Università di Pisa

Facoltà di Scienze Matematiche, Fisiche e Naturali
Corso di Laurea in Matematica

Decoding Reed Solomon and BCH Codes beyond their Error-Correcting Radius: an Euclidean Approach

Tesi di Laurea Magistrale

Candidato
Alessandro Neri

Relatore
Prof.ssa Patrizia Gianni
Università di Pisa

Controrelatore
Prof. Roberto Dvornicich
Università di Pisa

Anno Accademico 2013/2014

Contents

Introduction **iii**

1 Coding Theory **1**
1.1 What is a code 2
1.2 Detection and correction of the errors 4
1.3 Linear codes 5
1.4 Cyclic codes 5
1.5 BCH codes 8
1.6 Reed-Solomon codes 9
1.7 Decoding BCH codes 9

2 The linear division problem **14**
2.1 Formalization of the problem 15
2.2 The study of the divisibility ideal 18
2.3 Bound on the cardinality of $\mathcal{V}(\mathcal{I}_{\mathcal{F},H})$ 22
2.3.1 A conjecture on the bound 26

3 List Decoding **28**
3.1 Decoding BCH codes 29
3.2 Unique decoding 30
3.3 The decoding function 31
3.4 Fundamental Theorems for list decoding 35
3.5 Decoding binary BCH codes 38
3.6 Decoding up to a certain error-threshold 44

4 Bounds on list decoding **49**
4.1 General bounds 50
4.2 Madhu Sudan bound 52
4.3 Reduction to a packing set problem 54
4.4 Bounds for some special cases 57

5 Computational aspects of list decoding **60**
5.1 Calculation of the divisibility ideal 60
5.2 Computing the divisibility set 63
5.3 Some special cases 63
5.3.1 Binary BCH codes with $e = t + 1$ 63
5.3.2 RS and binary BCH codes with $e = t + 1$ 64
5.4 Sudan and Guruswami-Sudan algorithms 73
5.5 Comparison between the algorithms 76

6 Examples **77**
6.1 Binary BCH code with $n = 31$ and $\delta = 11$ 77
6.2 $[15, 7, 9]$ RS code 78
6.3 $[15, 7, 9]$ RS code 81
6.4 $[12, 7, 6]$ RS code 83
6.5 $[10, 2, 9]$ RS code 84

Introduction

Coding theory and error-correcting codes naturally arise as a consquence of tha practical problem of data transmission. Data can be corrupted during the transmission, so that the information received can present some errors. Therefore the aim of error correcting codes is both detecting and correcting errors that occur after the transmission of data.

Formally, given a finite set $\mathcal{Q}$, called *alphabet*, a code is a subset C of $\mathcal{Q}^n$, whose elements are called *codewords*. At the beginning, coding theory developed in the direction of unique decoding, in which an algorithm is expected to output a single codeword. However it was realized early on that unambiguous decoding is not the only useful notion of recovery from error. In the 1950's Elias [4] proposed the notion of *list decoding*. List decoding generalizes the notion of error-correction, when the number of errors is potentially very large, and main idea behind it is that the decoding algorithm instead of outputting a single possible message outputs a list of possibilities one of which is correct.
Algorithmically, this problem is stated as follows:

Definition. The *list decoding* problem for a code C is a process that takes as input a received word $x \in \mathcal{Q}^n$ and an error bound e, and then outputs a list of all codewords $c^{(1)}, \ldots, c^{(s)} \in C$ that differ from x in at most e places.

From 1995 to 2007, more efficient list-decoding algorithms were progressivly developed. In particular there are two relevant algorithms. The first known polynomial time list decoding algorithm for Reed–Solomon codes was developed by Madhu Sudan in 1995. An improvement on it was made by Madhu Sudan and his then doctoral student Venkatesan Guruswami in 1999, and it corrects up to $(1 - \sqrt{R})n$ errors, where R is the code *rate*.

In this work we present a different algorithm for list decoding of Reed Solomon and BCH codes, based on the Euclidean division algorithm. For both BCH and Reed Solomon codes the decoding procedure involves the computation of the so-called *error-locator polynomial*. When we try to correct a number of errors which does not exceed the error-correction radius the error-locator polynomial is the unique (up to scalar) kernel generator of the

syndrome matrix.

If we try to correct more errors than the error-correcting radius, such a kernel will have dimension greater than 1. Another property of the error-locator polynomial is that it divides the polynomial $x^n - 1$. Hence, given a basis $\{f_0, \ldots, f_k\}$ of such a kernel, the problem that naturally arises is the problem of determining all the linear combinations of the f_is that divide $x^n - 1$.

For this purpose, after a first introductory chapter about general notions on algebraic coding theory, the second chapter is about the *linear division problem*. Given a set of univariate polynomials $f_0, f_1, \ldots, f_k$ and H, with coefficients in a finite field $\mathbb{F}_q$, we would like to find all the linear combinations of the f_is that divide H. This problem can be reduced to the problem of finding the variety of a particular ideal of the ring $\mathbb{F}_q[a_1, \ldots, a_k]$, the so-called *divisibility ideal*.

In the third chapter we deal with the list decoding problem by using the results of linear division. Given a received word r and an integer e greater than the error correcting radius, we try to find all the codewords with distance at most e from r. Hence we define a *decoding function* $\mathcal{D}_{r,e}$ which takes a zero of the divisibility ideal, and it outputs another word c. We show that for binary BCH codes and for Reed Solomon codes such a word c is a codeword with distance at most e from r. Moreover such a function is surjective onto the set $L(r, e) = \{c \in C \mid d(c, r) \leq e\}$. In particular the algorithm developed here permits to solve the list decoding problem.

In the fourth chapter we find some bounds on the number of codewords the algorithm outputs, i.e. on the cardinality of the set $L(r, e)$. Some of those bounds are well-known in coding theory, others are obtained by using algebraic and combinatorial techniques. The motivation is that we need the combinatorial guarantee that any Hamming ball of radius e around a received word r has a small number of codewords. This is because the list size itself is clearly a lower bound on the running time of the algorithm, and we want to be sure that the problem of list decoding is treatable.

The algorithm that we propose, unlike Guruswami-Sudan algorithm, works for every kind of Reed Solomon code and for every binary BCH code, without any condition on the rate and on the number of errors that we would like to recover. However, in the general case, this algorithm has a high computational cost, due to the fact that it involves the computation of a Gröbner basis. In the fifth chapter we indeed analyze the main computational aspects of this approach, and we propose an alternative way to solve the problem without involving Gröbner basis, that works only for some special cases. In these cases it seems that the algorithm is very efficient, especially for binary BCH codes.

In the last chapter we present some examples that show how the algorithm works step by step. Some of these examples follows by introducing random errors in a transmitted codeword, while some others are constructed ad hoc, in order to find as many codewords as possible in a Hamming ball of radius e, as shown in [9].

Chapter 1

Coding Theory

Information passes from a source to a sink via a conduit or channel. In our view of communication we are allowed to choose exactly the way information is structured at the source and the way it is handled at the sink, but the behaviour of the channel is not in general under our control. The unreliable channel may take many forms. We may communicate through space, such as talking across a noisy room, or through time, such as writing a book to be read many years later. The uncertainties of the channel, whatever it is, allow the possibility that the information will be damaged or distorted in passage.

Communication across space has taken various sophisticated forms in which coding has been used successfully. Indeed Shannon, Hamming, and many of the other originators of mathematical communication theory worked for Bell Telephone Laboratories. They were specifically interested in dealing with errors that occur as messages pass across long telephone lines.

The usual way to represent, manipulate, and transmit information is to use bit strings, that is, sequences of zeros and ones. It is extremely difficult, and often impossible, to prevent errors when data are stored, retrieved, operated on, or transmitted. Errors may occur from noisy communication channels, electrical interference, human error, or equipment error. Similarly, errors are introduced into data stored over a long period of time on magnetic tape as the tape deteriorates.

The objective of error-correcting codes is to protect a message going through a noisy channel by introducing redundancy to the message. In a nutshell, an error-correcting code is just a pair of (encoding/decoding) mappings that convert a message to and from a codeword.

1.1 What is a code

Coding theory, the study of codes, including error detecting and error correcting codes, has been studied extensively for the past sixty years. It has become increasingly important with the development of new technologies for data communications and data storage. In this chapter we will study both error detecting and error correcting codes.

We are concerned here with block coding. That is, we transmit blocks of symbols block coding of fixed length n from a fixed alphabet $\mathcal{Q}$. These blocks are the codewords, and that codeword transmitted at any given moment depends only upon the present message, not upon any previous messages or codewords. Our encoder has no memory. We also assume that each codeword from the code, that is the set of all possible codewords, is as likely to be transmitted as any other.

Let us formalize the concept of error-correcting codes. From now on we assume that information is coded using an *alphabet* $\mathcal{Q}$, with $|\mathcal{Q}| = q$.

Definition 1.1.1. A *q-ary code* C is a non-empty subset of $\mathcal{Q}^n$. The elements $c \in C$ are the *codewords* and n is called the *block length* or just *length*.
If $|C| = 1$ we call the code *trivial*.
If $q = 2$ the code is called *binary*.

Definition 1.1.2. If $x \in \mathcal{Q}^n$, $y \in \mathcal{Q}^n$, then the *Hamming distance* $d(x, y)$ of x and y is a function

$$d : \mathcal{Q}^n \times \mathcal{Q}^n \longrightarrow \mathbb{N}$$

defined by

$$d(x, y) := |\{i | 1 \leq i \leq n, x_i \neq y_i\}| .$$

The *weight* $w(x)$ of x is a function

$$w : \mathcal{Q}^n \longrightarrow \mathbb{N}$$

defined by

$$w(x) := d(x, 0).$$

Theorem 1.1.3. *Let $d(x, y)$ represent the Hamming distance between the strings x and y of length n. Then:*

1. *$d(x, y) \geq 0$ for all x, $y \in \mathcal{Q}^n$;*
2. *$d(x, y) = 0$ if and only if $x = y$;*
3. *$d(x, y) = d(y, x)$ for all x, $y \in \mathcal{Q}^n$;*

4. $d(x, y) \leq d(x, z) + d(z, y)$ *for all* x, y, $z \in \mathcal{Q}^n$.

□

The Hamming distance is indeed a metric on $\mathcal{Q}^n$. If we are using a channel with the property that an error in position i does not influence other positions and a symbol in error can be each of the remaining $q - 1$ symbols with equal probability, then Hamming distance is a good way to measure the error content of a received message.

Suppose that when a codeword x from a code C is sent, the string r is received. If the transmission was error-free, then r would be the same as x. But if errors were introduced by the transmission, for instance by a noisy line, then r is not the same as x. How can we detect errors? And how can we correct errors, that is, how can we recover x?

One approach would be to compute the Hamming distance between r and each of the codewords in C. Then to decode r, we take the codeword of minimum Hamming distance from r, if such a codeword is unique. If the distance between the closest codewords in C is large enough and if sufficiently few errors were made in transmission, this codeword should be x, the codeword sent. This type of decoding is called *nearest neighbour decoding*, or *minimum distance decoding*.

Now we introduce some concepts that play an essential role in coding theory.

Definition 1.1.4. The *minimum distance* d of a nontrivial code C is

$$d := \min \{d(x, y) | x \in C, y \in C, x \neq y\} .$$

The *minimum weight* w of C is given by

$$w := \min \{w(x, 0) | x \in C, x \neq 0\} .$$

Sometimes we shall be interested in knowing how far a received word can be from the closest codeword. For this purpose we introduce a counterpart of minimum distance.

Definition 1.1.5. If $C \subset \mathcal{Q}^n$ is a code, then the *covering radius* $\rho(C)$ of C is defined by

$$\rho(C) := \max \{\min \{d(x, c) | c \in C\} | x \in \mathcal{Q}^n\} .$$

Remark 1.1.6. We denote the *sphere* with radius ρ and center x by $B_\rho(x)$, i.e. the set $\{y \in \mathcal{Q}^n | d(x, y) \leq \rho\}$.
We can see that if $\bar{\rho} := \bar{\rho}(C)$ is the largest integer such that the spheres $B_{\bar{\rho}}(\mathbf{c})$

with $\mathbf{c} \in \mathbf{C}$ are disjoint, then $d = 2\bar{\rho} + 1$ or $d = 2\bar{\rho} + 2$.
The covering radius is the smallest number $\rho := \rho(C)$ such that the spheres $B_\rho(\mathbf{c})$ with $\mathbf{c} \in \mathbf{C}$ cover the set $\mathcal{Q}^n$.
If $\bar{\rho} = \rho$, then the code C is called *perfect*.

1.2 Detection and correction of the errors

We now turn our attention to the problem of error-detecting and error-correcting codes.

Theorem 1.2.1. *A q-ary code C can detect up to k errors in any codeword if and only if $d \geq k + 1$, where d denotes the minimum distance of the code C.*

Proof. Suppose that C is a q-ary code with minimum distance $d \geq k + 1$. Suppose that a codeword x is transmitted and is received with k or fewer errors. Since the minimum distance between codewords is at least $k + 1$, the vector received cannot be another codeword. Hence, the receiver can detect these errors.

