_i \otimes x_i \text{ and } Y = \sum_{0 \le j \le 3} B_j \otimes y_j,$$

where $x_i,\ y_j \in \mathbb{F}^{m \times m}$ are uniquely determined since A and B are bases. Then,

$$\begin{aligned} XY &= (\sum_{0 \le i \le 3} A_i \otimes x_i)(\sum_{0 \le j \le 3} B_j \otimes y_j) \\ &= \sum_{i,j} (A_i B_j) \otimes (x_i y_j) \\ &= A_0 \otimes (x_0 y_0) + A_1 \otimes (x_1(y_0 + y_3)) + A_2 \otimes (x_2(y_0 + y_2)) + A_3 \otimes (x_3(y_0 + y_1)) \\ &+ B_1 \otimes ((x_0 + x_2)y_1) + B_2 \otimes ((x_0 + x_1)y_2) + B_3 \otimes ((x_0 + x_3)y_3) \\ &= A_0 \otimes P_1 + A_1 \otimes P_2 + A_2 \otimes P_3 + A_3 \otimes P_4 + B_1 \otimes P_5 + B_2 \otimes P_6 + B_3 \otimes P_7. \end{aligned}$$

where the second last equality follows from Table 5.2. Note that the $m \times m$ matrices P_k's contains 1 addition of the form $x_0 + x_i$ or $y_0 + y_j$ and 1 multiplication. So, we have 7 multiplications to compute all P_k's. In [2], it is proven that the x_i's and y_j's can be evaluated using 10 additions. Also, it is shown that the 1 addition inside P_k's can be evaluated using those 10 additions. Let $M = XY$ be the desired product. Now, observe that

$$M_{11} = P_1 + P_2 + P_6 + P_7\ ,\ M_{12} = P_4 - P_6,$$

$$M_{21} = -P_3 + P_7\ ,\ M_{22} = P_1 + P_3 + P_4 + P_5.$$

Thus, we have 8 more additions. So, in total of 7 multiplications and 18 additions are required to multiply two $n \times n$ matrices where $n = 2m$ for some integer m. This algorithm explained above is known as Strassen's algorithm.

Definition 5.4.2. *$T(n)$ is the minimum number of arithmetic operations required to compute the product of two $n \times n$ matrices.*

18 additions of $m \times m$ matrices gives us in total of $18m^2$ additions. Similarly, 7 multiplication of $m \times m$ matrices are required. So, we have the upper bound

$$T(n) \le 7T(\frac{n}{2}) + 18(\frac{n}{2})^2.$$

Solving this with $T(1) = 1$, give us $T(n) \le 7n^{log7} - 6n^2$. Note that $log7 < 2.81 < 3$, i.e., this is a huge improvement since we now can say that it is $\mathcal{O}(n^{log7})$ instead of $\mathcal{O}(n^3)$.

Definition 5.4.3. $w = \inf\{h \in \mathbb{R} \mid \textit{multiplication in } \mathbb{F}^{n\times n} \textit{ has cost } \mathcal{O}(n^h)\}$.

This is a very active research area and so far it is known that $w < 2.38$. We finish this discussion and state the open problem.

Open Problem 5.4.4. *Determine the exponent w of matrix multiplication.*

We write $< e,h,l >$ for the matrix multiplication $\mathbb{F}^{e\times h} \times \mathbb{F}^{h\times l} \to \mathbb{F}^{e\times l}$. Thus,

$$< e,h,l >= \sum_{i,j,m} u_{ij} \otimes v_{jm} \otimes w_{mi} \in \mathbb{F}^{e\times h} \otimes \mathbb{F}^{h\times l} \otimes \mathbb{F}^{l\times e}$$

Clearly, $< e,e,e >= \mathbb{F}^{e\times e}$, and $< e,h,l > \otimes < e',h',l' > \cong < ee',hh',ll' >$. We showed that $Rk(< 2,2,2 >) \leq 7$. We will give another approach which help us to see the tensor rank. Consider $\begin{pmatrix} a & b \\ c & d \end{pmatrix}\begin{pmatrix} x & y \\ z & t \end{pmatrix}$. This can be seen as $\begin{pmatrix} a & 0 & b & 0 \\ 0 & a & 0 & b \\ c & 0 & c & 0 \\ 0 & d & 0 & d \end{pmatrix}\begin{pmatrix} x \\ y \\ z \\ t \end{pmatrix}$. We have the basis

$$B = \{\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}\}.$$

This can be split into 8 rank one matrices. Thus, $trk(M^{2\times 2}(\mathbb{F})) \leq 8$. However, we showed above that it is at most 7. We can actually create a tensor to represent this and show that its rank is at most 7. See the first basis element. The entry in the first row and first column is 1. So, we have $e_1 \otimes e_1 \otimes e_1$ as the first pure tensor. The first component represent which basis element it corresponds to, second one represents in which column it lies and the last one represents the row it lies on. Using this approach, we get
$T = e_1 \otimes e_1 \otimes e_1 + e_1 \otimes e_2 \otimes e_2 + e_2 \otimes e_3 \otimes e_1 + e_2 \otimes e_4 \otimes e_2 +$
$e_3 \otimes e_1 \otimes e_3 + e_3 \otimes e_2 \otimes e_4 + e_4 \otimes e_3 \otimes e_3 + e_4 \otimes e_4 \otimes e_4$.

Theorem 5.4.5. $trk(T) \leq 7$.

Proof. $T = (e_1 + e_4) \otimes (e_1 + e_4) \otimes (e_1 + e_4) + (e_2 + e_4) \otimes e_1 \otimes (e_2 - e_4)$
$+ e_1 \otimes (e_3 - e_4) \otimes (e_3 + e_4) + e_4 \otimes (-e_1 + e_2) \otimes (e_1 + e_3) + (e_1 + e_3) \otimes e_4 \otimes (-e_1 + e_3)$
$+ (-e_1 + e_2) \otimes (e_1 + e_3) \otimes e_4 + (e_3 - e_4) \otimes (e_2 + e_4) \otimes e_1$. □

In [26], it is proven that $trk(T) = 7$.

6. FINITE GEOMETRIC APPROACH

So far, the only geometric object which is related to rank-metric codes was tensors. In this chapter, we will talk about the relation between rank-metric codes and some other geometric objects.

6.1 Semifields

Definition 6.1.1. *A finite semifield $(S,+,\circ)$ is a structure satisfying*

- $|S| \geq 2$,
- $(S,+)$ *is an abelian group,*
- $(a+b)\circ c = a\circ c + b\circ c$,
- $a\circ(b+c) = a\circ b + a\circ c$,
- $x\circ y = 0 \implies x = 0$ *or* $y = 0$,
- $\exists 1 \in S$ *such that* $a\circ 1 = 1\circ a = a$ *for all* $a,b,c \in S$.

Note that, semifields are division algebras which are not necessarily associative or commutative. So, we define the center of a semifield as a measure of its closeness to being a field.

Definition 6.1.2. $Z(S) = \{x \in S |$ *x commutes and associates with all elements of S*$\}$.

We have the following theorem by [6].

Theorem 6.1.3. *If $[S : Z(S)] = 2$, then S is a field.*

Given a semifield we can define its left, right and middle nucleus.

