Universidade de Aveiro
2022

Vanessa Filipa de Sousa Santana

Códigos convolucionais para codificação em rede com múltiplos envios

Convolutional codes for multi-shot network coding

Universidade de Aveiro
2022

Vanessa Filipa de Sousa Santana

Códigos convolucionais para codificação em rede com múltiplos envios

Convolutional codes for multi-shot network coding

Tese apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Doutor em Matemática, Programa Doutoral em Matemática da Universidade de Aveiro, realizada sob a orientação científica da Doutora Maria Raquel Rocha Pinto, Professora Associada do Departamento de Matemática da Universidade de Aveiro e do Doutor Diego Oscar Napp Avelli, Professor Associado do Departamento de Matemática da Universidade de Alicante, Espanha.

o júri

presidente

Prof. Doutor Eduardo Anselmo Ferreira da Silva
Professor Catedrático, Universidade de Aveiro

vogais

Doutor Joan Josep Climent Coloma
Professor Catedrático, Universidade d'Alacant

Doutora Verónica Requena Arevalo
Professora Contratada Doctora, Universidade d'Alacant

Doutora Maria Raquel Rocha Pinto (Orientadora)
Professora Associada, Universidade de Aveiro

Doutor Paolo Vettori
Professor Auxiliar, Universidade de Aveiro

Doutora Marisa Lapa Toste
Professora Adjunta, Escola Superior de Tecnologia e Gestão de Oliveira do Hospital

agradecimentos

Este projeto foi um grande desafio lançado por uma professora por quem tenho muito carinho, a Doutora Enide Andrade, e que desafio foi … Não podia deixar de lhe agradecer.

Trabalhar e estudar ao mesmo tempo não é nada fácil nem para nós nem para os nossos orientadores e eu não podia ter tido mais sorte. A Professora Doutora Maria Raquel Rocha Pinto e Professor Doutor Diego Oscar Napp Avelli, aceitaram-me como sua orientanda e, apesar das dificuldades, nunca desistiram de mim.

Tive também oportunidade de estar inserida num grupo fantástico que me recebeu de braços abertos e me fez sempre sentir em casa, o Doutor Paulo Almeida, a Doutora Marisa Toste, o Doutor Paolo Vettori e a Doutora Rita Simões.

Não podia deixar de agradecer ao Departamento de Matemática, pelos meios disponibilizados para a concretização deste projeto, e ao CIDMA por me ter aceite e por ter permitido que eu fosse partilhando o meu trabalho com todos os membros do grupo.

Por fim queria agradecer às duas pessoas mais importantes da minha vida, o meu marido e a minha filha, os dois melhores abraços e sorrisos do mundo.

palavras-chave

codificação, redes, códigos convolucionais, distância de característica, concatenação.

resumo

Nesta tese, pretendemos mostrar uma visão geral da área de códigos multi-shot na codificação em redes. Para o efeito, iremos rever as abordagens e resultados propostos até agora e apresentar definições um pouco mais gerais de códigos a blocos e códigos convolucionais que permitem uma ampliação das definições de códigos de métrica rank que já existem na literatura. Também apresentamos, dentro desta nova estrutura, a noção de distância de coluna de um código convolucional de métrica rank. Investigamos as suas propriedades e derivamos um limite superior para o valor da mesma, que nos permite estender as noções de MDP e Strongly MDS para os códigos de métrica rank.

Iremos também focar-nos no desenvolvimento de codificadores de canal como mecanismo que permite uma melhor recuperação dos dados perdidos durante o processo de transmissão. Também nos concentramos na construção de novas classes de códigos convolucionais MRD. Em particular, pretendemos estender as construções apresentadas por Napp, Pinto, Rosenthal e Vettori, com o intuito de incrementar o grau do código e, consequentemente, melhorar a sua capacidade corretora.

Como alternativa aos códigos convolucionais de métrica rank, apresentamos um novo esquema usando concatenação de um código convolucional de métrica Hamming (como código externo) e um código a bloco de métrica rank (como um código interno). O código concatenado proposto é definido sobre o corpo finito base, com o intuito de reduzir a complexidade do processo de codificação e decodificação e, além disso, usa a definição mais geral de código de métrica rank, tornando o processo mais natural.

keywords coding, network, convolutional codes, rank distance, concatenation.

abstract

In this thesis, we aim to provide a general overview of the area of multi-shot codes for network coding. We will review the approaches and results proposed so far and present slightly more general definitions of rank metric block and convolutional codes that allows a wider set of rates than the definitions of rank metric codes that exist in the literature. We also present, within this new framework, the notion of column rank distance of a rank metric convolutional code. We investigate it properties and derive an upper-bound that allows us to extend the notions of Maximum Distance Profile and Strongly-Maximum Distance Separable convolutional codes to some rank metric codes analogues.

We focused on the development of channel encoders as a mechanism that allows the recovery of the data lost during the transmission. We also concentrate on the construction of novel classes of MRD convolutional codes. In particular we aim at extending the constructions presented by Napp, Pinto, Rosenthal and Vettori, in order to increase the degree of the code and consequently it error correction capability.

As alternative to rank metric convolutional codes, we present a novel scheme by concatenation of a Hamming metric convolutional code (as outer code) and a rank metric block code (as a inner code). The proposed concatenated code is defined over the base finite field instead of over several extension finite fields and pretend to reduce the complexity of encoding and decoding process and moreover use the more general definition of rank metric code in order to be more natural.

Contents

Notation

$\mathcal{C}$	the code
$\mathbb{F}_q$	finite field
$\mathbb{F}_q[D]$	polynomials with coefficients in $\mathbb{F}_q$
$\mathbb{F}_{q^m}$	extension field
$\mathbb{F}_{q^m}[D]$	polynomials with coefficients in $\mathbb{F}_{q^m}$
$intdeg(A(D))$	internal degree of A(D)
$extdeg(A(D))$	external degree of A(D)
$u, u(D)$	message
$v, v(D)$	codeword
$V, V(D)$	codeword in matrix form
$G, G(D)$	encoder of a block code/convolutional code
$H, H(D)$	parity-check matrix of a block code/convolutional code
G_c^j	truncated sliding generator matrices
H_c^j	truncated sliging parity check matrices
(n, k, δ)	parameters of a convolutional code
(n, k)	parameters of a rank metric block code over extention fields
$(n \times m, k)$	parameters of a rank metric block code over the base field
$(n \times m, k, \delta)$	parameters of a rank metric convolutional code
(n_o, k_o, δ)	parameters of the outer convolutional code
(n_I, k_I)	parameters of the inner rank metric block code
$\mathrm{wt}(v)$	Hamming weight of v
$d_H(a - b)$	Hamming distance between a and b
$d_H(\mathcal{C})$	Hamming distance of a block code
$d_{free}(\mathcal{C})$	the free Hamming distance of a convolutional code
$d_j^c(\mathcal{C})$	the j-th column distance of a convolutional code
$\mathrm{rank}(A)$	rank of the matrix A
$d_{\mathrm{rank}}(A, B)$	rank distance between A and B
$d_{\mathrm{rank}}(\mathcal{C})$	the rank distance of a rank metric block code
$rwt(A(D))$	rank weight of a polynomial matrix A(D)
$d_{\mathrm{SR}}(A(D), B(D))$	sum rank distance between A(D) and B(D)
$d_{\mathrm{SR}}(\mathcal{C})$	the sum rank distance of a rank metric convolutional code
$d_j^{cr}(\mathcal{C})$	the j-th column sum rank distance of a rank metric convolutional code
G_{GC}	Gabidulin codes encoder
A	deficiency channel matrix

Chapter 1

Introduction

In this chapter we present the general ideas of this thesis. We describe the context in which these ideas have been developed and motivate the main research goals pursued in this work. We first start introducing the area of coding theory as it has been studied in the last decades. We then present a more modern subarea called network coding that is devoted to investigate coding theory when the information is being sent through a network. Within this context we describe the main research goal proposed in this thesis, namely, the study of an algebraic framework for the use of convolutional codes for network coding. We conclude this introductory chapter by providing a brief description of each chapter and therefore outlining the structure of the thesis.

1.1 Classical Coding Theory

The information suffer specific transformations during its transmission process from a source to the receiver. This process is represented in the Figure 1.1. When an information source sends a message, a process, which we called source encoder, divides the message into blocks. Each of these blocks is transformed into its digital form , *i.e.*, into a set of symbols that we often call alphabet, forming an algebraic structure, which usually corresponds to a field or a ring. By means of this process, the initial message becomes a source message. After that, a certain amount of redundancy is added by the channel encoder to each block in order to create a longer block that we will call codeword . The set of all possible codewords will form the code.

A codeword is then transmitted over a transmission channel, or simply stored in memory. During the transmission process or during the storage errors can occur. To recover the original message, a channel decoder is activated. This decoder will use the redundancy added to detect and correct the errors, wherever it is possible, and retrieve the most likely codeword. Finalizing the process, a source decoder determines and reconstruct the source message and deliver it to the destination.

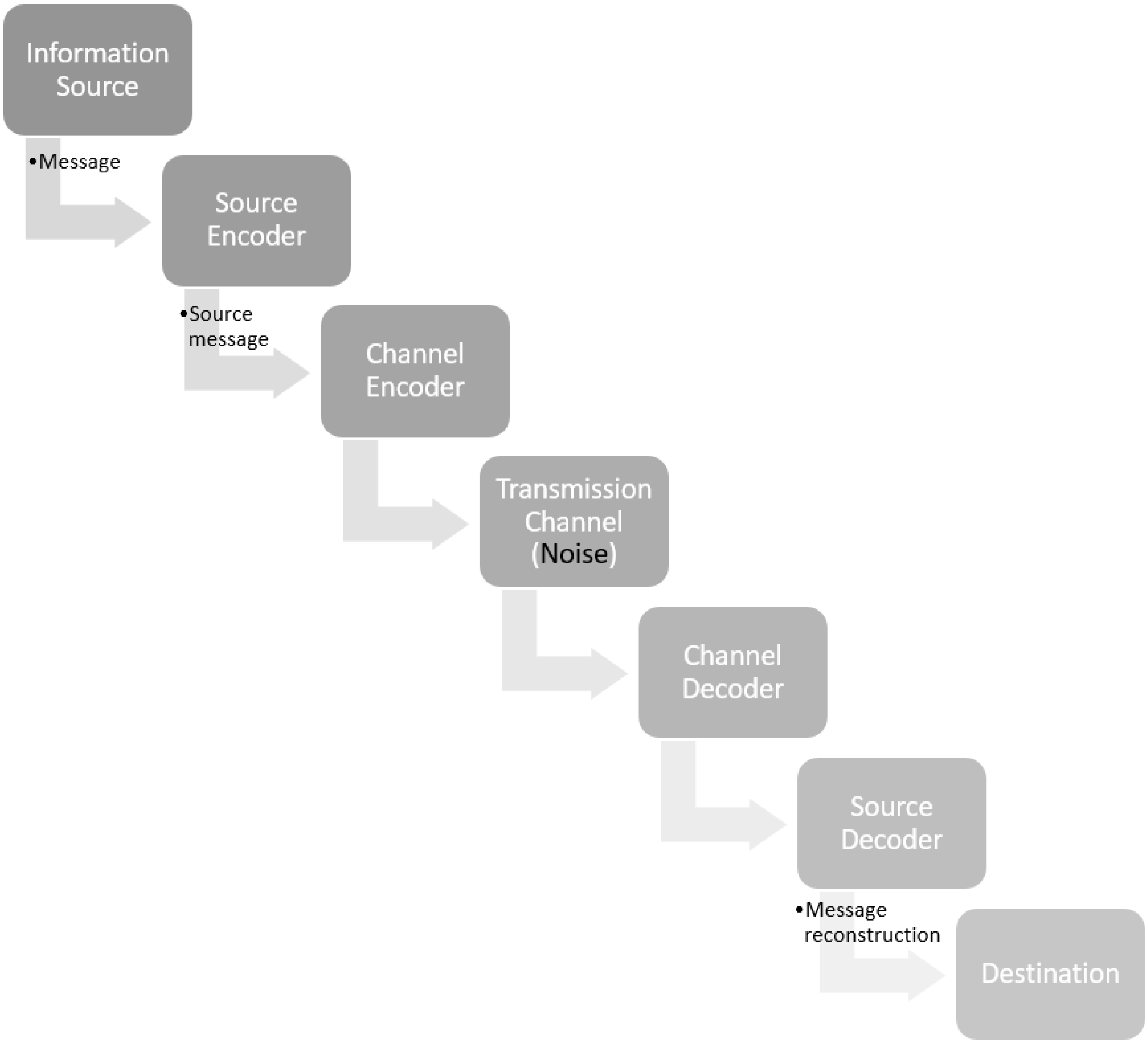

Figure 1.1: Transmission process

In this thesis we focused on the development of channel encoders as a mechanism that allows the recovery of the data against errors that occur during the transmission and assume that the information is already given in terms of elements of a finite field. The aim of coding theory is to develop the methods to detect and correct these errors. For instance, CD players, computer hard drives or video streaming applications would not be possible without the development of coding theory. Error correction codes are used in everyday practical applications and have been the foundation of the revolutionary growth in digital communications and storage.

When we refer to coding theory, we refer to the study of the codes properties and their specific applications. Coding theory is the mathematical theory for algebraic and combinatorial codes that has emerged out of the need for better communication. Its rich inter-dependency with other areas of mathematics and its applications to a number of areas such as cryptography, electrical engineering, and theoretical computer science have brought forward coding theory as a highly important area of applicable discrete mathematics.

The study of error correcting codes started in the late 40's by the hand of Shannon and Hamming, and since then it became a very active area of research. Shannon, Hamming and Golay were the pioneers in the area and developed the first studies and ideas that are still used nowadays. Hamming's codes [13] were the first codes, but many other authors developed variations. The first followers of Hamming were Hocquenghem [14], in 1959, and Bose and Ray-Chaudhari [5], in 1960. They introduced the BCH codes, a generalization of the Hamming codes for multiple-error correction over the binary field. Also in 1960, Reed and Solomon [33] built a class of codes for nonbinary channels, named Reed-Solomon codes. Over the years new codes have been discovered and a well-developed algebraic theory of linear block codes has been developed [6, 15, 24, 35].

The encoding process depends heavily on the type of channel in which the information is being sent. The most studied channel is the q-ary symmetric channel. In this channel, the sender typically sends a vector of bits (zeros and ones) to the receiver. The channel is noisy, *i.e.*, each symbol can be exchanged by other symbol in the alphabet with a probability of p or otherwise is received correctly. This type of channel appear frequently in communication channels such as telephone lines or disk drive storage. In this scenario we have one-sender and one-receiver. However, modern communications often requires the transmission of data from multiple sources to multiple receivers or computers.

There exists two types of error-correcting codes: block codes and convolutional codes. Block codes provide the framework to encode and decode vectors of information of fixed size. Convolutional codes were introduced around the mid-20th century as an alternative to block codes [8]. The main difference between block codes and convolutional codes is the way they encode the data. While block codes encode a fixed number of bits, convolutional encoders of convolutional codes take the string of information bits globally and process a continuous sequence of data. The data that is being encoded at a certain time depends on previous information, *i.e.*, the encoder has memory. In other words, block codes are convolutional codes without memory.

In contrast to block codes which are limited to the transmission of codewords in blocks, convolutional codes can naturally deal with streams of information. When the sender needs to encode a sequence of data, convolutional codes have the advantage that they tend to be much easier to implement than comparable block codes [17]. For this reason they have many practical and interesting applications [17].

1.2 Modern Coding Theory

Contemporary communication and computation environments are network communication channels, *i.e.*, the transmission is sent through a network where there may be many information sources and possibly many receivers. The internet, wireless network communication, and cloud computing are examples of these new communication scenarios where a transmitter sends packets to several users through a series of intermediate nodes.

These relatively new channels are based on certain mathematical models that pose interesting challenges for researchers. Network communication models can be represented using a directed graph carrying an information flow with possibly several information sources and sinks. The seminal paper [22] provided the mathematical foundations for this communication scenario: an algebraic theory of network coding. Network coding has since then became a very active are for research among researchers in mathematics, coding theory, and cryptography.

Traditional approaches to the operation of packet networks treat the data in a very simplistic way by just routing the information along network pathways. The channel that can better illustrate this situation is the Internet. On the Internet, the information is transmitted in the form of packets through a series of nodes or routers that just forward the received packets. The packets have headers and number sequences describing their position within a given stream. Moreover the integrity of the packets is controlled via cyclic redundancy check (CRC) codes. So, the receiver knows exactly when there are the missing or corrupted packets. In order to warrant reliable transmission of data the TCP/IP protocol asks for the retransmission of lost or corrupted packets. This ensures the reception of the correct information by the receiver, however it enlarges the transmission time and has unsatisfactory performance when data need to be transmitted from one server to multiple receivers or in real-time applications such like video-calling.

Network coding challenges this conventional approach and is based on a simple, yet far-reaching, idea: rather than simply routing packets, intermediate nodes are permitted to combine packets in order to add redundancy that allows the error correction of the transmitted packets. The award-winning paper [1] presented the butterfly network example that illustrates how linear network coding can outperform routing and we explain next.

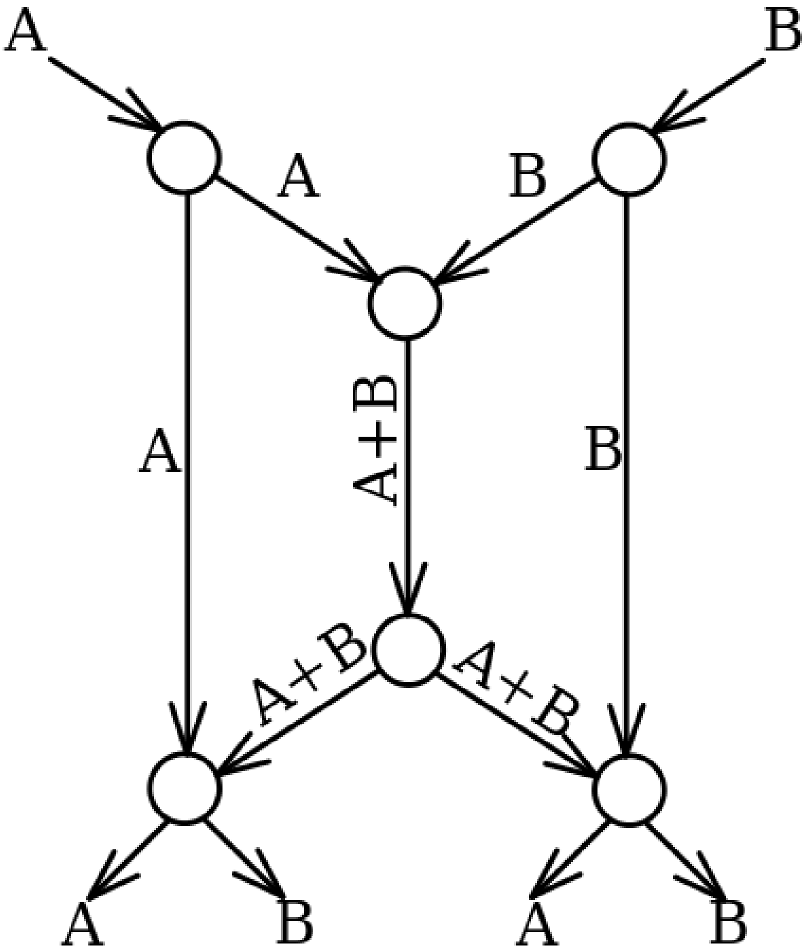

Figure 1.2: The butterfly example

Two source nodes, the senders, see Figure 1.2, have information A and B that must be transmitted to two receivers, *i.e.*, each receiver node must receive both A and B. Each edge can carry only a single value. If only routing were allowed, then the central link would be only able to carry A or B, but not both. Suppose we send A through the center; then the left destination would receive A twice and not know B at all. No routing scheme can transmit both A and B simultaneously to both destinations and therefore it takes four time instances to transmit A and B to the receivers. However, we can reduce this time if we can use a simple code: We just need to encode A and B summing them up and send ”A+B”. That is, A and B can be sent to both destinations simultaneously by transmitting the sum of the symbols through the two relay nodes. The left destination receives A and A + B, and can calculate B by subtracting the two values. Similarly, the right destination will receive B and A + B, and will also be able to determine both A and B. Therefore, with network coding, it takes only three time slots and improves the throughput.

The communication between transmitter and receiver occurs in a series of rounds or ”shots” during each generation where the transmitter injects a number of fixed-length packets into the network, each of which may be regarded as a row vector over a finite field $\mathbb{F}_q$. These packets propagate through the network, possibly passing through a number of intermediate nodes between transmitter and receiver. Whenever an intermediate node has an opportunity to send a packet, it creates an $\mathbb{F}_q$-linear combination of the packets it has available and transmits this random linear combination. Finally, the receiver collects such randomly generated packets and tries to infer the set of packets injected into the network.

Let us denote by $\{p_1, p_2, \dots, p_K\} \subset \mathbb{F}_q^M$ the packets or vectors that are injected into the network. If there are not errors during the transmission process, then the receivers will obtain linear combinations of the p_i, *i.e.*, $y_j = \sum_{i=1}^K a_{j,i}p_i$ where $a_{j,i} \in \mathbb{F}_q$ are coefficients generated by the nodes of the network. If we choose to consider the injection of erroneous packets, denoted by $e_t \in \mathbb{F}_q^M$, $t = 1, \dots T$, this model is enlarged to include error packets as

$$y_i = \sum_{i=1}^{K} a_{j,i}p_i + \sum_{t=1}^{T} s_{j,t}e_t$$

with $s_{j,t} \in \mathbb{F}_q$. In a matrix form, the transmission model may be written as

$$Y = PA + ES$$

where $Y = (y_1, \dots, y_M)$, $P = [p_1 \cdots p_K] \in \mathbb{F}_q^{M \times K}$, $E = [e_1, \dots, e_T]$, $A \in \mathbb{F}_q^{K \times N}$ and $S \in \mathbb{F}_q^{T \times N}$. Note that if no errors occur during the transmission, then the receiver gets a linear combination of the packets (columns of P), *i.e.*, the $\mathbb{F}$-linear subspace generated by the columns of P remains invariant over a noiseless transmission. Hence, in this communication model, a projective geometry, *i.e.*, the set of all subspaces of a finite vector space, serves as communication alphabet. If these subspaces are represented with matrices (which rows or columns represent a basis of such subspace), then the code is equipped with the rank metric in order to correct the errors. This gives rise to the so-called rank metric codes that will be formally defined in Chapter 2. More details on network coding can be found in [12].

1.3 Novel contributions of the Thesis

Most of the existing literature in the area of network coding is concerned with the situation in which the network is used only once to propagate the information, *i.e.*, a fixed number of packets are encoded and sent via the network at one time instant. We call the codes used in such a scenario one-shot network codes. To achieve a reliable communication over network channels, matrix codes can be employed by the hand of the so-called rank metric codes. Rank metric codes such as Gabidulin codes are known to be able to protect packets in such a scenario.

The rank metric codes were introduced by Delsarte in [7] and were further developed by Gabidulin, in 1985 [9], by Ruth, in 1991, and generated a vivid interest among researchers (see for instance [4, 22, 28]) and became very popular in the last decade, due mainly to their application to random network coding [22].

However, if one needs to transmit a lot of information and requires to use the network several instants, then one can improve the error-correction capability of the code by creating correlation among the transmitted data in the different shots. These codes are referred as multi-shot network codes. This new class of codes has recently attracted much attention due to possible application in the fast-growing areas of distributed storage and streaming communications [3, 4, 25, 31] (video traffic has had an explosive growth and already accounts for over 50% of the traffic on networks). As coding techniques for streaming are fundamentally different from the classical ones fascinating new open problems have appeared in the design of multi-shot network codes. In this thesis we propose the use of convolutional codes for this scenario.

The work in [34] was pioneer in this direction by presenting the first class of convolutional codes for multi-shot network coding. However, the results were only valid for unit memory codes and in [4] a new class of convolutional codes was introduced but only for delay-free networks and restricted parameters, see also [3, 25]. In [28], a general framework was proposed without restrictions on the parameters and in [2, 30] a new metric was introduced to cope with delay networks. Also, In [29] some new ideas regarding more efficient Viterbi decoding algorithms were proposed using the idea that one can reduce the number of branch metrics to be calculated in the trellis. All these ideas surely contribute to the maturity of the area of rank metric convolutional codes, which is in its infancy. However, although some preliminary ideas have been developed there is no general theory till date for this area. The main goal of this thesis is the development of a new mathematical theory which will lead to improved design techniques and more efficient convolutional codes in network environments.

1.4 Structure of the Thesis

This thesis is divided into 5 chapters. Then we give a brief outline of the contents of each chapter.

Chapter 2 - Hamming and rank block codes and convolutional codes

This chapter presents some preliminaries about both block and convolutional codes and their Hamming distance properties. Most of the definitions and results were from [9, 17, 19, 36]. Further, we explain that when these codes are used over channels, that can be represented by networks, we need to introduce a different metric, called rank metric. We presented novel definitions and properties of rank metric codes. Some of these results have been already published in [28, 30] and presented in Fifth International Castle Meeting on Coding Theory and Applications (5ICMCTA), Estonia, in 2017 [27].

In particular, we introduce slightly more general definitions of rank metric block and convolutional codes that allows a wider set of rates than the definitions of rank metric codes that exist in the literature. We also present, within this new framework, the notion of column rank distance of a rank metric convolutional code. We investigate their properties and derive new upper-bounds that allows us to extend the notions of Maximum Distance Profile and Strongly-Maximum Distance Separable convolutional codes to some rank metric codes analogues.

Chapter 3 - Constructions of MRD convolutional codes

In this chapter we concentrate on the construction of novel classes of MRD convolutional codes. In particular we aim at extending the constructions presented by Napp, Pinto, Rosenthal and Vettori [28] where the degree of the MRD convolutional codes were restricted to one.