Now suppose that C can detect up to k errors and that $d \leq k$. Then there are two codewords in C that differ in no more than k positions. It is then possible for k errors to be introduced when one of these codewords is transmitted so that the other codeword is received, contradicting the fact that C can detect up to k errors. □

When errors are detected, all we can do to obtain the correct codeword is to ask for retransmission and hope that no errors will occur when this is done. However, there are codes that can not only detect but can also correct errors. We now turn our attention to these codes, called error correcting codes.

Theorem 1.2.2. *A q-ary code C can correct up to k errors in any codeword if and only if $d \geq 2k + 1$, where d denotes the minimum distance of the code C.*

Proof. Suppose that C is a q-ary code with $d \geq 2k + 1$. Suppose that a codeword x is transmitted and received with k or fewer errors as the vector r, so that $d(x, r) \leq k$. To see that C can correct these errors, note that if y is a codeword other than x, then $d(r, y) \geq k + 1$. In fact, if $d(r, y) \leq k$, then by the triangle inequality $d(x, y) \leq d(x, r) + d(r, y) \leq k + k = 2k$, contradicting the assumption that $d \geq 2k + 1$.

Conversely, suppose that C can correct up to k errors. If $d \leq 2k$, then there are two codewords that differ in $2k$ positions. Changing k of the bits in one of these codewords produces a bit string that differs from each of these two codewords in exactly k positions, thus making it impossible to correct these k errors. □

1.3 Linear codes

We now turn to the problem of constructing codes which have some algebraic structure. The first idea is to take a group $\mathcal{Q}$ as alphabet and to take a subgroup C of $\mathcal{Q}^n$ as code. But it is not enough; we require more structure. In the following $\mathcal{Q}$ is the finite field $\mathbb{F}_q$, with $q = p^r$ and p is a prime. Then $\mathcal{Q}^n$ is an n-dimensional vector space, namely $\mathbb{F}_q^n$.

Definition 1.3.1. A *q-ary linear code* C is a linear subspace of $\mathbb{F}_q^n$. If C has dimension k then C is called $[n, k]$ code.

From now on we shall use $[n, k, d]$ code as the notation for a k-dimensional linear code of length n and minimum distance d.

Definition 1.3.2. A *generator matrix* G for a linear code C is a $k \times n$ matrix for which the rows are a basis of C.

If G is a generator matrix for C, then $C = \{aG \mid a \in \mathcal{Q}^k\}$.

If we want to know how many errors a code C corrects, we have to calculate the minimum distance. If C has minimum distance $d = 2e + 1$, then it corrects up to e errors in a received word.

In general if C has M words one must check $\binom{M}{2}$ pairs of codewords to find d. For linear codes the work is easier.

Theorem 1.3.3. *For a linear code C the minimum distance is equal to the minimum weight.*

Proof. $d(x, y) = d(x - y, 0) = w(x - y)$ and if $x \in C$, $y \in C$ then $x - y \in C$. □

1.4 Cyclic codes

Now we study a special class of linear codes, the so-called cyclic codes.

Definition 1.4.1. A linear code C is called *cyclic* if

$$\forall\ (c_0, c_1, \ldots, c_{n-1}) \in C \quad \Rightarrow \quad (c_{n-1}, c_0, c_1, \ldots, c_{n-2}) \in C.$$

Here we present the theory for cyclic codes.

The most important tool in the description of the cyclic codes is the following group isomorphism between $\mathbb{F}_q^n$ and a group of polynomials.
The multiples of $x^n - 1$ form a principal ideal in the polynomial ring $\mathbb{F}_q[x]$. The residue class ring $\mathbb{F}_q[x]/(x^n - 1)$ has the set of polynomials

$$\left\{a_0 + a_1 x + ... + a_{n-1}x^{n-1} | a_i \in \mathbb{F}_q, 0 \leq i < n\right\}$$

as a system of representatives. If we consider $\mathbb{F}_q^n$ only as an additive group, clearly it is isomorphic to this ring. From now on we make the following identification

$$\begin{array}{c} \mathbb{F}_q^n \overset{\sim}{\longleftrightarrow} \mathbb{F}_q[x]\big/(x^n - 1) \\ (a_0, a_1, \ldots, a_{n-1}) \longleftrightarrow a_0 + a_1 x + \cdots + a_{n-1}x^{n-1} \end{array}$$

In the following we shall also use the multiplicative structure which we have introduced, namely the multiplication of polynomials modulo $(x^n - 1)$.

Theorem 1.4.2. *A linear code C in $\mathbb{F}_q$ is cyclic if and only if C is an ideal in $\mathbb{F}_q[x]/(x^n - 1)$.*

Proof. If C is an ideal in $\mathbb{F}_q[x]/(x^n - 1)$ and $c(x) = c_0 + c_1 x + ... + c_{n-1}x^{n-1}$ is any codeword, then $xc(x)$ is also a codeword, i.e. $(c_{n-1}, c_0, c_1, \ldots, c_{n-2}) \in C$.

Conversely, if C is cyclic, then for every codeword $c(x)$ the word $xc(x)$ is also in C. Therefore $x^i c(x)$ is in C for every i, and since C is linear $a(x)c(x)$ is in C for every polynomial $a(x)$. Hence C is an ideal. □

From now we only consider cyclic codes of length n over $\mathbb{F}_q$ with $(n, q) = 1$. Since $\mathbb{F}_q[x]/(x^n - 1)$ is a principal ideal ring, every cyclic code C consists of the multiple of a polynomial $g(x)$ which is the monic polynomial of lowest degree in the ideal (not the zero polynomial).

Definition 1.4.3. This polynomial $g(x)$ is called the *generator polynomial* of the cyclic code.

Remark 1.4.4. The generator polynomial is a divisor of $x^n - 1$, since otherwise the g.c.d. of $x^n - 1$ and $g(x)$ would be a polynomial in C of degree lower than the degree of $g(x)$.

Let $x^n - 1 = f_1(x)f_2(x)\cdots f_t(x)$ be the decomposition of $x^n - 1$ into irreducible factors. Because of $(n, q) = 1$ these factors are different. We can now find all cyclic codes of length n by picking up one of the 2^t factors of $x^n - 1$ as generator polynomial $g(x)$ and defining the corrispondent code to be the ideal $(g(x))$ in $\mathbb{F}_q[x]/(x^n - 1)$.

Let $g(x)$ be the generator polynomial of a cyclic code C of length n. If $g(x)$ has degree $n-k$, then C is an $[n,k]$ code and the codewords $g(x)$, $xg(x)$, $\dots$, $x^{k-1}g(x)$ form a basis for C. Hence, if $g(x) = g_0 + g_1x + \cdots + g_{n-k}x^{n-k}$, then

$$G = \begin{bmatrix} g_0 & g_1 & \cdots & g_{n-k} & 0 & 0 & \cdots & 0 \\ 0 & g_0 & \cdots & g_{n-k-1} & g_{n-k} & 0 & \cdots & 0 \\ 0 & 0 & \cdots & & & & \cdots & 0 \\ 0 & 0 & \cdots & & & g_0 & g_1 & \cdots & g_{n-k} \end{bmatrix}$$

is a generator matrix for C. This means that we encode an information sequence $(a_0, a_1, \dots, a_{k-1})$ as aG which is the polynomial

$$(a_0 + a_1x + \cdots + a_{k-1}x^{k-1})g(x).$$

A more convenient form of the generator matrix is obtained by writing, for $i \geq n-k$, $x^i = g(x)q_i(x) + r_i(x)$, where $r_i(x)$ is a polynomial of degree $< n-k$. The polynomials $x^i - r_i(x)$ are codewords of C and form a basis for the code, which yields a generator matrix of C in standard form, with $G = (P \quad I_k)$. In this case $(a_0, a_1, \dots, a_{k-1})$ is encoded as follows: divide $(a_0 + a_1x + \cdots + a_{k-1}x^{k-1})x^{n-k}$ by $g(x)$ and subtract the remainder from $(a_0 + a_1x + \cdots + a_{k-1}x^{k-1})x^{n-k}$, thus obtaining a codeword. Tecnically this is a very easy way to encode information.

Remark 1.4.5. In general a cyclic code can be specified by requiring that all codewords have certain prescribed zeros.
In fact, it is sufficent to take one zero β_i of each irreducible factor f_i of the generator polynomial $g(x)$ and require that all codewords have, in a suitable extension field of $\mathbb{F}_q$, these points as zeros.
If we start with any set $\alpha_1, \dots, \alpha_s$ of zeros of $x^n - 1$ and define a code C by

$$c(x) \in C \iff c(\alpha_i) = 0 \ \ \forall i = 1, \dots, s$$

then C is cyclic and the generator polynomial of C is the least common multiple of the minimal polynomials of $\alpha_1, \alpha_2, \dots, \alpha_s$.

We know that exists an integer m such that all these zeros lie in $\mathbb{F}_{q^m}$, an extension field that we can represent as a vector space $\mathbb{F}_q^m$. For every i we can consider the $m \times n$ matrix with the vector representations of $1, \alpha_i, \alpha_i^2, \dots, \alpha_i^{n-1}$ as columns and put all these together to form the $sm \times n$ matrix H with entries in $\mathbb{F}_q$ as follows

$$H = \begin{bmatrix} 1 & \alpha_1 & \alpha_1^2 & \cdots & \alpha_1^{n-1} \\ 1 & \alpha_2 & \alpha_2^2 & \cdots & \alpha_2^{n-1} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & \alpha_s & \alpha_s^2 & \cdots & \alpha_s^{n-1} \end{bmatrix}.$$

Clearly, if $c = (c_0, c_1, \ldots, c_{n-1})$, we have $cH^T = 0$ iff $c(\alpha_i) = 0$ for $i = 1, 2, \ldots, s$. Observe that the rows of H are not necessarily indipendent.

1.5 BCH codes

An important class of cyclic codes, still used a lot in practice, was discovered by R. C. Bose and D. K. Ray-Chaudhuri (1960) and independently by A. Hocquenghem (1959). The codes are known as BCH codes.

Definition 1.5.1. A cyclic code of length n over $\mathbb{F}_q$ is called a *BCH code of designed distance* δ if its generator $g(x)$ is the least common multiple of the minimal polynomials of $\beta^l, \beta^{l+1}, \ldots, \beta^{l+\delta-2}$ for some l, where β is a primitive nth root of unity. Usually we shall take $l = 1$. If $n = q^m - 1$, i.e. β is a primitive element of $\mathbb{F}_{q^m}$, then the BCH code is called *primitive*.

The terminology "designed distance"is explained by the following

Theorem 1.5.2. *The minimum distance of a BCH code with designed distance d is at least d.*

Proof. In the same way as in the previous section we form the $m(d-1) \times n$ matrix H:

$$H := \begin{bmatrix} 1 & \beta^l & \beta^{2l} & \ldots & \beta^{(n-1)l} \\ 1 & \beta^{l+1} & \beta^{2(l+1)} & \ldots & \beta^{(n-1)(l+1)} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & \beta^{l+d-2} & \beta^{2(l+d-2)} & \ldots & \beta^{(n-1)(l+d-2)} \end{bmatrix}$$

where eache entry is interpreted as a column vector of length m over $\mathbb{F}_q$. A word $\mathbf{c}$ is in the BCH code iff $\mathbf{cH^T} = \mathbf{0}$. The $m(d-1)$ rows of H are not necessarily independent. Consider any $d-1$ columns of H and let $\beta^{i_1 l}, \ldots, \beta^{i_{d-1} l}$ be the top elements in these columns. The determinant of the submatrix of H obtained in this way is a Vandermonde determinant and it is equal to

$$\beta^{(i_1+\cdots+i_{d-1})l} \prod_{r>s} (\beta^{i_r} - \beta^{i_s}) \neq 0.$$

This determinant is clearly not zero, because β is a primitive nth root of unity. Therefore any $d-1$ columns of H are linearly independent and hence a codeword $\mathbf{c} \neq \mathbf{0}$ has weight $\geq d$. □

Remark 1.5.3. This theorem is usually called *BCH bound*. From now we only consider BCH codes with $l = 1$.

1.6 Reed-Solomon codes

One of the simplest examples of BCH codes, namely the case $n = q-1$, turns out to have many important applications.

Definition 1.6.1. A *Reed-Solomon code* (RS code) is a primitive BCH code of length $n = q - 1$ over $\mathbb{F}_q$ The generator of such a code has the form $g(x) = \prod_{i=1}^{d-1}(x - \alpha^i)$ where α is a primitive element in $\mathbb{F}_q$.

By the BCH bound we know that the minimum distance of an RS code with this generator $g(x)$ is at least d. Moreover in this special case we have the following theorem proved in [13].

Theorem 1.6.2. *A Reed-Solomon code over $\mathbb{F}_q$ with designed distance d has minimum distance exactly d.*

□

A Reed-Solomon code can be represented in another way. This representation is very useful because it gives a very efficient encoding procedure for the code. The proof of the following theorem can be found in [13].