Definition 6.1.4. *Given a semifield S, we have*

$$N_l(S) = \{x \in S |\ x \circ (y \circ z) = (x \circ y) \circ z,\ \forall y, z \in S\}.$$

$$N_m(S) = \{y \in S |\ x \circ (y \circ z) = (x \circ y) \circ z,\ \forall x, z \in S\}.$$

$$N_r(S) = \{z \in S |\ x \circ (y \circ z) = (x \circ y) \circ z,\ \forall x, y \in S\}.$$

Similarly, the middle and right nucleus of a rank-metric code C are defined as follows in [21].

Definition 6.1.5. *Given a rank-metric code $C \subseteq \mathbb{F}_q^{n\times m}$, we have*

$$N_m(C) = \{A \in \mathbb{F}_q^{n\times n} |\ AM \in C\ \forall M \in C\}.$$

$$N_r(C) = \{B \in \mathbb{F}_q^{m\times m} |\ MB \in C\ \forall M \in C\}.$$

In Definition 3.2.11, we defined the equivalence of two rank-metric codes C and C' with respect to an isometry on $\mathbb{F}_q^{n\times m}$. The following well-known theorem gives us another equivalence criteria.

Theorem 6.1.6. *If f is an isometry of $\mathbb{F}_q^{n\times m}$ with $n, m \geq 2$, then there exists matrices $X \in GL(n,q)$ and $Y \in GL(m,q)$ and $Z \in \mathbb{F}_q^{n\times m}$ such that $f(A) = XA^{\alpha}Y + Z$ for all $A \in \mathbb{F}_q^{n\times m}$ where $\alpha \in Aut(\mathbb{F}_q)$. In particular, $Z = 0$ if f is additive.*

Using the theorem, the equivalence of rank-metric codes can be defined as the following. Two rank metric codes C_1 and C_2 are equivalent if there are $X \in GL(n,q)$, $Y \in GL(m,q)$, $Z \in \mathbb{F}_q^{n\times m}$ and $\alpha \in Aut(\mathbb{F}_q)$ such that

$$C_2 = \{XA^{\alpha}Y + Z : A \in C_1\}.$$

Clearly, if C_1 and C_2 are linear, then we can assume $Z = 0$. In the next theorem, we will start to see the connection between rank-metric codes and semifields.

Theorem 6.1.7. *If C_1 and C_2 are equivalent linear rank-metric codes, then their middle(similarly right) nucleus are also equivalent.*

Proof. $A \in N_m(C_1) \iff AM \in C_1\ ,\ \forall M \in C_1 \iff X(AM)^{\alpha}Y \in C_2\ ,\ \forall M \in C_1$ $\iff XA^{\alpha}M^{\alpha}Y \in C_2\ ,\ \forall M \in C_1 \iff (XA^{\alpha}X^{-1})(XM^{\alpha}Y) \in C_2\ ,\ \forall M \in C_1$. Thus,

$$A \in N_m(C_1) \iff XA^{\alpha}X^{-1} \in N_m(C_2)$$

since by Theorem 6.1.6, we have $XM^{\alpha}Y \in C_2$, i.e, $XA^{\alpha}X^{-1} \in N_m(C_2)$ due to the

assumption that the codes C_1 and C_2 are equivalent. Again, by Theorem 6.1.6, we have the desired result. □

Now, we will define what does it mean to have an isotopism between semifields. Note that given a map F and $x \in S$, x^F denotes the image of x under the map F.

Definition 6.1.8. *Consider two finite semifields $(S,+,\circ)$ and $(S',+,\star)$. The 3-tuple (F,G,H) where all three components are nonsingular linear maps from S to S' is called an isotopism between S and S' if*

$$x^F \star y^G = (x \circ y)^H \ \forall x,y \in S.$$

Semifield theory is mainly studied by looking at the isotopism classes of semifields. Note that if a linear code C defines a semifield, then the middle and right nucleus are invariant under isotopy. However, if the code in non-linear, then they are not invariant.

By relaxing some of the properties in the definition of a semifield, we can get some other geometric objects.

Definition 6.1.9. *If we remove the left distributivity law in the definition of a semifield, then it is called a finite (right) quasifield $(Q,+,\circ)$.*

We will state a very important classification theorem using quasifields in the following parts. It will come in handy to define the kernel of a quasifield now.

Definition 6.1.10.

$$KerQ = \{s \in Q : s\circ(a+b) = s\circ a + s\circ b, s\circ(a\circ b) = (s\circ a)\circ b \text{ for all } a,b \in Q\}.$$

Definition 6.1.11. *Without the multiplicative identity element, $(S,+,\circ)$ is called a pre-semifield.*

The reason we like working with pre-semifields instead of semifields is that sometimes it is very difficult to find a reasonable formula for the multiplication. However, we need to find a way to get semifields starting from pre-semifields so that it makes sense to work with pre-semifields. There is way introduced by Kaplansky in [11]. This is also known as Kaplansky's trick.

Theorem 6.1.12. *Given a pre-semifield $(S,+,\circ)$ and $a,b,x \in S$, we have $(S,+,\star)$ as a semifield by defining*

$$(a\circ x)\star(x\circ b) = a\circ b.$$

Proof. See that $x \circ x$ is the multiplicative identity, and thus we are done. □

Similarly, using Kaplansky's trick, one can show that any pre-semifield is isotopic to a semifield under the isotopism (R, L, id) where $R, L : (S, +, \circ) \to (S, +, \star)$ denotes the right and left multiplication by x, respectively.
We mentioned that semifields are algebras which are not necessarily associative or commutative. We precisely defined what an algebra is in Definition 5.1.14. Let A be an n-dimensional algebra over $\mathbb{F}$ and $\{e_1, \ldots, e_n\}$ be a $\mathbb{F}-$basis for S. Then, we can define

$$x \circ y = \sum_{i,j=1}^{n} x_i y_j (e_i \circ e_j) = \sum_{i,j=1}^{n} x_i y_j (\sum_{k=1}^{n} a_{ijk} e_k).$$

Here, $x = \sum_{i=1}^{n} x_i e_i$ and $y = \sum_{i=1}^{n} y_i e_i$ with $x_i, y_i \in \mathbb{F}$ and some constant $a_{ijk} \in \mathbb{F}$. These constants $a_{ijk} \in \mathbb{F}$ are called structure constants. In [12], it was observed that the action of the symmetric group S_3 over the indices of the structure constants give us 5 more semifields. In total, we get 6 semifields, not necessarily different. The set of those 6 semifields is given by $\{S, S^{(12)}, S^{(13)}, S^{(23)}, S^{(123)}, S^{(132)}\}$ and it is called the S_3-orbit of S. That give us the set of isotopism classes of semifield S. The set is known as the Knuth orbit and defined as follows:

Definition 6.1.13. *Knuth orbit of a semifield S is*

$$\{[S], [S^{(12)}], [S^{(13)}], [S^{(23)}], [S^{(123)}], [S^{(132)}]\}$$

To understand the behaviour, we will consider S^{12}.