These constructions have been briefly presented by the author of this thesis in three local conferences, namely: in the Systems and Control Group Workshop in Aveiro (Portugal) in 2019, on the Annual meeting of the Center for Research and Development in Mathematics and Applications (CIDMA) in Aveiro (Portugal) in 2020 and on Research summit UA in Aveiro (Portugal) in 2021.

Chapter 4 - Concatenated code

In this chapter we address the problem of building multi-shot codes. We present a concatenation of a convolutional code and a rank metric block code as a alternative approach of rank metric convolutional codes. We present a novel scheme by concatenation of a Hamming metric convolutional code (as outer code) and a rank metric block code (as a inner code). As opposed to the scheme presented in [32] the proposed concatenated code is defined over the base finite field instead of over several extension finite fields. This will reduce the complexity of encoding and decoding process and moreover use the more general definition of rank metric code as presented in Chapter 2.

Chapter 5 - Conclusions

In the last chapter we summarize the main results obtained in our work and discuss some interesting avenues for future investigations.

Chapter 2

Hamming and rank metric block and convolutional codes

In this chapter we introduce both block and convolutional codes and present their basic properties. These codes could be equipped with different metrics depending on the channel in which the information is being sent. In this chapter we study their Hamming and rank distance properties.

The distance of a code measures its capability of error detection and error correction. In the context of the Hamming metric, the free distance is defined for block and convolutional codes. As opposite to block codes, several types of distances can be defined for convolutional codes. The column distance is another important notion of distance of a convolutional code that will be considered. These distances have been thoroughly investigated and their properties have been fully understood. This is not the case within the context of the rank metric. In this chapter we introduce the novel notion of column rank distance and study its properties. We show that it is also possible to derive upper bounds on the column rank distances in a similar way as it is done for the Hamming distance case.

2.1 Hamming metric codes

When we want to transmit digital data over a noisy channel, errors may occur and therefore we need to develop a mechanism allowing recovery against these errors. Normally, the data that we want to transmit is represented by a vector of bits or elements in a finite field. This vector is encoded into a codeword, by adding redundancy. After transmission of this codeword through a channel, the receiver attempts to reconstruct the original sent codeword. It starts by examining the received word (a possibly corrupted version of this codeword), and then makes a decision by selecting, in the set of all possible codewords, the codeword which is more similar to the received word. This process is called the decoding.

The set of all possible codewords is called the code. When the encoding map is linear, the obtained code is also said to be linear. In the case that the code is formed by fixed vectors we are in the context of block codes, whereas when the set of information data is formed by sequences of vectors we are in the context of convolutional codes.

Let $\mathbb{F}_q$ be the finite field constituted by q elements. Next, we formally introduce both block codes and convolutional codes. More details and properties of block codes can be found in [16, 24] and of convolutional codes in [16, 17, 26].

2.1.1 Block codes

A linear *block code* of rate k/n is an $\mathbb{F}_q$-subspace of $\mathbb{F}_q^n$ of dimension k. A full row rank matrix $G \in \mathbb{F}_q^{k\times n}$ such that:

$$\mathcal{C} = \mathrm{Im}_{\mathbb{F}_q} G = \left\{ uG : u \in \mathbb{F}_q^k \right\}$$

is called an *encoder* of $\mathcal{C}$. Two encoders of $\mathcal{C}$ differ by left multiplication by an invertible matrix and are called equivalent encoders. The elements of $\mathcal{C}$ are called *codewords*.

Linear block codes can also be described by means of them*parity-check* matrices. A parity-check matrix of a linear block code $\mathcal{C}$ is an $(n-k)\times k$ full row rank matrix such that

$$\mathcal{C} = \ker_{\mathbb{F}_q} H = \left\{ v \in \mathbb{F}_q^n : Hv^T = 0 \right\}.$$

Parity-check matrices can be used in order to decide whether a specific vector is a codeword of $\mathcal{C}$, as a particular vector v is a codeword of $\mathcal{C}$ if and only if $Hv^T = 0$. For this reason they are commonly used in decoding algorithms.

If a codeword is sent over a noisy channel then some components of the codeword may arrive corrupted. In order to allow a block code to correct such errors we need to define a distance for $\mathcal{C}$. The Hamming distance is a metric for comparing two vectors of data a and b that measures the number of coordinates in which the two vectors differ, denoted by $d_H(a,b)$. Hence,

$$d_H(a,b) = \mathrm{wt}(a-b),$$

where $\mathrm{wt}(a-b)$ is the Hamming weight of the vector $a-b$, i.e., the number of nonzero components of $a-b$.

The free Hamming distance of a block code $\mathcal{C}$ is given by

$$d_H(\mathcal{C}) = \min_{v \in \mathcal{C}, v \neq 0} \mathrm{wt}\left(v\right).$$

When the linear block code is equipped with the Hamming distance we shall refer to it as simply Hamming metric codes. Moreover, if no confusion arises we refer to the free Hamming distance of $\mathcal{C}$ simply as the Hamming distance of $\mathcal{C}$. The Hamming distance is keyed to the error-correcting capability of the code. In fact, after channel transmission, a code can detect s errors in a received word w if the Hamming distance of $\mathcal{C}$ is greater or equal than $s+1$, since this means that w cannot be a sent codeword. Moreover, the code can always correct s errors if the Hamming distance is greater or equal than $2s+1$, because this means that there is a unique codeword that differ from w in s positions. This is stated in the following theorem.

Theorem 2.1. *[24, Theorem 2] Let $\mathcal{C}$ be a code with Hamming distance d. Then,*

1. *$\mathcal{C}$ can always detect $d-1$ errors;*
2. *$\mathcal{C}$ can always correct $\lfloor \frac{d-1}{2} \rfloor$ errors;*
3. *$\mathcal{C}$ can always correct $d-1$ erasures, where erasures are errors whose location is known.*

Note that in an erasure we assume to know the location of the error but not its exactly value.

In this way, the larger the distance the more errors and erasures the code can correct. Hence, the main problem in coding theory is to build codes having the largest possible distance for a given rate k/n together with the creation of an efficient decoding algorithm.

The Hamming distance of a block code of rate k/n is upper bounded by

$$d_H(\mathcal{C}) \leq n - k + 1.$$

This bound is called the *Singleton bound* . A block code of rate k/n whose Hamming distance attains the Singleton bound is called *maximum distance separable* (MDS) code.

2.1.2 Convolutional codes

In this section we consider a different class of linear Hamming codes, the convolutional codes. These codes are constituted by polynomial vectors. Let $\mathbb{F}_q[D]$ be the ring of polynomials with coefficients in a finite field $\mathbb{F}_q$. We start this section by recalling some notions and results on matrices over $\mathbb{F}_q[D]$ that will be useful in the definition of convolutional codes.

Definition 2.2. *A matrix $U(D) \in \mathbb{F}_q[D]^{k\times k}$ is said to be unimodular if it has a polynomial inverse, i.e., if there exists $V(D) \in \mathbb{F}_q[D]^{k\times k}$ such that $U(D)V(D) = V(D)U(D) = I$.*

A matrix $U(D) \in \mathbb{F}_q[D]^{k\times k}$ is unimodular if and only if its determinant belongs to $\mathbb{F}_q \backslash \{0\}$, [10, 20].

Definition 2.3. *Let $A(D) \in \mathbb{F}_q[D]^{k\times n}$ be a full row rank matrix. $A(D)$ is said to be left prime if*

$$A(D) = X(D)\tilde{A}(D),$$

$X(D) \in \mathbb{F}_q[D]^{k\times k}$ and $\tilde{A}(D) \in \mathbb{F}_q[D]^{k\times n}$, then $X(D)$ is unimodular.

Next theorem gives some characterizations of left prime matrices.

Theorem 2.4. *[20] Let $A(D) \in \mathbb{F}_q[D]^{k\times n}$. The following statements are equivalent:*

1. *$A(D)$ is left prime;*
2. *$A(D)$ admits a polynomial right inverse;*
3. *the greatest common divisor of the $k \times k$ minors of $A(D)$ is 1.*

The following result follows immediately from the above theorem.

Corollary 2.5. *Let $A(D) \in \mathbb{F}_q[D]^{k\times n}$ be a left prime matrix and $U(D) \in \mathbb{F}_q[D]^{k\times k}$ be a unimodular matrix. Then $U(D)A(D)$ is left prime.*

Polynomial matrices that differ by left multiplication by unimodular matrices are said to be (left) equivalent. Among equivalent polynomial matrices we will consider the ones that have least sum of the its row degrees. The degree of a row of a polynomial matrix is defined as the maximum degree of the entries of the row.

Definition 2.6. *Let $A(D) \in \mathbb{F}_q[D]^{k\times n}$.*

1. *The internal degree of $A(D)$ is the maximum degree of all $k \times k$ minors of $A(D)$ and it is represented by intdeg$(A(D))$;*

2. *The external degree of $A(D)$ is the sum of the row degrees of $A(D)$, and it is represented by extdeg$(A(D))$.*

It is clear that the internal degree of a polynomial matrix is smaller or equal than its external degree.

Definition 2.7. *Let $A(D) \in \mathbb{F}_q[D]^{k\times n}$ be a full row rank matrix. $A(D)$ is said to be row reduced if intdeg$(A(D))$ =extdeg$(A(D))$.*

The following theorem gives an efficient way to check if a matrix is row reduced.

Theorem 2.8. *[20] Let $A(D) \in \mathbb{F}_q[D]^{k\times n}$ be a full row rank matrix and $[A]^{hc} \in \mathbb{F}_q^{k\times n}$ be the matrix with the i-th row constituted by the coefficients of D^{ν_i}, where ν_i is the row degree of the i-th row of $A(D)$. Then $A(D)$ is row reduced if and only if $[A]^{hc}$ is full row rank.*

Example 2.9. *The matrix*

$$A(D) = \begin{bmatrix} 1+D+D^2 & 1+D & 1+D^2 \\ D & 1+D & 1 \end{bmatrix} \in \mathbb{F}_2[D]^{2\times 3}$$

has row degrees 2 and 1 and it is row reduced because

$$[A]^{hc} = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$$

is full row rank. In fact, extdeg$A(D)) = 2+1 = 3$ and intdeg$A(D)) = 3$ since the full size minors of $A(D)$ are

$$det \begin{bmatrix} 1+D+D^2 & 1+D \\ D & 1+D \end{bmatrix} = 1+D+D^2+D^3,$$

$$det \begin{bmatrix} 1+D+D^2 & 1+D^2 \\ D & 1 \end{bmatrix} = 1+D^2+D^3$$

$$det \begin{bmatrix} 1+D & 1+D^2 \\ 1+D & 1 \end{bmatrix} = D^2+D^3.$$

Next theorems present some results about row reduced matrices.

Theorem 2.10. *[20] Let $A(D) \in \mathbb{F}_q[D]^{k \times n}$ be a full row rank matrix. Then there exists a unimodular matrix $U(D) \in \mathbb{F}_q[D]^{k \times k}$ such that $U(D)A(D)$ is row reduced.*

Theorem 2.11. *[20] Let $A(D), B(D) \in \mathbb{F}_q[D]^{k \times n}$ be two row reduced matrices such that*

$$A(D) = U(D)B(D),$$

for some unimodular matrix $U(D) \in \mathbb{F}_q[D]^{k \times k}$. Then $A(D)$ and $B(D)$ have the same row degrees, up to a permutation.

Next we define and give some results on convolutional codes. There are several possible definitions of convolutional codes, however, in this work, we consider the one provided in [19].

Definition 2.12. *A convolutional code $\mathcal{C}$ of rate k/n is an $\mathbb{F}_q[D]$-submodule of $\mathbb{F}_q[D]^n$ with rank k. If $G(D) \in \mathbb{F}_q[D]^{k \times n}$ is a full row rank matrix such that*

$$\mathcal{C} = \mathrm{Im}_{\mathbb{F}_q[D]} G(D) = \left\{ u(D)G(D) : u(D) \in \mathbb{F}_q[D]^k \right\},$$

then $G(D)$ is called an encoder *of $\mathcal{C}$.*

Any other encoder $\tilde{G}(D)$ of $\mathcal{C}$ differ from $G(D)$ by left multiplication by a unimodular matrix, *i.e.*, $\tilde{G}(D) = U(D)G(D)$, for some unimodular matrix $U(D) \in \mathbb{F}_q[D]^{k \times k}$. Therefore, if $\mathcal{C}$ admits a left prime convolutional encoder then all its encoders are left prime (see Corollary 2.5). Such a code is called *observable* .

A convolutional code always admits a *minimal* encoder, an encoder in row reduced form (see Theorem 2.10). The sum of the row degrees of a minimal encoder attains its minimum among all the encoders of $\mathcal{C}$. Such sum is usually denoted by δ and called the *degree* of $\mathcal{C}$. A rate k/n convolutional code $\mathcal{C}$ of degree δ is called an (n, k, δ) convolutional code [26].

As we mentioned above, the distance is the most important parameter to evaluate the performance of a code and, for the case of convolutional codes the most important distances are the free distance and the column distance [17]. Such distances are again important measures of the capability of error detection and correction of the code: the free distance allows the correction of errors when the complete sequence of information is known and the column distance is used for correction per time intervals.

The *free Hamming distance*, or simply the Hamming distance, of a convolutional code $\mathcal{C}$ is given by

$$d_{free}(\mathcal{C}) = \min_{v(D)\in\mathcal{C},\ v(D)\neq 0} \operatorname{wt}\big(v(D)\big),$$

where $\operatorname{wt}\big(v(D)\big)$ is the Hamming weight of a polynomial vector

$$v(D) = \sum_{i\in\mathbb{N}} v_i D^i \in \mathbb{F}_q[D]^n,$$

defined as

$$\operatorname{wt}\big(v(D)\big) = \sum_{i\in\mathbb{N}} \operatorname{wt}(v_i).$$

Rosenthal and Smarandache [18] showed that the free Hamming distance of an (n, k, δ) convolutional code is upper bounded by

$$d_{free}(\mathcal{C}) \leq (n-k)\left(\left\lfloor \frac{\delta}{k} \right\rfloor + 1\right) + \delta + 1.$$

This bound was called the *generalized Singleton bound*. An (n, k, δ) convolutional code whose free Hamming distance is equal to the generalized Singleton bound is called *maximum distance separable* (MDS) convolutional code [18].

Let us now introduce the notion of column distances of convolutional codes $\mathcal{C}$. As opposite to the block code case, not all convolutional codes admit a representation in terms of the parity-check matrix. Next theorem characterizes the class of convolutional codes that admit such representation.

Theorem 2.13. *[19] Let $\mathcal{C}$ be a convolutional code of rate k/n. Then there exists a full row rank matrix $H(D) \in \mathbb{F}_q[D]^{(n-k)\times n}$ such that:*

$$\mathcal{C} = \ker H(D) = \{w(D) \in \mathbb{F}_q[D]^n \,:\, w(D)^T H(D) = 0\},$$

if and only if $\mathcal{C}$ is observable.

A full row rank matrix $H(D) \in \mathbb{F}_q[D]^{(n-k)\times n}$ such that $\mathcal{C} = \ker H(D)$ is called a *parity-check matrix* of $\mathcal{C}$. From the above theorem it follows that only the observable codes admit parity-check matrices. We will see that these matrices have an important role in the study of the column distances of a convolutional code. Thus, we will restrict the study of column distances to observable convolutional codes.

Let $\mathcal{C}$ be an observable convolutional code. The *j-th column distance*of $\mathcal{C}$, for $j \in \mathbb{N}_0$, is given by

$$d_j^c(\mathcal{C}) = \min\{\mathrm{wt}(v(D)|_{[0,j]}) : v(D) \in \mathcal{C},\ v_0 \neq 0\},$$

where $v(D) = \sum_{i\in\mathbb{N}} v_i D^i$ and $v(D)_{|[0,j]} = \sum_{i=0}^{j} v_i D^i$.

Let

$$G(D) = \sum_{j=0}^{\nu} G_j D^j \in \mathbb{F}_q[D]^{k\times n}, G_j \in \mathbb{F}_q^{k\times n}, G_\nu \neq 0$$

be an encoder of $\mathcal{C}$ and

$$H(D) = \sum_{j=0}^{\mu} H_j D^j \in \mathbb{F}_q[D]^{(n-k)\times n}, H_i \in \mathbb{F}_q^{(n-k)\times n}, H_\mu \neq 0$$

be a parity-check matrix of $\mathcal{C}$. For every $j \in \mathbb{N}_0$, the truncated sliding generator matrices $G_j^c \in \mathbb{F}_q^{(j+1)k\times(j+1)n}$ and the truncated sliding parity-check matrices $H_j^c \in \mathbb{F}_q^{(j+1)(n-k)\times(j+1)n}$ are given by

$$G_j^c = \begin{bmatrix} G_0 & G_1 & \cdots & G_j \\ & G_0 & \cdots & G_{j-1} \\ & & \ddots & \vdots \\ & & & G_0 \end{bmatrix}$$

and

$$H_j^c = \begin{bmatrix} H_0 & & & \\ H_1 & H_0 & & \\ \vdots & \vdots & \ddots & \\ H_j & H_{j-1} & \dots & H_0 \end{bmatrix},$$

respectively. When $j > \nu$, we consider $G_j = 0$ and when $j > \mu$, we define $H_j = 0$.

Note that since any encoder $G(D) = \sum_{j=0}^{\nu} G_j D^j \in \mathbb{F}_q[D]^{k\times n}$ of $\mathcal{C}$ is left prime, it follows that there exists $B(D) \in \mathbb{F}_q[D]^{n\times k}$ such that $G(D)B(D) = I$ (see Theorem 2.4), and therefore $G(0)B(0) = I$, which means that $G_0 = G(0)$ is full row rank. Thus, for every $j \in \mathbb{N}_0$, the *j-th column distance* of $\mathcal{C}$ is given by

$$\begin{aligned} d_j^c(\mathcal{C}) &= \min\{\mathrm{wt}([v_0 v_1 \cdots v_j]) : [v_0 v_1 \cdots v_j] = [u_0 u_1 \cdots u_j] G_j^c, u_i \in \mathbb{F}_q^k, u_0 \neq 0\} \\ &= \min\{\mathrm{wt}(v) : v = (v_0, \dots, v_j) \in \mathbb{F}_q^{(j+1)n}, v(H_j^c)^T = 0, v_0 \neq 0\}. \end{aligned}$$

It is clear that the column distances are invariants of the code, *i.e.*, they do not depend on the encoder that was selected. For the sake of simplicity we sometimes write d_j^c instead of $d_j^c(\mathcal{C})$. The following result provides an upper bound on the column distances of an (n,k,δ) convolutional code. Some other properties related to these bounds are also presented.

Proposition 2.14. *[19, Proposition 2.2] Let $\mathcal{C}$ be an (n,k,δ) convolutional code. For every $j \in \mathbb{N}_0$, we have*

$$d_j^c \leq (n-k)(j+1)+1.$$

Corollary 2.15. *[19, Corollary 2.3] Let $\mathcal{C}$ be an (n,k,δ) convolutional code. If $d_j^c = (n-k)(j+1)+1$ then $d_i^c = (n-k)(i+1)+1$, for every $i \leq j$.*

The following result shows the first possible time instant that an (n,k,δ) convolutional code can achieve the generalized Singleton bound for this class of codes.

Proposition 2.16. *[19, Proposition 2.7] Let $\mathcal{C}$ be an MDS (n,k,δ) convolutional code with column distances d_j^c and free distance d_{free}. Let $M = \min\{j \in \mathbb{N}, d_j^c = d_{free}\}$. Then,*

$$M \geq \left\lfloor \frac{\delta}{k} \right\rfloor + \left\lceil \frac{\delta}{n-k} \right\rceil .$$

These results leads to the following definitions.

Definition 2.17. *[19, Definition 2.8 and 2.9] Let $\mathcal{C}$ be an (n,k,δ) convolutional code and $M = \lfloor \frac{\delta}{k} \rfloor + \lceil \frac{\delta}{n-k} \rceil$. Then, $\mathcal{C}$ is called strongly-MDS, if*

$$d_M^c = (n-k)\left(\left\lfloor \frac{\delta}{k} \right\rfloor + 1\right) + \delta + 1$$

and is said to have a maximum distance profile (MDP) if

$$d_j^c = (n-k)(j+1)+1, \ \textit{for}\ j = 1, \dots, L,$$

where

$$L = \left\lfloor \frac{\delta}{k} \right\rfloor + \left\lfloor \frac{\delta}{n-k} \right\rfloor ,$$

that is

$$L = \begin{cases} M & \textit{if}\ (n-k)|\delta \\ M-1 & \textit{otherwise} \end{cases} .$$

MDP convolutional codes are very appealing for error correction as fast growth of column distances is an attractive property for codes to be used with sequential decoding. The maximal possible growth in the column distances implies that they can correct the maximal number of errors and erasures per time interval. Strongly MDS means that they achieve the maximum error correction capability as fast as possible. This property is particularly useful for low-delay streaming applications.

2.2 Rank metric codes

In the previous section we considered codes that were equipped with the Hamming distance as this is the appropriate distance in the context of one-sender to one-receiver communication channels. In such a channel the Hamming distance characterizes the error correction capabilities of the code.

However, network coding theory is concerned with multicast communications where the data transmission is addressed to a group of destination computers simultaneously and therefore there can be multiple receivers and even multiple senders. In this context codes are not equipped with the Hamming distance anymore but rather with a different metric: the rank metric, see [22] for more details.

This is a comparably fresh area of coding theory that differs from the classical one in that a network takes the role of the traditional single-link communication. As this specific area of coding theory is comparably new, many fundamental results that have been fully understood in the context of the Hamming distance have not yet been intensively investigated or even defined in the rank metric setting.

The theory of network coding developed so far is concerned to large extent with the so-called one-shot network coding, where the coding is performed over one use of the network. In this scenario linear block codes can be used to protect the information sent through the network. However, coding can also be performed over multiple uses of the network, giving rise to the multi-shot network coding. The potential of using multi-shot network coding was already observed in the seminal paper [22]. The general idea stems from the fact that creating dependencies among the transmitted codewords of different shots can improve the error-correction capabilities. In this section we introduce rank metric convolutional codes as a natural class of codes for multi-shot network coding. We will base our work on the definition of rank metric codes developed by [28].

2.2.1 Rank metric block codes

A rank metric code C is defined as any nonempty subset of $\mathbb{F}_q^{n\times m}$. Let $A, B \in \mathbb{F}_q^{n\times m}$. Gabidulin [9] defines *rank distance* between A and B as

$$d_{\text{rank}}(A, B) = \text{rank}(A - B).$$

Then, any subset of $\mathbb{F}_q^{n\times m}$ equipped with this distance is a *rank metric code.*

Although rank metric codes in $\mathbb{F}_q^{n\times m}$ are usually constructed as block codes of length n over the extension field $\mathbb{F}_q^m$ (see Remark 2.19 below). We consider in this thesis a more general construction as defined in [28]. An *$(n \times m, k)$ linear rank metric code* $\mathcal{C} \subset \mathbb{F}_q^{n\times m}$ of rate $k/nm < 1$ is the image of a monomorphism $\varphi : \mathbb{F}_q^k \to \mathbb{F}_q^{n\times m}$. We write $\varphi = \psi \circ \gamma$ as a composition of an isomorphism ψ and a monomorphism γ:

$$\begin{aligned} \varphi : \mathbb{F}_q^k &\xrightarrow{\gamma} \quad \mathbb{F}_q^{nm} \quad \xrightarrow{\psi} \quad \mathbb{F}_q^{n\times m} \\ u &\longmapsto v = uG \longmapsto V = \psi(v), \end{aligned}$$

where $G \in \mathbb{F}_q^{k\times nm}$ is full row rank and the rows of V are simply the n consecutive blocks of v with m elements.

The rank distance of $\mathcal{C}$, $d_{\text{rank}}(\mathcal{C})$, is defined as

$$d_{\text{rank}}(\mathcal{C}) = \min_{U,V\in\mathcal{C}} d_{\text{rank}}\big(U - V\big) = \min_{V\in\mathcal{C},V\neq 0} d_{\text{rank}}\big(V\big),$$

or simply the minimum rank distance between two different codewords. In the following, for the sake of simplicity we will assume that $n \leq m$ (but analogous results can be given for the other case). Linear rank metric codes also have a Singleton-like bound which provides a limit for the value of the code distance.

Theorem 2.18. *[28, Theorem 1] The rank distance of an $(n\times m, k)$ linear rank metric code is upper bounded by*

$$d_{\text{rank}}(\mathcal{C}) \leq n - \left\lfloor \frac{k-1}{m} \right\rfloor = n - \left\lceil \frac{k}{m} \right\rceil + 1.$$

A code $\mathcal{C}$ that attains the Singleton-like bound is called *maximum rank distance code* (MRD code). The first MRD codes over a finite field $\mathbb{F}_q$ have been constructed by Delsarte and Gabidulin [7, 9]. In the literature these codes are often called (generalized) Gabidulin codes.

Remark 2.19. *As mentioned above, linear rank metric codes are typically defined over the extension field $\mathbb{F}_{q^m}$ using an isomorphism ϕ between $\mathbb{F}_{q^m}$ and $\mathbb{F}_q^m$. More concretely, a linear rank metric code is typically defined via*

$$\mathcal{C} = \mathrm{Im}_{\mathbb{F}_q^m} G = \left\{ uG : u \in \mathbb{F}_{q^m}^k \right\} \subset \mathbb{F}_{q^m}^n,$$

with $G \in \mathbb{F}_{q^m}^{k \times n}$. Then, the rank metric code is $\phi(\mathcal{C})$. Gabidulin MRD codes (and most of the existing rank metric codes) are defined within this framework. Note that in this setting the rate is km/nm whereas in the more general framework described above the rate is k/mn.