Theorem 1.6.3. *Let C be a Reed-Solomon code of length $n = q - 1$ over $\mathbb{F}_q$ and designed distance d, and let $k := n - d$. Then*

$$C = \left\{(c_0, \ldots, c_{n-1}) \mid c_i = p(\alpha^i), 0 \leq i < n, p \in \mathbb{F}_q[x]_{\leq k}\right\},$$

where α is a primitive element in $\mathbb{F}_q$.

□

1.7 Decoding BCH codes

Once again consider a primitive BCH code of length $n = q^m - 1$ over $\mathbb{F}_q$ with designed distance $\delta = 2t + 1$ and let β be a primitive nth root of unity in $\mathbb{F}_{q^m}$. We consider a codeword $c(x)$ and assume that the received word is

$$r(x) = r_0 + r_1 x + \cdots + r_{n-1}x^{n-1}.$$

Let $\varepsilon(x) := r(x) - c(x) = \varepsilon_0 + \varepsilon_1 x + \cdots + \varepsilon_{n-1}x^{n-1}$ be the error vector. We denote with

$$M := \{i | \varepsilon_i \neq 0\},$$

the set of the positions where an error occur, and with

$$e := |M|$$

the number of errors that occur in $r(x)$.

Definition 1.7.1. The polynomial $\sigma(z) \in \mathbb{F}_{q^m}[z]$ defined by

$$\sigma(z) := \prod_{i \in M} (1 - \beta^i z)$$

is called *error-locator polynomial* for the received word r.

Definition 1.7.2. The polynomial $\omega(z) \in \mathbb{F}_{q^m}[z]$ defined by

$$\omega(z) := \sum_{i \in M} \varepsilon_i \beta^i \prod_{j \in M \setminus \{i\}} (1 - \beta^j z)$$

is called *error-evaluator polynomial* for the received word r. Observe that $\deg \omega < \deg \sigma = e$.

Definition 1.7.3. Let $r(x) = c(x) + \varepsilon(x)$ be a received word. We define the *syndrome* s_i for $i = 1, \ldots, 2t$ by

$$s_i = r(\beta^i).$$

Remark 1.7.4. Since $r(x) = c(x) + \varepsilon(x)$ and $c(x)$ is a codeword, $c(\beta^i) = 0$ for $i = 1, \ldots, 2t$, hence

$$s_i = r(\beta^i) = \varepsilon(\beta^i) = \sum_{i=0}^{n-1} \varepsilon_j (\beta^i)^j.$$

Suppose that, for a received word $r(x)$, the e errors occur in location corresponding to indexes $i_1, \ldots, i_e$. for ease of notation we reformulate the syndromes as

$$s_i = \sum_{j=1}^{e} E_j a_j^i, \tag{1.1}$$

for $i = 1, \ldots, 2t$, where we have put $E_j := \varepsilon_{i_j}$ and $a_j := \beta^{i_j}$.

With this notation we can also reformulate the error-locator and the error-evaluator polynomial as

$$\sigma(z) := \prod_{i=1}^{e} (1 - a_i z) \tag{1.2}$$

$$\omega(z) := \sum_{i=1}^{e} E_i a_i \prod_{j \neq i} (1 - a_j z) \tag{1.3}$$

Definition 1.7.5. Let $r(x)$ be a received word with syndromes s_i for $i = 1, \ldots, 2t$. Then the polynomial

$$S(z) := \sum_{i=1}^{2t} s_i z^{i-1} = \sum_{i=1}^{2t} z^{i-1} \sum_{j=1}^{e} E_j a_j^i$$

is called *syndrome polynomial* of the received word r.
The *syndrome series* of r is the formal power series defined by

$$\widehat{S}(z) := \sum_{i=1}^{+\infty} z^{i-1} \sum_{j=1}^{e} E_j a_j^i$$

Proposition 1.7.6. *Let $r(x)$ be a received word and let $\sigma(z)$, $\omega(z)$ be respectively the error-locator and the error-evaluator polynomial for the received word r. Then an error occurs in position i if and only if $\sigma(\beta^{-i}) = 0$, and in that case the error is*

$$\varepsilon_i = -\frac{\omega(\beta^{-i})}{\sigma'(\beta^{-i})}.$$

Proof. It is an easy calculation. □

Thus if we can find $\sigma(z)$ and $\omega(z)$, then the errors can be corrected.

From now on we assume that $e \leq t$, because if $e > t$ we do not expect to be able to correct the errors.

Theorem 1.7.7. *Let $r(x)$ be a received word and let $\sigma(z)$, $\omega(z)$ be respectively the error-locator and the error-evaluator polynomial for the received word r. Then*

$$\frac{\omega(z)}{\sigma(z)} = \widehat{S}(z),$$

where $\widehat{S}(z)$ is the syndrome series of r.

Proof. It is sufficient to observe that

$$\begin{aligned} \frac{\omega(z)}{\sigma(z)} &= \sum_{i=1}^{e} \frac{E_i a_i}{1 - a_i z} = \sum_{i=1}^{e} E_i a_i \sum_{l=0}^{\infty} (a_i z)^l \\ &= \sum_{l=0}^{\infty} z^l \sum_{i=1}^{e} E_i a_i^{l+1} = \widehat{S}(z), \end{aligned}$$

where all calculations are with formal power series over $\mathbb{F}_{q^m}$. □

Corollary 1.7.8. *Let $r(x)$ be a received word and let $\sigma(z)$, $\omega(z)$ be respectively the error-locator and the error-evaluator polynomial for the received word r. Then*

$$\omega(z) \equiv S(z)\sigma(z) \qquad \mod (z^{2t}), \tag{1.4}$$

where $S(z)$ is the syndrome polynomial of r.

Theorem 1.7.9. *The polynomials $\sigma(z)$ and $\omega(z)$ coincide with the polynomials $\bar{\sigma}(z)$ and $\bar{\omega}(z)$ such that $\deg \bar{\omega}(z) < \deg \bar{\sigma}(z)$ and $\deg \bar{\sigma}(z)$ is as small as possible under the condition*

$$\frac{\bar{\omega}(z)}{\bar{\sigma}(z)} \equiv S(z) \qquad \mod (z^{2t})$$

Proof. Let $\sigma(z) = \sum_{i=0}^{e} \sigma_i z^i$. Then

$$\omega(z) \equiv \left(\sum_{l=0}^{2t-1} s_{l+1} z^l\right)\left(\sum_{i=0}^{e} \sigma_i z^i\right) \equiv \sum_{k=0}^{2t-1}\left(\sum_{i+l=k} s_{l+1}\sigma_i\right) z^k \qquad \mod (z^{2t}).$$

Beacuse $\deg \omega(z) < e$ we have

$$\sum_{i+l=k} s_{l+1}\sigma_i = 0, \qquad \text{for } e \le k \le 2t-1.$$

This is a system of $2t - e$ linear equations for the unknows $\sigma_1, \ldots, \sigma_e$ (because we know that $\sigma_0 = 1$). Let $\bar{\sigma}(z) = \sum_{i=0}^{e} \bar{\sigma}_i z^i$ (where $\bar{\sigma}_0 = 1$) be the polynomial of lowest degree found by solving these equations (we know there is at least the solution $\sigma(z)$). For $e \le k \le 2t-1$ we have

$$0 = \sum_l s_{k-l+1}\bar{\sigma}_l = \sum_{i\in M}\sum_l \varepsilon_i \beta^{(k-l+1)i}\bar{\sigma}_l = \sum_{i\in M} \varepsilon_i \beta^{i(k+1)}\bar{\sigma}(\beta^{-i}).$$

We can interpret the right-hand side as a system of linear equations for $\varepsilon_i\bar{\sigma}(\beta^{-i})$ with coefficients $\beta^{i(k+1)}$. So the determinant of coefficients is a Vandermonde determinant, hence it is $\neq 0$. So $\varepsilon_i\bar{\sigma}(\beta^{-i}) = 0$ for $i \in M$. Since $\varepsilon_i \neq 0$ for $i \in M$ we have $\bar{\sigma}(\beta^{-i}) = 0$ for $i \in M$ and then $\sigma(z)$ divides $\bar{\sigma}(z)$, i.e. $\bar{\sigma}(z) = \sigma(z)$. So indeed, the solution $\bar{\sigma}(z)$ of lowest degree solves our problem and we have seen that finding it amounts to solving a system of linear equations. □

The advantage of this approach is that the decoder has an algorithm that does not depend on e. Of course, in practice it is even more important to find a fast algorithm that actually does what we have only considered

form a theoretical point of view. Such an algorithm was designed by E. R. Berlekamp and is often referred to as the *Berlekamp-decoder*.

Now, given a received word $r(x)$, suppose we know that in r occur exactly e errors, with $e \leq t$. We have seen in the previous proof that the coefficients of the error-locator polynomial satisfy some linear conditions. We can state this fact with the following result.

Theorem 1.7.10. *The vector $(\sigma_e, \ldots, \sigma_0)^T$ given by the coefficients of the error locator polynomial $\sigma(z)$ is the unique, non-trivial solution of the linear system*

$$\begin{bmatrix} s_1 & s_2 & \cdots & s_{e+1} \\ s_2 & s_3 & \cdots & s_{e+2} \\ \vdots & \vdots & & \vdots \\ s_{2t-e} & s_{2t-e+1} & \cdots & s_{2t} \end{bmatrix} \begin{bmatrix} x_e \\ x_{e-1} \\ \vdots \\ x_1 \\ x_0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix},$$

with $x_0 = 1$.

Chapter 2

The linear division problem

In the prevoius chapter we have seen that, when the number of errors e that occur in a received word is less or equal to the error-correcting bound t, the error-locator polynomial is given by the unique non-trivial solution of the linear system

$$\begin{bmatrix} s_1 & s_2 & \cdots & s_{e+1} \\ s_2 & s_3 & \cdots & s_{e+2} \\ \vdots & \vdots & & \vdots \\ s_{2t-e} & s_{2t-e+1} & \cdots & s_{2t} \end{bmatrix} \begin{bmatrix} x_e \\ x_{e-1} \\ \vdots \\ x_1 \\ x_0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \tag{2.1}$$

with $x_0 = 1$.

We would like to investigate in what happens when we try to correct more than t errors. If we suppose that the received word is affected by e errors with $e > t$ we still have that the error-locator polynomial is a solution of the linear system 2.1. In this case we lose the unicity of the solution. In fact the matrix S_e is a $(2t - e) \times (e + 1)$ with $e > t$, so if we put $s := e - t$, we have that $S_e \in M(t + s, t - s + 1, \mathbb{F}_{q^m})$, and $\dim \ker S_e \geq 2s + 1$.

Therefore, given a basis for $\ker S_e$, the error-locator polynomial will be a linear combination of the elements of such a basis. Since one of the properties of the error-locator polynomial is that it divides $x^{q^m-1} - 1$, from this research a new problem naturally arises. It is the more general *linear division problem*. In the following we accurately formulate this problem and we try to develop a theory for trying to solve it.

So, in this chapter we deal with the problem of divisibility of a polynomial $H(x)$ with coefficients in a finite field $\mathbb{F}_q$. Let k, e be positive integers with $k \leq e$ and suppose we have $k + 1$ polynomials $f_0(x), f_1(x), \ldots, f_k(x) \in \mathbb{F}_q[x]$, and a polynomial $H(x) \in \mathbb{F}_q[x]$. We would like to find all the polynomials

$F(x)$ of the form

$$F(x) = \sum_{i=0}^{k} \alpha_i f_i(x)$$

such that $\alpha_i \in \mathbb{F}_q$ for $i = 0, \ldots, k$, $\deg F = e$ and $F(x)$ divides $H(x)$. Clearly we can suppose $\deg f_i \leq e$.

Let V be the set of all polynomials in $\mathbb{F}_q[x]$ with degree at most e, i.e.

$$V = \{p(x) \in \mathbb{F}_q[x] \mid \deg p \leq e\} = \mathbb{F}_q[x]_{\leq e}.$$

Clearly V is a vector space over $\mathbb{F}_q$ of dimension $e+1$. Let $W = \langle f_0, \ldots, f_k \rangle$ be the linear subspace of V generated by the f_is. Observe that one of the f_i is necessarily of degree e, otherwise such polynomial F does not exist. Moreover we can choose the $f_i(x)$ such that

$$e = \deg f_0 > \deg f_1 > \cdots > \deg f_k$$

and $lc(f_i) = 1$ for all $i = 0, 1, \ldots, k$, where $lc(f_i)$ denotes the leading coefficient of f_i. So we can look for all the polynomials of the form

$$F(x) := f_0(x) + \sum_{i=1}^{k} \alpha_i f_i(x) \qquad \alpha_i \in \mathbb{F}_q.$$

2.1 Formalization of the problem

Now we want develop a theory that permit us to deal with such a problem. Let us introduce some instruments that will be useful to formalize and solve the linear division problem.