Theorem 6.1.14. *$S^{(12)}$ fixes the middle nucleus and interchanges the left and right nucleus.*

Proof. $N_l : x \circ (y \circ z) = (x \circ y) \circ z \ \forall y, z \in S \to (z \circ y) \circ x = z \circ (y \circ x) \ \forall y, z \in S \ : N_r$.
$N_m : x \circ (y \circ z) = (x \circ y) \circ z \ \forall x, z \in S \longrightarrow (z \circ y) \circ x = z \circ (y \circ x) \ \forall x, z \in S \ : N_m$.
$N_r : x \circ (y \circ z) = (x \circ y) \circ z \ \forall x, y \in S \longrightarrow (z \circ y) \circ x = z \circ (y \circ x) \ \forall x, y \in S \ : N_l$. □

Question 6.1.15. *What happens if we apply this action to rank-metric codes? Can we derive a similar theory just like for the semifields to the rank-metric codes?*

Given a semifield S, we can construct a spread $\mathbb{S}$ of $PG(2n-1, q)$ as follows:
$\mathbb{S} = \{\mathbb{S}_a |\ a \in S \cup \{\infty\}\}$ where $\mathbb{S}_a = \{(x, a \circ x) : x \in S\}$ and $\mathbb{S}_\infty = \{(0, x) : x \in S\}$. We can also define a spread of $PG(2n-1, q)$ using a spreadset C in $\mathbb{F}_q^{n \times n}$. First, let us define what is a spreadset.

Definition 6.1.16. *A spreadset $C \leq \mathbb{F}_q^{n \times n}$ is a set of q^n nonsingular matrices such*

that the difference of any two of them is nonsingular. In other words, we have $0 \in C$, $|C| = q^n$, *and* $det(A-B) \neq 0$ *for all* $A, B \in C$.

Given a semifield S, we can create the spreadset C defined above.

Definition 6.1.17. *If the spreadset* C *is closed under addition, then it is called a semifield spread set.*

Now, the spread $\mathbb{S}$ becomes

$$\mathbb{S} = \{\mathbb{S}(A) \mid A \in C\} \cup \{\mathbb{S}(\infty)\}.$$

Here, $\mathbb{S}(A) = \{(x, xA) \ : \ x \in \mathbb{F}_q^n\}$ and $\mathbb{S}(\infty) = \{(0, y) \ : \ y \in \mathbb{F}_q^n\}$. Again, clearly $\mathbb{S}$ is an $(n-1)$-spread of $PG(2n-1, q)$ since it is a partition of the points of the space by subspaces of projective dimension $n-1$.

Remark 6.1.18. *Let* S *be a semifield and* C *be its corresponding spreadset. Since* $\mathbb{F}_q^{n \times n} \cong End_{\mathbb{F}_q}(\mathbb{F}_q^n) := E$, *we can see the elements of* C *as* $\mathbb{F}_q$*-linear maps on* $\mathbb{F}_q^n$.

Note that two spreadsets are equivalent if their corresponding spreads are isomorphic. Similar to the equivalence of rank-metric codes, we can define the equivalence of semifield spreadsets. The following theorem is taken from [18] and proven there.

Theorem 6.1.19. *Semifield spreadsets* $C, C' \subseteq E$ *are equivalent if and only if there exists invertable elements* $X, Y \in E$ *and* $\alpha \in Aut(\mathbb{F}_q)$ *such that*

$$C' = \{XA^{\alpha}Y : A \in C\}.$$

Note 6.1.20. *This is actually the same definition as the equivalence of rank-metric codes. We do not have the "+Z" term here since as stated in Theorem 6.1.6,* $Z = 0$ *if we have additivity. This is the case since semifield spreadsets are closed under addition by Definition 6.1.17.*

Consider C a semifield spreadset. Thus, by definition, C is an additive subgroup of E. Then, we can assume that it is an $\mathbb{F}_p$-subspace for some subfield $\mathbb{F}_p$ of $\mathbb{F}_q$ where $q = p^t$. By Remark 1.1.11, we have

$$\dim_{\mathbb{F}_p} C = (\dim_{\mathbb{F}_q} C)(\dim_{\mathbb{F}_p} \mathbb{F}_q) = n \cdot t = nt.$$

Therefore, we have the following theorem which we will prove in a moment.

Theorem 6.1.21. *The semifield* S *whose corresponding spreadset is* C, *defines an* $\mathbb{F}_p$*-linear set* L *in* $PG(n^2-1, q)$ *of rank* nt.

We will use the field reduction map that was introduced in Chapter 3.

$$F_{r,t,p} : PG(r-1,p^t) \longrightarrow PG(rt-1,p)$$
$$\text{all points} \longmapsto D_{r,t,p}$$

Here, $D_{r,t,p}$ represents the Desarguesian spread. We will come back to it later. Let U be a subspace of D, i.e. $U \leq D$, such that $B_U = \{x \in D |\ x \cap U \neq \emptyset\} \subseteq D$. Then, we take its pre-image and call it the linear set $L(U)$.

Definition 6.1.22. *If U has dimension d in $PG(rt-1,p)$, then the linear set $L(U)$ is said to be of rank $d+1$.*

Remark 6.1.23. *Note that $L(U)$ is scattered if and only if U intersect x in at most a point for all $x \in D$. A scattered linear set has rank at most $\frac{rt}{2}$. If that bound is met, then it is called maximum scattered set, and they have some very interesting relations with MRD codes. For more on this subject, see [4].*

Proof of Theorem 6.1.21. Note that $PG(n^2-1,q) = PG(n^2-1,p^t)$. By the field reduction $F_{n^2,t,p}$, it is mapped to $PG(n^2t-1,p)$. Recall that $\dim_{\mathbb{F}_p} C = nt$. Thus, C has dimension $nt-1$ in $PG(n^2t-1,p)$. Then, $B_C = \{x \in D |\ x \cap C = \emptyset\}$. Its pre-image is the set $L = \{\mathbb{F}_{p^t} u |\ u \in C - \{0\}\}$ which has rank $(nt-1)+1 = nt$, as desired. □

Note 6.1.24. *We talked about how to use field reduction map to get rank-metric codes from the vector codes before. See Figure 3.1 and how we applied it to the Example 3.3.6.*

Consider the tensor space $\mathcal{T}$ defined in Section 3. Now, we will define a contraction of an element in $\mathcal{T}$ just like we defined the contraction spaces of 3-tensors to get information about rank-metric codes.

Definition 6.1.25. *We call any nonzero vector of V_i nonsingular. Let $T \in \mathcal{T}$. We call $v_i^D(T) \in V_1 \otimes \ldots \otimes V_{i-1} \otimes V_{i+1} \otimes \ldots \otimes V_m$ the contraction of $T \in \mathcal{T}$ by $v_i^D \in V_i^D$. We call a tensor T nonsingular if the contraction $v_i^D(T)$ is nonsingular for all $i \in [m]$.*

Now, we can define the contraction spaces just as we did in Definition 3.0.3. It is basically the subspace spanned by all the contractions of T.

Example 6.1.26. *Let us find the $i-th$ contraction space of a nonsingular tensor $T \in \mathcal{T}$. It is equal to*

$$cs_1(T) = < v_i^D(T) \mid v_i^D \in V_i^D > .$$

Consider any nonzero element T' of $cs_1(T)$. Then, T' can be written as a linear combination of contractions of T. Hence, T' is a contraction itself. Since T is

nonsingular, T' is nonsingular as well. Thus, any element of the contraction space is also a nonsingular tensor.