Example 2.20. *The so-called Gabidulin codes were introduced by Delsarte and Gabidulin in [7, 9] and represent the first general constructions of MRD codes for any n and k. Their generator matrices are defined via the Moore matrices:*

$$G_{GC} = \begin{bmatrix} \alpha_1^{q^0} & \alpha_2^{q^0} & \dots & \alpha_n^{q^0} \\ \alpha_1^{q^1} & \alpha_2^{q^1} & \dots & \alpha_n^{q^1} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_1^{q^{k-1}} & \alpha_2^{q^{k-1}} & \dots & \alpha_n^{q^{k-1}} \end{bmatrix} \in \mathbb{F}_{q^m}^{k \times n},$$

where $\alpha_1, \dots, \alpha_n \in \mathbb{F}_{q^m}$ are linearly independent over $\mathbb{F}_q$. This in particular implies that $m \geq n$. The linear independence of the $\alpha_i's$ over $\mathbb{F}_q$ is equivalent to the linear independence of any k columns of the generator matrix G_{GC} over $\mathbb{F}_{q^m}$ which guarantees that the code is MRD. Note again that using the isomorphic matrix representation we can interpret the codewords of $\mathcal{C}$ as matrices in $\mathbb{F}_q^{m \times n}$.

In this thesis, we will only focus on erasures occurrence. Similarly to the case of Hamming metric codes, the analogue of an erasure in the context of network coding is rank deficiency. Adapting the proposal of [2] to the rank metric code definition presented by [28] this very especial type of errors in the network can be described as follows: Let $v \in \mathbb{F}_q^{nm}$, or equivalently $V \in \mathbb{F}_q^{n \times m}$, be called channel packet, then V represents the n packets of length m to be sent through the network. Following the approach of [4] and consider a rank-deficiency channel for one shot given by

$$x = Av^T,$$

where $x \in \mathbb{F}_q^{nm}$ represents the received packets and A the deficiency channel matrix. The channel matrix A corresponds to the overall linear transformations applied by the network over the base field F_q and it is known by the receiver. For perfect communications we have that $\mathrm{rank}(A) = n$, but channel interference may cause a rank deficient channel matrix. We will call $n - \mathrm{rank}(A)$ the rank deficiency of the channel which, in a practical way, represents the number of packets lost during the transmission.

2.2.2 Rank metric convolutional codes

In this section we will consider rank metric convolutional codes whose codewords are polynomials matrices in $\mathbb{F}_q[D]^{n\times m}$. These codes were introduced in [28] where it was also defined the notion of sum rank distance. We will introduce the definition of column sum rank distances of a rank metric convolutional code and derive an upper bound on this distance.

A *rank metric convolutional code* $\mathcal{C} \subset \mathbb{F}_q[D]^{n\times m}$ is the image of an homomorphism $\varphi : \mathbb{F}_q[D]^k \to \mathbb{F}_q[D]^{n\times m}$. We write $\varphi = \psi \circ \gamma$ as a composition of a monomorphism γ and an isomorphism ψ:

$$\begin{aligned} \varphi : \mathbb{F}_q[D]^k &\xrightarrow{\gamma} \mathbb{F}_q[D]^{nm} \xrightarrow{\psi} \mathbb{F}_q[D]^{n\times m} \\ u(D) &\mapsto v(D) = u(D)G(D) \mapsto V(D), \end{aligned} \tag{2.1}$$

where $G(D) \in \mathbb{F}_q^{k\times nm}$ is a full row rank polynomial matrix, called *encoder* of $\mathcal{C}$. We will consider that the isomorphism ψ is such that $V_{i,j}(D) = v_{mi+j}(D)$, i.e., the rows of $V(D)$ are the n consecutive blocks of $v(D)$ each one with m elements .

Again, two encoders of $\mathcal{C}$ differ by left multiplication by a unimodular matrix and therefore $\mathcal{C}$ always admits minimal encoders (i.e., row reduced encoders).

The degree δ of a rank metric convolutional code $\mathcal{C}$ is the sum of the row degrees of a minimal encoder of $\mathcal{C}$, *i.e.*, the minimum value of the sum of the row degrees of its encoders. A rank metric convolutional code $\mathcal{C}$ is said to be *delay-free* if it has an encoder $G(D)$ with constant term $G(0)$ full row rank. Note that since any other encoder of $\mathcal{C}$, $\tilde{G}(D)$, is such that $\tilde{G}(D) = U(D)G(D)$ for some unimodular matrix $U(D)$, it follows that all encoders of $\mathcal{C}$ have constant term full row rank. This means that if $V(D) = \varphi(u(D))$ for some $u(D) = \sum_{i\in\mathbb{N}_0} u_i D^i$, with $u_\ell \neq 0$ and $u_i = 0$ for $i < \ell$, then the same happens for $V(D) = \sum_{i\in\mathbb{N}_0} V_i D^i$, i.e., $V_\ell \neq 0$ and $V_i = 0$ for $i < \ell$.

A rank metric convolutional code $\mathcal{C}$ of degree δ, defined as in (2.1), is called an $(n \times m, k, \delta)$-rank metric convolutional code.

When dealing with rank metric codes, a different measure of distance must be considered. The *rank weight* of a polynomial matrix $A(D) = \sum_{i\in\mathbb{N}} A_i D^i \in \mathbb{F}_q[D]^{n\times m}$, is given by

$$\operatorname{rwt}\big(A(D)\big) = \sum_{i \in \mathbb{N}} \operatorname{rank}(A_i).$$

If $B(D) = \sum_{i\in\mathbb{N}} B_i D^i \in \mathbb{F}_q[D]^{n\times m}$, the *sum rank distance* between $A(D)$ and $B(D)$ is defined as

$$\begin{aligned} d_{\mathrm{SR}}\big(A(D), B(D)\big) &= \operatorname{rwt}\big(A(D) - B(D)\big) \\ &= \sum_{i \in \mathbb{N}} \operatorname{rank}(A_i - B_i). \end{aligned}$$

Lemma 2.21. *[28, Lemma 2] The sum rank distance d_{SR} is a distance in $\mathbb{F}_q[D]^{n\times m}$.*

Next we will focus on two rank distances definitions for rank metric convolutional codes: the sum rank distance [28] and the column rank distance [27].

The *sum rank distance* of a rank metric convolutional code $\mathcal{C}$ is defined as

$$d_{\mathrm{SR}}(\mathcal{C}) = \min_{V(D),U(D)\in\mathcal{C}, V(D)\neq U(D)} d_{\mathrm{SR}}(V(D), U(D)).$$

As $\mathcal{C}$ is linear, $V(D) - U(D) \in \mathcal{C}$ for any $V(D), U(D) \in \mathcal{C}$, and therefore it follows that

$$d_{\mathrm{SR}}(\mathcal{C}) = \min_{0\neq V(D)\in\mathcal{C}} \operatorname{rwt}\big(V(D)\big).$$

Next theorem establishes an upper bound on the sum rank distance of a rank metric convolutional code. Analogously, as for the free Hamming distance of a convolutional code, this bound is referred as the generalized Singleton bound for rank metric convolutional codes. We will assume that $n \leq m$ for simplicity, but similar results can be given for the case in which $n > m$.

Theorem 2.22. *[28, Theorem 3] Let $\mathcal{C}$ be an $(n \times m, k, \delta)$ rank metric convolutional code. Then, the sum rank distance of $\mathcal{C}$ is upper bounded by*

$$d_{\mathrm{SR}}(\mathcal{C}) \leq n\left(\left\lfloor \frac{\delta}{k} \right\rfloor + 1\right) - \left\lceil \frac{k(\lfloor \frac{\delta}{k} \rfloor + 1) - \delta}{m} \right\rceil + 1. \tag{2.2}$$

This result can also be founded in [28, Theorem 3]. An $(n \times m, k, \delta)$ rank metric convolutional code whose sum rank distance attains the generalized Singleton bound is called *Maximum Rank Distance* (MRD) convolutional code . The minimal encoders of MRD convolutional codes have a well-established set of row degrees as stated in the following lemma.

Corollary 2.23. *[28, Corollary 4]. Let $\mathcal{C}$ be an $(n \times m, k, \delta)$ rank metric convolutional code and $G(D) \in \mathbb{F}_q[D]^{k \times n}$ be a minimal encoder of $\mathcal{C}$. Then $G(D)$ has $k\left(\left\lfloor \frac{\delta}{k} \right\rfloor + 1\right) - \delta$ rows of degree $\left\lfloor \frac{\delta}{k} \right\rfloor$ and $\delta - k\left\lfloor \frac{\delta}{k} \right\rfloor$ rows of degree $\left\lfloor \frac{\delta}{k} \right\rfloor + 1$.*

It is not known the existence of MRD $(n \times m, k, \delta)$ rank metric convolutional codes for any given set of parameters $n, m, k, \delta \in \mathbb{N}$. Napp, Pinto, Rosenthal and Vettori [28] proposed the first construction of $(n \times m, k, \delta)$ MRD rank metric convolutional codes for $m \geq \delta + k$. We will present first the particular case $k = 1$, $n = m$ and $m > \delta$, for clarity, and after that the general case, for any value of k .

Let $A \in \mathbb{F}_q^{m \times m}$ be a matrix with irreducible characteristic polynomial $\chi(\lambda)$. Then the matrices A^i, $i = 0, 1, \ldots, m-1$ are $\mathbb{F}_q$-linearly independent and

$$\mathbb{F}_q[A] = \left\{ \sum_{i=0}^{m-1} u_i A^i : u_i \in \mathbb{F}_q, i = 0, \ldots, m-1 \right\} \cong \mathbb{F}_{q^m}$$

is a field. Moreover, let δ be an integer smaller than m and define the matrix

$$G(D) = \sum_{i=0}^{\delta} \psi^{-1}(A^i) D^i \in \mathbb{F}_q[D]^{1 \times m^2}. \tag{2.3}$$

Then $G(D)$ is an encoder of an MRD $(m \times m, 1, \delta)$ rank metric convolutional code [28].

Remark 2.24. *An $(n \times m, 1, \delta)$ MRD rank metric convolutional, with $n < m$, can be easily constructed following the same idea as the construction above, by multiplying the matrices A^i, $i = 0, 1, \ldots, m-1$ by a full row rank matrix $X \in \mathbb{F}_q^{n \times m}$, in the definition of the encoder $G(D)$. More precisely, the rank metric convolutional code with encoder*

$$G(D) = \sum_{i=0}^{\delta} \psi^{-1}(XA^i) D^i \in \mathbb{F}_q[D]^{1 \times m^2},$$

with $X \in \mathbb{F}_q^{n \times m}$ a full row rank matrix, is an $(n \times m, 1, \delta)$ MRD rank metric convolutional.

The following example shows the construction of an $(4 \times 4, 1, 3)$ MRD convolutional code.

Example 2.25. *Consider the companion matrix A of the irreducible polynomial $\chi(\lambda) = \lambda^4 + \lambda + 1 \in \mathbb{F}_2[\lambda]$, i.e.,*

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \end{bmatrix} \in \mathbb{F}_2^{4\times 4}.$$

The rank metric convolutional code with encoder

$$G(D) = G_0 + G_1 D + G_2 D^2 + G_3 D^3$$

with

$$G_0 = \psi^{-1}(I) = \left[\begin{array}{cccc|cccc|cccc|cccc} 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \end{array}\right],$$

$$G_1 = \psi^{-1}(A) = \left[\begin{array}{cccc|cccc|cccc|cccc} 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \end{array}\right],$$

$$G_2 = \psi^{-1}(A^2) = \left[\begin{array}{cccc|cccc|cccc|cccc} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \end{array}\right],$$

and

$$G_3 = \psi^{-1}(A^3) = \left[\begin{array}{cccc|cccc|cccc|cccc} 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \end{array}\right],$$

is an $(4 \times 4, 1, 3)$ MRD convolutional code.

In the same paper [28], the authors show a more general construction of an $(n \times m, k, \delta)$ MRD convolutional codes with $m \geq \delta + k$. For that they still consider a matrix $A \in \mathbb{F}_q^{m\times m}$ with irreducible characteristic polynomial $\chi(\lambda)$ and a full row rank matrix $X \in \mathbb{F}_q^{n\times m}$. The matrix

$$G(D) = \sum_{i=0}^{\lfloor \frac{\delta}{k} \rfloor + 1} G_i D^i \in \mathbb{F}_q[D]^{k\times nm} \tag{2.4}$$

with

$$G_i = \begin{bmatrix} \psi^{-1}(XA^{ki}) \\ \psi^{-1}(XA^{ki+1}) \\ \vdots \\ \psi^{-1}(XA^{ki+k-1}) \end{bmatrix}, \ 0 \leq i \leq \left\lfloor \frac{\delta}{k} \right\rfloor,$$

and

$$G_{\lfloor \frac{\delta}{k} \rfloor + 1} = \begin{cases} 0 & \text{if } k \text{ divides } \delta, \\ \begin{bmatrix} \psi^{-1}(XA^{k\lfloor \frac{\delta}{k} \rfloor + k}) \\ \vdots \\ \psi^{-1}(XA^{k+\delta-1}) \\ 0 \\ \vdots \\ 0 \end{bmatrix} & \text{otherwise.} \end{cases} \tag{2.5}$$

is an encoder of and $(n \times m, k, \delta)$ MRD rank metric convolutional code, when $m \geq \delta + k$.

The next example illustrates the above construction.

Example 2.26. *[28] Let us consider the same matrix*

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \end{bmatrix} \in \mathbb{F}_2^{4\times 4}$$

as in Example 2.25, and the full row rank matrix

$$X = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \in \mathbb{F}_2^{3\times 4}.$$

Let $\delta = 2$ (note that $m = 4 \geq \delta + k = 2 + 2$). Note that the XA^i is the matrix constituted by the first 3 rows of A^i. The rank metric convolutional code with encoder $G(D) = G_0 + G_1 D$ where

$$G_0 = \begin{bmatrix} \psi^{-1}(X) \\ \psi^{-1}(XA) \end{bmatrix} = \left[\begin{array}{cccc|cccc|cccc} 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \end{array} \right],$$

$$G_1 = \begin{bmatrix} \psi^{-1}(XA^2) \\ \psi^{-1}(XA^3) \end{bmatrix} = \left[\begin{array}{cccc|cccc|cccc} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \end{array} \right],$$

is a $(3 \times 4, 2, 2)$ MRD convolutional code.

Next we will define a new notion of sum rank distance, called *column rank distance*, which can be seen as the analog of column distance for the rank metric case. We will restrict to $(n \times m, k, \delta)$ delay-free rank metric convolutional codes.

Definition 2.27. *Let $\mathcal{C}$ be an $(n \times m, k, \delta)$ delay-free rank metric convolutional code. For $j \in \mathbb{N}$ we define the j-th column rank distance of $\mathcal{C}$ as*

$$d_j^{cr}(\mathcal{C}) = \min\{\mathrm{rwt}(V(D)_{|[0,j]}) : V(D) \in \mathcal{C} \text{ and } V_0 \neq 0\},$$

where $V(D) = \sum_{i \in \mathbb{N}} V_i D^i$ and $V(D)_{|[0,j]} = \sum_{i=0}^{j} V_i D^i$.

Similarly to the Hamming case, the column rank distances are upper bounded at each time instant j as the following result shows. Again when no confusion arises we write d_j^{cr} for $d_j^{cr}(\mathcal{C})$.

Theorem 2.28. *Let $\mathcal{C}$ be an $(n \times m, k, \delta)$ delay-free rank metric convolutional code. Then, the j-th column rank distance of $\mathcal{C}$ is upper bounded by*

$$d_j^{cr}(\mathcal{C}) \leq j\left(n - \left\lfloor \frac{k}{m} \right\rfloor\right) + n - \left\lfloor \frac{k-1}{m} \right\rfloor.$$

Proof. Let $G(D) = \sum_{i \in \mathbb{N}} G_i D^i$ be an encoder of $\mathcal{C}$. Since G_0 is full row rank (because $\mathcal{C}$ is delay-free) it contains an invertible $k \times k$ submatrix. We will assume, without loss of generality, that the $k \times k$ submatrix of G_0 is constituted by the first k columns.

We will prove the theorem by induction on j. For $j = 0$ let $u_0 \in \mathbb{F}^k$ be such that $v_0 = u_0 G_0$ has the first $k-1$ entries equal to zero, *i.e.*, $\mathrm{wt}(v_0) \leq nm - k + 1$, and let $V_0 = \psi(v_0)$. Then, the first $\left\lfloor \frac{k-1}{m} \right\rfloor$ rows of V_0 are equal to zero and therefore $\mathrm{rwt}(V_0) \leq n - \left\lfloor \frac{k-1}{m} \right\rfloor$ and so $d_0^{cr} \leq n - \left\lfloor \frac{k-1}{m} \right\rfloor$.

Let us suppose now that $d_j^{cr} \leq j\left(n - \left\lfloor \frac{k}{m} \right\rfloor\right) + n - \left\lfloor \frac{k-1}{m} \right\rfloor$ and let us prove that $d_{j+1}^{cr} \leq (j+1)\left(n - \left\lfloor \frac{k}{m} \right\rfloor\right) + n - \left\lfloor \frac{k-1}{m} \right\rfloor$. Let $u(D) \in \mathbb{F}[D]^k$, $v(D) = u(D)G(D)$ and $V(D) = \psi(v(D)) = \sum_{i \in \mathbb{N}} V_i D^i \in \mathcal{C}$ be such that $\mathrm{rwt}(V(D)_{|[0,j]}) = d_j^{cr}$. Moreover, since the $k \times k$ submatrix of G_0 constituted by the first k columns is invertible, we can consider u_{j+1} such that $v_{j+1} = u_{j+1}G_0 + u_{j-1}G_1 + \cdots + u_0 G_{j+1}$ has the first k entries equal to zero. Then,

$$\begin{aligned}
d_{j+1}^{cr} &\leq \mathrm{rwt}((V(D))_{|[0,j+1]}) \\
&= d_j^{cr} + \mathrm{rwt}(V_{j+1}) \\
&\leq j\left(n - \left\lfloor \frac{k}{m} \right\rfloor\right) + n - \left\lfloor \frac{k-1}{m} \right\rfloor + n - \left\lfloor \frac{k}{m} \right\rfloor \\
&= (j+1)\left(n - \left\lfloor \frac{k}{m} \right\rfloor\right) + n - \left\lfloor \frac{k-1}{m} \right\rfloor.
\end{aligned}$$

This concludes the proof. □

With a similar reasoning as in the proof of the above theorem we can prove that if the j-th column rank distance of a rank metric convolutional code achieves the corresponding bound then the same happens for all the i-th column rank distance, for $i < j$.

Theorem 2.29. *Let $\mathcal{C}$ be an $(n \times m, k, \delta)$ delay-free rank metric convolutional code. If $d_j^{cr}(\mathcal{C}) = j\left(n - \lfloor \frac{k}{m} \rfloor\right) + n - \lfloor \frac{k-1}{m} \rfloor$ for some $j \in \mathbb{N}$, then*

$$d_i^{cr}(\mathcal{C}) = i\left(n - \left\lfloor \frac{k}{m} \right\rfloor\right) + n - \left\lfloor \frac{k-1}{m} \right\rfloor,$$

for all $i \leq j$.

Proof. It is enough to prove that $d_j^{cr} = j\left(n - \lfloor \frac{k}{m} \rfloor\right) + n - \lfloor \frac{k-1}{m} \rfloor$ implies that $d_{j-1}^{cr} = (j-1)\left(n - \lfloor \frac{k}{m} \rfloor\right) + n - \lfloor \frac{k-1}{m} \rfloor$. Let us assume that $d_{j-1}^{cr} < (j-1)\left(n - \frac{k}{m}\right) + n - \lfloor \frac{k-1}{m} \rfloor$ and let $u(D) \in \mathbb{F}[D]^k$, $v(D) = u(D)G(D)$ and $V(D) = \text{rmat}_{n \times m}(v(D)) = \sum_{i \in \mathbb{N}} V_i D^i \in \mathcal{C}$ be such $\text{rwt}(V(D))_{|[0,j-1]} = d_{j-1}^{cr}$. Let u_j be such that $v_j = u_0 G_j + u_1 G_{j-1} + \cdots + u_{j-1} G_1 + u_j G_0$ has the first k entries equal to zero. Then, $\text{rank}(V_j) \leq n - \lfloor \frac{k}{m} \rfloor$ and, therefore, $\text{rwt}(V(D)_{[0,j]}) < j\left(n - \lfloor \frac{k}{m} \rfloor\right) + n - \lfloor \frac{k-1}{m} \rfloor$. Consequently, $d_j^{cr} < j\left(n - \lfloor \frac{k}{m} \rfloor\right) + n - \lfloor \frac{k-1}{m} \rfloor$ □

It is obvious that the sequence of column rank distances of the code is nondecreasing and that they cannot grow over the Singleton bound for rank metric convolutional codes given in (2.2). This implies that there exists an integer M such that $d_M^{cr} = d_j^{cr}$, for $j > M$. If the code is MRD then M is precisely determined as stated in the next proposition.

Proposition 2.30. *Let $\mathcal{C}$ be an $(n \times m, k, \delta)$ MRD delay-free rank metric convolutional code with column rank distances d_j^{cr}, for $j \in \mathbb{N}_0$ and sum rank distance d_{SR}. Let $M = \min\{j \in \mathbb{N}, d_j^{cr} = d_{\text{SR}}\}$. Then,*

$$M \geq \left\lceil \frac{n \left\lfloor \frac{\delta}{k} \right\rfloor + \left\lfloor \frac{\delta - k \left\lfloor \frac{\delta}{k} \right\rfloor}{m} \right\rfloor}{n - \left\lfloor \frac{k}{m} \right\rfloor} \right\rceil.$$

Proof. M is such that

$$d_{\text{SR}} = n\left\lfloor \frac{\delta}{k} \right\rfloor + \left\lfloor \frac{\delta - k\left\lfloor \frac{\delta}{k} \right\rfloor}{m} \right\rfloor + n - \left\lfloor \frac{k-1}{m} \right\rfloor = d_M^{cr} \leq M\left(n - \left\lfloor \frac{k}{m} \right\rfloor\right) + n - \left\lfloor \frac{k-1}{m} \right\rfloor.$$

Let $\tilde{M} = \frac{n\left\lfloor\frac{\delta}{k}\right\rfloor + \left\lfloor\frac{\delta - k\left\lfloor\frac{\delta}{k}\right\rfloor}{m}\right\rfloor}{n - \left\lfloor\frac{k}{m}\right\rfloor}$. We will show that $\tilde{M}\left(n - \left\lfloor\frac{k}{m}\right\rfloor\right) + n - \left\lfloor\frac{k-1}{m}\right\rfloor = d_{\mathrm{SR}}$ and therefore $M \geq \lceil\tilde{M}\rceil$.

We will consider two cases, when $m \mid k$ and when $m \nmid k$.

Case 1: $m \mid k$. In this case it holds that

$$\begin{aligned}\tilde{M}\left(n - \left\lfloor\frac{k}{m}\right\rfloor\right) + n - \left\lfloor\frac{k-1}{m}\right\rfloor &= n\left\lfloor\frac{\delta}{k}\right\rfloor + \left\lfloor\frac{\delta - k\left\lfloor\frac{\delta}{k}\right\rfloor}{m}\right\rfloor + n - \left\lfloor\frac{k-1}{m}\right\rfloor \\ &= n\left(\left\lfloor\frac{\delta}{k}\right\rfloor + 1\right) - \frac{k}{m}\left\lfloor\frac{\delta}{k}\right\rfloor + \left\lfloor\frac{\delta}{m}\right\rfloor - \left\lfloor\frac{k-1}{m}\right\rfloor.\end{aligned}$$

Then, since $\left\lfloor\frac{k-1}{m}\right\rfloor = \frac{k}{m} - 1$, we have that

$$\begin{aligned}\tilde{M}\left(n - \left\lfloor\frac{k}{m}\right\rfloor\right) + n - \left\lfloor\frac{k-1}{m}\right\rfloor &= n\left(\left\lfloor\frac{\delta}{k}\right\rfloor + 1\right) - \frac{k}{m}\left(\left\lfloor\frac{\delta}{k}\right\rfloor + 1\right) + \left\lfloor\frac{\delta}{m}\right\rfloor + 1 \\ &= n\left(\left\lfloor\frac{\delta}{k}\right\rfloor + 1\right) - \left\lceil\frac{k(\left\lfloor\frac{\delta}{k}\right\rfloor + 1) - \delta}{m}\right\rceil + 1 \\ &= d_{\mathrm{SR}}.\end{aligned}$$

Case 2: $m \nmid k$. In this case it follows that

$$\begin{aligned}\tilde{M}\left(n - \left\lfloor\frac{k}{m}\right\rfloor\right) + n - \left\lfloor\frac{k-1}{m}\right\rfloor &= n\left(\left\lfloor\frac{\delta}{k}\right\rfloor + 1\right) + \left\lfloor\frac{\delta - k\left\lfloor\frac{\delta}{k}\right\rfloor}{m}\right\rfloor - \left\lfloor\frac{k-1}{m}\right\rfloor \\ &= n\left(\left\lfloor\frac{\delta}{k}\right\rfloor + 1\right) - \left\lceil\frac{k\left(\left\lfloor\frac{\delta}{k}\right\rfloor + 1\right) - \delta - k}{m}\right\rceil - \left\lfloor\frac{k-1}{m}\right\rfloor\end{aligned}$$

$$\begin{aligned}&= n\left(\left\lfloor\frac{\delta}{k}\right\rfloor + 1\right) - \left(\left\lceil\frac{k\left(\left\lfloor\frac{\delta}{k}\right\rfloor + 1\right) - \delta}{m}\right\rceil - \left\lceil\frac{k}{m}\right\rceil\right) - \left\lfloor\frac{k-1}{m}\right\rfloor \\ &= n\left(\left\lfloor\frac{\delta}{k}\right\rfloor + 1\right) - \left\lceil\frac{k\left(\left\lfloor\frac{\delta}{k}\right\rfloor + 1\right) - \delta}{m}\right\rceil + 1 \\ &= d_{\mathrm{SR}},\end{aligned}$$

because $\left\lceil\frac{k}{m}\right\rceil = \left\lfloor\frac{k}{m}\right\rfloor - 1$ and $\left\lfloor\frac{k}{m}\right\rfloor = \left\lfloor\frac{k-1}{m}\right\rfloor$. Hence, in both cases we obtain that $M = \lceil\tilde{M}\rceil$. □

Definition 2.31. *Let $\mathcal{C}$ be an $(n \times m, k, \delta)$-rank metric convolutional code. $\mathcal{C}$ is called* Strongly MRD *(sMRD) if*

$$d_M^{cr}(\mathcal{C}) = d_{\mathrm{SR}}(\mathcal{C}),$$

where

$$M = \left\lceil \frac{n\left\lfloor \frac{\delta}{k}\right\rfloor + \left\lfloor \frac{\delta - k\left\lfloor \frac{\delta}{k}\right\rfloor}{m}\right\rfloor}{n - \left\lfloor \frac{k}{m}\right\rfloor} \right\rceil.$$

Thus, sMRD codes are MRD codes. For $(n \times m, k, \delta)$ rank metric convolutional codes such that $k = 1$ or $m \geq \delta + k$ the above definition has the following form.