Definition 2.1.1. Given a finite dimensional subspace W of $\mathbb{F}_q[x]$, the *degree* of W is defined as

$$\deg W := \max_{p(x) \in W} \deg p(x)$$

Clearly, since W is finite dimensional, $\deg W < +\infty$.

Remark 2.1.2. Such definition is extended also to the subspaces W ov V.

Definition 2.1.3. A basis $\mathcal{F} = \{f_0, \ldots, f_k\}$ of a subspace W of degree e of $\mathbb{F}_q[x]$ is called *leading basis* iff $e = \deg f_0$ and $\deg f_i < e$ for all $i = 1, \ldots, k$.

From now on we work with the polynomials $f_0(x), f_1(x), \ldots f_k(x)$, that are linearly independent over $\mathbb{F}_q$, and where $\mathcal{F} := \{f_0(x), f_1(x), \ldots f_k(x)\}$ is a leading basis for the subspace W generated by the f_is.

Definition 2.1.4. A basis $\mathcal{F} = \{f_0, \ldots, f_k\}$ of a subspace W of degree e of $\mathbb{F}_q[x]$ is called *escalier basis* iff

$$e = \deg f_0 > \deg f_1 > \cdots > \deg f_k$$

and $lc(f_i) = 1$ for all $i = 0, 1, \ldots, k$.

The following is a very easy result of linear algebra.

Theorem 2.1.5. *Every finite dimensional subspace W of $\mathbb{F}_q[x]$ admit an escalier basis.*

□

From now on suppose that $\gcd(H(x), H'(x)) = 1$, i.e. that all the roots of $H(x)$ are distinct.

Consider the multivariate polynomial

$$F(a_1, \ldots, a_k, x) := f_0(x) + \sum_{i=1}^{k} a_i f_i(x) \in \mathbb{F}_q[a_1, \ldots, a_k, x] =: \mathbb{F}_q[\underline{a}, x]$$

where $\underline{a} = (a_1, \ldots, a_k)$. Consider $F(\underline{a}, x)$ embedded in the ring $\mathbb{F}_q(\underline{a})[x]$, where $\mathbb{F}_q(\underline{a})$ denotes the fraction field of $\mathbb{F}_q[\underline{a}]$. This ring is an euclidean domain, so there exist unique polynomials $q(\underline{a}, x)$, $r(\underline{a}, x) \in \mathbb{F}_q(\underline{a})[x]$ such that

$$H(x) = F(\underline{a}, x)q(\underline{a}, x) + r(\underline{a}, x)$$

with $\deg_x r < \deg_x F = e$.

Proposition 2.1.6. *Let $H(x) \in \mathbb{F}_q[x]$, $F(\underline{a}, x) \in \mathbb{F}_q[\underline{a}, x]$ be two polynomials with $\deg_x F = e < n = \deg_x H$, and $lc_x(F) \in \mathbb{F}_q^*$. Then, the quotient $q(\underline{a}, x)$ and the remainder $r(\underline{a}, x)$ of the division between H and F lie in $\mathbb{F}_q[\underline{a}, x]$*

Proof. The proof is very easy. In fact in the division algorithm at each step the only division that is needed is the division by the leading coefficient of $F(\underline{a}, x)$ in seen as a polynomial in x. Since by hypothesis $lc_x(F) \in \mathbb{F}_q^*$, we trivially obtain that $q(\underline{a}, x)$ and $r(\underline{a}, x)$ belong to $\mathbb{F}_q[\underline{a}, x]$. □

Lemma 2.1.7. *Let $H(x) \in \mathbb{F}_q[x]$, $F(\underline{a}, x) \in \mathbb{F}_q[\underline{a}, x]$ be two polynomials with $\deg_x F = e < n = \deg_x H$, and $lc_x(F) \in \mathbb{F}_q^*$. Suppose that*

$$F(\underline{a}, x) = f_0(x) + \sum_{i=1}^{k} a_i f_i(x)$$

Then $\deg_{\underline{a}} q(\underline{a}, x) \leq n - e$ and $\deg_{\underline{a}} r(\underline{a}, x) \leq n - e + 1$, where $q(\underline{a}, x)$ and $r(\underline{a}, x)$ are the quotient and the remainder of the division between $H(x)$ and $F(\underline{a}, x)$, and $\deg_{\underline{a}}$ is the total degree with respect to the variables $(a_1, \ldots, a_k)$.

Proof. Perform the division algorithm step by step. At each step the degrees in $\underline{a}$ of provisional remainder and provisional quotient increase at most by 1. At the first step $\deg_{\underline{a}} q = 0$ and $\deg_{\underline{a}} r \leq 1$. Since the algorithm terminates after $n - e + 1$ steps we easily conclude. □

Let $\underline{\alpha} \in \overline{\mathbb{F}}_q^k$. The polynomial $F(\underline{\alpha}, x)$ divides $H(x)$ if and only if $r(\underline{\alpha}, x) = 0$. So, if we write

$$r(\underline{a}, x) = \sum_{i=0}^{e-1} r_i(\underline{a}) x^i$$

and we define the ideal $\mathcal{I}_{\mathcal{F},H}$ of $\mathbb{F}_q[\underline{a}]$

$$\mathcal{I}_{\mathcal{F},H} := (r_0(\underline{a}), \ldots, r_{e-1}(\underline{a}))\,,$$

where $\mathcal{F} = \{f_0(x), \ldots, f_k(x)\}$, our goal is equivalent to finding the variety

$$\mathcal{V}(\mathcal{I}_{\mathcal{F},H}) = \left\{ \underline{\alpha} \in \overline{\mathbb{F}}_q^k \mid p(\underline{\alpha}) = 0 \ \ \forall\, p \in \mathcal{I}_{\mathcal{F},H} \right\}.$$

In fact

$$\underline{\alpha} \in \mathcal{V}(\mathcal{I}_{\mathcal{F},H}) \Longleftrightarrow F(\underline{\alpha}, x) \text{ divides } H(x).$$

Definition 2.1.8. The ideal $\mathcal{I}_{\mathcal{F},H}$ defined above is called *divisibility ideal of H by $\mathcal{F}$.*

Definition 2.1.9. Let $\mathcal{F} = \{f_0, \ldots, f_k\}$ be a leading basis of a subspace W of degree e of V. Then the set

$$\Sigma_{\mathcal{F},H} := \left\{ F(\underline{\alpha}, x) = f_0(x) + \sum_{i=1}^{k} \alpha_i f_i(x) \;\middle|\; \underline{\alpha} \in \mathcal{V}(\mathcal{I}_{\mathcal{F},H}) \right\}$$

is called *divisibility set of H by $\mathcal{F}$.*

Proposition 2.1.10. *Let $\mathcal{F} = \{f_0, \ldots, f_k\}$ be a leading basis of a subspace W of degree e of V. Then*

$$\Sigma_{\mathcal{F},H} = \left\{ p(x) \in W \;\middle|\; \deg p = e, lc(p) = 1,\ p(x) | H(x) \right\}.$$

Proof. It easily follows by the definition of the set $\Sigma_{\mathcal{F},H}$. □

Corollary 2.1.11.

$$|\mathcal{V}(\mathcal{I}_{\mathcal{F},H})| = |\Sigma_{\mathcal{F},H}| = \left| \left\{ p(x) \in W \;\middle|\; \deg p = e, lc(p) = 1,\ p(x) | H(x) \right\} \right|$$

Proof. It is an immediate consequence of the previous proposition. □

What happens if we choose another leading basis $\mathcal{G}$ of the subspace W? Clearly $\mathcal{V}(\mathcal{I}_{\mathcal{G},H}) \neq \mathcal{V}(\mathcal{I}_{\mathcal{F},H})$. But how are they related? The answers to those questions are given by the following result.

Proposition 2.1.12. *Let $\mathcal{G} = \{g_0, \dots, g_k\}$ and $\mathcal{F} = \{f_0, \dots, f_k\}$ be two leading basis of a subspace W of degree e with $g_0(x) = \lambda f_0(x) + \sum_{i=1}^{k} c_i^0 f_i(x)$ and $g_j(x) = \sum_{i=1}^{k} c_i^j f_i(x)$ for $j = 1, \dots k$. Let C be the change-of-basis matrix from $\mathcal{G} \setminus \{g_0\}$ to $\mathcal{F} \setminus \{f_0\}$, and let $b \in \mathbb{F}_q^k$ be the vector defined by $b_i = c_i^0$. Then $\underline{\alpha} \in \mathcal{V}(\mathcal{I}_{\mathcal{G},H})$ if and only if $\frac{1}{\lambda}\left(C^T\underline{\alpha} + b\right) \in \mathcal{V}(\mathcal{I}_{\mathcal{F},H})$.*

Proof. It is an easy calculation. We can rewrite the polynomial $G(\underline{a}, x)$ in terms of the $f_i(x)$.

$$\begin{aligned} G(\underline{a}, x) = g_0(x) + \sum_{i=1}^{k} a_i g_i(x) = f_0(x) + \sum_{i=1}^{k} c_i^0 f_i(x) + \sum_{j=1}^{k} a_j \sum_{i=1}^{k} c_i^j f_i(x) = \\ = \lambda f_0(x) + \sum_{i=1}^{k} c_i^0 f_i(x) + \sum_{i=1}^{k} \left(\sum_{j=1}^{k} a_j c_i^j \right) f_i(x) = \\ = \lambda f_0(x) + \sum_{i=1}^{k} \left[\left(\sum_{j=1}^{k} a_j c_i^j \right) + c_i^0 \right] f_i(x) = \lambda F(\underline{y}, x). \end{aligned}$$

where $y_i = \frac{1}{\lambda}\left[\left(\sum_{j=1}^{k} a_j c_i^j\right) + c_i^0\right]$. Observing that the vector $\underline{y} = \frac{1}{\lambda}\left(C^T\underline{a} + b\right)$ we conclude the proof.

□

Corollary 2.1.13. *Let $\mathcal{F}$ and $\mathcal{G}$ are two leading basis of a subspace W of degree e of V. Then there exists a bijection between $\mathcal{V}(\mathcal{I}_{\mathcal{F},H})$ and $\mathcal{V}(\mathcal{I}_{\mathcal{G},H})$. In particular $|\mathcal{V}(\mathcal{I}_{\mathcal{F},H})| = |\mathcal{V}(\mathcal{I}_{\mathcal{G},H})|$, even if they are not finite set. However in the next section we show that actually $\mathcal{V}(\mathcal{I}_{\mathcal{F},H})$ is a finite set.*

Remark 2.1.14. If $\mathcal{F}$ and $\mathcal{G}$ are two leading basis of W then $\Sigma_{\mathcal{F},H} = \Sigma_{\mathcal{G},H}$.

2.2 The study of the divisibility ideal

In this section we are going to study some properties of the divisibility ideal. The first of these properties is the Krull dimension of $\mathcal{I}_{\mathcal{F},H}$.

Theorem 2.2.1. *Let W be a subspace of degree e of $V = \mathbb{F}_q[x]_{\leq e}$ with leading basis $\mathcal{G} = \{g_0, \dots, g_k\}$ and let $H(x)$ a polynomial of degree n greater than e. Then the Krull dimension of the divisibility ideal $\mathcal{I}_{\mathcal{G},H}$ is* 0.

Proof. For a polynomial ring, we have that

$$\dim \mathcal{I}_{\mathcal{G},H} = 0 \iff |\mathcal{V}(\mathcal{I}_{\mathcal{G},H})| < +\infty,$$

where

$$\mathcal{V}(\mathcal{I}_{\mathcal{G},H}) = \left\{ \underline{\alpha} \in \overline{\mathbb{F}}_q^k \mid p(\underline{\alpha}) = 0 \ \ \forall\, p(\underline{a}) \in \mathcal{I}_{\mathcal{G},H} \right\}.$$

Consider an escalier basis $\mathcal{F} = \{f_0, \ldots, f_k\}$ of W. Then

$$\mathcal{V}(\mathcal{I}_{\mathcal{F},H}) = \frac{1}{\lambda}\left(C^T \mathcal{V}(\mathcal{I}_{\mathcal{G},H}) + b\right)$$

where C is the change-of-basis matrix between $\mathcal{G} \setminus \{g_0\}$ and $\mathcal{F} \setminus \{f_0\}$, and b is the vector of the b_i defined by $f_0 = g_0 + \sum_i b_i g_i$. Then, since $|\mathcal{V}(\mathcal{I}_{\mathcal{G},H})| = |\mathcal{V}(\mathcal{I}_{\mathcal{F},H})|$, we can show that $|\mathcal{V}(\mathcal{I}_{\mathcal{F},H})| < +\infty$.