Hence, given any nonsingular tensor T, we have a set of nonsingular tensors in tensor product where the $i-th$ factor is left out. Then, the dimension of the $i-th$ contraction space of T is equal to dimension of V_i, i.e., the factor that is left out. So, we have the following corollary.

Corollary 6.1.27. *Given a nonsingular tensor T, we have $cs_i(T) = dim(V_i) = d_i$. Thus, T is concise.*

We can combine Definition 3.2.3 and 3.2.9 to make the following observation.

Note 6.1.28. *$rank(T) = rank(cs_1(T))$ for all $i \in [m]$.*

Given a semifield, we can define a tensor associated to it. We did it in Example 5.3.6. Now, we will state a theorem using Definition 1.1.20.

Theorem 6.1.29 (Theorem 4.2 and Theorem 4.3 in [16])**.** *Consider a semifield S and two presemifields S_1 and S_2.*

(1) S_1 and S_2 are isotopic if and only if $T^G_{S_1} = T^G_{S_2}$. This is given by the relation

$$(A, B, K) : S_1 \to S_2 \iff (A^D, B^D, K^{-1}) : T_{S_1} \to T_{S_2}.$$

(2) $T_S \in U \otimes V \otimes W$ is nonsingular.

(3) Given a nonsingular tensor T, there exists a presemifield S' such that $T = T_{S'}$.

The important takeaway from this theorem is the following remark.

Remark 6.1.30. *There exists a correspondence between set of isotopism classes of semifields and the G-orbits of nonsingular tensors in $U \otimes V \otimes W$. In other words,*

$$[S] \iff T^G_S.$$

Note 6.1.31. *Observe that Remark 6.1.30 also means that if we have two tensors $T_1, T_2 \in \mathcal{T}$ that are in the same G-orbit, then it is equivalent to say that $i-th$ contraction spaces $cs_i(T_1)$ and $cs_i(T_2)$ are also in the same G-orbit.*

Consider a semifield S and its associated nonsingular tensor $T_S \in U \otimes V \otimes W$. Then the following definition is natural.

Definition 6.1.32. *A point corresponding to the semifield S is*

$$P_S = PG(\langle T_S \rangle) \in PG(U \otimes V \otimes W).$$

Lemma 6.1.33. *If $P_{S_1} = P_{S_2}$, then the semifields S_1 and S_2 are isotopic.*

Proof. $P_{S_1} = P_{S_2}$ means there exists a nonzero scalar $c \in \mathbb{F}$ such that $T_{S_1} = cT_{S_2}$. By Theorem 6.1.29, consider $(id, id, x \longmapsto c^{-1}x)$ so that $(id, id, x \longmapsto cx)$ gives two semifield in the same isotopism class. □

We need the following concepts from projective geometry to continue. Recall that just as in Definition 1.1.5, we can define $\Gamma L(V)$ as the group of nonsingular semilinear transformations of V. So, $\Gamma L(V)$ is just the semidirect product of $GL(V)$ and $Aut(\mathbb{F})$. That means, any $f \in \Gamma L(V)$ is of the form (A, σ) where $A \in GL(V)$ and $\sigma \in Aut(\mathbb{F})$. The following definition is well-known.

Definition 6.1.34. *Any $f \in \Gamma L(V)$ induces a collineation(bijective morphism) α of $PG(V)$. If α is induced by $f = (A, id)$, then α is called a projectivity and the group of all projectivities is denoted by $PGL(V)$. Similarly, if $f = (A, \neq id)$, then $\alpha \in P\Gamma L(V)$.*

In Definition 1.1.20, we defined $G = GL(U) \times GL(V) \times GL(W)$. Thus, G is a subgroup of $GL(U \otimes V \otimes W)$. Now, consider a subgroup M of $PGL(U \otimes V \otimes W)$ such that any element of M is induced by (A, id) where $A \in G$.

Theorem 6.1.35 (Theorem 4.7 in [16])**.** $S_1 \cong S_2 \iff P_{S_1}^M = P_{S_2}^M$.

Proof. By Theorem 6.1.29, $S_1 \cong S_2 \iff T_{S_1}^G = T_{S_2}^G$. By how we constructed M and Lemma 6.1.33, we have $S_1 \cong S_2 \iff T_{S_1}^G = T_{S_2}^G \iff P_{S_1}^M = P_{S_2}^M$. □

Last but not least, we define the tensor rank of a semifield.

Definition 6.1.36. *Tensor rank of a semifield S is the rank of the corresponding tensor T_S.*

The big idea that help us understand and develop the correspondence between rank-metric codes, tensors and algebras was this definition and the theory behind it. This is examined in [16] where tensor rank also turns out to be an invariant under semifield isotopy.

Theorem 6.1.37. *Tensor rank of semifields is an invariant under isotopism of semifields.*

Proof. Let $S_1 \cong S_2$ given by (A, B, K). By Theorem 6.1.29, (A^D, B^D, K^{-1}) maps T_{S_1} to T_{S_2}. Since A, B, K are nonsingular, ranks are also the same. Thus, the ranks of the corresponding tensors are also the same, as desired. □

Now, let us talk about how to compute the tensor rank of semifields, so that we can apply this theory to MRD codes, i.e, to a family of rank-metric codes. Here we refer to the paper [19]. In that paper, the first examples of semifields of the same order with different tensor ranks have been given. Similarly, they prove that there are semifields of order 81 whose tensor rank is less than the tensor rank of a finite field of size 81. In terms of complexity, this is huge result since one can use semifields instead of finite fields in some occasions to provide faster algorithms. This is the first such known example in the literature. In the same paper, they considered $T \in \mathcal{T}$ such that we have its decomposition into R pure tensors. That is,

$$T = \sum_{r=1}^{R} v_{r1} \otimes v_{r2} \otimes \ldots \otimes v_{rm}.$$

Then, one can define m linear codes with the generator matrices

$$G_i = \begin{pmatrix} | & | & \ldots & | \\ v_{1i} & v_{2i} & \ldots & v_{Ri} \\ | & | & \ldots & | \end{pmatrix}.$$

The following theorems summarizes this relation and proofs can be found therein.

Theorem 6.1.38 ([19])**.** *Let C_i be the codes that are generated by G_i.*

(1) C_i is a code of length R.

(2) $dim(C_i) \leq d_i$.

(3) $d(C_i) \geq \min\{\max\{dim(v^D(cs_j(T))) : i \neq j\} \quad v^D \in V_i^D\}$.

(4) $trk(T) \geq N_q[dim(C_i), t_i]$ where t_i is the minimal tensor rank belonging to the elements of the $i-th$ contraction space.

Lastly, we propose two open questions using the theorem above.