Definition 2.32. *Let $\mathcal{C}$ be an $(n \times m, k, \delta)$-rank metric convolutional code. Then*

1. *if $k = 1$, $\mathcal{C}$ is Strongly MRD if*

$$d_M^{cr}(\mathcal{C}) = n(\delta + 1),$$

where $M = \delta$.

2. *if $m \geq \delta + k$, $\mathcal{C}$ is Strongly MRD if*

$$d_M^{cr}(\mathcal{C}) = n\left(\left\lfloor \frac{\delta}{k}\right\rfloor + 1\right),$$

where $M = \left\lfloor \frac{\delta}{k}\right\rfloor$.

The rank metric convolutional codes defined in (2.3) and (2.4) are sMRD codes as shown in the next theorems.

Theorem 2.33. *Let $A \in \mathbb{F}_q^{m\times m}$ be a matrix with irreducible characteristic polynomial $\chi(\lambda)$, and $\mathcal{C}$ the $(m \times m, 1, \delta)$ rank metric convolutional code with encoder*

$$G(D) = \sum_{i=0}^{\delta} \psi^{-1}(A^i)D^i \in \mathbb{F}_q[D]^{1\times m^2}.$$

Then $\mathcal{C}$ is an sMRD rank metric convolutional code.

Proof. Note that since $G_0 \neq 0$, $G(D)$ is delay-free. To show that $\mathcal{C}$ is sMRD we have to prove that $d_M^{cr}(\mathcal{C}) = m(\delta + 1)$, for $M = \delta$. Let us consider a message $u(D) = \sum_{i\in\mathbb{N}_0} u_i D^i \in \mathbb{F}_q[D]$ with $u_0 \neq 0$ and $V(D) = \varphi(u(D))$. Then

$$V(D)_{|[0,\delta]} = \sum_{i=0}^{\delta} V_i D^i,$$

where

$$V_i = \sum_{j=0}^{i} u_{i-j} A^j,$$

$i = 0, 1, \ldots, \delta$. Thus V_i is a nonzero element of $\mathbb{F}_q[A]$ (since it is a nontrivial linear combination of $I, A, \ldots, A^{m-1}$) and therefore is invertible, which means that $\text{rank}(V_i) = m$ and therefore $\text{rwt}(V_{|[0,\delta]}) = m(\delta + 1)$. So we conclude that $d^{cr}_M(\mathcal{C}) = m(\delta + 1)$, for $M = \delta$ and therefore $\mathcal{C}$ is sMRD. □

Next we will show that the rank metric convolutional code with encoder defined in (2.4) is an sMRD code. First we prove that such code is delay-free.

Lemma 2.34. *Let $A \in \mathbb{F}_q^{m\times m}$ be a matrix with irreducible characteristic polynomial $\chi(\lambda)$, $X \in \mathbb{F}_q[D]^{n\times m}$ a full row rank matrix and $\mathcal{C}$ the $(n \times m, k, \delta)$ rank metric convolutional code (where $m \geq \delta + k$) with encoder*

$$G(D) = \sum_{i=0}^{\lfloor \frac{\delta}{k} \rfloor + 1} G_i D^i \in \mathbb{F}_q[D]^{k\times nm},$$

with

$$G_i = \begin{bmatrix} \psi^{-1}(XA^{ki}) \\ \psi^{-1}(XA^{ki+1}) \\ \vdots \\ \psi^{-1}(XA^{ki+k-1}) \end{bmatrix}, \ 0 \leq i \leq \left\lfloor \frac{\delta}{k} \right\rfloor, \ \textit{and}$$

$$G_{\lfloor \frac{\delta}{k} \rfloor + 1} = \begin{cases} 0 & \textit{if } k \textit{ divides } \delta, \\ \begin{bmatrix} \psi^{-1}(XA^{k\lfloor \frac{\delta}{k} \rfloor + k}) \\ \vdots \\ \psi^{-1}(XA^{k+\delta-1}) \\ 0 \\ \vdots \\ 0 \end{bmatrix} & \textit{otherwise.} \end{cases}$$

Then $\mathcal{C}$ is a delay-free rank metric convolutional code.

Proof. Let us assume that G_0 is not full row rank. Then there exists a nonzero vector $\begin{bmatrix} a_0 & a_1 & \cdots & a_{k-1} \end{bmatrix} \in \mathbb{F}_q$ such that

$$\begin{bmatrix} a_0 & a_1 & \cdots & a_{k-1} \end{bmatrix} \begin{bmatrix} \psi^{-1}(X) \\ \psi^{-1}(XA) \\ \vdots \\ \psi^{-1}(XA^{k-1}) \end{bmatrix} = 0.$$

Consequently,

$$a_0X + a_1XA + \cdots + a_{k-1}XA^{k-1} = 0,$$

which implies that

$$a_0I + a_1A + \cdots + a_{k-1}A^{k-1} = 0,$$

and so we conclude that $I, A, \ldots, A^{k-1}$ are not $\mathbb{F}_q$-linearly independent, which is not true. So, we conclude that G_0 is full row rank and therefore $\mathcal{C}$ is delay-free. □

Theorem 2.35. *Let $A \in \mathbb{F}_q^{m\times m}$ be a matrix with irreducible characteristic polynomial $\chi(\lambda)$, $X \in \mathbb{F}_q[D]^{n\times m}$ a full row rank matrix and $\mathcal{C}$ the $(n \times m, k, \delta)$ rank metric convolutional code (where $m \geq \delta + k$) with encoder*

$$G(D) = \sum_{i=0}^{\left\lfloor \frac{\delta}{k} \right\rfloor + 1} G_i D^i \in \mathbb{F}_q[D]^{k\times nm},$$

with

$$G_i = \begin{bmatrix} \psi^{-1}(XA^{ki}) \\ \psi^{-1}(XA^{ki+1}) \\ \vdots \\ \psi^{-1}(XA^{ki+k-1}) \end{bmatrix}, \ 0 \leq i \leq \left\lfloor \frac{\delta}{k} \right\rfloor, \ \textit{and}$$

$$G_{\left\lfloor \frac{\delta}{k} \right\rfloor + 1} = \begin{cases} 0 & \textit{if } k \textit{ divides } \delta, \\ \begin{bmatrix} \psi^{-1}(XA^{k\left\lfloor \frac{\delta}{k} \right\rfloor + k}) \\ \vdots \\ \psi^{-1}(XA^{k+\delta-1}) \\ 0 \\ \vdots \\ 0 \end{bmatrix} & \textit{otherwise.} \end{cases}$$

Then $\mathcal{C}$ is an sMRD rank metric convolutional code.

Proof. Let $M = \left\lfloor \frac{\delta}{k} \right\rfloor$. We have to show that $d_M^{cr}(\mathcal{C}) = n\left(\left\lfloor \frac{\delta}{k} \right\rfloor + 1\right)$. Let $u(D) = \sum_{i\in\mathbb{N}_0} u_iD^i \in \mathbb{F}_q[D]^k$ with $u_0 \neq 0$. Let us represent $u_i = \begin{bmatrix} u_{i,0} & u_{i,1} & \cdots & u_{i,k-1} \end{bmatrix}$ and let $v(D) = u(D)G(D)$ and $V(D) = \varphi(u(D)) = \psi(v(D))$. Then $v(D) = \sum_{i\in\mathbb{N}_0} v_iD^i$ and

$V(D) = \sum_{i \in \mathbb{N}_0} V_i D^i$ are such that

$$\begin{aligned}
v_i &= \sum_{h=0}^{i} u_{i-h} G_h \\
&= \sum_{h=0}^{i} \sum_{l=0}^{k-1} u_{i-h,l} \psi^{-1}(XA^{kh+l}) \\
&= \sum_{h=0}^{i} \sum_{l=kh}^{kh+k-1} u_{i-h,l-kh} \psi^{-1}(XA^{l}) \\
&= \sum_{h=0}^{i} \sum_{l=kh}^{kh+k-1} u_{i-\lfloor \frac{l}{k} \rfloor, l-k\lfloor \frac{l}{k} \rfloor} \psi^{-1}(XA^{l}) \\
&= \sum_{l=0}^{ki+k-1} u_{i-\lfloor \frac{l}{k} \rfloor, l-k\lfloor \frac{l}{k} \rfloor} \psi^{-1}(XA^{l}) \\
&= \psi^{-1}(X \sum_{l=0}^{ki+k-1} u_{i-\lfloor \frac{l}{k} \rfloor, l-k\lfloor \frac{l}{k} \rfloor} A^{l}) \\
&= \psi^{-1}(XB_i),
\end{aligned}$$

where $B_i = \sum_{l=0}^{ki+k-1} u_{i-\lfloor \frac{l}{k} \rfloor, l-k\lfloor \frac{l}{k} \rfloor} A^{l}$, and

$$V_i = XB_i.$$

Note that for $i = 0, 1, \dots, \lfloor \frac{\delta}{k} \rfloor$, B_i is a nontrivial linear combination of some matrices of the form A^j, $j \in \{0, 1, \dots, m-1\}$, since $u_0 \neq 0$. This means that for $i \in \{0, 1, \dots, \lfloor \frac{\delta}{k} \rfloor\}$, B_i is a nonzero element of the field $\mathbb{F}_q[A]$ and therefore is full rank, and consequently XB_i is full row rank, i.e. $\operatorname{rank}(XB_i) = n$. Thus

$$\begin{aligned}
\operatorname{rwt}(V_{|[0, \lfloor \frac{\delta}{k} \rfloor]}) &= \sum_{i=0}^{\lfloor \frac{\delta}{k} \rfloor} \operatorname{rank}(V_i) \\
&= \sum_{i=0}^{\lfloor \frac{\delta}{k} \rfloor} \operatorname{rank}(XB_i) \\
&= n \left(\left\lfloor \frac{\delta}{k} \right\rfloor + 1 \right).
\end{aligned}$$

So we conclude that $d_M^{cr}(\mathcal{C}) = n\left(\lfloor \frac{\delta}{k} \rfloor + 1\right)$ and therefore $\mathcal{C}$ is sMRD. □

Let $\mathcal{C}$ be an $(n \times m, k, \delta)$ sMRD rank metric convolutional code. From the proof of the above theorem we have that if $\tilde{M} = \frac{n\lfloor \frac{\delta}{k} \rfloor + \left\lfloor \frac{\delta - k\lfloor \frac{\delta}{k} \rfloor}{m} \right\rfloor}{n - \lfloor \frac{k}{m} \rfloor}$ is an integer, i.e., if $\tilde{M} = M$, then

$$d_M^c = d_{\mathrm{SR}} = M\left(n - \left\lfloor\frac{k}{m}\right\rfloor\right) + n - \left\lfloor\frac{k}{m}\right\rfloor + 1,$$

and therefore, by Theorem 2.28, we conclude that all column rank distances attain the optimal value, i.e.,

$$d_j^{cr} = \begin{cases} j\left(n - \left\lfloor\frac{k}{m}\right\rfloor\right) + n - \left\lfloor\frac{k}{m}\right\rfloor + 1 & \text{for } j < M \\ M\left(n - \left\lfloor\frac{k}{m}\right\rfloor\right) + n - \left\lfloor\frac{k}{m}\right\rfloor + 1 & \text{for } j \geq M \end{cases},$$

In case $\tilde{M}$ is not an integer we have that $d_M^{cr} < M\left(n - \left\lfloor\frac{k}{m}\right\rfloor\right) + n - \left\lfloor\frac{k}{m}\right\rfloor + 1$ and therefore the i-th column distances, for $i < M$, may not attain their optimal value. Next we consider rank metric convolutional codes that have optimal column rank distance as long as it is possible.

Definition 2.36. *Let $\mathcal{C}$ be an $(n \times m, k, \delta)$-rank metric convolutional code. $\mathcal{C}$ is called* Maximum Rank Profile *(MRP) if*

$$d_j^{cr}(\mathcal{C}) = j\left(n - \left\lfloor\frac{k}{m}\right\rfloor\right) + n - \left\lfloor\frac{k-1}{m}\right\rfloor,$$

for $j = 1, 2, \ldots, L$ where

$$L = \left\lfloor \frac{n\left\lfloor\frac{\delta}{k}\right\rfloor + \left\lfloor\frac{\delta - k\left\lfloor\frac{\delta}{k}\right\rfloor}{m}\right\rfloor}{n - \left\lfloor\frac{k}{m}\right\rfloor} \right\rfloor.$$

Note that $L = M$ or $L = M - 1$. The following lemma is an immediate consequence of Definitions 2.32 and 2.36.

Lemma 2.37. *Let $\mathcal{C}$ be an $(n \times m, k, \delta)$-rank metric convolutional code such that $L = M$. The $\mathcal{C}$ is is sMRD if and only if it is MRP.*

The next result in an immediate consequence of the above lemma.

Corollary 2.38. *Let $\mathcal{C}$ be an $(n \times m, k, \delta)$-rank metric convolutional code. If $k = 1$ or $m \geq \delta + k$, then $\mathcal{C}$ is sMRD if and only if it is MRD.*

Proof. If $\delta = 0$, $M = L = 0$ (this is the trivial situation in that $\mathcal{C}$ is a rank metric code).

Let us assume that $\delta > 0$. If $k = 1$, then $M = L = n\delta$. If $m \geq \delta + k$ we have that $\left\lfloor\frac{k}{m}\right\rfloor = 0$. On the other hand, $\delta = k\left\lfloor\frac{\delta}{k}\right\rfloor + r$ where $0 \leq r < k$ implies that $0 \leq \delta - \left\lfloor\frac{\delta}{k}\right\rfloor < k$

and consequently $\left\lfloor \frac{\delta - k\left\lfloor \frac{\delta}{k}\right\rfloor}{m} \right\rfloor = 0$. This yields $\tilde{M} = \left\lfloor \frac{n\left\lfloor \frac{\delta}{k}\right\rfloor + \left\lfloor \frac{\delta - k\left\lfloor \frac{\delta}{k}\right\rfloor}{m}\right\rfloor}{n - \left\lfloor \frac{k}{m}\right\rfloor} \right\rfloor = \frac{n\left\lfloor \frac{\delta}{k}\right\rfloor}{n} = n\left\lfloor \frac{\delta}{k}\right\rfloor$ and therefore $M = L = n\left\lfloor \frac{\delta}{k}\right\rfloor$. □

We immediately conclude that the rank metric convolutional codes defined in (2.3) and (2.4) are also MRP.

Although general decoding algorithms are not analyzed in this thesis, for the case of rank metric convolutional codes, that it is an interesting topic of research that is left for future work.

Chapter 3

Constructions of MRD convolutional codes

In this chapter we will present novel constructions of MRD rank metric convolutional codes. As referred in Chapter 2, the only known constructions of $(n \times m, k, \delta)$ MRD convolutional codes are the ones defined in (2.4) (and in (2.3) for the particular case $k = 1$). These constructions were proposed by Napp, Pinto, Rosenthal and Vettori [28], and are restricted to codes such that $\delta \leq m - k$. Inspired by their work, we define new constructions of $(n \times m, k, \tilde{\delta})$ MRD convolutional codes with larger values for the code degree. The proposed constructions are based on the idea of extending the encoders of the constructions defined in (2.4) by adding terms of higher degree which coefficients are obtained from coefficients of lower degree. We will first consider the case $k = 1$ for clarity and after that present the more general case.

As we saw in previous chapters the degree δ of the code has influence on the distance that a code can achieve. Therefore, larger values of the code degree may lead to larger distances and, consequently, to the increase of the correcting capability of the code.

3.1 Construction 1

In this section we propose a new construction of an $(m \times m, 1, \delta)$ MRD rank metric convolutional code that fulfills the condition $\delta \leq 2m - 1$. For the purpose, we will define an encoder of the code which can be seen as an extension of the encoder of an $(m \times m, k, \delta)$ MRD convolutional as defined in (2.3), which is presented next.

Let $A \in \mathbb{F}_q^{m \times m}$ be a matrix with irreducible characteristic polynomial. Recall that

$$\mathbb{F}_q[A] = \left\{ \sum_{i=0}^{m-1} u_i A^i \, : \, u_i \in \mathbb{F}_q, i = 0, 1, \ldots, m-1 \right\}$$

is a field. Moreover, since $I, A, \ldots, A^{m-1}$ are linearly independent, any nontrivial linear combination of $I, A, \ldots, A^{m-1}$ is full rank.

Theorem 3.1. *Let $A \in \mathbb{F}_q^{m\times m}$ be a matrix with irreducible characteristic polynomial and δ be an integer smaller than $2m$. The $(m \times m, 1, \delta)$ convolutional code $\mathcal{C}$ with encoder*

$$G(D) = \sum_{i=0}^{m-1} \psi^{-1}(A^i)D^i + \sum_{i=0}^{\delta-m} \psi^{-1}(A^{m-1-i})D^{m+i} \in \mathbb{F}_q[D]^{1\times m^2}, m \leq \delta \leq 2m-1. \quad (3.1)$$

is MRD.

Proof. It is clear that $G(D)$ is row reduced since $G_\delta \neq 0$, and therefore $\mathcal{C}$ has degree δ. To prove that $\mathcal{C}$ is MRD we have to show that $d_{\mathrm{SR}}(\mathcal{C}) = m(\delta+1)$.

Let $u(D) = \sum_{i=0}^{l} u_i D^i \in \mathbb{F}_q[D]^k, l \in \mathbb{N}$, be a nonzero vector and $V(D) = \varphi(u(D)) \in \mathbb{F}_q[D]^{n\times m}$. Let us see that $\operatorname{rank}(V(D)) \geq m(\delta+1)$.

Without loss of generality, we can assume that $u_0 \neq 0$. If $u(D)$ has degree zero, i.e., if $l = 0$, then $V_j = u_0 A^j$, for $0 \leq j \leq m-1$ and $V_j = u_0 A^{2m-j-1}$, for $m-1 \leq j \leq \delta$, and therefore V_j is full rank for $0 \leq j \leq \delta$, which means that $\operatorname{rwt}(V(D)) = m(\delta+1)$.

Let us now assume that $l > 0$. It follows that $V_j = \sum_{i=0}^{j} u_{j-i}A^i$ with $0 \leq j \leq m-1$ is a nonzero element of $\mathbb{F}_q[A]$, i.e., V_j is full rank. So, we have that

$$\sum_{j=0}^{m-1} rank(V_j) = m^2.$$

Let us now consider V_j, $m \leq j \leq \delta$. Note that $V_j = 0$ or V_j is full row rank for $j = m, \ldots, \delta$, since they are elements of $\mathbb{F}_q[A]$. Thus, if $V_j \neq 0$ for $m \leq j \leq \delta$, then $\operatorname{rwt}(V(D)) \geq \sum_{j=0}^{\delta} \operatorname{rank}(V_j) = m(\delta+1)$. If this is not the case, let V_j, with $j = m+r$ and $0 \leq r \leq \delta - m - 1$, be the first V_j equal to zero. Then,

$$V_{m+r} = \sum_{i=0}^{m-1} u_{m+r-i}A^i + \sum_{i=0}^{r} u_{r-i}A^{m-1-i} = \sum_{i=0}^{m-1} \hat{u}_i A^i = 0,$$

where

$$\hat{u}_i = u_{m+r-i},$$

for $i = 0, 1, \ldots, m-r-2$ and

$$\hat{u}_i = u_{m+r-i} + u_{r-m+i+1},$$

for $i = m-r-1, m-r, \ldots, m-1$. In particular, $\hat{u}_{m-1-r} = u_0 + u_{2r+1}$. Then it follows that $\hat{u}_i = 0$, for $i = 0, 1, \ldots, m-1$ because $I, A, A^2, \ldots, A^{m-1}$ are linearly independent and therefore, $u_{2r+1} = -u_0$ is nonzero, since $u_0 \neq 0$. Moreover $u_i = 0$, for $i = 2r+2, \ldots, m+r$.

Consequently, if we consider any V_j, with $j = m+r+s$ and $s \leq \delta - (m+r)$, we have that

$$V_{m+r+s} = \sum_{i=0}^{m-1} u_{m+r+s-i}A^i + \sum_{i=0}^{r+s} u_{r+s-i}A^{m-1-i} = \sum_{i=0}^{m-1} \tilde{u}_i A^i,$$

where $\hat{u}_{m-1-r-s} = u_0 + u_{2r+1+2s}$. Since $u_{2r+1+2s} = 0$ it follows that $\hat{u}_{m-1-r-s} \neq 0$ and therefore V_{m+r+1} is full rank.

Therefore, V_j is also full rank, for $m+r+1 \leq j \leq \delta$, and consequently

$$\sum_{j=0}^{\delta} rank(V_j) = m^2 + m(\delta - m) = m\delta.$$

Moreover, $V_{l+\delta} = u_l A^\delta$ is full rank, since $u_l \neq 0$.

Then,

$$\operatorname{rwt}(V(D)) = \sum_{j=0}^{\delta+l} rank(V_j) \geq m(\delta+1).$$

Consequently, $\mathcal{C}$ is MRD. □

Example 3.2. *Consider the companion matrix*

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \end{bmatrix} \in \mathbb{F}_2^{4\times 4}$$

of the irreducible polynomial $\chi(\lambda) = \lambda^4 + \lambda + 1 \in \mathbb{F}_2[\lambda]$.

The rank metric convolutional code with encoder

$$G(D) = G_0 + G_1 D + G_2 D^2 + G_3 D^3 + G_3 D^4 + G_2 D^5 + G_1 D^6 + G_0 D^7,$$

where

$$G_0 = \psi^{-1}(I) = \left[\begin{array}{cccc|cccc|cccc|cccc} 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \end{array}\right],$$

$$G_1 = \psi^{-1}(A) = \left[\begin{array}{cccc|cccc|cccc|cccc} 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \end{array}\right],$$

$$G_2 = \psi^{-1}(A^2) = \left[\begin{array}{cccc|cccc|cccc|cccc} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \end{array}\right],$$

and

$$G_3 = \psi^{-1}(A^3) = \left[\begin{array}{cccc|cccc|cccc|cccc} 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \end{array}\right],$$

is an $(4 \times 4, 1, 7)$ *MRD rank metric convolutional code.*

Note that the construction defined in (2.3) only allowed to obtain $(m \times m, 1, \delta)$ MRD rank metric convolutional codes for $\delta \leq m - 1$. The construction proposed in this section allows to obtain $(m \times m, 1, \delta)$ MRD rank metric convolutional codes for $\tilde{\delta} \leq 2m - 1$.

Remark 3.3. *An* $(n \times m, 1, \delta)$ *MRD rank metric convolutional, with* $n \leq m$ *and* $\delta \leq 2m - 1$*, can be constructed by multiplying the matrices* A^i*,* $i = 0, 1, \dots, m-1$*, in the definition of the encoder* $G(D)$ *in (3.1) by a full row rank matrix* $X \in \mathbb{F}_q^{n \times m}$*, i.e., the rank metric convolutional code with encoder*

$$G(D) = \sum_{i=0}^{m-1} \psi^{-1}(XA^i)D^i + \sum_{i=0}^{\delta-m} \psi^{-1}(XA^{m-1-i})D^{m+i} \in \mathbb{F}_q[D]^{1 \times nm},$$

is an $(n \times m, 1, \tilde{\delta})$ *MRD rank metric convolutional.*

3.2 Construction 2

In this section and in the following one we present more general constructions of $(n \times m, k, \delta)$ MRD convolutional codes for larger values of k than the one presented in Section 3.1. In this section we will consider the case $k|\delta$ and in the next section we will present the opposite case.

Let us consider an $(n \times m, k, \delta)$ MRD convoltutional code as defined in 2.4, with $n \leq m$, $k < nm$, $m \geq \delta + k$ and such that $k|\delta$.

Let $A \in \mathbb{F}_q^{m \times m}$ be a matrix with irreducible characteristic polynomial and $X \in \mathbb{F}_q^{n \times m}$ a full row rank matrix.

Let

$$G_i = \begin{bmatrix} \psi^{-1}(XA^{ki}) \\ \psi^{-1}(XA^{ki+1}) \\ \vdots \\ \psi^{-1}(XA^{ki+k-1}) \end{bmatrix}, \ 0 \leq i \leq \frac{\delta}{k},$$

and

$$G_i = \begin{bmatrix} 0 & & 1 \\ & \iddots & \\ 1 & & 0 \end{bmatrix} G_{2\frac{\delta}{k}+1-i},$$

for $\frac{\delta}{k}+1 \leq i \leq 2\frac{\delta}{k}+1$.

Let us define the polynomial matrix

$$G(D) = \sum_{i=0}^{2\frac{\delta}{k}+1} G_i D^i \in \mathbb{F}_q[D]^{k\times nm}, \tag{3.2}$$

and let $\mathcal{C}$ be the rank metric convolutional code with encoder $G(D)$. Next lemma shows that $\mathcal{C}$ has degree $2\delta+k$.