Let $\mathbb{F} = \mathbb{F}_{q^m}$ be the splitting field of $H(x)$ over $\mathbb{F}_q$. If we prove that $\mathcal{V}(\mathcal{I}_{\mathcal{F},H}) \subseteq \mathbb{F}^k$ then it follows that $|\mathcal{V}(\mathcal{I}_{\mathcal{F},H})| \leq q^m k < +\infty$ and we conclude the proof. Observe that, if $\underline{\alpha} \in \mathcal{V}(\mathcal{I}_{\mathcal{F},H})$ then $F(\underline{\alpha}, x)$ divides $H(x)$, that splits into linear factors in $\mathbb{F}[x]$. Hence $F(\underline{\alpha}, x) \in \mathbb{F}[x]$. We put $n_i := \deg f_i$. Then

$$n_0 > n_1 > \cdots > n_k$$

and

$$f_i(x) = x^{n_i} + \sum_{j=0}^{n_i - 1} f_i^j x^j.$$

Since

$$F(\underline{\alpha}, x) = f_0(x) + \sum_{i=0}^{k} \alpha_i f_i(x),$$

the coefficient of x_1^n is $f_0^{n_1} + \alpha_1$ and it is in $\mathbb{F}$, then also $\alpha_1 \in \mathbb{F}$. The coefficient of x_2^n is $f_0^{n_2} + \alpha_1 f_1^{n_2} + \alpha_2$ and it is in $\mathbb{F}$, then also $\alpha_2 \in \mathbb{F}$, and so on. Hence $\alpha_i \in \mathbb{F}$ for every $i = 1, \ldots, k$, and we conclude that $\underline{\alpha} \in \mathbb{F}^k$. □

Corollary 2.2.2. *Let W be a subspace of degree e of $V = \mathbb{F}_q[x]_{\leq e}$ with leading basis $\mathcal{G} = \{g_0, \ldots, g_k\}$ and let $H(x)$ a polynomial of degree n greater than e, whose splitting field is $\mathbb{F}_{q^m}$. Then $\mathcal{V}(\mathcal{I}_{\mathcal{G},H}) \subseteq \mathbb{F}_{q^m}^k$.*

Another property of the divisibility ideal is that it is a radical ideal. In order to prove this we study first the case $k = 1$ and then the general case. Before showing such a property we need the following lemma.

Lemma 2.2.3. *Let $F(a, x) = f_0(x) + a f_1(x)$ be a polynomial in $\mathbb{F}_q[a, x]$ with $f_0(x), f_1(x) \in \mathbb{F}_q[x]$ and $\deg f_0 > \deg f_1$. Let $H(x) \in \mathbb{F}_q[x]$ be a polynomial with all distinct roots. If the polynomial $F(a, x)$ divides $H(x)$ modulo (a) then it does not divide $H(x)$ modulo (a^2).*

Proof. By hypothesis we know that there exists a polynomial $q(a,x)$ such that

$$H(x) \equiv \sigma(a,x)q(a,x) \mod (a).$$

If we write

$$q(a,x) = q_0(x) + aq_1(x) + a^2q_2(a,x)$$

we obtain, modulo a,

$$\begin{aligned} H(x) &\equiv (f_0(x) + af_1(x)) \left(q_0(x) + aq_1(x) + a^2q_2(a,x)\right) \\ &\equiv f_0(x)q_0(x). \end{aligned}$$

So $H(x) = f_0(x)q_0(x)$. Since the polynomial H has all distinct roots, hence $\gcd(f_0, q_0) = 1$.
Now suppose that $F(a,x)$divides $H(x)$ modulo (a^2), i.e.

$$\begin{aligned} H(x) &\equiv (f_0(x) + af_1(x)) \left(q_0(x) + aq_1(x) + a^2q_2(a,x)\right) \\ &\equiv f_0(x)q_0(x) + a\left(f_1(x)q_0(x) + f_0(x)q_1(x)\right). \end{aligned}$$

Hence

$$a\left(f_1(x)q_0(x) + f_0(x)q_1(x)\right) \equiv 0 \mod (a^2)$$

that means

$$f_1(x)q_0(x) = -f_0(x)q_1(x).$$

Therefore $f_0(x)$ divides $f_1(x)q_0(x)$, and $\gcd(f_0, q_0) = 1$. So we obtain that $f_0(x)$ divides $f_1(x)$ and, at the same time, by hypothesis $\deg f_0 > \deg f_1$, contradicting our assumption. □

Now we are ready to prove the following theorem.

Theorem 2.2.4. *Suppose we have a leading basis $\mathcal{F} = \{f_0, f_1\}$ of a subspace W of dimension 2, and let $H(x) \in \mathbb{F}_q[x]$ be a polynomial with all distinct roots. Then the divisibility ideal $\mathcal{I}_{\mathcal{F},H} \subseteq \mathbb{F}_q[a]$ is a radical ideal.*

Proof. Observe that, in order to prove that the ideal $\mathcal{I}_{\mathcal{F},H}$ is radical, it is sufficent to show that the variety $\mathcal{V}(\mathcal{I}_{\mathcal{F},H})$ does not contain points of multiplicity greater than one. We know that, if $\alpha \in \mathcal{V}(\mathcal{I})$ then $\alpha \in \mathbb{F}_q$. Without loss of generality we can suppose $\alpha = 0$. In fact, if $\alpha \neq 0$ it is sufficient to change basis in $\mathcal{G} = \{f_0(x) + \alpha f_1(x), f_1(x)\}$, that is a leading basis too. We know that there exist two polynomials $q(a,x)$, $r(a,x) \in \mathbb{F}_q[a][x]$ such that

$$H(x) = F(a,x)q(a,x) + r(a,x).$$

Since $0 \in \mathcal{V}(\mathcal{I}_{\mathcal{F},H})$, we have

$$H(x) \equiv F(a,x)q(a,x) \mod (a).$$

Now suppose that the multiplicity of 0 in $\mathcal{V}(\mathcal{I}_{\mathcal{F},H})$ is greater than 1. Then a^2 divides $r(a,x)$, i.e. $r(a,x) = a^2\tilde{r}(a,x)$.
Hence, $H(x) \equiv F(a,x)h(a,x)$ modulo (a^2), in contradiction with the previous lemma. □

We are ready to generalize this result, in the case $k > 1$. We are working in the ring

$$A := \mathbb{F}_q[a_1, \ldots, a_k] = \mathbb{F}_q[\underline{a}].$$

Lemma 2.2.5. *Let $\mathfrak{q} \subset A$ be a primary ideal such that $\sqrt{\mathfrak{q}} = (a_1, \ldots, a_k) =: \mathfrak{m}$ and $\mathfrak{q} \neq \mathfrak{m}$. Let $H(x) \in \mathbb{F}_q[x]$ be a polynomial with all distinct roots and let $\mathcal{F}$ be a leading basis for a subspace W of degree e of V. Put*

$$F(\underline{a}, x) = f_0(x) + \sum_{i=1}^{k} a_i f_i(x).$$

If $F(\underline{a}, x)$ divides $H(x)$ modulo $\mathfrak{m}$, then $F(\underline{a}, x)$ does not divide $H(x)$ modulo $\mathfrak{q}$.

Proof. Without loss of generality we can suppose $\mathfrak{m}^2 \subseteq \mathfrak{q} \subsetneq \mathfrak{m}$. There exist two polynomials $q(\underline{a}, x)$, $r(\underline{a}, x) \in \mathbb{F}_q[\underline{a}][x]$ such that

$$H(x) = F(\underline{a}, x)q(\underline{a}, x) + r(\underline{a}, x)$$

with

$$q(\underline{a}, x) = q_0(x) + a_1 q_1(x) + \cdots + a_k q_k(x) + \bar{q}(\underline{a}, x), \quad \bar{q}(\underline{a}, x) \in \mathfrak{m}^2.$$

By hypothesis $H(x) \equiv f_0(x)q_0(x)$ modulo $\mathfrak{m}$. Hence

$$H(x) = f_0(x)q_0(x).$$

Now suppose that $H(x) \equiv F(\underline{a}, x)q(\underline{a}, x)$ modulo $\mathfrak{q}$. Then

$$H(x) \equiv f_0 q_0 + a_1(f_0 q_1 + f_1 q_0) + \cdots + a_k(f_0 q_k + f_k q_0) \mod \mathfrak{q}$$

that is true if and only if

$$a_1(f_0 q_1 + f_1 q_0) + \cdots + a_k(f_0 q_k + f_k q_0) \equiv 0 \mod \mathfrak{q}.$$

Since $\mathfrak{q} \neq \mathfrak{m}$, at least one of the a_i is not in $\mathfrak{q}$. For this i $f_0 q_i + f_i q_0 = 0$, i.e.

$$f_0 q_i = -f_i q_0, \quad \deg f_0 > \deg f_i$$

and we conclude in the same way as in the previous lemma. □

Finally we can generalize the Theorem 2.2.4, when $k > 1$.

Theorem 2.2.6. *Let $\mathcal{F} = \{f_0, f_1, \ldots, f_k\}$ be a leading basis for a subspace W of degree e and dimension $k+1$, and let $H(x) \in \mathbb{F}_q[x]$ be a polynomial with all distinct roots. Then the divisibility ideal $\mathcal{I}_{\mathcal{F},H} \subseteq \mathbb{F}_q[\underline{a}]$ is a radical ideal.*

Proof. Consider the splitting field $\mathbb{F}_{q^m}$ of $H(x)$ over $\mathbb{F}_q$. We can see $\mathcal{I}_{\mathcal{F},H}$ as an ideal of $\mathbb{F}_{q^m}[\underline{a}]$. Let

$$\mathcal{I}_{\mathcal{F},H} = \bigcap_{j=1}^{s} \mathfrak{q}_j$$

be the primary decomposition of the divisibility ideal $\mathcal{I}_{\mathcal{F},H}$ in $\mathbb{F}_{q^m}[\underline{a}]$. Then the radical of $\mathcal{I}_{\mathcal{F},H}$ satisfies

$$\sqrt{\mathcal{I}_{\mathcal{F},H}} = \bigcap_{j=1}^{s} \sqrt{\mathfrak{q}_j}.$$

The ideal $\mathcal{I}_{\mathcal{F},H}$ is $0-$dimensional, so $\sqrt{\mathfrak{q}_j} = \mathfrak{m}_j$, with $\mathfrak{m}_j$ maximal ideal for every $j = 1, \ldots, s$. Furthermore, since $\mathcal{V}(\mathcal{I}_{\mathcal{F},H}) \subseteq \mathbb{F}_{q^m}^k$, $|\mathcal{V}(\mathfrak{m}_j)| = 1$ for every j.
Suppose that there exist j such that $\mathfrak{q}_j \subsetneq \mathfrak{m}_j$ and let $\underline{\alpha} = (\alpha_1, \ldots, \alpha_k) \in \mathbb{F}_q^k$ be the unique point in $\mathcal{V}(\mathfrak{m}_j)$. As in the Theorem 2.2.4, we can suppose $\underline{\alpha} = (0, \ldots, 0)$. In fact, if $\underline{\alpha} \neq 0$, it is sufficient to change basis in

$$\mathcal{G} = \{f_0(x) + \alpha_1 f_1(x) + \ldots + \alpha_k f_k(x), f_1(x), \ldots, f_k(x)\}$$

that is a leading basis too. Hence $\mathfrak{m}_j = (a_1, \ldots, a_k)$ and $F(\underline{a}, x)$ divides $H(x)$ modulo $\mathfrak{m}_j$. So by definition of $\mathcal{I}_{\mathcal{F},H}$, $F(\underline{a}, x)$ divides $H(x)$ modulo $\mathcal{I}_{\mathcal{F},H}$. But $\mathcal{I}_{\mathcal{F},H} \subseteq \mathfrak{q}_j \subsetneq \mathfrak{m}_j$, hence $F(\underline{a}, x)$ should divide $H(x)$ modulo $\mathfrak{q}_j$, in contradiction with the previous lemma. □

2.3 Bound on the cardinality of $\mathcal{V}(\mathcal{I}_{\mathcal{F},H})$

In this section we are interested in determining some bounds on the cardinality of $\mathcal{V}(\mathcal{I}_{\mathcal{F},H})$ when $H(x) = x^{q-1} - 1$ that depend only on q, k and e. So, from now on, we work with

$$H(x) := x^{q-1} - 1.$$

By Corollary 2.2.2 $\mathcal{V}(\mathcal{I}_{\mathcal{F},H}) \subseteq \mathbb{F}_q^k$, so a first bound that we find is

$$|\mathcal{V}(\mathcal{I}_{\mathcal{F},H})| \leq q^k. \tag{2.2}$$

But we can do better.

Lemma 2.3.1. *If $k = 1$ then $|\mathcal{V}(\mathcal{I}_{\mathcal{F},H})| \leq q - e$.*

Proof. By euclidean division in $\mathbb{F}_q(a)[x]$ there exist unique polynomials $q(a, x)$, $r(a, x)$, that by Proposition 2.1.6 are in $\mathbb{F}_q[a][x]$, such that

$$x^{q-1} - 1 = F(a, x)q(a, x) + r(a, x)$$

with $r(a, x) = \sum_{i=0}^{e-1} r_i(a)x^i$. So

$$\mathcal{I}_{\mathcal{F},H} = (r_0(a), \ldots, r_{e-1}(a)) = (p(a)),$$

where $p(a) = \gcd\{r_0(a), \ldots, r_{e-1}(a)\}$. By Lemma 2.1.7 $\deg p \leq \deg r_i \leq q - 1 - e + 1 = q - e$ and we conclude the proof. □

We can use this lemma to improve the bound (2.2) as follows.