Open Problem 6.1.39. *Find the exact equalities in Theorem 6.1.38.*

Open Problem 6.1.40. *In Theorem 6.1.38, the codes are defined by a decomposition of a tensor, not by a tensor. What would the parameters be in that case and what can be said about the uniqueness keeping in mind that decomposition of tensors are not unique.*

6.2 Gabidulin Codes and the Case m=n=d

Let us start by recalling the isomorphism $\mathbb{F}_q^{n\times m} \cong \mathbb{F}_{q^m}^n$. Given $(v_1,\ldots,v_n) \in \mathbb{F}_{q^m}^n$, we denote the $k \times n$ Moore matrix by

$$M_k(v_1,\ldots,v_n) := \begin{pmatrix} v_1 & v_2 & \ldots & v_n \\ v_1^q & v_2^q & \ldots & v_n^q \\ & & \vdots & \\ v_1^{q^{k-1}} & v_2^{q^{k-1}} & \ldots & v_n^{q^{k-1}} \end{pmatrix}.$$

Definition 6.2.1. *Let $u_1,\ldots,u_n \in \mathbb{F}_{q^m}$ be linearly independent vectors over $\mathbb{F}_q$ and form the Moore matrix $G = M_k(u_1,\ldots,u_n)$. We define the Gabidulin code $C \subseteq \mathbb{F}_{q^m}^n$ as the linear code whose generator matrix is G. Clearly, it has dimension k. The code C can be seen as a rank-metric code in $\mathbb{F}_q^{n\times m}$ by the fact that $\mathbb{F}_q^{n\times m} \cong \mathbb{F}_{q^m}^n$. It has been proven in [7] that a Gabidulin code $C \subseteq \mathbb{F}_{q^m}^n$ of dimension k over $\mathbb{F}_{q^m}$ has minimum distance $n-k+1$, i.e, it is an MRD code.*

Since the relation between the Gabidulin codes and the linearized polynomials are well-known, we will define and use linearized polynomials here and show how they are related to MRD codes.

Definition 6.2.2. *Let $t \in \mathbb{N}$. A linearized polynomial(q-polynomial) in $\mathbb{F}_{q^n}[X]$ is of the form*

$$a_0X + a_1X^q + \ldots + a_iX^{q^i} + \ldots = \sum_{i=0}^{t} a_iX^{q^i}$$

Let $L_{n,q}[X]$ be the set of all linearized polynomials in $\mathbb{F}_{q^n}[X]$. Then we have

$$L_{n,q}[X] \Big/ (X^{q^n} - X) \cong End_{\mathbb{F}_q}(\mathbb{F}_{q^n}).$$

$\mathbb{F}_{q^n}$ is the splitting field of the polynomials $x^{q^n} - x$. The elements of $\mathbb{F}_{q^n}$ are precisely the roots of $x^{q^n} - x$. Now consider

$$\sum_{i=0}^{t} a_iX^{q^i} \in \mathbb{F}_{q^n}[X] \Big/ (X^{q^n} - X).$$

The Gabidulin codes are defined as follows.

Definition 6.2.3. *Given n and d, we let $k = n-d+1$ and $m = n$. Then,*

$$G = \{a_0X + a_1X^q + \ldots + a_{k-1}X^{q^{k-1}} : a_0, \ldots, a_{k-1} \in \mathbb{F}_{q^n}\}.$$

Now, observe that for each $f \in G$, f has at most q^{k-1} roots. There are k coefficients where each can take q^n values. Thus, we have

$$|G| = (q^n)^k = q^{nk} = q^{n(m-d+1)}.$$

So, the Gabidulin codes are $\mathbb{F}_{q^n}$-linear MRD codes. Now, let us define the middle and right nucleus of a Gabidulin code. Again if the code defines a semifields, then they will correspond to the middle and right nucleus of a semifield.

Definition 6.2.4.

$$N_r(C) = \{g : g \circ f \in C \text{ for all } f \in C\} \cong \mathbb{F}_{q^n}$$

$$N_m(C) = \{g : f \circ g \in C \text{ for all } f \in C\} \cong \mathbb{F}_{q^n}.$$

It is good to know that your nucleus is isomorphic to $\mathbb{F}_{q^n}$ when you are dealing with some classification problems, or any kind of problem in general. We state the following remark before going further to the special case $m = n = d$.

Remark 6.2.5. *The existence of linear MRD codes for all q, m, n are known. It is also shown in [25] that there are some MRD codes which are different from the Gabidulin codes.*

For the rest of this section, consider $m = n = d$, i.e., $k = 1$. Then $C = \{a_0X : a_0 \in \mathbb{F}_{q^n}\}$. That is, $|C| = q^n$. Since $m = n = d$, we get an MRD code C in $\mathbb{F}_q^{n\times n}$ of minimum distance n. Hence, C is a set of matrices such that $det(A-B) \neq 0$ for all $A \neq B$ in C. Let $B \in C$. Now, consider the set

$$C - B = \{A - B : A \in C\}.$$

So, if we replace C with $C - B$, then we can have the zero matrix as an element of C since we can choose $A = B$. Thus, all nonzero matrices in C are invertable. Similarly, if we replace C with $B^{-1}C$ for some $B \neq 0$, then we can have the identity matrix an element of C. All this transformations are isometries, so the rank distance is preserved. So far, we have $|C| = q^n$ and $det(A) \neq 0$ for all nonzero $A \in C$. That means, we get a spreadset in $\mathbb{F}_q^{n\times n}$. Conversely, a spreadset in $\mathbb{F}_q^{n\times n}$ defines an MRD code C. To see this, we will show that we get a quasifield and prove a theorem

afterwards. Observe that $C - \{0\}$ acts transitively without the fixed points. That is, the pointwise stabilizer is trivial. So, suppose we fix a nonzero vector $v_0 \in \mathbb{F}_{q^n}$. Then, for all $v \in \mathbb{F}_{q^n}$, there exists $A(v) \in C$ such that $v_0 A(v) = v$. Without loss of generality, let $v_0 = e_1 = (1, 0, \ldots, 0)$. Thus, the first row of $A(v)$ is equal to v itself. In particular, we write $C = \{A(v) : v \in \mathbb{F}_{q^n}\}$ where $A(0) = 0$. In summary, given any two distinct nonzero elements v, w in $\mathbb{F}_{q^n}$, there is an unique matrix A such that $Av = w$. Consider the standard vector addition $\mathbb{F}_{q^n}$ and define the multiplication $\circ$ by

$$v \circ w = vA(w).$$

This multiplication gives us a quasifield. If Q is a finite quasifield, then $ker Q$ is a finite field. We can see Q as a finite dimensional left vector space over $ker Q$. Therefore, we have the following correspondence.

Theorem 6.2.6. *MRD codes in $\mathbb{F}^{n \times n}$ with minimum distance n corresponds to finite quasifields Q with $dim_{\mathbb{F}} Q = n$. In particular, additive MRD codes corresponds to finite semifields S with $dim_{\mathbb{F}} S = n$.*

Proof. Observe that Definition 6.1.16 is the same as the definition of an MRD code since $det(A - B) \neq 0$ means that the matrices are full rank, and thus the minimum distance is n. Above we showed how to get a quasifield from a spreadset. Thus, we are done with the first part. Now, let Q be a quasifield and $C = \{A(u) : u \in Q\}$ be additively closed. Thus, for each $u, v \in Q$, since C is additive, there is a unique $w \in Q$ such that $A(u) + A(v) = A(w)$. Then, we have

$$w = e_Q \circ w = e_Q A(w) = e_Q A(u) + e_Q A(v) = e_Q \circ u + e_Q \circ v = u + v.$$

Now, see that $x \circ u + x \circ v = xA(u) + xA(v) = xA(w) = x \circ (u + v)$. Thus, Q is left distributive and hence Q is a semifield. □

Remark 6.2.7. *Note that we have some inequivalent quasifields and semifields as shown in [18].*

Quasifields are strongly related to translation planes. Translation planes are precisely the affine planes which can be coordinatized by quasifields. Given a quasifield Q, we define a map $T : Q \times Q \times Q \to Q$ by $T(a, b, c) = ab + c$ for all $a, b, c \in Q$. (Q, T) satisfies the axioms of a planar ternary ring(PTR). Associated to (Q, T) is its projective plane. The details were shown in [8].