Lemma 3.4. *Let m, n, k and δ be integers such that $n \leq m$, $k < nm$, $m \geq \delta + k$ and such that $k|\delta$. Let $A \in \mathbb{F}_q^{m\times m}$ be a matrix with irreducible characteristic polynomial and $X \in \mathbb{F}_q^{n\times m}$ a full row rank matrix. Let $\mathcal{C}$ be the rank metric convolutional code with encoder $G(D)$ as defined in (3.2). Then $\mathcal{C}$ is an $(n \times m, k, 2\delta + k)$ rank metric convolutional code.*

Proof. Let us see that $G(D)$ is row reduced. Note that:

$$[G]^{hc} = G_{2\frac{\delta}{k}+1} = \begin{bmatrix} 0 & & 1 \\ & \iddots & \\ 1 & & 0 \end{bmatrix} G_0.$$

Let $\begin{bmatrix} \alpha_0 & \alpha_1 & \cdots & \alpha_{k-1} \end{bmatrix} \in \mathbb{F}_q$ such that:

$$\begin{bmatrix} \alpha_0 & \alpha_1 & \cdots & \alpha_{k-1} \end{bmatrix} G_0 = 0.$$

Then,

$$\alpha_0 X A^0 + \alpha_1 X A^1 + \cdots + \alpha_{k-1} X A^{k-1} = 0,$$

which implies that

$$\alpha_0 A^0 + \alpha_1 A^1 + \cdots + \alpha_{k-1} A^{k-1} = 0,$$

because X is full row rank, and therefore, since $A^0, A^1, \ldots A^{k-1}$ are linearly independent, we have that $\alpha_0 = \alpha_1 = \cdots = \alpha_{k-1} = 0$ and consequently G_0 is full row rank and so it is $[G]^{hc} = G_{2\frac{\delta}{k}+1}$. This means that $G(D)$ is row reduced (see Theorem 2.8). Thus the degree of $\mathcal{C}$ is equal to the external degree of $G(D)$ which is $k\left(2\frac{\delta}{k}+1\right) = 2\delta+k$. □

Next theorem shows that $\mathcal{C}$ is an $(n \times m, k, 2\delta + k)$ MRD rank metric convolutional code.

Theorem 3.5. *Let m, n, k and δ be integers such that $n \leq m$, $k < nm$, $m \geq \delta + k$ and such that $k|\delta$. Let $A \in \mathbb{F}_q^{m \times m}$ be a matrix with irreducible characteristic polynomial and $X \in \mathbb{F}_q^{n \times m}$ a full row rank matrix. Let $\mathcal{C}$ be the rank metric convolutional code with encoder $G(D)$ as defined in (3.2). Then $\mathcal{C}$ is an $(n \times m, k, 2\delta + k)$ MRD rank metric convolutional code.*

Proof. In order to make the statement holds true, we have to prove that

$$d_{SR}(\mathcal{C}) = n\left(\frac{2\delta + k}{k} + 1\right) = 2n\left(\frac{\delta}{k} + 1\right).$$

(see Theorem 2.2). For that, we will show that $d_{\mathrm{SR}}(V(D)) \geq 2n\left(\frac{\delta}{k} + 1\right)$ for any nonzero $V(D) \in \mathcal{C}$. Let $u(D) = \sum_{i=0}^{l} u_i D^i \in \mathbb{F}_d[D]^k$ be a nonzero vector, $v(D) = u(D)G(D) \in \mathbb{F}_q[D]^{nm}$ and $V(D) = \psi(v(D)) \in \mathcal{C}$. We can assume, without loss of generality, that $u_0 \neq 0$.

We will first consider the case in which $u(D)$ has degree zero, i.e., $u(D) = u_0$. Then

$$v(D) = \sum_{i=0}^{2\frac{\delta}{k}+1} u_0 G_i D^i,$$

and

$$\begin{aligned}
V(D) &= \psi(\sum_{i=0}^{2\frac{\delta}{k}+1} u_0 G_i D^i) \\
&= \sum_{i=0}^{2\frac{\delta}{k}+1} \psi(u_0 G_i D^i) \\
&= \sum_{i=0}^{\frac{\delta}{k}} (u_0^0 X A^{ki} + u_0^1 X A^{ki+1} + \cdots + u_0^{k-1} X A^{ki+k-1}) D^i + \\
&\quad + \sum_{i=0}^{\frac{\delta}{k}} (u_0^0 X A^{\delta-ki+k-1} + u_0^1 X A^{\delta+ki+k-2} + \cdots + u_0^{k-1} X A^{\delta-k}) D^{\frac{\delta}{k}+i+1} \\
&= X(\sum_{i=0}^{\frac{\delta}{k}} (u_0^0 A^{ki} + u_0^1 A^{ki+1} + \cdots + u_0^{k-1} A^{ki+k-1}) D^i + \\
&\quad + \sum_{i=0}^{\frac{\delta}{k}} (u_0^0 A^{\delta-ki+k-1} + u_0^1 A^{\delta+ki+k-2} + \cdots + u_0^{k-1} A^{\delta-k}) D^{\frac{\delta}{k}+i+1}),
\end{aligned}$$

i.e., $V(D) = \sum_{i=0}^{2\frac{\delta}{k}+1} V_i D^i$ with $V_i = XB_i$ and $i = 0, 1, \dots, 2\frac{\delta}{k} + 1$, where

$$B_i = u_0^0 A^{ki} + u_0^1 A^{ki+1} + \cdots + u_0^{k-1} A^{ki+k-1}, \ \ i = 0, 1, \dots, \frac{\delta}{k},$$

and

$$B_{\frac{\delta}{k}+i+1} = u_0^0 A^{\delta-ki+k-1} + u_0^1 A^{\delta+ki+k-2} + \cdots + u_0^{k-1} A^{\delta-k} \ \ i = 0, 1, \dots, \frac{\delta}{k}.$$

Since $u_0^s \neq 0$ for some $s \in \{0, 1, \dots, k-1\}$, then B_i is a nontrivial linear combination of $I, A, \dots, A^{m-1}$. Therefore B_i is full rank, $i \in \{0, 1, \dots, 2\frac{\delta}{k} + 1\}$, and consequently $V_i = XB_i$ is full row rank for $i \in \{0, 1, \dots, 2\frac{\delta}{k} + 1\}$. Thus,

$$\begin{aligned} \operatorname{rwt}(V(D)) &= \sum_{i=0}^{2\frac{\delta}{k}+1} \operatorname{rank}(V_i) \\ &= n\left(2\frac{\delta}{k} + 2\right) \\ &= 2n\left(\frac{\delta}{k} + 1\right). \end{aligned}$$

Let us now consider any message $u(D) = \sum_{i=0}^{l} u_i D^i$, with $u_0 \neq 0$ and $u_l \neq 0$, with $l > 1$. Let us represent $u_i = \begin{bmatrix} u_i^0 & u_i^1 & \cdots & u_i^{k-1} \end{bmatrix}$ and let $v(D) = u(D)G(D) = \sum_{i \in \mathbb{N}_0} v_i D^i \in \mathbb{F}_q[D]^{nm}$ and $V(D) = \psi(v(D)) = \sum_{i \in \mathbb{N}_0} V_i D^i$. Then,

$$v_i = \sum_{j=0}^{i} u_{i-j} G_j, \ \ 0 \leq i \leq \frac{\delta}{k},$$

and, consequently, using the same reasoning as in the proof of Theorem 2.35,

$$\begin{aligned} V_i &= \psi(v_i) \\ &= \left(\sum_{h=0}^{ki+k-1} u_{i-\lfloor \frac{h}{k} \rfloor}^{h-k\lfloor \frac{h}{k} \rfloor} X A^h \right) \\ &= XB_i, \end{aligned}$$

for $0 \leq i \leq \frac{\delta}{k}$, where $B_i = \sum_{h=0}^{ki+k-1} u_{i-\lfloor \frac{h}{k} \rfloor}^{h-k\lfloor \frac{h}{k} \rfloor} A^h$.

Note that B_i is a nontrivial linear combination of $I, A, A^2, \dots A^{m-1}$, because $u_0^s \neq 0$ for some $s \in \{0, 1, \dots, k-1\}$. Thus, B_i is full rank and consequently $V_i = XB_i$ is full row rank, because X is a full row rank matrix, for $i = 0, 1, \dots, \frac{\delta}{k}$.

The next $\frac{\delta}{k}+1$ vector coefficients of $v(D)$ are defined as

$$\begin{aligned}
v_{\frac{\delta}{k}+i} &= \sum_{j=0}^{\frac{\delta}{k}+i} u_{\frac{\delta}{k}+i-j}G_j \\
&= \sum_{j=0}^{\frac{\delta}{k}-i} u_{\frac{\delta}{k}+i-j}G_j + \sum_{j=\frac{\delta}{k}-i+1}^{\frac{\delta}{k}} u_{\frac{\delta}{k}+i-j}G_j + \sum_{j=\frac{\delta}{k}+1}^{\frac{\delta}{k}+i} u_{\frac{\delta}{k}+i-j}G_j \\
&= \sum_{j=0}^{\frac{\delta}{k}-i} u_{\frac{\delta}{k}+i-j}G_j + \sum_{j=\frac{\delta}{k}-i+1}^{\frac{\delta}{k}} u_{\frac{\delta}{k}+i-j}G_j + \sum_{j=\frac{\delta}{k}+1}^{\frac{\delta}{k}+i} u_{\frac{\delta}{k}+i-j}G_{\frac{\delta}{k}-((\frac{\delta}{k}-j+1)-1)} \\
&= \sum_{j=0}^{\frac{\delta}{k}-i} u_{\frac{\delta}{k}+i-j}G_j + \sum_{j=\frac{\delta}{k}-i+1}^{\frac{\delta}{k}} u_{\frac{\delta}{k}+i-j}G_j + \sum_{j=\frac{\delta}{k}+1}^{\frac{\delta}{k}+i} u_{\frac{\delta}{k}+i-j}\begin{bmatrix}0 & & 1\\ & \iddots & \\ 1 & & 0\end{bmatrix}G_{2\frac{\delta}{k}-j+1} \\
&= \sum_{j=0}^{\frac{\delta}{k}-i} u_{\frac{\delta}{k}+i-j}G_j + \sum_{j=\frac{\delta}{k}-i+1}^{\frac{\delta}{k}} u_{\frac{\delta}{k}+i-j}G_j + \sum_{j=\frac{\delta}{k}-i+1}^{\frac{\delta}{k}} u_{j+i-\frac{\delta}{k}-1}\begin{bmatrix}0 & & 1\\ & \iddots & \\ 1 & & 0\end{bmatrix}G_j \\
&= \sum_{j=0}^{\frac{\delta}{k}-i} u_{\frac{\delta}{k}+i-j}G_j + \sum_{j=0}^{i-1} u_{i+j}G_{\frac{\delta}{k}-j} + \sum_{j=0}^{i-1} u_{i-j+1}\begin{bmatrix}0 & & 1\\ & \iddots & \\ 1 & & 0\end{bmatrix}G_{\frac{\delta}{k}-j} \\
&= \sum_{j=0}^{\frac{\delta}{k}-i} u_{\frac{\delta}{k}+i-j}G_j + \sum_{j=0}^{i-1}(u_{i+j}+\hat{u}_{i-j+1})G_{\frac{\delta}{k}-j},
\end{aligned}$$

where $\hat{u}_{i-j+1} = u_{i-j+1}\begin{bmatrix}0 & & 1\\ & \iddots & \\ 1 & & 0\end{bmatrix} = \begin{bmatrix}u^{k-1}_{i-j+1} & u^{k-2}_{i-j+1} & \cdots & u^{0}_{i-j+1}\end{bmatrix}$, $j = 0, 1, \ldots, i-1$ and $i = 1, 2, \ldots$. Thus,

$$\begin{aligned}
V_{\frac{\delta}{k}+i} &= \sum_{j=0}^{\frac{\delta}{k}-i}(u^0_{\frac{\delta}{k}+i-j}XA^{kj} + u^1_{\frac{\delta}{k}+i-j}XA^{kj+1} + \cdots + u^{k-1}_{\frac{\delta}{k}+i-j}XA^{kj+k-1}) + \\
&+ \sum_{j=0}^{i-1}((u^0_{i+j} + u^{k-1}_{i-j-1})XA^{\delta-kj} + (u^1_{i+j} + u^{k-2}_{i-j-1})XA^{\delta-kj+1} + \cdots + \\
&+ (u^{k-1}_{i+j} + u^0_{i-j-1})XA^{\delta-kj+k-1} \\
&= XB_{\frac{\delta}{k}+i}, \qquad (3.3)
\end{aligned}$$

where

$$\begin{aligned} B_{\frac{\delta}{k}+i} = & \sum_{j=0}^{\frac{\delta}{k}-i}(u^0_{\frac{\delta}{k}+i-j}A^{kj} + u^1_{\frac{\delta}{k}+i-j}A^{kj+1} + \cdots + u^{k-1}_{\frac{\delta}{k}+i-j}A^{kj+k-1}) + \\ & + \sum_{j=0}^{i-1}((u^0_{i+j} + u^{k-1}_{i-j-1})XA^{\delta-kj} + (u^1_{i+j} + u^{k-2}_{i-j-1})XA^{\delta-kj+1} + \cdots + \\ & + (u^{k-1}_{i+j} + u^0_{i-j-1})A^{\delta-kj+k-1}. \end{aligned}$$

If $B_{\frac{\delta}{k}+i} \neq 0$, $i = 1, 2, \ldots, \frac{\delta}{k} + 1$, then $B_{\frac{\delta}{k}+i}$ is full rank because it is an element of $\mathbb{F}_q[A]$, and therefore $V_{\frac{\delta}{k}+i} = XB_{\frac{\delta}{k}+i}$ is full row rank and $\sum_{i=\frac{\delta}{k}+1}^{2\frac{\delta}{k}+1} \operatorname{rank}(V_i) = n(\frac{\delta}{k} + 1)$. So, we have that

$$\operatorname{rwt}(V(D)) \geq \sum_{i=0}^{2\frac{\delta}{k}+1} \operatorname{rank}(V_i) = 2n(\frac{\delta}{k} + 1).$$

Let us assume that there exists $V_{\frac{\delta}{k}+i}$, as defined in (3.3) equal to zero, for some $i \in \{1, 2, \ldots, \frac{\delta}{k} + 1\}$. Let i be such that $V_{\frac{\delta}{k}+i} = 0$ and $V_{\frac{\delta}{k}+j} \neq 0$, for $j = 1, \ldots, i-1$. Then

$$\sum_{j=1}^{i-1} \operatorname{rank}(V_{\frac{\delta}{k}+j}) = ni.$$

Moreover,

$$u_{\frac{\delta}{k}+i} = u_{\frac{\delta}{k}+i-1} = \cdots = u_{2i} = 0,$$

and for $j = 0, 1, \ldots, i-1$,

$$u^0_{i+j} + u^{k-1}_{i-j-1} = u^1_{i+j} + u^{k-2}_{i-j-1} = \cdots = u^{k-1}_{i+j} + u^0_{i-j-1} = 0.$$

In particular,

$$u^0_{2i-1} + u^{k-1}_0 = u^1_{2i-1} + u^{k-2}_0 = \cdots = u^{k-1}_{2i-1} + u^0_0 = 0,$$

which means that

$$u_{2i-1} = -u_0 \begin{bmatrix} 0 & & 1 \\ & \iddots & \\ 1 & & 0 \end{bmatrix} \neq 0,$$

since $u_0 \neq 0$.

Let us now consider one of the coefficients of degree $\frac{\delta}{k} + i + r$, for some $r \in$

$\{1, 2, \dots, \frac{\delta}{k} - i\}$. Then, it follows that,

$$v_{\frac{\delta}{k}+i+r} = \sum_{j=0}^{\frac{\delta}{k}-(i+r)} u_{\frac{\delta}{k}+i+r-j} G_j + \sum_{j=0}^{i+r-1} u_{i+r+j} G_{\frac{\delta}{k}-j} + \sum_{j=0}^{i+r-1} \hat{u}_{i+r-1-j} G_{\frac{\delta}{k}-j},$$

where $\hat{u}_{i+r-1-j} = u_{i+r-1-j} \begin{bmatrix} 0 & & 1 \\ & \iddots & \\ 1 & & 0 \end{bmatrix}$, for $0 \leq j \leq i + r - 1$. Thus,

$$\begin{aligned} V_{\frac{\delta}{k}+i+r} &= \sum_{j=0}^{\frac{\delta}{k}-(i+r)} \left(\sum_{\ell=0}^{k-1} u^{\ell}_{\frac{\delta}{k}+i+r-j} X A^{kj+\ell} \right) + \sum_{j=0}^{i+r-1} \left(\sum_{\ell=0}^{k-1} (u^{\ell}_{i+r+j} + u^{k-1-\ell}_{i+r-1-j}) X A^{\delta-kj+\ell} \right) \\ &= X B_{\frac{\delta}{k}+i+r}, \end{aligned}$$

where

$$B_{\frac{\delta}{k}+i+r} = \sum_{j=0}^{\frac{\delta}{k}-(i+r)} \left(\sum_{\ell=0}^{k-1} u^{\ell}_{\frac{\delta}{k}+i+r-j} A^{kj+\ell} \right) + \sum_{j=0}^{i+r-1} \left(\sum_{\ell=0}^{k-1} (u^{\ell}_{i+r+j} + u^{k-1-\ell}_{i+r-1-j}) A^{\delta-kj+\ell} \right).$$

Since $u_{2i-1} \neq 0$, $B_{\frac{\delta}{k}+i+r}$ is a nontrivial linear combination of $I, A, \dots, A^{m-1}$ then, $B_{\frac{\delta}{k}+i+r}$ is full rank and consequently $X B_{\frac{\delta}{k}+i+r}$ is full row rank.

So, we conclude that all V_j for $\frac{\delta}{k}+i+1 \leq j \leq 2\frac{\delta}{k}$ are full row rank and consequently,

$$\begin{aligned} \sum_{j=0}^{2\frac{\delta}{k}+1} \operatorname{rank}(V_j) &= \sum_{j=0}^{\frac{\delta}{k}} \operatorname{rank}(V_j) + \sum_{j=\frac{\delta}{k}+1}^{\frac{\delta}{k}+i-1} \operatorname{rank}(V_j) + \sum_{j=\frac{\delta}{k}+i+1}^{2\frac{\delta}{k}} \operatorname{rank}(V_j) \\ &= n\left(\frac{\delta}{k}+1\right) + ni + n\left(\frac{\delta}{k} - i\right) \\ &= 2n\left(\frac{\delta}{k}\right) + n. \end{aligned}$$

Moreover,

$$\begin{aligned} v_{l+2\frac{\delta}{k}+1} &= u_l G_{2\frac{\delta}{k}+1} \\ &= u_l \begin{bmatrix} 0 & & 1 \\ & \iddots & \\ 1 & & 0 \end{bmatrix} G_0 \end{aligned}$$

and, therefore,

$$V_{l+2\frac{\delta}{k}+1} = \sum_{j=0}^{k-1} \hat{u}_l^j X A^{k-1-j} = X B_{l+2\frac{\delta}{k}+1},$$

with $B_{l+2\frac{\delta}{k}+1} = \sum_{j=0}^{k-1} \hat{u}_l^j A^{k-1-j}$, which is full rank since $u_l \neq 0$ and therefore $V_{l+2\frac{\delta}{k}+1}$ is full row rank. Thus,

$$\text{rwt}(V(D)) \geq \sum_{j=0}^{2\frac{\delta}{k}+1} \text{rank}(V_j) + \text{rank}(V_{l+2\frac{\delta}{k}+1}) = 2n\left(\frac{\delta}{k}+1\right).$$

Consequently, $\mathcal{C}$ is MRD. □

Next example illustrates the above theorem.

Example 3.6. *Consider the companion matrix*

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \end{bmatrix} \in \mathbb{F}_2^{4\times 4}.$$

of the irreducible polynomial $\chi(\lambda) = \lambda^4 + \lambda + 1 \in \mathbb{F}_2[\lambda]$. and the full row rank matrix

$$X = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \in \mathbb{F}_2^{3\times 4}.$$

Let $\delta = 2$ and $k = 2$ (note that $m \geq \delta + k$ and $k|\delta$).

The rank metric convolutional code with encoder $G(D) = G_0 + G_1 D + G_2 D^2 + G_3 D^3$ with

$$G_0 = \begin{bmatrix} \psi^{-1}(X) \\ \psi^{-1}(XA) \end{bmatrix} = \left[\begin{array}{cccc|cccc|cccc} 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \end{array}\right],$$

$$G_1 = \begin{bmatrix} \psi^{-1}(XA^2) \\ \psi^{-1}(XA^3) \end{bmatrix} = \left[\begin{array}{cccc|cccc|cccc} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \end{array}\right],$$

$$\begin{aligned} G_2 &= \begin{bmatrix} 0 & & 1 \\ & \iddots & \\ 1 & & 0 \end{bmatrix} G_1 \\ &= \begin{bmatrix} \psi^{-1}(XA^3) \\ \psi^{-1}(XA^2) \end{bmatrix} = \left[\begin{array}{cccc|cccc|cccc} 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \end{array}\right] \end{aligned}$$

and

$$\begin{aligned} G_3 &= \begin{bmatrix} 0 & & 1 \\ & \iddots & \\ 1 & & 0 \end{bmatrix} G_0 \\ &= \begin{bmatrix} \psi^{-1}(XA) \\ \psi^{-1}(X) \end{bmatrix} = \left[\begin{array}{cccc|cccc|cccc} 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \end{array}\right], \end{aligned}$$

is a $(3 \times 4, 2, 6)$ *MRD convolutional code.*

The construction presented in this section allows to obtain an $(n \times m, k, 2\delta + k)$ MRD convolutional code for $m \geq \delta + k$ and such that $k|\delta$. In the next section we generalize this construction for the case in which $k \nmid \delta$.

3.3 Construction 3

Let us consider an $(n \times m, k, \delta)$ MRD convolutional code as defined in (2.4) with $n \leq m$, $k < nm$, $m \geq \delta + k$ and such that $k \nmid \delta$.

Let $A \in \mathbb{F}_q^{m \times m}$ be a matrix with irreducible characteristic polynomial and $X \in \mathbb{F}_q^{n \times m}$ a full row rank matrix.

Let

$$G_i = \begin{bmatrix} \psi^{-1}(XA^{ki}) \\ \psi^{-1}(XA^{ki+1}) \\ \vdots \\ \psi^{-1}(XA^{ki+k-1}) \end{bmatrix}, \ 0 \leq i \leq \left\lfloor \frac{\delta}{k} \right\rfloor \tag{3.4}$$

$$G_{\lfloor \frac{\delta}{k} \rfloor + 1} = \begin{bmatrix} \psi^{-1}(XA^{k\lfloor \frac{\delta}{k} \rfloor + k}) \\ \vdots \\ \psi^{-1}(XA^{k+\delta-1}) \\ \psi^{-1}(XI) \\ \dots \\ \psi^{-1}(XA^{k-1-(\delta - k\lfloor \frac{\delta}{k} \rfloor)}) \end{bmatrix}, \tag{3.5}$$

and

$$G_i = \begin{bmatrix} 0 & & 1 \\ & \iddots & \\ 1 & & 0 \end{bmatrix} G_{2\left\lfloor \frac{\delta}{k} \right\rfloor + 3 - i}, \tag{3.6}$$

for $\left\lfloor \frac{\delta}{k} \right\rfloor + 2 \leq i \leq 2\left\lfloor \frac{\delta}{k} \right\rfloor + 3$.

Let $\mathcal{C}$ be the rank metric convolutional code with encoder

$$G(D) = \sum_{i=0}^{2\left\lfloor \frac{\delta}{k} \right\rfloor + 3} G_i D^i \in \mathbb{F}[D]^{k \times nm}. \tag{3.7}$$

Next lemma states that $\mathcal{C}$ has degree $k\left(2\left\lfloor \frac{\delta}{k} \right\rfloor + 3\right)$.

Lemma 3.7. *Let m, n, k and δ be integers such that $n \leq m$, $k < nm$, $m \geq \delta + k$ and such that $k \nmid \delta$. Let $A \in \mathbb{F}_q^{m \times m}$ be a matrix with irreducible characteristic polynomial and $X \in \mathbb{F}_q^{n \times m}$ a full row rank matrix. Let $\mathcal{C}$ be the rank metric convolutional code with encoder $G(D)$ as defined in (3.7). Then $\mathcal{C}$ is an $\left(n \times m, k, k\left(2\left\lfloor \frac{\delta}{k} \right\rfloor + 3\right)\right)$ rank metric convolutional code.*

Proof. By a similar reasoning as in the proof of Lemma 3.4 we show that $[G]^{hc} = \begin{bmatrix} 0 & & 1 \\ & \iddots & \\ 1 & & 0 \end{bmatrix} G_0$ is full row rank and so we conclude that $G(D)$ is row reduced and therefore the degrre of the code is equal to $\operatorname{extdeg}(G(D)) = k\left(2\left\lfloor \frac{\delta}{k} \right\rfloor + 3\right)$. □

Next theorem shows that $\mathcal{C}$ is an $\left(n \times m, k, k\left(2\left\lfloor \frac{\delta}{k} \right\rfloor + 3\right)\right)$ MRD rank metric convolutional code.

Theorem 3.8. *Let m, n, k and δ be integers such that $n \leq m$, $k < nm$, $m \geq \delta + k$ and such that $k \nmid \delta$. Let $A \in \mathbb{F}_q^{m \times m}$ be a matrix with irreducible characteristic polynomial and $X \in \mathbb{F}_q^{n \times m}$ a full row rank matrix. Let $\mathcal{C}$ be the rank metric convolutional code with encoder $G(D)$ as defined in (3.7). Then $\mathcal{C}$ is an $\left(n \times m, k, k\left(2\left\lfloor \frac{\delta}{k} \right\rfloor + 3\right)\right)$ MRD rank metric convolutional code.*

Proof. $\mathcal{C}$ is an MRD rank metric convolutional code if $d_{\text{SR}}(\mathcal{C}) = 2n\left(\left\lfloor \frac{\delta}{k} \right\rfloor + 2\right)$ (see Theorem 2.2). Thus we have to prove that $\operatorname{rank}(V(D)) \geq 2n\left(\left\lfloor \frac{\delta}{k} \right\rfloor + 2\right)$ for all nonzero $V(D) \in \mathcal{C}$.

Let us consider $u(D) = \sum_{i=0}^{l} u_i D^i$, with $u_0 \neq 0$, $v(D) = u(D)G(D) \in \mathbb{F}[D]^{nm}$ and $V(D) = \psi(v(D)) \in \mathcal{C}$.