Proposition 2.3.2. $|\mathcal{V}(\mathcal{I}_{\mathcal{F},H})| \leq (q - e)q^{k-1}$.

Proof. It follows from the prevoius Lemma. For every possible choice of $a_2 = \alpha_2, \ldots, a_k = \alpha_k$ we have

$$x^{q-1} - 1 = F(a_1, \alpha_2, \ldots, \alpha_k, x)q(a_1, \alpha_2, \ldots, \alpha_k, x) + r(a_1, \alpha_2, \ldots, \alpha_k, x)$$

and we conclude observing that, by the previous Lemma, we have only $q - e$ possible values for a_1. □

A result due to L. Caniglia, A. Galligo, and J. Heintz. ([3]) improves the bound obtained again.

Theorem 2.3.3. *(Caniglia, Galligo, Heintz 1989 [3]) Let $\mathbb{F}$ be a field and $I \subseteq \mathbb{F}[\underline{a}] := \mathbb{F}[a_1, \ldots, a_k]$ be a zero-dimensional ideal generated by polynomials $\{r_1, \ldots, r_s\}$ of degrees $d_1 \geq \cdots \geq d_s$. Then*

$$\dim_{\mathbb{F}}(\mathbb{F}[\underline{a}]/I) \leq d_1 \cdot \ldots \cdot d_k.$$

Corollary 2.3.4. $|\mathcal{V}(\mathcal{I}_{\mathcal{F},H})| \leq (q - e)^k$.

Proof. The divisibility ideal $\mathcal{I}_{\mathcal{F},H}$ is generated by $r_0, \ldots, r_{e-1}$ that, by Lemma 2.1.7, are all of degree not greater than $q - e$. Then, since $\mathcal{I}_{\mathcal{F},H}$ is zero-dimensional, by the previous theorem

$$\dim_{\mathbb{F}_q}(\mathbb{F}_q[\underline{a}]/\mathcal{I}_{\mathcal{F},H}) \leq (q - e)^k.$$

Observing that $|\mathcal{V}(\mathcal{I}_{\mathcal{F},H})| \leq \dim_{\mathbb{F}_q}(\mathbb{F}_q[\underline{a}]/\mathcal{I}_{\mathcal{F},H})$, we conclude the proof. □

Another simple bound is the following.

Lemma 2.3.5. $|\mathcal{V}(\mathcal{I}_{\mathcal{F},H})| \leq \binom{q-1}{e}$.

Proof. The number of all the polynomials of degree e that divide $x^{q-1}-1$ is equal to the number of the subsets of $\mathbb{F}_q^*$ of cardinality e, i.e. the ways to choose the e roots, that is $\binom{q-1}{e}$. Furthermore, if $F(\alpha_1,\ldots,\alpha_k,x) = F(\beta_1,\ldots,\beta_k,x)$ then $(\alpha_1,\ldots,\alpha_k) = (\beta_1,\ldots,\beta_k)$. In fact,

$$F(\underline{\alpha},x) - F(\underline{\beta},x) = f_0(x) + \sum_{i=1}^{k} \alpha_i f_i(x) - f_0(x) - \sum_{i=1}^{k} \beta_i f_i(x)$$
$$= \sum_{i=1}^{k} (\alpha_i - \beta_i) f_i(x) = 0$$

implies $\alpha_i = \beta_i$ for every $i = 1,\ldots,k$ because of the linear independence of the f_i.

So, for every polynomial $G(x)$ of degree e that divides $x^{q-1}-1$, there is at most one $\underline{\alpha} = (\alpha_1,\ldots,\alpha_k)$ such that $F(\underline{\alpha},x) = G(x)$, and we conclude the proof.

□

To improve the bound shown above we need the following Lemma.

Lemma 2.3.6. *Let $\mathcal{T}_e$ be the set defined by*

$$\mathcal{T}_e := \left\{ p(x) \in \mathbb{F}_q[x] \mid \deg p = e,\ lc(p) = 1,\ p(x) | x^{q-1} - 1 \right\}.$$

If $e < q-1$, then $\mathcal{T}_e$ is a set of generators for the vector space $\mathbb{F}_q[x]_{\leq e}$.

Proof. We prove this lemma proceding by induction on e.

For $e = 1$ observe that $q > 2$. Then we can choose $\alpha_1, \alpha_2 \in \mathbb{F}_q$ such that $\alpha_1, \alpha_2 \neq 0$ and $\alpha_1 \neq \alpha_2$. We have that $(x-\alpha_1), (x-\alpha_2) \in \mathcal{T}_e$ and so also $(x-\alpha_1) - (x-\alpha_2) = \alpha_2 - \alpha_1$ is in $\mathcal{T}_e$. But $\alpha_2 - \alpha_1 \neq 0$, hence $\{x-\alpha_1, \alpha_2-\alpha_1\}$ is a basis of $\mathbb{F}_q[x]_{\leq 1}$.

Suppose that $\mathcal{T}_{e-1}$ generates $\mathbb{F}_q[x]_{\leq e-1}$ and let us show that $\mathcal{T}_{e-1} \subset \langle \mathcal{T}_e \rangle$. Let

$$p(x) = \prod_{i=1}^{e-1} (x - \alpha_i)$$

be a polynomial in $\mathcal{T}_{e-1}$. Since $e < q-1$ we can take $\beta, \gamma \in \mathbb{F}_q^*$ such that $\beta, \gamma \notin \{\alpha_1,\ldots,\alpha_{e-1}\}$. So the two polynomials

$$p_1(x) := (x-\beta)p(x), \qquad p_2(x) := (x-\gamma)p(x)$$

are included in $\mathcal{T}_e$, and

$$p_1(x) - p_2(x) = (x-\beta)p(x) - (x-\gamma)p(x) = (\gamma-\beta)p(x),$$

with $\gamma - \beta \neq 0$, so $p(x) \in \langle \mathcal{T}_e \rangle$. Therefore $\mathcal{T}_{e-1} \subset \langle \mathcal{T}_e \rangle$, whence $\langle \mathcal{T}_{e-1} \rangle \subset \langle \mathcal{T}_e \rangle$. By induction hypothesis $\langle \mathcal{T}_{e-1} \rangle = \mathbb{F}_q[x]_{\leq e-1}$ and so we can conclude that

$$\langle \mathcal{T}_e \rangle \supseteq \langle \mathcal{T}_{e-1} \cup \{p_1(x)\} \rangle = \langle \mathbb{F}_q[x]_{\leq e-1} \cup \{p_1(x)\} \rangle = \mathbb{F}_q[x]_{\leq e}$$

□

Theorem 2.3.7. *If $e > k$ then $|\mathcal{V}(\mathcal{I}_{\mathcal{F},H})| \leq \begin{pmatrix} q-1 \\ e \end{pmatrix} - (e-k)$*

Proof. It follows from Lemma 2.3.5 and from the fact that $\dim\langle f_0, \ldots, f_k \rangle = k+1$ and $\dim\langle \mathcal{T}_e \rangle = e+1$. So we need at least $e-k$ polynomials in $\mathcal{T}_e$ to complete $\langle f_0, \ldots, f_k \rangle$ as a basis of $\mathbb{F}_q[x]_{\leq e}$. □

Let us try to investigate what happens in some particular cases.

Remark 2.3.8. If $e = k$, then the subspace $W = \langle \mathcal{F} \rangle = \mathbb{F}_q[x]_{\leq e}$, and so $\mathcal{T}_e \subset W$. But $|\mathcal{T}_e| = \binom{q-1}{e}$, hence $|\mathcal{V}(\mathcal{I}_{\mathcal{F},H})| = \begin{pmatrix} q-1 \\ e \end{pmatrix}$.

Remark 2.3.9. If $e = q-2$, then $|\mathcal{T}_e| = q-1$. By Lemma 2.3.6 $\mathcal{T}_e$ generates the whole vector space $\mathbb{F}_q[x]_{\leq e}$, so the set $\mathcal{T}_e$ must be a basis of $\mathbb{F}_q[x]_{\leq e}$. Hence in the subspace $W = \langle \mathcal{F} \rangle$ we can find at most $k+1$ polynomials that are also in $\mathcal{T}_e$. So $|\mathcal{V}(\mathcal{I}_{\mathcal{F},H})| \leq k+1$.

Proposition 2.3.10. *If $k = 1$ there exist $f_0(x), f_1(x) \in \mathbb{F}_q[x]$ such that $|\mathcal{V}(\mathcal{I}_{\mathcal{F},H})| = q - e$.*

Proof. We choose

$$f_1(x) = \prod_{i=1}^{e-1}(x - \alpha_i),$$

where the α_i are all distinct and are in $\mathbb{F}_q^*$, and

$$f_0(x) = x f_1(x).$$

Then

$$F(a,x) = f_0(x) + a f_1(x) = (x-a)\prod_{i=1}^{e-1}(x-\alpha_i),$$

and $F(\beta, x)$ divides $x^{q-1} - 1$ for every $\beta \in \mathbb{F}_q^* \setminus \{\alpha_1, \ldots, \alpha_{e-1}\}$. Hence $|\mathcal{V}(\mathcal{I}_{\mathcal{F},H})| = q - e$. □

Proposition 2.3.11. *There exist $k+1$ polynomials $f_0(x), \dots, f_k(x) \in \mathbb{F}_q[x]$ such that*

$$|\mathcal{V}(\mathcal{I}_{\mathcal{F},H})| = \binom{q-1-e+k}{k}.$$

Proof. We choose

$$f_k(x) = \prod_{i=1}^{e-k}(x-\alpha_i),$$

where $\alpha_i \in \mathbb{F}_q^*$ for every $i = 1, \dots, e-k$ and $\alpha_i \neq \alpha_j$ for every $i \neq j$. Moreover we can take, for $j = 0, \dots, k-1$,

$$f_j(x) = x^{k-j} f_k(x).$$

Then

$$F(\underline{a}, x) = f_k(x)(x^k + a_1 x^{k-1} + \ldots + a_k),$$

and so $F(\underline{a}, x)$ divides $x^{q-1} - 1$ if and only if $x^k + a_1 x^{k-1} + \ldots + a_k$ divides

$$\frac{x^{q-1}-1}{f_k(x)} = \prod_{\alpha \in \mathbb{F}_q^* \setminus \{\alpha_1, \ldots \alpha_{e-k}\}} (x-\alpha) =: p(x).$$

The number of polynomials of degree k that divide $p(x)$ is equal to the number of ways to choose the k roots from the remaining $q-1-e+k$, i.e. $\binom{q-1-e+k}{k}$. Since $\langle x^k, x^{k-1}, \dots, 1\rangle = \mathbb{F}_q[x]_{\leq k}$, we obtain every such a polynomial with a suitable choice of $\underline{a} = \underline{\alpha}$. □

2.3.1 A conjecture on the bound

We have shown some bound on the cardinality of $\mathcal{V}(\mathcal{I}_{\mathcal{F},H})$ when $H(x) = x^{q-1} - 1$. Now we try to formalize better this problem in order to find the best possible bound.

Let $\mathbb{F}_q$ be a finite field and let e, k be positive integers with $k \leq e < q-1$. Let $\mathcal{T}_e$ be the set

$$\mathcal{T}_e = \left\{ p(x) \in \mathbb{F}_q[x] \mid \deg p = e \, , lc(p) = 1, \; p(x) | x^{q-1} - 1 \right\}.$$

We would like to find the value

$$M(e,k,q) := \max\left\{ |W \cap \mathcal{T}_e| \;\mid W \text{ is a subspace of } \mathbb{F}_q[x]_{\leq e}, \; \dim W = k+1 \right\}.$$

By using the results proved in the previous section we can determine exactly the value $M(e,k,q)$ in three simple cases.

- When $e = k$ then we have

$$M(e, e, q) = \binom{q-1}{e}.$$

  In fact, since $e = k$, we can take $W = \mathbb{F}_q[x]_{\leq e}$ and so we obtain $W \cap \mathcal{T}_e = \mathcal{T}_e$.

- When $k = 1$ then, by Lemma 2.3.1 and by Proposition 2.3.10, we have

$$M(e, 1, q) = q - e.$$

- If $e = q - 2$ then, as we have seen in Remark 2.3.9 we have

$$M(q-2, k, q) = k + 1.$$

For the general case we conjecture that the maximum $M(e, k, q)$ is obtained by taking a polynomial $p(x)$ of degree $e - k$ that divides $x^{q-1} - 1$ and choosing the subspace

$$W = \langle p(x), xp(x), \ldots, x^k p(x) \rangle.$$

From this we would obtain the following conjecture.

Conjecture 1. *Let q be a power of a prime, and let e, k positive integers such that $0 < k \leq e < q - 1$. Then*

$$M(e, k, q) = \binom{q-1-e+k}{k}.$$

Chapter 3

List Decoding

In this chapter we introduce the new notion for decoding error-correcting codes called *list decoding*. List decoding generalizes the notion of error-correction, when the number of errors is potentially very large.