Remark 6.2.8. *A projective plane is a translation plane with respect to a line at infinity if and only if any (or all) of its associated PTR's are right quasifields.*

Following remark shows the connection of translation planes with spreads.

Remark 6.2.9. *Consider a spread S in $PG(3,q)$. S is a set of q^2+1 lines with no two intersecting. Andre-Bruck-Bose(ABB) construction produces a translation plane $P(S)$ of order q^2 as follows:*
Firstly, we embed $PG(3,q)$ as a hyperplane of $PG(4,q)$. Then, define the incidence structure $A(S) = (Points, Lines, I)$ where

- *Points = points of $PG(4,q)$ not on $PG(3,q)$.*
- *Lines = planes of $PG(4,q)$ meeting $PG(3,q)$ in a line.*
- *I = symmetric containment.*

Then $P(S)$ is just the projective completion of $A(S)$. In general, ABB construction shows that the study of translation planes correspond with the study of $(n-1)$-spreads of $PG(2n-1,q)$.

We summarize what we showed so far in the following note.

Note 6.2.10. *We have a close relation between quasifields and translation planes and this relation can be used to determine if a given projective plane is a translation plane or not. In addition to that, translation planes corresponds with the $(n-1)$-spreads of $PG(2n-1)$ which can also be constructed using a spreadset. Thus, all of them are connected. Lastly, in Theorem 6.2.6 it has been shown that quasifield are related to MRD codes, not to mention their relation with semifields.*

The following corollary is obvious now.

Corollary 6.2.11. *MRD codes C (m=n=d) and the following objects are equivalent:*

- *quasifields Q*
  When C is $\mathbb{F}_q$-linear, then Q is a semifield.
- *Spreadsets*

We propose an open problem, and an interesting question.

Open Problem 6.2.12. *For $m = n = 2,3$ and $d = 2$, classification of $\mathbb{F}_q$-linear codes in $\mathbb{F}_q^{n\times n}$ is solved. However if $q = p^e$ for a prime p and integer $e > 1$, the classification is open for all n.*

Question 6.2.13. *Note 6.2.10 shows that there are obvious similarities between Semifield Theory and Rank-metric codes. Given the best of our knowledge, it seems as a missing piece in the literature. Try to find as many correspondence as possible to develop and improve this corresponce.*

7. EXISTENCE OF MTR CODES

In this section, we will go back to Note 3.2.16 and summarize what has been proven so far. Then, we will state the open problem and provide some ideas on attacking that problem. The following has been said but not explicitly stated in [23].

Theorem 7.0.1. *Given $d \geq k$, there exists an $\mathbb{F}_q - [n \times m, k, d]$ MTR code for every $m, n \geq d$.*

Depending on that, they move on to the case $d < k$. We analyzed the known results in Section 4 and summarize them in the following remark.

Remark 7.0.2. *Suppose we have $d < k$.*

- *Proposition 4.2.11 $\implies$ existence of MTR codes for $n+m \geq R+d$.*
- *Lemma 4.2.5 $\implies$ existence of MTR codes where $m \geq k$ and $n \geq d$.*
- *Lemma 4.2.5 $\implies$ existence of MTR codes where $m \geq d$ and $n \geq k$.*
- *Theorem 2.1.6 $\implies$ there can not exist MTR codes for*

$$k > \min\{n(m-d+1), m(n-d+1)\}.$$

- *Theorem 4.3.3 $\implies$ existence of MTR codes that are also MRD, i.e, satisfying the analogue of the Singleton bound, Theorem 2.1.6.*

Thus, the only remaining case to check is the following.

Open Problem 7.0.3. *Given $k > n, m > d$ and $R+d > n+m$, decide whether there are $\mathbb{F}_q - [n \times m, k, d]$ MTR codes with tensor rank R. If yes, characterize them.*

In this section, we will try to attack on this problem 7.0.3. On the way, we will find correspondences with the previous parts as we did throughout the thesis. Let us start with the existence problem. We will choose the following parameters and try

to create an MTR code with those parameters.

$$n = 5,\ m = 3,\ k = 6,\ d = 2,\ R = 7,\ q = 3.$$

So, we are looking for an $\mathbb{F}_3 - [5 \times 3, 6, 2]$ MTR code. We are going to follow the geometric approach using Segre embedding.

Definition 7.0.4. *Segre embedding is given by the following map*

$$\sigma_{d_1,\ldots,d_k} : PG(d_1, q) \times \ldots \times PG(d_k, q) \longrightarrow PG(m, q)$$

that maps the k-tuple of points $(\langle v_1\rangle, \ldots, \langle v_k\rangle)$ *to the point* $\langle v_1 \otimes \ldots \otimes v_k\rangle$ *where* $m = (d_1 + 1)\ldots(d_k + 1) - 1$. *Its image is called the Segre variety* $S_{d_1,\ldots,d_k}$.

Clearly from the definition, we can see the points of the Segre variety as rank 1 matrices. Thus, to achieve an $\mathbb{F}_3 - [5 \times 3, 6, 2]$ code, we need to consider

$$\sigma_{4,2} : PG(4, 3) \times PG(2, 3) \longrightarrow PG(14, 3).$$

Since, we want the dimension to be 6, we will look for subspaces of projective dimension 5 in $PG(14, 3)$ such that it has no intersection with the Segre variety $S_{4,2}$ so that minimum distance of the code is at least 2. Let us talk about our strategy to find such a code.

(1) Look at all spaces spanned by 7 points of $S_{4,2}$ so that $trk(C) = 7$. It is important to note that Theorem 3.2.5 gives $trk(C) \leq 7$. However, since 7 is the minimal possibility given the Theorem 3.2.10, we will have $trk(C) = 7$.

(2) From the spaces created in (1), select the ones which have dimension 6 so that we have the dimension $k = 6$.

(3) Inside each of those selected spaces, try to find 5 dimensional spaces disjoint from the Segre variety $S_{4,2}$. These are the codes we are looking for.

The following algorithm can be used without any restrictions on the parameters to create MTR codes. We apply the 3 steps.

Algorithm 2: Creation of MTR codes with tensor rank R

Result: This algorithm will create an $\mathbb{F}_q - [n \times m, k, d]$ MTR code.

We first create the finite field we are working on, $\mathbb{K} := F_q$;

Define the Segre Variety $S_{n,m}$;

Create the points of the Segre Variety and write them as a list, call it P ;

Now, we need to look all the subspaces spanned by R points of P;

From each of those R points, we will check if their span is a k-space;

Denote the list of those k-spaces as L;

For each k-space in L, we will look for $k-1$ dimensional subspaces disjoint from the Segre variety $S_{n,m}$

A big problem in the above algorithm occurs when we are creating all the subspaces spanned by R points of P. Even for the small parameters for which we are trying to find an example, the size of P is 1573. Since, we need to choose $R = 7$ points from those 1573, we get a memory error. The reason is that $\binom{1573}{7}$ is too large. We write this in the following remark.