If $u(D) = u_0$ has degree zero then,

$$v(D) = \sum_{i=0}^{2\left\lfloor\frac{\delta}{k}\right\rfloor+3} u_0 G_i D^i$$

and

$$V(D) = \psi(u_0 G_i) D^i = \sum_{i=0}^{2\left\lfloor\frac{\delta}{k}\right\rfloor+3} X B_i,$$

where

$$B_i = u_0^0 A^{ki} + u_0^1 A^{ki+1} + \cdots + u_0^{k-1} A^{ki+k-1},$$

for $i = 0, 1, \dots, \left\lfloor\frac{\delta}{k}\right\rfloor$,

$$B_{\left\lfloor\frac{\delta}{k}\right\rfloor+1} = u_0^0 A^{k\left\lfloor\frac{\delta}{k}\right\rfloor+k} + u_0^1 A^{k\left\lfloor\frac{\delta}{k}\right\rfloor+k+1} + \cdots + u_0^{\delta-k\left\lfloor\frac{\delta}{k}\right\rfloor-1} A^{k+\delta-1} + u_0^{\delta-k\left\lfloor\frac{\delta}{k}\right\rfloor} I +$$

$$+ u_0^{\delta-k\left\lfloor\frac{\delta}{k}\right\rfloor+1} A + \cdots + u_0^{k-1} A^{k\left\lfloor\frac{\delta}{k}\right\rfloor-1}$$

and

$$B_i = u_0^0 A^{k\left(2\left\lfloor\frac{\delta}{k}\right\rfloor+3-i\right)+k-1} + u_0^1 A^{k\left(2\left\lfloor\frac{\delta}{k}\right\rfloor+3-i\right)+k-2} + \cdots + u_0^{k-1} A^{k\left(2\left\lfloor\frac{\delta}{k}\right\rfloor+3-i\right)},$$

$\left\lfloor\frac{\delta}{k}\right\rfloor + 1 \leq i \leq 2\left\lfloor\frac{\delta}{k}\right\rfloor + 3$.

B_i is a nontrivial linear combination of the matrices $I, A, \dots, A^{m-1}$ for $i = 0, 1, \dots, 2\left\lfloor\frac{\delta}{k}\right\rfloor + 3$ since $u_0^s \neq 0$ for some $s \in \{0, 1, \dots, k-1\}$, and therefore $V_i = XB_i$ is full row rank for $i = 0, 1, \dots, 2\left\lfloor\frac{\delta}{k}\right\rfloor + 3$. So, we conclude that

$$\begin{aligned}\operatorname{rwt}(V(D)) &= \sum_{i=0}^{2\left\lfloor\frac{\delta}{k}\right\rfloor+3} \operatorname{rank}(V_i) \\ &= n\left(2\left\lfloor\frac{\delta}{k}\right\rfloor + 4\right) \\ &= 2n\left(\left\lfloor\frac{\delta}{k}\right\rfloor + 2\right).\end{aligned}$$

Let us consider $u(D) = \sum_{i=0}^{l} u_i D^i$ with $u_0 \neq 0$, $u_l \neq 0$ and $l \geq 1$. Then,

$$v_i = \sum_{j=0}^{i} u_{i-j} G_j, 0 \leq i \leq \left\lfloor\frac{\delta}{k}\right\rfloor$$

and $V_i = \psi(v_i)$ will be given by

$$V_i = \sum_{h=0}^{ki+k-1} u_{i-\lfloor\frac{h}{k}\rfloor}^{h-k\lfloor\frac{h}{k}\rfloor} XA^h$$

and since $k\left\lfloor\frac{\delta}{k}\right\rfloor - 1 < \delta$ and, by hypothesis, $\delta + k \leq m$, then $k\left\lfloor\frac{\delta}{k}\right\rfloor + k - 1 < \delta + k$. Therefore, V_i is full row rank because it is a nontrivial linear combination of elements of $\mathbb{F}[A]$, for all $i \leq \left\lfloor\frac{\delta}{k}\right\rfloor$

Let us now consider $V_{\left\lfloor\frac{\delta}{k}\right\rfloor+1}$, which is given by,

$$V_{\left\lfloor\frac{\delta}{k}\right\rfloor+1} = \sum_{j=0}^{\delta-k\left\lfloor\frac{\delta}{k}\right\rfloor-1} \hat{u}_j XA^{k\left\lfloor\frac{\delta}{k}\right\rfloor+k+j} + \sum_{j=\delta-k\left\lfloor\frac{\delta}{k}\right\rfloor}^{k-1} \hat{u}_j XA^{\delta-k\left\lfloor\frac{\delta}{k}\right\rfloor-j} + \sum_{j=1}^{\left\lfloor\frac{\delta}{k}\right\rfloor}\left[\sum_{h=0}^{k-1} \hat{u}_{kj+k} XA^{k(\left\lfloor\frac{\delta}{k}\right\rfloor+1-j)+h}\right]$$

$$+ \sum_{j=0}^{k-\delta+k\left\lfloor\frac{\delta}{k}\right\rfloor-1} \hat{u}_{k(\left\lfloor\frac{\delta}{k}\right\rfloor+1)+j} XA^j + \sum_{j=k-\delta+k\left\lfloor\frac{\delta}{k}\right\rfloor}^{k-1} \hat{u}_{k(\left\lfloor\frac{\delta}{k}\right\rfloor+1)+j} XA^j,$$

where $\hat{u}_{ki+j} = u_i^j$.

This vector can be written as

$$\begin{aligned}
V_{\left\lfloor\frac{\delta}{k}\right\rfloor+1} = {} & \sum_{j=0}^{\delta-k\left\lfloor\frac{\delta}{k}\right\rfloor-1} \hat{u}_j XA^{k\left\lfloor\frac{\delta}{k}\right\rfloor+k+j} + \sum_{j=1}^{\left\lfloor\frac{\delta}{k}\right\rfloor}\left[\sum_{h=0}^{k-1} \hat{u}_{kj+k} XA^{k(\left\lfloor\frac{\delta}{k}\right\rfloor+1-j)+h}\right] \\
& + \left(\sum_{j=\delta-k\left\lfloor\frac{\delta}{k}\right\rfloor}^{k-1} \hat{u}_j XA^{\delta-k\left\lfloor\frac{\delta}{k}\right\rfloor-j} + \sum_{j=0}^{k-\delta+k\left\lfloor\frac{\delta}{k}\right\rfloor-1} \hat{u}_{k(\left\lfloor\frac{\delta}{k}\right\rfloor+1)+j} XA^j\right) + \\
& + \sum_{j=k-\delta+k\left\lfloor\frac{\delta}{k}\right\rfloor}^{k-1} \hat{u}_{k(\left\lfloor\frac{\delta}{k}\right\rfloor+1)+j} XA^j \\
= {} & \sum_{j=0}^{\delta-k\left\lfloor\frac{\delta}{k}\right\rfloor-1} \hat{u}_j XA^{k\left\lfloor\frac{\delta}{k}\right\rfloor+k+j} + \sum_{j=1}^{\left\lfloor\frac{\delta}{k}\right\rfloor}\left[\sum_{h=0}^{k-1} \hat{u}_{kj+k} XA^{k(\left\lfloor\frac{\delta}{k}\right\rfloor+1-j)+h}\right] \\
& + \left(\sum_{h=0}^{k-1-\big(\delta-k\left\lfloor\frac{\delta}{k}\right\rfloor)\big)} \hat{u}_{\delta-k\left\lfloor\frac{\delta}{k}\right\rfloor-h} XA^h + \sum_{j=0}^{k-1-\big(\delta-k\left\lfloor\frac{\delta}{k}\right\rfloor)\big)} \hat{u}_{k\big(\left\lfloor\frac{\delta}{k}\right\rfloor+1\big)+j} XA^j\right) + \\
& + \sum_{j=k-\delta+k\left\lfloor\frac{\delta}{k}\right\rfloor}^{k-1} \hat{u}_{k(\left\lfloor\frac{\delta}{k}\right\rfloor+1)+j} XA^j,
\end{aligned}$$

and, therefore,

$$\begin{aligned}
V_{\left\lfloor \frac{\delta}{k} \right\rfloor+1} = & \sum_{j=0}^{\delta-k\left\lfloor \frac{\delta}{k} \right\rfloor-1} \hat{u}_j XA^{k\left\lfloor \frac{\delta}{k} \right\rfloor+k+j} + \sum_{j=1}^{\left\lfloor \frac{\delta}{k} \right\rfloor} \left[\sum_{h=0}^{k-1} \hat{u}_{kj+k} XA^{k(\left\lfloor \frac{\delta}{k} \right\rfloor+1-j)+h} \right] \\
& + \left(\sum_{j=0}^{k-1-\left(\delta-k\left\lfloor \frac{\delta}{k} \right\rfloor\right))} \left[\hat{u}_{\delta-k\left\lfloor \frac{\delta}{k} \right\rfloor-j} + \hat{u}_{k\left(\left\lfloor \frac{\delta}{k} \right\rfloor+1\right)+j} \right] XA^j \right) + \\
& + \sum_{j=k-\delta+k\left\lfloor \frac{\delta}{k} \right\rfloor}^{k-1} \hat{u}_{k(\left\lfloor \frac{\delta}{k} \right\rfloor+1)+j} XA^j.
\end{aligned}$$

Because we know that $u_0 \neq 0$ then, at least, one of the $\hat{u}_r$, for $r \in \{0, \ldots, k-1\}$ is nonzero. Therefore, if $r \leq \delta - k\left\lfloor \frac{\delta}{k} \right\rfloor - 1$, $V_{\left\lfloor \frac{\delta}{k} \right\rfloor+1}$ is full row rank. Otherwise it is possible to $V_{\left\lfloor \frac{\delta}{k} \right\rfloor+1}$ to be equal zero.

The next $\left\lfloor \frac{\delta}{k} \right\rfloor + 2$ vector coefficients of $v(D)$ are represented by

$$\begin{aligned}
v_{\left\lfloor \frac{\delta}{k} \right\rfloor+1+i} = {} & u_{\left\lfloor \frac{\delta}{k} \right\rfloor+1+i} G_0 + \cdots + u_{2i} G_{\left\lfloor \frac{\delta}{k} \right\rfloor+1-i} + u_{2i-1} G_{\left\lfloor \frac{\delta}{k} \right\rfloor+2-i} + \cdots + u_i G_{\left\lfloor \frac{\delta}{k} \right\rfloor+1} \\
& + u_{i-1} \hat{G}_{\left\lfloor \frac{\delta}{k} \right\rfloor+1} + u_{i-2} \hat{G}_{\left\lfloor \frac{\delta}{k} \right\rfloor} + \cdots + u_0 \hat{G}_{\left\lfloor \frac{\delta}{k} \right\rfloor+2-i},
\end{aligned}$$

for $1 \leq i \leq \left\lfloor \frac{\delta}{k} \right\rfloor + 2$.

Let us consider the transformation $u_i^j = \hat{u}_{ki+j}$. Then

$$\begin{aligned}
V_{\left\lfloor \frac{\delta}{k} \right\rfloor+1+i} = & \sum_{j=0}^{k-1-(\delta-k\left\lfloor \frac{\delta}{k} \right\rfloor)} \left(\hat{u}_{k(\left\lfloor \frac{\delta}{k} \right\rfloor+1+i)+j} + \hat{u}_{ki+\delta-k\left\lfloor \frac{\delta}{k} \right\rfloor+j} + \hat{u}_{k(i-1)+j} \right) XA^j \\
& + \sum_{j=1}^{\left\lfloor \frac{\delta}{k} \right\rfloor+1-i} \left[\sum_{h=0}^{k-1} \hat{u}_{k(\left\lfloor \frac{\delta}{k} \right\rfloor+1+i-j)+h} XA^{kj+h} \right] + \\
& + \sum_{j=0}^{k-1} \left(\hat{u}_{k(2i-1)+j} + \hat{u}_{k-1-j} \right) XA^{k(\left\lfloor \frac{\delta}{k} \right\rfloor+2-i)+j} \\
& + \cdots + \sum_{j=0}^{\delta-k\left\lfloor \frac{\delta}{k} \right\rfloor-1} \left(\hat{u}_{ki+j} + \hat{u}_{k(i-1)+k-1-j} \right) XA^{k\left\lfloor \frac{\delta}{k} \right\rfloor+k+j}.
\end{aligned}$$

Then if $V_{\left\lfloor \frac{\delta}{k} \right\rfloor+1+i} = 0$, it follows that:

- $u_{\left\lfloor \frac{\delta}{k} \right\rfloor+1+i-j} = 0$, for all $j \in \left\{1, \ldots, \left\lfloor \frac{\delta}{k} \right\rfloor + 1 - i\right\}$

- if $\hat{u}_s$ is the first element of the vector u_0 different from zero (i.e., $u_0^t \neq 0$ for $t < s$ and $u_0^s = 0$), then $\hat{u}_{2ki-(s+1)} \neq 0$

- $\hat{u}_{ki+j} + \hat{u}_{k(i-1)+k-1-j} = 0$

- $\hat{u}_{k(\lfloor \frac{\delta}{k} \rfloor+1+i)+j} + \hat{u}_{ki+\delta-k\lfloor \frac{\delta}{k} \rfloor+j} + \hat{u}_{k(i-1)+j} = 0$

Now, if we choose any V_j, with $j = \lfloor \frac{\delta}{k} \rfloor + 1 + i + r$ for $1 \leq r \leq \lfloor \frac{\delta}{k} \rfloor + 2 - i$, then it follows that,

$$\begin{aligned}
V_{\lfloor \frac{\delta}{k} \rfloor+1+i+r} = & \sum_{j=0}^{k-1-(\delta-k\lfloor \frac{\delta}{k} \rfloor)} \left(\hat{u}_{k(\lfloor \frac{\delta}{k} \rfloor+1+i+r)+j} + \hat{u}_{k(i+r)+\delta-k\lfloor \frac{\delta}{k} \rfloor+j} + \hat{u}_{k(i+r-1)+j}\right) XA^j + \\
& + \sum_{j=1}^{\lfloor \frac{\delta}{k} \rfloor+1-(i+r)} \left[\sum_{h=0}^{k-1} \hat{u}_{k(\lfloor \frac{\delta}{k} \rfloor+1+i+r-j)+h} XA^{kj+h}\right] + \\
& + \sum_{j=0}^{k-1} \left(\hat{u}_{k(2(i+r)-1)+j} + \hat{u}_{k-1-j}\right) XA^{k(\lfloor \frac{\delta}{k} \rfloor+2-(i+r))+j} + \\
& + \cdots + \sum_{j=0}^{\delta-k\lfloor \frac{\delta}{k} \rfloor-1} \left(\hat{u}_{k(i+r)+j} + \hat{u}_{k(i+r-1)+k-1-j}\right) XA^{k\lfloor \frac{\delta}{k} \rfloor+k+j}.
\end{aligned}$$

Note that

$$(\hat{u}_s+\hat{u}_{2k(i+r)-(s+1)})XA^{k(\lfloor \frac{\delta}{k} \rfloor+2-(i+r))+k-1-j} = X((\hat{u}_s+\hat{u}_{2k(i+r)-(s+1)})A^{k(\lfloor \frac{\delta}{k} \rfloor+2-(i+r))+k-1-j})$$

is a term of $V_{\lfloor \frac{\delta}{k} \rfloor+1+i+r}$ and since $2k(i+r) - (s+1) > k(2i-1) + k - 1$ it follows that $\hat{u}_{2k(i+r)-(s+1)} = 0$ and therefore $X((\hat{u}_s + \hat{u}_{2k(i+r)-(s+1)})A^{k(\lfloor \frac{\delta}{k} \rfloor+2-(i+r))+k-1-j})$ is full row rank and consequently $V_{\lfloor \frac{\delta}{k} \rfloor+1+i+r}$ is also full row rank.

Therefore, all V_j for $\lfloor \frac{\delta}{k} \rfloor + i + 2 \leq j \leq 2\lfloor \frac{\delta}{k} \rfloor + 3$ and $1 \leq i \leq \lfloor \frac{\delta}{k} \rfloor + 2$ are full row rank and consequently,

$$\begin{aligned}
\sum_{j=0}^{2\lfloor \frac{\delta}{k} \rfloor+3} \operatorname{rank}(V_j) &= n\left(\left\lfloor \frac{\delta}{k} \right\rfloor + 1\right) + n\left(\left\lfloor \frac{\delta}{k} \right\rfloor + 1\right) \\
&= 2n\left(\left\lfloor \frac{\delta}{k} \right\rfloor + 1\right).
\end{aligned}$$

Moreover,

$$V_{l+2\lfloor\frac{\delta}{k}\rfloor+3} = \psi(v_{l+2\lfloor\frac{\delta}{k}\rfloor+3}) = \psi\left(u_l \begin{bmatrix} 0 & & 1 \\ & \iddots & \\ 1 & & 0 \end{bmatrix} G_0\right) = \sum_{j=0}^{k-1} \hat{u}_{kl+j} X A^{k-1-j}$$

and

$$V_{l+2\lfloor\frac{\delta}{k}\rfloor+3-1} = \psi(v_{l+2\lfloor\frac{\delta}{k}\rfloor+3-1}) = \psi\left(u_l \begin{bmatrix} 0 & & 1 \\ & \iddots & \\ 1 & & 0 \end{bmatrix} G_1 + u_{l-1} \begin{bmatrix} 0 & & 1 \\ & \iddots & \\ 1 & & 0 \end{bmatrix} G_0\right)$$

are full row rank matrices since $u_l \neq 0$. Then,

$$\sum_{j=0}^{2\lfloor\frac{\delta}{k}\rfloor+3+l} \operatorname{rank}(V_j) \geq 2n\left(\left\lfloor\frac{\delta}{k}\right\rfloor+1\right)+2n = 2n\left(\left\lfloor\frac{\delta}{k}\right\rfloor+2\right).$$

Consequently, $\mathcal{C}$ is MRD. □

Next example presents an MRD rank metric convolutional code that it is not possible to build using the construction of Section 3.2.

Example 3.9. *Consider the companion matrix*

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \end{bmatrix} \in \mathbb{F}_2^{4\times 4}.$$

of the irreducible polynomial $\chi(\lambda) = \lambda^4+\lambda+1 \in \mathbb{F}_2[\lambda]$ and the full row rank matrix

$$X = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \in \mathbb{F}_2^{3\times 4}.$$

Let $\delta = 1$ and $k = 3$ (note that $m \geq \delta + k$ and that $k \nmid \delta$).

The rank metric convolutional code with encoder $G(D) = G_0+G_1D+G_2D^2+G_3D^3$ with

$$G_0 = \begin{bmatrix} \psi^{-1}(X) \\ \psi^{-1}(XA) \\ \psi^{-1}(XA^2) \end{bmatrix} = \left[\begin{array}{cccc|cccc|cccc} 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \end{array}\right],$$

$$G_1 = \begin{bmatrix} \psi^{-1}(XA^3) \\ psi^{-1}(X) \\ psi^{-1}(XA) \end{bmatrix} = \left[\begin{array}{cccc|cccc|cccc} 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \end{array}\right],$$

$$\begin{aligned} G_2 &= \begin{bmatrix} 0 & & 1 \\ & \iddots & \\ 1 & & 0 \end{bmatrix} G_1 \\ &= \begin{bmatrix} \psi^{-1}(XA) \\ \psi^{-1}(X) \\ \psi^{-1}(XA^3) \end{bmatrix} = \left[\begin{array}{cccc|cccc|cccc} 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \end{array}\right], \end{aligned}$$

and

$$\begin{aligned} G_3 &= \begin{bmatrix} 0 & & 1 \\ & \iddots & \\ 1 & & 0 \end{bmatrix} G_0 \\ &= \begin{bmatrix} \psi^{-1}(XA^2) \\ \psi^{-1}(XA) \\ \psi^{-1}(X) \end{bmatrix} = \left[\begin{array}{cccc|cccc|cccc} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \end{array}\right], \end{aligned}$$

is a $(3 \times 4, 1, 6)$ *MRD convolutional code.*

3.4 Construction of a $(5 \times 5, 1, 19)$ MRD rank metric convolutional code

In this section we consider $(5 \times 5, 1, \delta)$ rank metric convolutional codes. The above constructions only allowed to obtain $(5 \times 5, 1, \delta)$ MRD rank metric convolutional codes for $\delta \leq 9$. Next we present a construction of a $(5 \times 5, 1, 19)$ MRD rank metric convolutional code.

Let $A \in \mathbb{F}_q^{5\times 5}$ be a matrix with irreducible characteristic polynomial, consider the polynomial matrix

$$\begin{aligned} G(D) = {} & \psi^{-1}(I) + \psi^{-1}(A)D + \psi^{-1}(A^2)D^2 + \psi^{-1}(A^3)D^3 + \psi^{-1}(A^4)D^4 + \\ & + \psi^{-1}(I)D^5 + \psi^{-1}(A^2)D^6 + \psi^{-1}(A^4)D^7 + \psi^{-1}(A)D^8 + \psi^{-1}(A^3)D^9 + \\ & + \psi^{-1}(I)D^{10} + \psi^{-1}(A^3)D^{11} + \psi^{-1}(A)D^{12} + \psi^{-1}(A^4)D^{13} + \psi^{-1}(A^2)D^{14} + \\ & + \psi^{-1}(I)D^{15} + \psi^{-1}(A^4)D^{16} + \psi^{-1}(A^3)D^{17} + \psi^{-1}(A^2)D^{18} + \psi^{-1}(A)D^{19} \end{aligned} \tag{3.8}$$

and the rank metric convolutional code $\mathcal{C}$ with encoder $G(D)$.

Since $[G]^{hc} = \psi^{-1}(A) \neq 0$ (and $k = 1$), $G(D)$ is row reduced and therefore the degree of $\mathcal{C}$ is $\operatorname{extdeg}(G(D)) = 19$. Thus, to prove that $\mathcal{C}$ is an MRD rank metric convolutional code we have to show that

$$d_{\mathrm{SR}}(\mathcal{C}) = m(\delta + 1) = 5 \times (19 + 1) = 100.$$

Without loss of generality, let us consider $u(D) = u_0 + u_1 D + \cdots + u_l D^l$,with $u_0 \neq 0$ and $u_l \neq 0$, $v(D) = u(D)G(D)$ and $V(D) = \psi(v(D)) = \sum_{i=0}^{l+19} V_i D^i$.

If $l = 0$, $V_i = u_0 A^t$, for some $t \in \{0, 1, 2, 3, 4\}$, $i = 0, 1, \dots, 19$, and therefore all the coefficients of $V(D)$, V_i, $i = 0, 1, \dots, 19$, are full rank and we have that

$$\operatorname{rwt}(V(D)) = 20 \times 5 = 100.$$

Let us consider the case $l < 5$. Then

$$\begin{aligned} V_0 &= u_0 I \\ V_1 &= u_1 I + u_0 A \\ V_2 &= u_2 I + u_1 A + u_0 A^2 \\ V_3 &= u_3 I + u_2 A + u_1 A^2 + u_0 A^3 \\ V_4 &= u_4 I + u_3 A + u_2 A^2 + u_1 A^3 + u_0 A^4 \end{aligned}$$

Since $u_0 \neq 0$, each V_i, for $0 \leq i \leq 4$ is full rank,and then

$$\sum_{i=0}^{4} \operatorname{rank}(V_i) = 5 \times 5 = 25.$$

Moreover,

$$\begin{aligned}
V_5 &= u_0 I + u_4 A + u_3 A^2 + u_2 A^3 + u_1 A^4 \\
V_6 &= u_1 I + (u_0 + u_4) A^2 + u_3 A^3 + u_2 A^4 \\
V_7 &= u_2 I + u_1 A^2 + u_4 A^3 + (u_0 + u_3) A^4 \\
V_8 &= u_3 I + u_0 A + u_2 A^2 + (u_1 + u_4) A^4 \\
V_9 &= u_4 I + u_1 A + u_3 A^2 + u_0 A^3 + u_2 A^4.
\end{aligned}$$

Note that V_5, V_8 and V_9 are full rank because they are a nontrivial linear combination of I, A, A^2, A^3, A^4 since $u_0 \neq 0$. However V_6 and V_7 can be zero. In these cases we have the following:

- If $V_6 = 0$ then $u_1 = u_2 = u_3 = u_0 + u_4 = 0$, which means that $u_1 = u_2 = u_3 = 0$ and $u_4 = -u_0 \neq 0$. This implies that V_7 is full rank.
- If $V_7 = 0$ then $u_1 = u_2 = u_4 = u_0 + u_3 = 0$, and therefore $u_1 = u_2 = u_4 = 0$ and $u_3 = -u_0 \neq 0$. Thus V_6 is full rank.

So, we conclude that

$$\sum_{i=5}^{9} \operatorname{rank}(V_i) \geq 20.$$

The next five coefficients of $V(D)$ are,

$$\begin{aligned}
V_{10} &= u_0 I + u_2 A + u_4 A^2 + u_1 A^3 + u_3 A^4 \\
V_{11} &= u_1 I + u_3 A + (u_0 + u_2) A^3 + u_4 A^4 \\
V_{12} &= u_2 I + (u_0 + u_4) A + (u_1 + u_3) A^3 \\
V_{13} &= u_3 I + u_1 A + (u_2 + u_4) A^3 + u_0 A^4 \\
V_{14} &= u_4 I + u_2 A + u_0 A^2 + u_3 A^3 + u_1 A^4.
\end{aligned}$$

Using the same reasoning as before we conclude that V_{10}, V_{13} and V_{14} are full rank and that

- $V_{11} = 0$ implies that $u_1 = u_3 = u_4 = u_0 + u_2 = 0$, and therefore $u_1 = u_3 = u_4 = 0$ and $u_2 = -u_0 \neq 0$. Thus V_{12} is full rank.
- $V_{12} = 0$ implies that $u_2 = u_0 + u_4 = u_1 + u_3 = 0$, and therefore $u_2 = 0$, $u_4 = -u_0 \neq 0$ and $u_3 = -u_1$. Thus V_{11} is full rank.