Suppose that to transmit information over a noisy channel, the transmitter sends a codeword of an error-correcting code. This transmitted word is corrupted by the noisy channel, and the receiver gets some corrupted word, the so-called *received word*. If the number of errors that occurs during transmission is very large, then the received word may actually be closer to some other codeword other than the transmitted one. Under the mandate of list decoding, the receiver is required to compile a list of all codewords within a reasonable size Hamming ball around the received word, and not just the nearest one. The list decoding is declared to be succesful if this list includes the transmitted word.

Let's try to explain better what is list decoding. One of the first observations that can be made about a code C with minimum distance $d = 2t + 1$ is that it can unambiguously correct t errors, i.e., given a any word $x \in \mathcal{Q}^n$, there exists at most one codeword $c \in C$ such that $d(x, c) \leq t$. It is also easy to find a word x such there exist two codewords at distance $\leq t + 1$ from it, so one can not improve the error bound for unambiguous decoding. However it was realized early on that unambiguous decoding is not the only useful notion of recovery from error. In the 1950's Elias [4] proposed the notion of *list decoding* in which a decoding algorithm is expected to output a list of all codewords within a given distance e from a received word $x \in \mathcal{Q}^n$. If the list of words output is relatively small, then one could consider this to be a reasonable recovery from error. Algorithmically, this problem is stated as follows:

Definition. The *list decoding* problem for a code C is a process that takes

as input a received word $x \in \mathcal{Q}^n$ and an error bound e, and then outputs a list of all codewords $c^{(1)}, \ldots, c^{(s)} \in C$ that differ from x in at most e places.

As usual, the goal is to solve the list decoding problem efficiently, i.e. in time polynomial in n.

3.1 Decoding BCH codes

In this section we are going to study a way to analize the problem of list decoding for BCH codes. So, from now on we work with a primitive BCH code C of length $n = q^m - 1$ and designed distance $\delta = 2t + 1$ over $\mathbb{F}_q$.

We observe that, in order to correct a received word, it is sufficient to find the error-locator and the error-evaluator polynomials $\sigma(z)$, $\omega(z)$. So we can transform our goal in the research of all the possible couples of error-locator and error-evaluator polynomials $\{(\sigma_i(z), \omega_i(z))\}_{i=1}^s$ such that the corrispondent list of codewords $\{c^{(i)}\}_{i=1}^s$ satisfies the list decoding problem.

For a BCH code we have seen in the previous chapter that, in order to find the error-locator and the error-evaluator polynomials, using the so-called key equation

$$\omega(z) \equiv S(z)\sigma(z) \qquad \mod (z^{2t}),$$

it is sufficient to solve the linear system

$$\begin{bmatrix} s_1 & s_2 & \cdots & s_{e+1} \\ s_2 & s_3 & \cdots & s_{e+2} \\ \vdots & \vdots & & \vdots \\ s_{2t-e} & s_{2t-e+1} & \cdots & s_{2t} \end{bmatrix} \begin{bmatrix} \sigma_e \\ \sigma_{e-1} \\ \vdots \\ \sigma_1 \\ \sigma_0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

Such a linear system is of the form

$$S_e \sigma = 0$$

with $S_e \in M(2t - e, e + 1, \mathbb{F}_{q^m})$ and $\sigma \in \mathbb{F}_{q^m}^{e+1}$.

Definition 3.1.1. The matrix $S_e \in M(2t - e, e + 1, \mathbb{F}_{q^m})$ defined above is called the *e-th syndrome matrix of the received word* r.

Now we would like to know what happens when $e \leq t$, i.e. when we can effectively correct the received word, and when $e > t$, i.e. when we are beyond the error-correcting bound.

3.2 Unique decoding

When the number of errors is not greater of the error-correction bound we have $S_e \in M(2t-e, e+1, \mathbb{F}_q)$, with $2t-e \geq e$. In this case $rk(S_e) \leq e$, because

$$e+1 = rk(S_e) + \dim \ker(S_e),$$

and we know that there exists $\sigma \in \ker(S_e)$.

The following proposition shows that $rk(S_e) = e$.

Proposition 3.2.1. *Let e be the number of errors, with $e \leq t$, and let S'_e be the principal square submatrix of S of order e, i.e. the submatrix*

$$S'_e = \begin{bmatrix} s_1 & s_2 & \cdots & s_e \\ s_2 & s_3 & \cdots & s_{e+1} \\ \vdots & & & \vdots \\ s_e & s_{e+1} & \cdots & s_{2e-1} \end{bmatrix}$$

of S_e obtained taking the first e rows and the first e columns of S_e. Thus S'_e is invertible.

Proof. Consider the two matrices

$$V = \begin{bmatrix} 1 & a_1 & \cdots & a_1^{e-1} \\ 1 & a_2 & \cdots & a_2^{e-1} \\ \vdots & & \ddots & \vdots \\ 1 & a_e & \cdots & a_e^{e-1} \end{bmatrix}, \quad D = \begin{bmatrix} E_1 a_1 & & & 0 \\ & E_2 a_2 & & \\ & & \ddots & \\ 0 & & & E_e a_e \end{bmatrix}.$$

V is a Vandermonde matrix with $a_i \neq a_j$ for all $i \neq j$, so it is invertible. Also D is invertible because it is a diagonal matrix with non-zero entries on the

principal diagonal. Moreover we have

$$
\begin{aligned}
V^T D V &= \begin{bmatrix} E_1 a_1 & E_2 a_2 & \cdots & E_e a_e \\ E_1 a_1^2 & E_2 a_2^2 & & E_e a_e^2 \\ \vdots & & \ddots & \vdots \\ E_1 a_1^e & E_2 a_2^e & \cdots & E_e a_e^e \end{bmatrix} V \\
&= \begin{bmatrix} \sum E_i a_i & \sum E_i a_i^2 & \cdots & \sum E_i a_i^e \\ \sum E_i a_i^2 & \sum E_i a_i^3 & & \sum E_i a_i^{e+1} \\ \vdots & & \ddots & \vdots \\ \sum E_i a_i^e & \sum E_i a_i^{e+1} & \cdots & \sum E_i a_i^{2e-1} \end{bmatrix} \\
&= \begin{bmatrix} s_1 & s_2 & \cdots & s_e \\ s_2 & s_3 & \cdots & s_{e+1} \\ \vdots & & & \vdots \\ s_e & s_{e+1} & \cdots & s_{2e-1} \end{bmatrix} = S'_e.
\end{aligned}
$$

Then S'_e is invertible. □

Therefore, when $e \leq t$ we have that $rk(S_e) = e$ and $\dim \ker(S_e) = 1$, so the error-locator polynomial σ is the unique (up to scalar) generator of $\ker(S_e)$.

3.3 The decoding function

Now suppose that the number of errors e that occur in the received word r is greater than the error-correction bound t. We put $e = t + s$ where s is a positive integer smaller than t. So the e-th syndrome matrix S_e has $t - s$ rows and $t + s + 1$ columns. For the rank-nullity theorem

$$t + s + 1 = rk(S_e) + \dim \ker(S_e)$$

and $rk(S_e) \leq t - s$, so $\dim \ker(S_e) \geq 2s + 1$. Also in this case, by the following proposition, we know exactly the rank of S_e.

Proposition 3.3.1. *Let $e > t$ be the number of errors, with $e = t + s$ and $s < t$. Then the principal square submatrix S'_e of order $t - s$ is invertible.*

Proof. The proof is the same of the case $e \leq t$. We have

$$S'_e = \widetilde{V}^T \widetilde{D} \widetilde{V},$$

where

$$\widetilde{V} = \begin{bmatrix} 1 & a_1 & \cdots & a_1^{t-s-1} \\ 1 & a_2 & \cdots & a_2^{t-s-1} \\ \vdots & & \ddots & \vdots \\ 1 & a_{t-s} & \cdots & a_{t-s}^{t-s-1} \end{bmatrix}, \quad \widetilde{D} = \begin{bmatrix} E_1 a_1 & & & 0 \\ & E_2 a_2 & & \\ & & \ddots & \\ 0 & & & E_{t-s} a_{t-s} \end{bmatrix}.$$

Since $\widetilde{V}$ and $\widetilde{D}$ are both invertible, we conclude the proof. □

Corollary 3.3.2. *Let $e > t$ be the number of errors, with $e = t+s$ and $s < t$. Then $rk(S_e) = t - s$ and $\dim\ker(S_e) = 2s + 1$.*

In this case $\sigma \in \ker(S_e)$, with $\dim\ker(S_e) = 2s + 1$. So there exist $f_0, f_1, \ldots, f_{2s} \in \mathbb{F}_{q^m}[z]$ such that $\mathcal{F} = \{f_0, f_1, \ldots, f_{2s}\}$ is an escalier basis of $\ker(S)$, i.e. $\ker(S) = \langle f_0, f_1, \ldots, f_{2s}\rangle$,

$$e = \deg f_0 > \deg f_1 > \cdots > \deg f_{2s},$$

and $lc(f_i) = 1 \ \ \forall\, i = 0, \ldots, 2s$.

What do we know about the error-locator polynomial $\sigma(z)$? We know that:

- $\deg\sigma = e$;
- all the roots of σ are in $\mathbb{F}_{q^m}$;
- σ has not multiple roots;
- 0 is not a root of σ.

The last three conditions are equivalent to the following

- σ divides the polynomial $z^{q^m-1} - 1$.

So, what we can do is listing all the possible error-locator polynomials in $\ker(S_e)$ that satisfies the conditions above, i.e. all the polynomials of the form

$$\sigma(a_1, \ldots, a_{2s}, z) := f_0(z) + \sum_{i=1}^{2s} a_i f_i(z)$$

such that $\sigma(\underline{a}, z)$ divides $z^{q^m-1} - 1$.

Let $H(z) = z^{q^m-1} - 1$, our goal is finding the variety $V(\mathcal{I}_{\mathcal{F},H})$ of the divisibility ideal $\mathcal{I}_{\mathcal{F},H}$.

Definition 3.3.3. In this case the divisibility set of $H(z) = z^{q^m-1} - 1$ by $\mathcal{F}$ is called *error-locator set of degree e of the received word r* and is denoted by

$$\Sigma(r, e) := \{\sigma(\underline{\alpha}, z) \mid \underline{\alpha} \in V(\mathcal{I}_{\mathcal{F},H})\}.$$

Remark 3.3.4. If we define the set

$$\mathcal{T}_e := \left\{ p(z) \in \mathbb{F}_{q^m}[z] \mid \deg p = e, p(z) | z^{q^m-1} - 1, lc(p) = 1 \right\},$$

then

$$\Sigma(r, e) = \ker(S_e) \cap \mathcal{T}_e.$$

Before understanding if a $\sigma \in \Sigma$ produces a word c that effectively is in the code C, we would like to understand how we can obtain such c from a polynomial $\sigma \in \Sigma$

Suppose we have a possible error-locator polynomial σ, how can we correct the received word r? We know the positions of the errors, but to correct r we also need the values E_i of the error vector E. Now we briefly present two ways to find E.

The first is by the Forney's equation. The key equation tells us that the error-evaluator polynomial ω satisfies

$$\sigma(z)S(z) \equiv \omega(z) \qquad \mod \left(z^{2t}\right).$$

But $\deg \omega < 2t$, so $\omega(z) = (\sigma(z)S(z) \mod (z^{2t}))$.

After calculating the polynomial ω, the Forney's equation gives us the values

$$E_i = -\frac{\omega(a_i^{-1})}{\sigma'(a_i^{-1})}.$$

The second method is based on the definition of the syndromes.

$$s_i = \sum_{j=1}^{e} E_j a_j^i \qquad i = 1, \ldots 2t.$$

If we take only the first e equations we obtain the linear system

$$\begin{bmatrix} a_1 & a_2 & \cdots & a_e \\ a_1^2 & a_2^2 & \cdots & a_e^2 \\ \vdots & & & \vdots \\ a_1^e & a_2^e & \cdots & a_e^e \end{bmatrix} \begin{bmatrix} E_1 \\ E_2 \\ \vdots \\ E_e \end{bmatrix} = \begin{bmatrix} s_1 \\ s_2 \\ \vdots \\ s_e \end{bmatrix} \tag{3.1}$$

Observe that the matrix A of the system is a quasi-Vandermonde matrix with determinant

$$\det A = \prod_{i=1}^{e} a_i \prod_{i<j} (a_i - a_j).$$

Since $a_i \neq 0$ for all $i = 1, \ldots, e$ and $a_i \neq a_j$ for all $i \neq j$, such determinant is not zero, so the linear system has a unique solution

$$E = \begin{bmatrix} E_1 \\ E_2 \\ \vdots \\ E_e \end{bmatrix} \in (\mathbb{F}_q)^e .$$

Now we would like to know if the two methods are equivalent, i.e. if the solutions obtained by using the two methods are the same.