Remark 7.0.5. *We used [20] for the above algorithm. However, GAP has a memory limit and even for the very small cases, the computer memory is not enough to run the above algorithm. Although it should be theoretically examined, the storage complexity seems to be very high.*

We propose the following open problem.

Open Problem 7.0.6. *Implement the algorithm in programming language C or Java since they are much more developed and faster to see if the above algorithm gives an output or not.*

In the thesis, we revisited various constructions of MTR codes and stated the remaining case in the beginning of this section. If somehow we solve the memory problem in a clever way and run the algorithm, then we will get a list of all possible MTR codes with given tensor rank. Analyzing that codes might give us a pattern so that a classification can be done. Firstly, we need to make sure there exists a solution to Problem 7.0.3. Thus, we go back to the paramaters choosen in the beginning and apply this algorithm by hand in the following example.

Example 7.0.7. *Since we get a problem while computing $\binom{1573}{7}$, we will just choose 7 random points from the set P until we get the span of those 7 points as a 6 dimensional space. Using "repeat-until", we got such a space very easily in GAP. Call that space A. In [20], there is a function $ShadowOfElement(IG, E, D)$ which returns the collection of elements of type D incident with E, where IG is the ambient incidence structure. Note that 1,2,... are types of points,lines and so on. Since we*

are looking for a 5-space that is disjoint from the Segre variety $S_{4,2}$ and $S_{4,2}$ lies on $PG(14,3)$, we should first create those 5-spaces by

$$shadow := ShadowOfElement(PG(14,3), A, 6)$$

and then choose a random element in shadow, say $C := Random(shadow)$ and try until $P \cap C = \emptyset$ so that the minimum distance is at least 2. If the minimum distance is exactly 2, then we get the desired code. After applying this algorithm, we have found that

$$C = \left\langle \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 2 & 2 & 2 \end{pmatrix}, \begin{pmatrix} 0 & 1 & 0 \\ 0 & 2 & 0 \\ 0 & 1 & 0 \\ 2 & 0 & 2 \\ 2 & 2 & 2 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \\ 2 & 0 & 0 \\ 2 & 1 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 2 \\ 1 & 0 & 1 \\ 1 & 1 & 2 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 2 \\ 0 & 2 & 2 \\ 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 2 & 0 & 0 \\ 2 & 1 & 1 \end{pmatrix} \right\rangle.$$

See that the second codeword in the basis has rank 2. Thus, $d(C) = 2$. So, we have found an $\mathbb{F}_3 - [5 \times 3, 6, 2]$ MTR code, as desired.

Of course, we need a better way to find all such codes. Now, we will develop a theory to construct a better approach.

Definition 7.0.8. *A frame of $PG(n, \mathbb{K})$ is an ordered tuple of $n+2$ points, no $n+1$ of them are contained in a hyperplane of $PG(n, \mathbb{K})$.*

Theorem 7.0.9. *If $B = \{e_1, \ldots, e_n\}$ is a standard basis for $\mathbb{K}^{n+1}$, then $(p_0, \ldots, p_n, p_{n+1})$ is a frame of $PG(n, K)$ where $e_{n+1} = e_1 + \ldots + e_n$ and $p_i = \langle e_i \rangle$ for $0 \leq i \leq n+1$. This frame is called the standard frame. Moreover, for each frame, we can find a suitable basis with respect to which the frame becomes a standard frame.*

Basically, a frame manages the role of a basis for projective spaces. This will play a crucial role. Consider the homomorphism

$$\psi \;\; GL(n,q) \times GL(m,q) \longrightarrow PGL(nm,q).$$

For a generator $g = (g_1, g_2)$ of $G = GL(n,q) \times GL(m,q)$, we will compute the image of a standard frame $\mathcal{F}$ of $PG(nm-1, q)$. Let $p \in \mathcal{F}$. Now, p is of length nm. So, we can create its $n \times m$ matrix representation, say M_p. Now, consider the image of M_p under g:

$$M_p^g = g_1 M_p g_2 = M_r.$$

Next, transform M_r back to a vector of length nm, call it r. Do this pro-

cess for all points $p \in \mathcal{F}$. Thus, we get $\mathcal{F}^g$. Do this for each generator of G. Put all of them in a list L. Again, we are going to get help from [20]. There is a function $VectorSpaceToElement(IG, X)$ that return to element of IG represented by the subspace spanned by X. Since $S_{n,m}$ lies on $PG(nm-1,q)$, we will apply this function to see each element $l \in L$ as a point in $PG(nm-1,q)$. Evaluate $VectorSpaceToElement(PG(nm-1,q), l)$ for each $l \in L$ and put them in a new list $\tilde{L}$. Elements of $\tilde{L}$ are points in $PG(nm-1,q)$.

Note 7.0.10. *Observe that number of generators of G will divide the list $\tilde{L}$ into blocks. For example, in our simple case, both $GL(5,3)$ and $GL(3,3)$ have 2 generators. Note that a frame in $PG(14,3)$ has 16 points. So, $\tilde{L}$ consists of $16 \times 2 \times 2$ points of $PG(14,3)$ and we can see them as divided into 4 blocks of size 16 each of which represents an image of a frame under one generator of the group $G = GL(5,3) \times GL(3,3)$. In general, if you have k generators of G, then the list $\tilde{L}$ looks like*

$$\tilde{L} = [\mathcal{F}^{g_1} \mid \mathcal{F}^{g_2} \mid \ldots \mid \mathcal{F}^{g_k}]$$

where each of the blocks $\mathcal{F}^{g_k}$ contains $nm+1$ points of $PG(nm-1,q)$.

The whole reason we followed this approach is to use a function in [20]. This function is $ProjectivityByImageOfStandardFrameNC(IG, List)$. After applying this function to each blocks of $\tilde{L}$, we get the set of projectivities, call it $Proj$. After using $H := Group(Proj)$ in [20], we get the projective collineation group in $PGL(nm,q)$ as desired. What we have achieved is stated in the following theorem.

Theorem 7.0.11. *$\mathcal{H} = \psi(G)$ is the subgroup of the setwise stabiliser of the points on the Segre Variety $S_{n,m}$.*

We end this subsection with the following remark that will motivate the algorithm we are going to propose in the next part.

Remark 7.0.12. *Geometrically, we can view rank-metric codes in $\mathbb{F}_q^{n\times m}$ as subspaces in $PG(nm-1,q)$ and $\mathbb{F}_{q^m}$ vector codes as subspaces in $PG(n-1,q^m)$. The set of rank 1 matrices corresponds to Segre variety in the first case, and a subgeometry in the second case. Then, the equivalence of rank-metric codes corresponds to equivalence under the setwise stabiliser of the Segre Variety inside the collineation group. In Chapter 6, we mentioned that Desarguesian spread of $PG(nm-1,q)$ can be formed by the field reduction map on the points of $PG(n-1,q^m)$. So, the set of 1-dimensional $\mathbb{F}_{q^m}$-subspaces corresponds to a Desarguesian spread D and Segre variety is partitioned by them.*

This remark emphasizes the importance of the field reduction map once more.