This means that

$$\sum_{i=10}^{14} \operatorname{rank}(V_i) \geq 20.$$

By reasoning in a similar way we prove that

$$\sum_{i=15}^{19} \operatorname{rank}(Vi) \geq 20.$$

Let us now analyze in more detail the cases $l = 1$, $l = 2$, $l = 3$ and $l = 4$.

If $l = 1$, it is easy to see that

$$\sum_{i=0}^{4} \operatorname{rank}(V_i) = \sum_{i=5}^{9} \operatorname{rank}(V_i) = \sum_{i=10}^{14} \operatorname{rank}(V_i) = \sum_{i=15}^{19} = \operatorname{rank}(V_i) = 25,$$

and therefore

$$\operatorname{rwt}(V(D)) \geq 100.$$

If $l = 2$, we have that

$$\sum_{i=0}^{4} \operatorname{rank}(V_i) = \sum_{i=5}^{9} \operatorname{rank}(V_i) = \sum_{i=15}^{19} = \operatorname{rank}(V_i) = 25,$$

$$\sum_{i=10}^{14} \operatorname{rank}(V_i) \geq 20$$

and

$$\operatorname{rank}(V_{20}) = 5.$$

So we conclude that

$$\operatorname{rwt}(V(D)) \geq 100.$$

If $l = 3$, then

$$\sum_{i=0}^{4} \operatorname{rank}(V_i) = 25,$$

$$\sum_{i=5}^{9} \operatorname{rank}(V_i) \geq 20,$$

$$\sum_{i=10}^{14} \operatorname{rank}(V_i) \geq 20,$$

$$\sum_{i=15}^{19} \operatorname{rank}(V_i) \geq 20$$

and since

$$\begin{aligned}V_{20} &= u_1 A + u_2 A^2 + u_3 A^3\\ V_{21} &= u_2 A + u_3 A^2\\ V_{22} &= u_3 A\end{aligned}$$

and $u_3 \neq 0$, it follows that

$$\operatorname{rank}(V_{20}) + \operatorname{rank}(V_{21}) + \operatorname{rank}(V_{22}) = 15,$$

and therefore

$$\operatorname{rwt}(V(D)) \geq 100.$$

Finally, if $l = 4$ then

$$\begin{aligned}\sum_{i=0}^{4} \operatorname{rank}(V_i) &= 25,\\ \sum_{i=5}^{9} \operatorname{rank}(V_i) &\geq 20,\\ \sum_{i=10}^{14} \operatorname{rank}(V_i) &\geq 20\\ \sum_{i=15}^{19} \operatorname{rank}(V_i) &\geq 20,\end{aligned}$$

and since

$$\begin{aligned}V_{21} &= u_2 A + u_3 A^2 + u_4 A^3\\ V_{22} &= u_3 A + u_4 A^2\\ V_{23} &= u_4 A\end{aligned}$$

and $u_4 \neq 0$, we have that

$$\operatorname{rank}(V_{21}) + \operatorname{rank}(V_{22}) + \operatorname{rank}(V_{23}) = 15,$$

and therefore

$$\operatorname{rwt}(V(D)) \geq 100.$$

Let us now examine the case $l \geq 5$.

Let $V(D) = \sum_{i=0}^{l+19} V_i D^i$. Using the same reasoning as before we conclude that

$\sum_{i=0}^{4} \operatorname{rank}(V_i) = 25$.

Let us consider the next five coefficients of $V(D)$,

$$\begin{aligned}
V_5 &= (u_0 + u_5)I + u_4 A + u_3 A^2 + u_2 A^3 + u_1 A^4 \\
V_6 &= (u_1 + u_6)I + u_5 A + (u_0 + u_4)A^2 + u_3 A^3 + u_2 A^4 \\
V_7 &= (u_2 + u_7)I + u_6 A + (u_1 + u_5)A^2 + u_4 A^3 + (u_0 + u_3)A^4 \\
V_8 &= (u_3 + u_8)I + (u_0 + u_7)A + (u_2 + u_6)A^2 + u_5 A^3 + (u_1 + u_4)A^4 \\
V_9 &= (u_4 + u_9)I + (u_1 + u_8)A + (u_3 + u_7)A^2 + (u_0 + u_6)A^3 + (u_2 + u_5)A^4
\end{aligned}$$

It is quite simple to see that we can have one of the $V_i's$ equal to zero. However, we will see that we cannot have two null $V_i's$. In fact, the u_i, $i = 0, 1, \ldots, 9$, such that, for instance,

$$V_5 = 0 \wedge V_7 = 0,$$

are such that

$$\begin{cases}
u_0 + u_5 = 0 \\
u_4 = 0 \\
u_3 = 0 \\
u_2 = 0 \\
u_1 = 0 \\
u_2 + u_7 = 0 \\
u_6 = 0 \\
u_1 + u_5 = 0 \\
u_4 = 0 \\
u_0 + u_3 = 0
\end{cases}$$

which is a system with unique solution, the trivial solution. However, this is not possible since $u_0 \neq 0$. So we conclude that it is not possible that $V_5 = V_7 = 0$.

Using the same reasoning we conclude that it is not true that

$$V_r = 0 \wedge V_s = 0,$$

for $r, s \in \{5, 6, \ldots, 9\}$, with $r \neq s$.

Therefore,

$$\sum_{i=5}^{9} \operatorname{rank}(Vi) \geq 20.$$

Let us now analyze the rank of the following coefficients of $V(D)$ by considering the cases $l > 5$ and $l = 5$, separately.

If $l > 6$, following the same reasoning we can observe that in the next five coefficients $V_{10}, V_{11}, V_{12}, V_{13}, V_{14}$ we see that we can have two $V_i's$ equal zero but not three and in the following five coefficients, $V_{15}, V_{16}, V_{17}, V_{18}, V_{19}$ we can have three coefficients equal to zero but not four. Thus,

$$\sum_{i=0}^{19} \text{rank}(Vi) \geq 25 + 20 + 15 + 10 = 70$$

Since

$$\begin{aligned}
V_{l+13} &= u_{l-6}A + (u_{l-1} + u_{l-5})A^2 + u_{l-4}A^3 + (u_l + u_{l-3})A^4 + u_{l-2}I \\
V_{l+14} &= u_{l-5}A + (u_l + u_{l-4})A^2 + u_{l-3}A^3 + u_{l-2}A^4 + u_{l-2}I \\
V_{l+15} &= u_{l-4}A + u_{l-3}A^2 + u_{l-2}A^3 + u_{l-1}A^4 + u_lI \\
V_{l+16} &= u_{l-3}A + u_{l-2}A^2 + u_{l-1}A^3 + u_lA^4 \\
V_{l+17} &= u_{l-2}A + u_{l-1}A^2 + u_lA^3 \\
V_{l+18} &= u_{l-1}A + u_lA^2 \\
V_{l+19} &= u_lA,
\end{aligned}$$

and $u_l \neq 0$, it follows that

$$\sum_{i=l+13}^{l+19} \text{rank}(V_i) \geq 30$$

because V_{l+13} and V_{l+14} cannot be simultaneously equal o zero and V_{l+15},V_{l+16},V_{l+17},V_{l+18} and V_{l+19} are nonzero. Therefore,

$$\text{rwt}(V(D)) \geq \sum_{i=0}^{19} \text{rank}(V_i) + \sum_{i=l+14}^{l+19} \text{rank}(V_i) \geq 100.$$

To finalize, it remains to analyze the case $l = 5$ and the case $l = 6$. Considering first the case $l = 5$, we already saw that

$$\sum_{i=0}^{9} \text{rank}(Vi) \geq 25 + 20 = 45.$$

The coefficients V_i, for $i = 10, 11, \ldots, 14$, are

$$\begin{aligned}
V_{10} &= (u_0 + u_5)I + u_2A + u_4A^2 + u_1A^3 + u_3A^4 \\
V_{11} &= u_1I + u_3A + u_5A^2 + (u_0 + u_2)A^3 + u_4A^4
\end{aligned}$$

$$\begin{aligned} V_{12} &= u_2 I + (u_0 + u_4)A + (u_1 + u_3)A^3 + u_5 A^4 \\ V_{13} &= u_3 I + (u_1 + u_5)A + (u_2 + u_4)A^3 + u_0 A^4 \\ V_{14} &= u_4 I + u_2 A + u_0 A^2 + (u_3 + u_5)A^3 + u_1 A^4. \end{aligned}$$

Using the same reasoning as before we can have one of the above coefficients equal to zero, but it is not possible to have two of these coefficients equal to zero. The same happens if we consider the coefficients $V_{15}, \dots, V_{19}$. Thus,

$$\sum_{i=0}^{19} \text{rank}(Vi) \geq 25 + 20 + 20 + 20 = 85.$$

However, the last $V_i's$ with i taking values between 20 and 24, will compensate for the possible rank loss in the previous $V_i's$, because we will have 5 full row rank matrices $V_i's$. Finally, it means

$$\sum_{i=0}^{24} \text{rank}(Vi) \geq 25 + 20 + 20 + 20 + 25 \geq 100.$$

The case $l = 6$ is similar to $l = 5$. So, we conclude that $\mathcal{C}$ is MRD.

This constructions allows to obtain a $(5 \times 5, 1, \tilde{\delta})$ MRD rank metric convolutional code for $\tilde{\delta} \leq 19$. For that we have to consider the corresponding encoder

$$\tilde{G}(D) = \sum_{i=0}^{\delta} G_i,$$

where G_i, $i = 0, 1, \dots, \delta$, are the first $\delta + 1$ coefficients of the matrix $G(D) = \sum_{i=0}^{19} G_i D^i$ defined in (3.8).

Generalizing, we believe that the statement holds true for all m, with m a prime number, as it is stated as the following conjecture.

Conjecture 3.10. *An $(m \times m, 1, m^2 - m - 1)$ rank metric convolutional code $\mathcal{C}$ with m a prime number and with encoder*

$$G(D) = \sum_{i=1}^{m-1} \left[\sum_{k=0}^{m-1} A^{ik(mod.\ m)} D^{(i-1)m+k} \right], \tag{3.9}$$

i.e.,

$$\begin{aligned} G(D) = {} & I + AD + A^2D^2 + \cdots + A_{m-1}D^{m-1} + \\ & + ID^m + A^2D^{m+1} + A^4D^{m+2} + \cdots + A^{m-2}D^{2m-1} + \\ & + ID^{2m} + A^3D^{2m+1} + \cdots + A^{m-3}D^{3m-1} + \\ & + \cdots + \\ & + ID^{(m-2)m} + A^{m-1}D^{(m-2)m+1} + \cdots + AD^{(m-1)m-1} \end{aligned}$$

is MRD and $d_{\mathrm{SR}}(\mathcal{C}) = m^2(m-1)$.

As above, for the cases in which the above result is true, the matrix

$$\tilde{G}(D) = \sum_{i=0}^{\delta} G_i,$$

where G_i, $i = 0, 1, \ldots, \delta$, are the first $\delta+1$ coefficients of the matrix $G(D) = \sum_{i=0}^{(m-1)m-1} G_iD^i$ defined in (3.9) is an encoder of an $(m \times m, 1, \tilde{\delta})$ MRD rank metric convolutional code for $\tilde{\delta} \leq m^2 - m - 1$.

Chapter 4

Concatenated code

In this chapter we address the problem of concatenation of a convolutional code and a rank metric code as an alternative approach for building multi-shot codes. In particular, we present a novel scheme of a concatenation of a Hamming metric convolutional code and a rank metric block code.

The work presented in this chapter is inspired by the work done by Napp, Pinto and Sidorenko [32]. There are however important differences between these two coding schemes that will be described in detail in this chapter. The main difference is in the way the information is concatenated. The novel scheme presented here is able to encode vectors with no restrictions on the length and requires vectors over smaller finite fields which lead to a reduction in the complexity of the encoding and decoding process. Moreover, we will show that the inner code (the rank metric) is able to recover lost packets that remain lost in the concatenation scheme introduced in [32]. We will illustrate the nuances of each coding procedure by presenting several examples for different parameters and erasure patterns.

4.1 Concatenation scheme over extension fields

In [32] the authors Napp, Pinto and Sidorenko proposed a concatenation scheme with a Hamming metric convolutional code as an outer code and a rank metric block code as an inner code. The way these two are concatenated is described below.

Let $\mathcal{C}_O$ be an (n_O, k_O, δ) convolutional code over an extension field of $\mathbb{F}_{q^{mk_I}}$ with encoder G_O, (Hamming) free distance $d_{free}(\mathcal{C}_O)$, (Hamming) column distance $d_j^c(\mathcal{C}_O)$. Using the framework described in Remark 2.19 we consider an (n_I, k_I) rank metric block code $\mathcal{C}_I$ with rank distance $d_{\text{rank}}(\mathcal{C}_I)$ and encoder G_I also over an extension field of $\mathbb{F}_{q^m}$.

Consider $u(D) = u_0 + u_1 D + u_2 D^2 + \cdots \in \mathbb{F}_{q^{mk_I}}[D]^{k_O}$ be the information vector. Then, the information vector we will be encoded through $G_O(D) \in \mathbb{F}_{q^{mk_I}}[D]^{k_O \times n_O}$ in order to obtain $v(D) \in \mathcal{C}_O$ defined as:

$$v(D) = v_0 + v_1 D + v_2 D^2 + \cdots = u(D) G_O(D) \in \mathcal{C}_O \subset \mathbb{F}_{q^{mk_I}}[D]^{n_O}$$

and we write $v_i = (v_i^0, \dots, v_i^{n_o - 1})$ with $v_i^j \in \mathbb{F}_{q^{mk_I}}$.

For a given basis of $\mathbb{F}_{q^{mk_I}}$ over $\mathbb{F}_{q^m}$ we can represent each $v_i^j \in \mathbb{F}_{q^{mk_I}}$ with a vector $v_i^j \in \mathbb{F}_{q^m}^{k_I}$ and

$$v_i = (v_i^0, \dots, v_i^{n_o - 1}) \in (\mathbb{F}_{q^m}^{k_I})^{n_o}.$$

With this identification

$$v(D) = v_0 + v_1 D + v_2 D^2 \cdots \in \mathbb{F}_{q^m}^{k_I}[D]^{n_o}.$$

Finally, the codewords $x(D)$ of the concatenated code $\mathcal{C}$ are obtained by concatenating at each time instant v_i^j with the rank metric code $\mathcal{C}_I$ through G_I in the following way,

$$x_i^j = v_i^j G_I \in \mathbb{F}_{q^m}^{n_I}$$

which yields

$$x_i = (x_i^0, \dots, x_i^{n_o - 1}) \in (\mathbb{F}_{q^m}^{n_I})^{n_o}$$

and, therefore, the codewords are given by

$$x(D) = x_0 + x_1 D + x_2 D^2 + \cdots \in \mathcal{C} \subset \mathbb{F}_{q^m}^{n_I}[D]^{n_o}.$$

It is important to note that within this setting in each shot (time instant) we encode $v_i^j \in \mathbb{F}_{q^m k_I}$ and send to the network $x_i^j = v_i^j G_I \in \mathbb{F}_{q^m}^{n_I}$. Again the vector x_i^j can be seen as a matrix $X_i^j \in \mathbb{F}_q^{n_I \times m}$. Then, the rows of X_i^j can be regarded as the packets that are being introduced into the network at each shot.

Notice that in this scheme v_i^j must have length multiple of m, considering v_i^j over $\mathbb{F}_q$, which is restrictive, and to transmit each v_i we need $n_O = k_I$ shots. Hence, for instance, we need $3n_O$ shots instants to completely send $v(D) = v_0 + v_1 D + v_3 D^2$. This is a bit counter-intuitive in the context of convolutional codes, because D is typically used as a delay operator of one instant.

Next, we illustrate how this concatenated scheme processes the data with a simple example.

Example 4.1. *Suppose we want to send a file with 18 elements of* $\mathbb{F}_q$. *If we want to apply the scheme of [32] we can divide the information as follows:*

$$u(D) = u_0 + u_1 D + u_2 D^2 \in \mathbb{F}_{q^6}[D].$$

Then, we have $m = 3$ *and* $k_I = 2$.

Suppose that to encode it we have a machine (the convolutional encoder) with the capacity of storing 2 elements of $\mathbb{F}_{q^6}$, i.e., $\delta = 2$. *Then, we can consider the following encoder*

$$G(D) = G_0 + G_1 D + G_2 D^2 \in \mathbb{F}_{q^6}^{1\times 4}[D].$$

In other words, we use an $(n_o = 4, k_o = 1, \delta = 2)$ *convolutional code* $\mathcal{C}_o$ *over* $\mathbb{F}_{q^6}$ *as outer code, and obtain:*

$$v(D) = v_0 + v_1 D + v_2 D^2 + v_3 D^3 + v_4 D^4 = u(D) G_O(D) \in \mathbb{F}_{q^6}^4[D]$$

as a codeword of $\mathcal{C}_o$. *We now apply the inner code in each* $v_i^j \in \mathbb{F}_{q^6}$ *where* $v_i = (v_i^0, v_i^1, v_i^2, v_i^3)$, *for* $0 \leq i \leq 4$ *and* $0 \leq j \leq 3$. *To this end, we regard* v_i^j *as an element of* $\mathbb{F}_{q^3}^2$. *If we use an* $(n_I = 3, k_I = 2)$ *rank metric block code over* $\mathbb{F}_{q^3}$ *with encoder* $G_I \in \mathbb{F}_{q^3}^{2\times 3}$ *we obtain:*

$$x_i^j = v_i^j G_I \in \mathbb{F}_{q^3}^3.$$

We identify (via an isomorphism between $\mathbb{F}_{q^3}^3$ *and* $\mathbb{F}_q^{3\times 3}$*) these vectors to matrices*

$$X_i^j \in \mathbb{F}_q^{3\times 3},$$

where each row represent a packet with 3 elements in $\mathbb{F}_q$ *and we send 3 packets at each shot. As we require 4 shots to send each* v_i *and we have 5* $v_i' s$, *we need 20 shots to send the encoded file.*

In the following subsection we propose a new framework in which we consider an encoder machine which will operate over a smaller finite field and the concatenation is performed via the v_i instead of v_i^j. Hence, for instance, we need $i + 1$ instants (shots) to completely send $v(D) = v_0 + v_1 D + \cdots + v_i D^i \in \mathcal{C}_O$ and therefore, in this regard, one can say that such concatenation scheme is very natural since D is now used as delay operator of one time instant and v_i has no restrictions on the length.

4.2 Novel concatenation scheme over the base field

In this section we shall consider the general definition of rank metric block codes as described in Subsection 2.2.1 and operate over the base field $\mathbb{F}_q$ instead of the extension fields of $\mathbb{F}_q$.

Let us now consider $\mathcal{C}_O$ an (n_O, k_O, δ) Hamming metric convolutional code with encoder G_O, free distance $d_{free}(\mathcal{C}_O)$, column distance $d_j^c(\mathcal{C}_O)$ and $\mathcal{C}_I$ an $(n_I \times m, k_I)$ rank metric block code with rank distance $d_{rank}(\mathcal{C}_I)$ and encoder G_I.

As in the previous scheme, the concatenated code $\mathcal{C}$ is obtained by using the Hamming metric convolutional code $\mathcal{C}_O$ as an outer code and the rank metric block code $\mathcal{C}_I$ as an inner code.

Let $u(D) = u_0 + u_1 D + u_2 D^2 + \cdots \in \mathbb{F}_q[D]^{k_O}$ be the information vector. Then, we encode it through $G_O(D) \in \mathbb{F}_q[D]^{k_O \times n_O}$ in order to obtain $v(D) \in \mathcal{C}_O$ as:

$$v(D) = v_0 + v_1 D + v_2 D^2 + \cdots = u(D) G_O(D) \in \mathcal{C}_O \subset \mathbb{F}_q[D]^{n_O}.$$

The codewords of the concatenated code $\mathcal{C}$ will be obtained through the composition of an isomorphism ψ and a monomorphism γ (see Chapter 2) with $G_I \in \mathbb{F}_q^{k_I \times n_I m}$ and $k_I = n_O$ in the following way:

$$x_i = \gamma(v_i) = v_i G_I \in \mathbb{F}_q^{n_I m}$$

which is transformed into a matrix by

$$X_i = \psi(x_i) \in \mathcal{C}_I \in \mathbb{F}_q^{n_I \times m}.$$

Finally, the codewords of the concatenation code $\mathcal{C}$ are

$$X(D) = X_0 + X_1 D + x_2 D^2 + \ldots \in \mathcal{C} \subset \mathbb{F}_q[D]^{n_I \times m}.$$

Again, the rows of the matrix X_i can be seen as the packets that are injected into the network at time instant i.

Of course, for a given basis of $\mathbb{F}_{q^m}$ over $\mathbb{F}_q$, we can identify each matrix $X_i \in \mathbb{F}_q^{n_I \times m}$ with a vector $x_i \in \mathbb{F}_{q^m}^{n_I}$ and $X(D) \in \mathbb{F}_q[D]^{n_I \times m}$ with a polynomial $x(D)$ in $\mathbb{F}_{q^m}[D]^{n_I}$.

Next we illustrate how this novel concatenated scheme processes the data.

Example 4.2. *In Example 4.1 we have illustrated how to send a file of 18 elements in $\mathbb{F}_q$ using the scheme in [32] with an $(4,1,2)$ outer convolutional code $\mathcal{C}_o$ over $\mathbb{F}_{q^6}$ and a $(3,2)$ inner rank metric block code $\mathcal{C}_I$ over $\mathbb{F}_{q^3}$.*

We now show how can we use the concatenation scheme described in this section to send the same file using now a $(2,1,12)$ as outer convolutional code and a $(3\times 3,2)$ as inner rank metric block code both over $\mathbb{F}_q$ instead of over $\mathbb{F}_{q^6}$ and $\mathbb{F}_{q^3}$.

Note that both convolutional codes have the same memory, namely, the encoder can store 2 elements of $\mathbb{F}_{q^6}$ or, equivalently, 12 elements of $\mathbb{F}_q$ which means that δ is, in this case, equal to 12.

The data can be divided as

$$u(D)=u_0+u_1D+u_2D^2+\cdots+u_{17}D^{17}\in\mathbb{F}_q[D].$$

An encoder of $\mathcal{C}_o$ can be taken to be of the following form

$$G(D)=G_0+G_1D+G_2D^2+\cdots+G_{12}D^{12}\in\mathbb{F}_q^{1\times 2}[D]$$

and then

$$v(D)=v_0+v_1D+v_2D^2+\cdots+v_{29}D^{29}=u(D)G(D)\in\mathcal{C}_o\subset\mathbb{F}_q^2[D].$$

In this setting we now encode each $v_i\in\mathbb{F}_q^2$ via an encoder of $\mathcal{C}_I$, $G_I\in\mathbb{F}_q^{2\times 3.3}$, to obtain

$$x_i=\gamma(v_i)=v_iG_I\in\mathbb{F}_q^9$$

and

$$X_i=\psi(x_i)\in\mathbb{F}_q^{3\times 3}.$$

The rows of X_i represent the packets that are sent at each shot. Hence, again, each packet has 3 elements of $\mathbb{F}_q$ and we send 3 packets at each shot.

Within this framework and this selection of parameters we need 29 shots to transmit the file.

4.3 Distance properties

In this section we will present the distance properties of the proposed concatenated code $\mathcal{C}$ described in the previous subsection. Without loss of generality we will assume throughout the chapter that $m>n_I$.

Let $X(D) \in \mathcal{C}$. Recall that

$$\mathrm{rwt}(X(D)) = \sum_{i \in \mathbb{N}_0} \mathrm{rank}(X_i),$$

and the sum rank distance of $\mathcal{C}$ is defined as

$$d_{\mathrm{SR}}(\mathcal{C}) = \min\{\mathrm{rwt}(X(D)), X(D) \in \mathcal{C}, X(D) \neq 0\}.$$

Theorem 4.3. *The sum rank distance of the concatenated code $\mathcal{C}$ satisfies:*

$$d_{\mathrm{SR}}(\mathcal{C}) \geq \left\lceil \frac{d_{\mathrm{free}}(\mathcal{C}_O)}{n_O} \right\rceil \times d_{\mathrm{rank}}(\mathcal{C}_I).$$

Proof. The sum rank distance $d_{SR}(\mathcal{C})$ is the minimum sum rank of a nonzero codeword $X(D)$ of $\mathcal{C}$. To take a nonzero codeword $X(D)$ we should take a nonzero codeword $v(D) = v_0 + v_1 D + v_2 D^2 + \cdots$ of the outer code $\mathcal{C}_O$, which has a at least $d_{free}(\mathcal{C}_O)$ nonzero components v_i^j. So there are at least $\left\lceil \frac{d_{\mathrm{free}}(\mathcal{C}_O)}{n_O} \right\rceil$ nonzero v_i's. After inner encoding each of these nonzero v_i we obtain the corresponding X_i that have rank of at least $d_{\mathrm{rank}}(\mathcal{C}_{\mathcal{I}})$. This concludes the proof. □

Let $X(D)|_{[0,j]} = X_0 + \cdots + X_j D^j$ be the j-th truncation of the codeword $X(D)$, then

$$\mathrm{rwt}(X(D)|_{[0,j]}) = \sum_{i=0}^{j} \mathrm{rank}(X_i).$$

The column rank distance of $\mathcal{C}$ is defined as

$$d_j^{cr}(\mathcal{C}) = \min\{\mathrm{rwt}(X(D)|_{[0,j]}), X(D) \in \mathcal{C} \wedge X_0 \neq 0\}.$$

Theorem 4.4. *The column sum rank distance of the concatenated code $\mathcal{C}$ satisfies:*

$$d_j^{cr}(\mathcal{C}) \geq \left\lceil \frac{d_j^c(\mathcal{C}_O)}{n_O} \right\rceil \times d_{\mathrm{rank}}(\mathcal{C}_I).$$

Proof. For any $v(D) \in \mathcal{C}_O$, with $v_0 \neq 0$ we have that $v(D)|_{[0,j]}$ has at least $d_j^c(\mathcal{C}_O)$ nonzero components. Then, $v(D)|_{[0,j]}$ has at least $\left\lceil \frac{d_c^j(\mathcal{C}_O)}{n_O} \right\rceil$ nonzero v_i's. Each v_i different from zero results into a x_i, or equivalently, into X_i that has rank of at least $d_{\mathrm{rank}}(\mathcal{C}_{\mathcal{I}})$. Then, the theorem follows. □

As explained before, Rosenthal and Smarandache [18] showed that the free distance of an (n, k, δ) convolutional code is upper bounded by the generalized Singleton bound,

$$d_{\text{free}}(\mathcal{C}) \leq (n-k)\left(\left\lfloor \frac{\delta}{k} \right\rfloor + 1\right) + \delta + 1.$$

and the column distance, see [11], is upper bounded by

$$d_j^c \leq (n-k)(j+1) + 1.$$

Note that the rank distance is upper bounded by the Hamming distance, *i.e.*, for all $x \in \mathbb{F}_{q^m}^n$ and a corresponding $X \in \mathbb{F}_q^{n \times m}$ we have that $\text{rwt}(X) \leq \text{wt}(x)$. Note that this fact is independent of the basis chosen when identifying x to X. Using this obvious fact, we derive the following results.

Theorem 4.5. *The sum rank distance of the concatenated code $\mathcal{C}$ satisfies*

$$d_{\text{SR}}(\mathcal{C}) \leq n_I \left(\left\lfloor \frac{\delta}{k_O} \right\rfloor + 1\right).$$

Proof. Let G(D) be a row reduced encoder of $\mathcal{C}_o$. Let us consider a row of $G_o(D)$ with degree $\left\lfloor \frac{\delta}{k_O} \right\rfloor$, which is a codeword of $\mathcal{C}_o$.That means that exists a codeword $v(D) = \sum_{i \geq 0} v_i D^i \in \mathcal{C}_O$ with maximum number of nonzero coefficients equal to $\left\lfloor \frac{\delta}{k_O} \right\rfloor + 1$. Each nonzero coefficient of $v(D)$, v_i, is encoded by the (n_I, k_I) inner code $\mathcal{C}_I$ which obviously yields a codeword with rank $\leq n_I$. Thus, $d_{\text{SR}}(\mathcal{C}) \leq n_I \left(\left\lfloor \frac{\delta}{k_O} \right\rfloor + 1\right)$. □

Analogously, we can derive an upper bound on the column rank distance of the concatenated code as we show in the next result.

Theorem 4.6. *The column sum rank distance of the concatenated code $\mathcal{C}$ satisfies*

$$d_j^{cr}(\mathcal{C}) \leq n_I (j+1).$$

Proof. Following the same reasoning as in the previous theorem, we know that the maximun number of nonzero coefficients of a truncated codeword of $\mathcal{C}_o$ is equal to $j+1$. In other words, there are at most $j+1$ nonzero $v_i's$ in $v(D)|_{[0,j]}$. Since each nonzero v_i leads to a nonzero X_i with $\text{rank}(X_i) \leq n_I$, we can conclude that $d_j^{cr} \leq (j+1)n_I$. □

4.4 Performance of the concatenated code

In this section we will evaluate the performance of the proposed concatenation scheme and then compare it to the one presented in [32] in terms of performance during the encoding and decoding process.

Consider packet loss only (no erroneous packets) in the network. As explained in [21] the inner (n_I, k_I) code with rank distance $d_I = n_I - k_I + 1$ is able to correct up to $d_I - 1$ lost packets. If the number ℓ of lost packets is more, then the inner decoder gives to the outer code the symbol of erasure. Assume that we transmit via network n_I linearly independent packets, and each packet can be lost with probability p independently on other packets. Then probability p_o of symbol erasure of the outer code is

$$p_o = \sum_{\ell=d_I}^{n_I} \binom{n_I}{\ell} p^{\ell}(1-p)^{n_I-\ell}. \tag{4.1}$$

After decoding sufficient inner codes, we decode outer convolutional code. Assume that blocks of the outer convolutional code are correctly decoded up to instance $t-1$ and there are erasures in the block t. At this stage we need to use the results of Napp, Pinto and Sidorenko [32] on the decoding of convolutional codes over the erasure channel that will help us to evaluate the performance of our code. This is presented next.

Let $\mathcal{C}_o$ be an (n_o, k_o, δ) convolutional code, $d^c_T(\mathcal{C}_O)$ be its T-th column distance and let $H(D) = H_0 + H_1 D + H_2 D^2 + \cdots + H_\gamma D^\gamma$ be a parity-check matrix of $\mathcal{C}_O$.

Assume that we have been able to correctly decode up to an instant $t-1$. Then, for each received codeword $v(D) = v_0 + v_1 D + v_2 D^2 + \cdots \in \mathcal{C}_O \subset \mathbb{F}[D]^{n_o}$ consider the system of linear equations

$$\left[\begin{array}{cccc|cccc} H_\gamma & \cdots & \cdots & H_1 & H_0 & & & \\ & H_\gamma & & & H_1 & H_0 & & \\ & & \ddots & & \vdots & \vdots & \ddots & \\ & & H_\gamma & \cdots & \underbrace{H_T}_{} & \underbrace{H_{T-1} \quad \cdots}_{H^c_T} & & H_0 \end{array}\right] \left[\begin{array}{c} v_{t-\gamma} \\ \vdots \\ v_{t-1} \\ \hline v_t \\ \vdots \\ v_{t+T} \end{array}\right] = 0 \tag{4.2}$$

where v_i, $t \leq i \leq t+T$ may contain some erasures on it and v_i, $t-\gamma \leq i < t$ are assumed to be correct.

If we consider the columns of H^c_T that correspond to the coefficients of the erased elements to form a new matrix, which will be denoted by $\widehat{H}_T^c$, then the remaining columns of H_T^c, denoted by $\widetilde{H}$, can help us to compute the independent terms of a system, *i.e.*, if $\widetilde{v}$ is the sub-vector of $v_{[t,\ldots,t+T]}$ corresponding to $\widetilde{H}$, which are assumed to be the known coefficients in (4.2), then we will be able to obtain the non-homogeneous linear system with $(T+1)(n-k)$ equations,

$$\widehat{H}_T^c Y = -\widetilde{H}\widetilde{v}, \tag{4.3}$$

where Y corresponds to the vector with the erasures in $v_{[t,\ldots,t+T]}$.
Note that this system has always a solution since $v(D) \in \ker H(D)$. Therefore, it will be possible to recover all the existing erasures in $v_{[t,\ldots,t+T]}$ if and only if the system (4.3) has a unique solution.

Lemma 4.7. *[32, Lemma 2] Let $\mathcal{C}_o$ be an (n_o, k_o, δ) Hamming metric convolutional code and let $d_T^c(\mathcal{C}_o)$ be its T-th column distance, $T \leq L$ and L has defined in Definition 2.17. Assume that we have been able to correctly decode up to an instant $t-1$. Let $E(t, t+T)$ be the number of erasures occurring in the time interval $[t, t+T]$. Then, we can recover v_t if*

$$E(t, t+T) \leq d_H^T(\mathcal{C}_o) - 1.$$

The authors in [32] presented a necessary conditions to recover all the erasures within an interval in case we have information about the number of erasures per time instant.

Lemma 4.8. *[32, Lemma 3] Let $d_0^c(\mathcal{C}_o), d_1^c(\mathcal{C}_o), \ldots, d_L^c(\mathcal{C}_o)$ be the distance profile of an (n_o, k_o, δ) Hamming metric convolutional code $\mathcal{C}_o$. Let E_i be the number of erasures at a time instant i. Assume that we have been able to correctly decode up to an instant $t-1$. Then, we can completely decode up to an instant $t+T$ where $T \leq L$ if*

$$\sum_{i=0}^{s} E_{T-i+t} \leq d_s^c(\mathcal{C}_o) - 1 \text{ for } s = 0, 1, \ldots, T.$$

Of course the best convolutional codes in this scenario is when they have the largest possible column distances, *i.e.*, is an MDP convolutional code, see Definition 2.17. According to the authors, the conditions of the previous results become also sufficient when considering MDP convolutional codes as we will show in the following theorem.

Theorem 4.9. *[32, Theorem 6] Let $\mathcal{C}_o$ be an MDP (n_o, k_o, δ) convolutional code. Assume that we have been able to correctly decode up to an instant $t-1$. Let E_i be the* *number of erasures at time instant i. Then, we can completely decode up to an instant $t+T$ where $T \leq L$ if and only if*

$$\sum_{i=0}^{s} E_{T-i+t} \leq (n_o - k_o)(s+1) \textit{ for } s = 0, 1, \ldots, T.$$

We will now consider the concatenated codes proposed above and show the conditions necessary, in both cases, to fully recover from the missing packets. The error-correcting capabilities of the concatenated code will depend on how the packet losses are distributed along $(X_t, \ldots, X_{t+T})$ and we will illustrate by showing some examples.

For the sake of simplicity, in order to measure the performance of the two concatenation schemes presented above we shall consider only lost packets over the network. If the inner rank metric code fails to recover the packets at a time instant, then it will deliver an erasure to the outer convolutional code. Then, using Lemma 4.7 and 4.8 and Theorem 4.9, the convolutional code will try to correct these erasures to recover the lost packets that the inner code could not recover.

As explained above, after decoding sufficient inner codes, we decode outer convolutional code. Assume that blocks of the outer convolutional code are correctly decoded up to instance $t-1$ and there are erasures in the block t. According to Lemma 4.7, we can recover tth block v_t using window of size $T+1$ of blocks $v_t, v_{t+1}, ...v_{t+T}$ if number of erasures in the window is less than $\hat{d} = D_H^T(\mathcal{C}_o)$, otherwise the decoder fails. Hence, the failure probability P_f is the probability to have at least $\hat{d}$ erasures in the $T+1$-window given by (4.4), where at least one erasure should be in the block v_t (see correction (4.5)). Since every symbol of the outer code can be independently erased with probability p_o, we obtain the failure probability

$$\begin{aligned} P_f(p) &= \sum_{\ell=\hat{d}}^{n_o(T+1)} \binom{n_o(T+1)}{\ell} p_o^{\ell}(1-p_o)^{n_o(T+1)-\ell} && (4.4)\\ &\quad -\sum_{\ell=\hat{d}}^{n_oT} \binom{n_oT}{\ell} p_o^{\ell}(1-p_o)^{n_oT-\ell}. && (4.5)\end{aligned}$$

In order to analyze the performance of the concatenation schemes, a model, such as the one described by the Elliot channel, would help to understand better the error correction capabilities of the code. This is left for future research. Next We illustrate the decoding process of both schemes in the following semple example.

Example 4.10. *Let us consider Example 4.1 and 4.2 and see how these two different concatenation schemes perform when we have lost packets during the transmission of a file over a network.*

Suppose that the lost packets pattern at each shot is the one represented in Figure 4.1. For the sake of simplicity we just analyze the situation after the first eight shots. Suppose that both $\mathcal{C}_I$ and $\mathcal{C}_O$ are, respectively, MRD and MDP, in both examples.

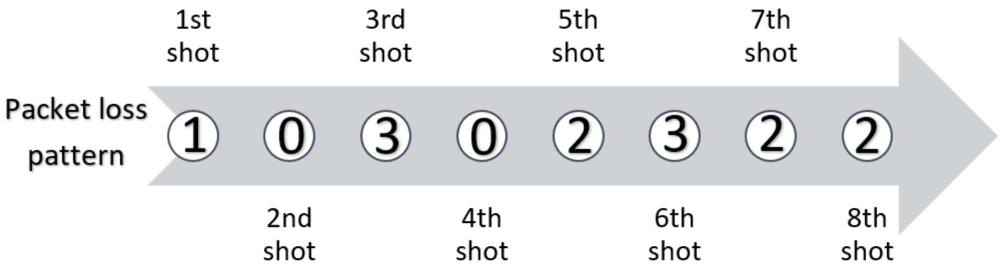

Figure 4.1: Lost packets at each shot

The concatenated code in Example 4.1 using [32] can recover, with $\mathcal{C}_I$, at most $n_i - k_i = 3 - 2 = 1$ lost packets at each shot whereas the inner rank metric code in Example 4.2 can correct up to $n_I - \lceil \frac{k_I}{m} \rceil = 3 - \lceil \frac{2}{3} \rceil = 2$ packets each shot.

Hence, the first scheme can fully recover X_0^0, X_0^1, X_0^3 and fails to retrieve $X_0^2, X_1^0, X_1^1, X_1^2$ and X_1^3. Then, the outer code receives v_0 with one erasure and v_1 with 4 erasures, as represented in the following figure,

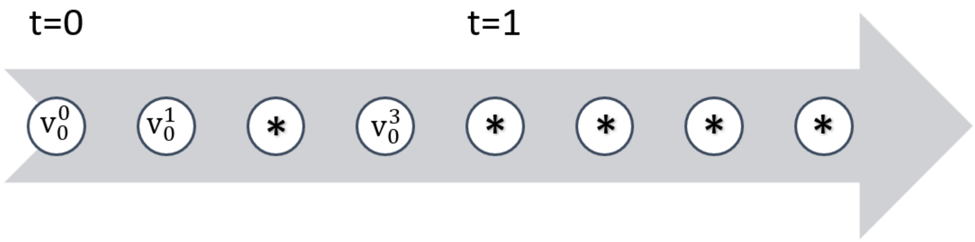

Figure 4.2: Erasure pattern

Taking into account that $d_0^c = 3$ and using Theorem 4.9 with $t = 0$ and $T = 0$ (consequently $s = 0$) we have that v_0 can be fully recovered as

$$E_0 = 1 < (n_O - k_O)(s + 1) = (4 - 1)(0 + 1) = 3.$$

Next, using again Theorem 4.9 with $t = 1$ and $T = 0$ (again $s = 0$) we have that

$$E_1 = 4 > (n_O - k_O)(s + 1) = (4 - 1)(0 + 1) = 3$$

and therefore v_1 cannot be recovered at time instant 8 (4 shots to receive v_0 and 4 shots to receive v_1).

In the other hand, for the scheme in Section 4.2 we have that X_0, X_1, X_3, X_4, X_6 and X_7 can be recovered and so v_0, v_1, v_3, v_4, v_6 and v_7 are received without erasures. The vectors v_2 and v_5 are considered by the outer code as erasures, as we can see in the Figure 4.3. Hence, $E_0 = E_1 = E_3 = E_4 = E_6 = E_7 = 0$ and $E_2 = E_5 = 1$.

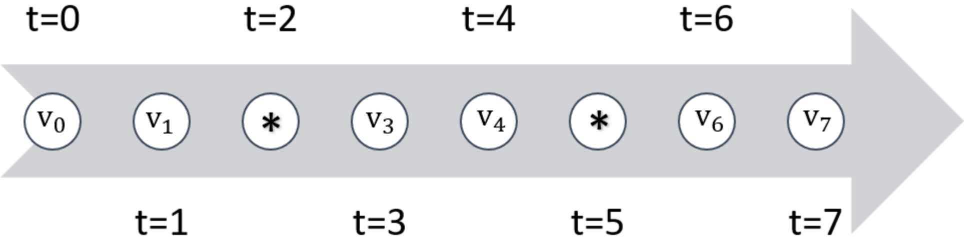

Figure 4.3: Erasure pattern

So, we have received the first erasure at time $t = 2$. Now the conditions of Theorem 4.9 with $t = 2$ and $T = 2$ are verified:
For $s = 0$,

$$E_4 = 0 \leq (n_O - k_O)(s + 1) = (2 - 1)(0 + 1) = 1 \times 1 = 1,$$

for $s = 1$,

$$E_3 + E_4 = 0 + 0 = 0 \leq (n_O - k_O)(s + 1) = (2 - 1)(1 + 1) = 1 \times 2 = 2,$$

and for $s = 2$,

$$E_2 + E_3 + E_4 = 1 + 0 + 0 = 1 \leq (n_O - k_O)(s + 1) = (2 - 1)(2 + 1) = 1 \times 3 = 3.$$

This means that we can correct v_2.

So, we can now assume we have no erasures up to time instant $t = 5$. Finally, sliding the correction window we now take $t = 5$ and $T = 2$ in Theorem 4.9 and verify

that the conditions are satisfied:
For $s = 0$,

$$E_7 = 0 \leq (n_O - k_O)(s+1) = (2-1)(0+1) = 1 \times 1 = 1,$$

for $s = 1$,

$$E_7 + E_6 = 0 + 0 = 0 \leq (n_O - k_O)(s+1) = (2-1)(1+1) = 2,$$

and for $s = 2$,

$$E_7 + E_6 + E_5 = 0 + 0 + 1 = 1 \leq (n_O - k_O)(s+1) = (2-1)(1+2) = 3.$$

This implies that we can recover v_5 *and therefore we can decode everything in the first eight instants of the transmission.*

Chapter 5

Conclusions

A great part of the existing literature about network coding is concerned with the situation in which the network is used only once to propagate the information, *i.e.*, a fixed number of packets are encoded and sent via the network at one time instant (one-shot network codes). In order to achieve a reliable communication over network channels, one-shot matrix codes called rank metric codes were constructed. However, if one needs to transmit a lot of information and needs to use the network several instants, then one can improve the error-correction capability of the code by creating correlation among the transmitted data in the different shots (Multi-shot network codes). This new class of codes has recently attracted much attention due to their application streaming communications.

In this thesis a number of problems regarding codes for multi-shot networks have been investigated. In particular, the thesis focus is twofold. The first one has to do with the rank metric analogues of Hamming metric convolutional codes. We first introduce a novel definition of rank metric convolutional codes and then we study their rank distance properties within this new setting. Several upper bounds are derived which allowed us to define define Strongly Maximum Rank Distance (sMRD) and Maximum Rank Distance Profile (MRP) convolutional codes.

Despite of the fact that the distance of a code is the most important single parameter of a code, very little was known about the rank distance properties of these codes. In this dissertation we have focused on the two distances that are considered the most relevant in the context of rank metric convolutional codes, namely, the free sum rank distance and the column rank distance.

Once we have established the proper notions in Chapter 2, we aim to derive concrete constructions of MRD convolutional codes in Chapter 3. Extending the previous constructions of these codes, we present novel and more general constructions for a wider set of parameters. More specifically, we intend to build codes with higher degree and, consequently, better error-correcting capability.

The second part of this dissertation propose a new coding framework for multi-shot networks as alternative to rank metric convolutional codes. The novel scheme presented here consists on using concatenation and requires vectors over smaller finite fields which reduce the complexity of the encoding and decoding process. This novel scheme is built by concatenating of a rank metric code as an inner code to an outer convolutional code.

We showed that our new concatenation scheme have some advantages with respect to previous coding solutions for sequential transmission over networks in multiple shots. In fact, we showed that the inner code (the rank metric) is able to recover lost packets that remain lost in other concatenation schemes.

The thesis raises several interesting follow-up questions. Is it possible to derive MRD and MRP convolutional code over more general set of parameters and smaller fields? This question remains widely open. Another challenging avenue of future research is to analyze the distance properties of the proposed codes in terms of different metrics, for instance, the injection metric [23]. Also it would be interesting to investigate the performance of the proposed concatenation scheme considering not only rank deficiencies but also other type of errors, such as the situation when injected error packets occur. For that a nontrivial decoding algorithm needs to be developed to deal with such errors. Some preliminary ideas and results regarding fast decoding algorithms have been presented in [29]. In this work the authors presented a new construction of maximum rank distance systematic rank metric convolutional codes was presented that allows to reduce the computational complexity of the decoding Viterbi algorithm. This result is achieved by lowering the number of branch metrics to be calculated and by setting to the highest value the metric of the remaining edges in the trellis.

Finally, another open issue that one can naturally raises is to investigate how would be the performance of the concatenation scheme of Chapter 4 in more real situations. For this purpose, one should develop a statistical model, *e.g.*, one may consider the Gilbert-Elliott channel, to simulate burst error patterns in transmission channels like the Internet. As the proposed scheme is very general, we would expect that a statistical analysis will allow us to derive more concrete parameters of the inner and outer codes in order to achieve a good performance in these channels. This is left as an interesting open problem.

Index

Bibliography

[1] R. Ahlswede, N. Cai, S. Li, and R. W. Yeung. Network information flow. *IEEE Trans. Inf. Th*, 46(4):1204–1216, 2000.

[2] P. Almeida and D. Napp. A new rank metric for convolutional codes. *Designs, Codes and Cryptography*, 89:53–73, 2021.

[3] J. Apostolopoulos, A. Badr, A. Khisti, and Wai-Tian. Tan. Layered constructions for low-delay streaming codes. *IEEE Trans. Inform. Theory*, 63(1):111–141, 2017.

[4] A. Badr, A. Khisti, and R. Mahmood. Convolutional codes with maximum column sum rank for network streaming. *IEEE Trans. Inform. Theory*, 62(6):3039–3052, 2016.

[5] R.C. Bose and D.K. Ray-Chaudhuri. On a Class of Error Correcting Binary Group Codes. *Information and Control*, 3:68–79, 1960.

[6] D.J. Costello and S. Lin. *Error Control Coding: Fundamentals and Applications.* Prentice-Hall, Inc, Englewood Cliffs, New Jersey, 1983.

[7] P. Delsarte. Bilinear forms over a finite field, with applications to coding theory. *Journal of Combinatorial Theory*, 25(3):226–241, 1978.

[8] P. Elias. Coding for noisy channels. In *IRE International Convention Record, pt. 4*, pages 37–46, 1955.

[9] E. M. Gabidulin. Theory of codes with maximum rank distance. *Problems of Information Transmission*, 21(1):1–12, 1985.

[10] F. Gantmacher. *The Theory of Matrices, vol.I.* Chelsea Publishing Company, New York, 1977.

[11] H. Gluesing-Luerssen, J. Rosenthal, and R. Smarandache. Strongly MDS convolutional codes. *IEEE Trans. Inform. Theory*, 52(2):584–598, 2006.

[12] M. Greferath, M. Pavčević, N. Silberstein, and M. Vázquez-Castro, editors. *Network Coding and Subspace Designs.* Springer International Publishing, 2018.

[13] R.W. Hamming. Error Detecting and Error Correcting Codes. *Bell System Technical Journal*, 29:147–160, 1950.

[14] A. Hocquenghem. Codes Correcteurs d'Errors. *Chiffres (Paris)*, 2:147–156, 1959.

[15] C. Huffman and V. Pless. *Handbook of Coding Theory, volumes 1,2.* Elsevier Sciences, North-Holland, 1998.

[16] W. Huffman, J. Kim, and P. Sole. *Concise Encyclopedia of Coding Theory.* Chapman & Hall, CRC Press, 2021.

[17] R. Johannesson and K. Sh. Zigangirov. *Fundamentals of Convolutional Coding.* IEEE Press, New York, 1999.

[18] J.Rosenthal and R.Smarandache. Maximum distance separable convolutional codes. *Appl. Algebra Engrg. Comm. Comput.*, 10(2):15–32, 1999.

[19] H. Gluesing-Luerssen; J.Rosenthal and R.Smarandache. Strongly-MDS convolutional codes. *IEEE Transations on Information theory*, 52(2), 2006.

[20] T. Kailath. *Linear Systems.* Englewood Cliffs, N.J., Prentice Hall, 1980.

[21] R. Koetter and F. Kschischang. Coding for errors and erasures in random network coding. *IEEE Trans. Inf. Th*, 54(8):3579–3591, 2008.

[22] R. Koetter, F. R. 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.

[23] F. Kschischang and D. Silva. On metrics for error correction in network coding. *IEEE Transactions on Information Theory*, 55(12):5479–5490, 2009.

[24] F. J. MacWilliams and N. J.A. Sloane. *The Theory of Error-Correcting Codes.* North Holland, Amsterdam, 1977.

[25] R. Mahmood. Rank metric convolutional codes with applications in network streaming. Master of applied science, 2015.

[26] R. J. McEliece. The algebraic theory of convolutional codes. In V. Pless and W.C.Human, editors, *Handbook of Coding Theory, volume 1*, pages 1065–1138. Elsevier, Amsterdam, 1998.

[27] D. Napp, R. Pinto, J. Rosenthal, and F. Santana. Column rank distances of rank metric convolutional codes. In A. Barbero, V. Skachek, and O. Ytrehus, editors, *Coding Theory and Applications*, pages 248–256. Springer International Publishing, 2017.

[28] D. Napp, R. Pinto, J. Rosenthal, and P. Vettori. Rank metric convolutional codes. *Proceedings of the 22nd International Symposium on Mathematical Theory of Network and Systems (MTNS), Minnesota, USA*, 2016.

[29] D. Napp, R. Pinto, J. Rosenthal, and P. Vettori. Faster decoding of rank metric convolutional codes. *Proceedings of the 23rd International Symposium on Mathematical Theory of Network and Systems (MTNS), Hong Kong*, pages 507–510, 2018.

[30] D. Napp and F. Santana. Multi-shot network coding. In Marcus Greferath, Mario Osvin Pavčević, Natalia Silberstein, and María Ángeles Vázquez-Castro, editors, *Network Coding and Subspace Designs*, pages 91–104. Springer International Publishing, Cham, 2018.

[31] R.W. Nóbrega and B.F. Uchoa-Filho. Multishot codes for network coding using rank-metric codes. In *Wireless Network Coding Conference (WiNC), 2010 IEEE*, pages 1–6, 2010.

[32] D. Napp; R. Pinto and V. Sidorenko. Concatenation of convolutional codes and rank metric codes for multi-shot network coding. *Springer Science+Business Media New York*, 86:303–318, 2017.

[33] I.S. Reed and G. Solomon. Polynomial Codes over Certain Finite Fields. *J. SIAM*, 8:300–304, 1960.

[34] V. Sidorenko, M. Stinner, and A. Wachter-Zeh. Convolutional codes in rank metric with application to random network coding. *IEEE Trans. Inform. Theory*, 61(6):3199–3213, 2015.

[35] J.H. van Lint. *Introduction to Coding Theory.* Springer-Verlag, Berlin, 1999.

[36] P. Verlinde. Error detecting and correcting codes. In Hossein Bidgoli, editor, *Encyclopedia of Information Systems*, pages 203–228. Elsevier, New York, 2003.