7.1 The Algorithm Snakes and Ladders

Consider Theorem 7.0.11. Now, the idea is to use this group $\mathcal{H}$ to reduce the total number of candidate subspaces in the original algorithm by taking one subspace from each orbit, i.e., a representative. That way, instead of having to check the required property for each subspace, we only need to verify the property for the representatives.
We have arranged our setup in a way that is suitable to use the Algorithm Snakes and Ladders presented in [1]. The algorithm constructs the orbit representatives level by level, and thus deserves the name given to it. Let us give some definitions.

Definition 7.1.1. *$G//X$ is the set of all G-orbits on X where G is a group such that G acts on the set X.*

We will try to compute the orbits of H on subsets of X.

Definition 7.1.2. *$P_k(X) = \{S \subseteq X \ : \ |S| = k\}$ and $P_{\leq k}(X) = \bigcup_{i=0}^{k} P_i(X)$.*

The following example is crucial to understand how the algorithm can be applied to our case.

Example 7.1.3 (Example 9.5.12 in [1])**.** *Construct and classify all $\mathbb{F}_2 - [8, 4, \geq 3]$ codes. Note that we are skipping the details since they are not related to the task at hand. In [1], they are looking for sets of 8 points in $PG(3,2)$, and say that in order to construct the codes, the orbits of $PGL(4,2)$ on $P_{\leq 8}(PG(3,2))$ should be computed.*

We already have the projectivity group $\mathcal{H}$ constructed in Theorem 7.0.11. Suppose we want the tensor rank to be R. Let $\mathcal{N}$ be the set of points of the Segre variety $S_{n,m}$ in $PG((n+1)(m+1)-1, q) = PG(nm+n+m, q)$. Thus, in order to construct the codes with tensor rank R, we need to compute the orbits of $\mathcal{H}$ on $P_{\leq R}(\mathcal{N})$. We end this thesis with the following question that will hopefully characterize the remaining class of MTR codes and answer the open problem 7.0.3.

Question 7.1.4. *Apply the algorithm Snakes and Ladders with the given parameters above. Once you have $\mathcal{H}//P_{\leq R}(\mathcal{N})$, try to find a relation so that a classification result can be obtained.*

This question can also be regarded as a future work since if answered positively, it will finish the classification of MTR codes.

BIBLIOGRAPHY

[1] A. Betten, M. Braun, H. Fripertinger, A. Kerber, A. Kohnert, and A. Wassermann. *Error-Correcting Linear Codes: Classification by Isometry and Applications (Algorithms and Computation in Mathematics, 18)*. Springer, 2006.

[2] P. Bürgisser, M. Clausen, M.A. Shokrollahi, and T. Lickteig. *Algebraic Complexity Theory (Grundlehren der mathematischen Wissenschaften, 315)*. Springer, 1997.

[3] J. Cramwinckel, E. Roijackers, R. Baart, E. Minkes, L. Ruscio, R. Miller, T. Boothby, C. Tjhai, and D. Joyner. GAP Package GUAVA, Version 3.15. https://www.gap-system.org/Packages/guava.html, 2019.

[4] B. Csajbók, G. Marino, O. Polverino, and F. Zullo. Maximum scattered linear sets and MRD-codes. *Journal of Algebraic Combinatorics*, 46(3-4):517–531, 2017. doi: 10.1007/s10801-017-0762-6.

[5] P. Delsarte. Bilinear forms over a finite field, with applications to coding theory. *Journal of Combinatorial Theory, Series A*, 25(3):226–241, 1978. doi: 10.1016/0097-3165(78)90015-8.

[6] LE. Dickson. On commutative linear algebras in which division is always uniquely possible. *Transactions of the American Mathematical Society*, 7(4):514–514, 1906. doi: 10.1090/s0002-9947-1906-1500764-6.

[7] E. Gabidulin. Theory of Codes with Maximum Rank Distance. *Problemy Peredachi Informatsii*, 21(1):3–16, 1985.

[8] M. Hall. *The Theory of Groups (Dover Books on Mathematics)*. Dover Publications, reprint edition, 2018.

[9] J. Håstad. Tensor rank is NP-complete. *Journal of Algorithms*, 11(4):644–654, 1990. doi: 10.1016/0196-6774(90)90014-6.

[10] M. Kaminski. A lower bound for polynomial multiplication. *Theoretical Computer Science*, 40:319–322, 1985. doi: 10.1016/0304-3975(85)90174-4.

[11] I. Kaplansky. Infinite-dimensional quadratic forms admitting composition. *Proceedings of the American Mathematical Society*, 4(6):956–956, 1953. doi: 10.1090/s0002-9939-1953-0059895-7.

[12] D. Knuth. *Finite Semifields and Projective Planes*. PhD thesis, California Institute of Technology, 1963.

[13] R. Koetter and M. Medard. An algebraic approach to network coding. *IEEE/ACM Transactions on Networking*, 11(5):782–795, 2003. doi: 10.1109/tnet.2003.818197.

[14] R. Koetter, FR. Kschischang, and D. Silva. A Rank-Metric Approach to Error Control in Random Network Coding. *IEEE Transactions on Information Theory*, 54(9):3951–3967, 2008. doi: 10.1109/tit.2008.928291.

[15] JB. Kruskal. Three-way arrays: rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics. *Linear Algebra and its Applications*, 18(2):95–138, 1977. doi: 10.1016/0024-3795(77)90069-6.

[16] M. Lavrauw. Finite semifields and nonsingular tensors. *Designs, Codes and Cryptography*, 68(1-3):205–227, 2012. doi: 10.1007/s10623-012-9710-6.

[17] M. Lavrauw and S. Ball. Arcs in finite projective spaces. *EMS Surveys in Mathematical Sciences*, 6(1):133–172, 2020. doi: 10.4171/emss/33.

[18] M. Lavrauw and O. Polverino. Finite Semifields and Galois Geometry. *Current Research Topics in Galois Geometry*, pages 131–160, 2011.

[19] M. Lavrauw and J. Sheekey. The Tensor Rank of Semifields of Order 16 and 81. *arXiv preprint arXiv:2102.01997*, 2021.

[20] M. Lavrauw, A. Betten, P. Cara, J. De Beule, J. Bamberg, and M. Neunhöffer. FinInG – Finite Incidence Geometry, Version 1.4.1. http://www.fining.org, 2018.

[21] G. Lunardon, R. Trombetti, and Y. Zhou. On kernels and nuclei of rank metric codes. *Journal of Algebraic Combinatorics*, 46(2):313–340, 2017. doi: 10.1007/s10801-017-0755-5.

[22] A Ravagnani. Galois Geometries and their applications. https://sites.google.com/view/galoisgeometriesapplications/seminars/previous-speakers?authuser=0, 09 2020.

[23] A. Ravagnani, A. Neri, E. Byrne, and J. Sheekey. Tensor Representation of Rank-Metric Codes. *SIAM Journal on Applied Algebra and Geometry*, 3(4):614–643, 2019. doi: 10.1137/19m1253964.

[24] J.L. Walker. *Codes and Curves (Student Mathematical Library, Vol. 7)*. Amer Mathematical Society, 2000.

[25] W. Willems, A. Wassermann, M. Kiermaier, and JDL. Cruz. Algebraic structures of MRD codes. *Advances in Mathematics of Communications*, 10(3):499–510, 2016. doi: 10.3934/amc.2016021.

[26] S. Winograd. On Multiplication of 2× 2 Matrices. *Linear Algebra and Its Applications*, 4(4):381–388, 1971.