Theorem 3.3.5. *Let C be a BCH code of length $n = q^m - 1$ and designed distance $\delta = 2t + 1$ over $\mathbb{F}_q$. Let $r \in \mathbb{F}_q^n$ be a received word and suppose that in r occur $e > t$ errors, with $e = t + s$ and e-th syndrome matrix $S_e \in M(t - s, t + s + 1, \mathbb{F}_q)$. Let $\mathcal{F} = \{f_0, \ldots, f_{2s}\}$, with $\ker(S_e) = \langle \mathcal{F} \rangle$ and let $\Sigma(r, e)$ be the error-locator set of degree e for r. Then, for every $\sigma \in \Sigma(r, e)$, the vector $\widetilde{E}$ obtained by Forney's equation is equal to the vector $\widehat{E}$ obtained solving the linear system 3.1.*

Proof. Let $\widehat{E}$ the error vector obtained by solving the linear system 3.1. Then it satisfies the equations

$$s_i = \sum_{j=1}^{e} \widehat{E}_j a_j^i \qquad i = 1, \ldots, 2t.$$

Now let us define

$$\widehat{\omega}(z) := \sum_{i=1}^{e} \widehat{E}_i a_i \prod_{j \neq i} (1 - a_j z).$$

Now observe that

$$\begin{aligned} \frac{\widehat{\omega}(z)}{\sigma(z)} &= \sum_{i=1}^{e} \frac{\widehat{E}_i a_i}{1 - a_i z} = \sum_{i=1}^{e} \widehat{E}_i a_i \sum_{l=0}^{\infty} (a_i z)^l \\ &= \sum_{l=0}^{\infty} z^l \sum_{i=1}^{e} \widehat{E}_i a_i^{l+1} = \sum_{l=0}^{2t-1} s_{l+1} z^l + \sum_{l=2t}^{\infty} B_l z^l, \end{aligned}$$

where $B_l = \sum_{i=1}^{e} \widehat{E}_i a_i^{l+1}$. So $\widehat{\omega}(z)$ satisfies the key equation

$$\widehat{\omega}(z) \equiv \sigma(z) S(z) \mod (z^{2t}).$$

Since $\deg \widehat{\omega}(z) < e < 2t$, it must be $\widehat{\omega} = (\sigma(z)S(z) \mod (z^{2t})) = \widetilde{\omega}(z)$. So, by Forney's equation we obtain

$$\widehat{E}_i = -\frac{\widehat{\omega}(a_i^{-1})}{\sigma'(a_i^{-1})} = -\frac{\widetilde{\omega}(a_i^{-1})}{\sigma'(a_i^{-1})} = \widetilde{E}_i$$

and we conclude the proof. □

Therefore we can define the notion of decoding by a polynomial σ.

Definition 3.3.6. Let r be a received word and $\sigma \in \Sigma(r, e)$ and let E be the vector obtained by one of the two methods shown above. Then the word obtained from r *decoding by* σ is the word $c = r - \varepsilon$, where ε is the error vector defined by $\varepsilon_{j_i} = E_i$ and $\varepsilon_l = 0$ for $l \notin \{j_1, \ldots, j_e\}$. Moreover we can define the *e-decoding function of* r to be the function

$$\mathcal{D}_{r,e} : \Sigma(r, e) \longrightarrow \mathbb{F}_q^n$$

such that

$$\mathcal{D}_{r,e}(\sigma) = c$$

Then, given the set $\Sigma(r, e)$ of all the possible error-locator polynomials, we would like to know if, decoding the received word r by some $\sigma \in \Sigma(r, e)$, we really obtain a word c of the code C, i.e. if the two conditions

1. $\deg \sigma = e$,
2. $\sigma(z)$ divides $z^{q^m-1} - 1$,

are also sufficient.

3.4 Fundamental Theorems for list decoding

We have two different answers for RS codes and BCH codes. In fact, while for BCH codes the two conditions above are not sufficient, for a Reed-Solomon code we have the following theorem.

Theorem 3.4.1 (Fundamental Theorem for RS list decoding)**.** *Let C be a Reed-Solomon code of length $n = q - 1$ and dimension $k + 1$ over $\mathbb{F}_q$, with $n-k = 2t+1$. Let $r \in \mathbb{F}_q^n$ be a received word and suppose that in r occur $e > t$ errors, with $e = t + s$ and e-th syndrome matrix $S_e \in M(t - s, t + s + 1, \mathbb{F}_q)$. Let $\mathcal{F} = \{f_0, \ldots, f_{2s}\}$ be an escalier basis for $\ker(S_e)$ and let $\Sigma(r, e)$ be the error-locator set of degree e for r. Then, for every $\sigma \in \Sigma(r, e)$, $\mathcal{D}_{r,e}(\sigma) \in C$.*

Proof. Since $\sigma \in \ker(S_e)$, then σ satisfy the key equation

$$\sigma(z)S(z) \equiv \omega(z) \mod (z^{2t}), \tag{3.2}$$

with $\deg \omega < e$.

Let $a_1, \ldots, a_e$ be the roots of σ. Since $\sigma \in \Sigma(r, e)$ then the roots are all distinct and they are in $\mathbb{F}_q^*$. Then the set of polynomials $\{p_i(z)\}_{i=1}^e$ defined by

$$p_i(z) := a_i \prod_{j \neq i} (1 - a_j z)$$

is a basis of the vector space $\mathbb{F}_q[z]_{<e}$. Then we can write

$$\omega(z) = \sum_{i=1}^{e} E_i p_i(z) = \sum_{i=1}^{e} E_i a_i \prod_{j \neq i} (1 - a_j z).$$

By the Key equation we have

$$\frac{\omega(z)}{\sigma(z)} \equiv S(z) \mod (z^{2t}) \tag{3.3}$$

and by the definition of ω and σ

$$\begin{aligned} \frac{\omega(z)}{\sigma(z)} &= \sum_{i=1}^{e} \frac{E_i a_i}{1 - a_i z} = \sum_{i=1}^{e} E_i a_i \sum_{\ell=0}^{\infty} a_i^{\ell} z^{\ell} \\ &= \sum_{\ell=0}^{\infty} z^{\ell} \sum_{i=1}^{e} E_i a_i^{\ell+1}. \end{aligned}$$

Hence

$$E_i a_i^{\ell+1} = s_{\ell+1} \qquad \text{for } \ell = 0, 1, \ldots, 2t-1.$$

Let ε the error vector defined by $\varepsilon_{j_i} = E_i$ and $\varepsilon_l = 0$ for $l \notin \{j_1, \ldots, j_e\}$, where the j_i are such that $a_i = \beta^{j_i}$ and β is a primitive element of $\mathbb{F}_q$.

We obtain that $\mathcal{D}_{r,e}(\sigma) = r - \varepsilon$ satisfies, for all $i = 0, \ldots, 2t-1$,

$$\mathcal{D}_{r,e}(\sigma)(\beta^i) = r(\beta^i) - \varepsilon(\beta^i) = s_i - \sum E_j a_j^i = s_i - s_i = 0,$$

so it belongs to the code C.

□

Remark 3.4.2. Consider the decoding function

$$\mathcal{D}_{r,e} : \Sigma(r, e) \longrightarrow \mathbb{F}_q^n$$

with $\mathcal{D}_{r,e}(\sigma) = c$, where c is the vector obtained decoding r by σ. The *Fundamental Theorem for RS List Decoding* states that for every σ in the error-locator set $\Sigma(r, e)$, $\mathcal{D}_{r,e}(\sigma)$ is a codeword. Moreover observe that, even if we imposed that the distance between r and the codeword c must have distance e, we can not be sure that $d\left(r, \mathcal{D}_{r,e}(\sigma)\right) = e$. But, by looking at how the decoding function works, we can state that

$$d\left(r, \mathcal{D}_{r,e}(\sigma)\right) \leq e.$$

In particular the decoding function is defined

$$\mathcal{D}_{r,e} : \Sigma(r, e) \longrightarrow B(r, e) \cap C.$$

Let C be a primitive BCH code of length $q^m - 1$ and designed distance $\delta = 2t + 1$, i.e. the BCH code over $\mathbb{F}_q$ generated by the polynomial

$$lcm(m_1(x), \ldots, m_d(x)),$$

where $m_i(x)$ is the minimal polynomial of α^i over $\mathbb{F}_q$ and α is a primitive element of $\mathbb{F}_{q^m}$.

While in the case $e \leq t$ everything done for RS codes is valid also for BCH codes, in the case $e > t$ we have some problems. The *Fundamental Theorem for RS list decoding* is not true for a BCH code. In fact, given a $\sigma \in \Sigma$, decoding the received word by σ there is a problem with the linear system

$$\begin{bmatrix} a_1 & a_2 & \cdots & a_e \\ a_1^2 & a_2^2 & \cdots & a_e^2 \\ \vdots & & \ddots & \vdots \\ a_1^e & a_2^e & \cdots & a_e^e \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_e \end{bmatrix} = \begin{bmatrix} s_1 \\ s_2 \\ \vdots \\ s_e \end{bmatrix}$$

$$AE = s.$$

This is a system of linear equations with coefficients in $\mathbb{F}_{q^m}$, but we need the error vector E to be in $(\mathbb{F}_q)^e$. So we need other conditions, for example we could require that

$$A^{-1}s \in (\mathbb{F}_q)^e$$

or, equivalently, that

$$\frac{\omega(a_i^{-1})}{\sigma'(a_i^{-1})} \in \mathbb{F}_q \qquad \forall\, i = 1, \ldots, e.$$

One of these conditions is necessary, but it is also sufficient, as the following theorem shows.

Theorem 3.4.3 (Fundamental Theorem for BCH list decoding)**.** *Let C be a BCH code of length $n = q^m - 1$ and designed distance $\delta = 2t + 1$ over $\mathbb{F}_q$. Let $r \in \mathbb{F}_q^n$ be a received word and suppose that in r occur $e > t$ errors, with $e = t + s$ and e-th syndrome matrix $S_e \in M(t - s, t + s + 1, \mathbb{F}_{q^m})$. Let $\mathcal{F} = \{f_0, \ldots, f_{2s}\}$ be an escalier basis for $\ker(S_e)$ and let $\Sigma(r, e)$ be the error-locator set of degree e for r. Given $\sigma(z) \in \Sigma(r, e)$, we put $\omega(z) := (\sigma(z)S(z) \mod (z^{2t}))$. Then, $\mathcal{D}_{r,e}(\sigma) \in C$ if and only if*

$$\frac{\omega(a_i^{-1})}{\sigma'(a_i^{-1})} \in \mathbb{F}_q \quad \forall\, i = 1, \ldots, e. \tag{3.4}$$

Proof. Obviously the condition

$$\frac{\omega(a_i^{-1})}{\sigma'(a_i^{-1})} \in \mathbb{F}_q \quad \forall\, i = 1, \ldots, e$$

is necessary, because otherwise $\mathcal{D}_{r,e}(\sigma) \notin \mathbb{F}_q^n$, and so it is not in C.

Moreover it is also sufficient. First of all observe that if (3.4) holds, then $\mathcal{D}_{r,e}(\sigma) \in \mathbb{F}_q^n$. Now, proceeding in the same way as in the proof of Theorem 3.4.1 we observe that

$$\mathcal{D}_{r,e}(\sigma)(\beta^i) = r(\beta^i) - \varepsilon(\beta^i) = s_i - \sum E_j a_j^i = s_i - s_i = 0.$$

From this we can conclude that $\mathcal{D}_{r,e}(\sigma) \in C$.

□

3.5 Decoding binary BCH codes

For a binary BCH code, when we try to decode a received word r by σ, we observe that, by the Forney's equation, we have

$$\frac{\omega(a_i^{-1})}{\sigma'(a_i^{-1})} = E_i \qquad \forall\, i = 1, \ldots, e.$$

Assuming that the received word r contains *exactly* e errors we have that $E_i \in \mathbb{F}_2^*$, i.e. $E_i = 1$ for every $i = 1, \ldots, e$. So $\omega(a_i) = \sigma'(a_i)$ in e points. Furthermore $\deg \omega < e$ and $\deg \sigma' < \deg \sigma = e$. Therefore

$$\omega(z) = \sigma'(z).$$

We can put it in the key equation obtaining

$$\sigma'(z) \equiv S(z)\sigma(z) \qquad \mod (z^{2t}).$$

If we write

$$\sigma(z) = \sigma_e(z^2) + z\sigma_o(z^2)$$

then $\sigma'(z) = \sigma_o(z^2)$, i.e. $\sigma'(z) = \sigma'_{e-1} z^{e-1} + \ldots + \sigma'_0$, with

$$\sigma'_k = \begin{cases} \sigma_{k+1} & \text{if } k \text{ is even} \\ 0 & \text{if } k \text{ is odd} \end{cases}$$

and the key equation holds for every coefficient of the two polynomials. So we have, for every $k = 0, \ldots, e-1$,

$$\sum_{i=0}^{k} \sigma_i s_{k-i+1} = (\sigma S)_k = \sigma'_k = \begin{cases} \sigma_{k+1} & \text{if } k \text{ is even} \\ 0 & \text{if } k \text{ is odd} \end{cases}.$$

Summarizing, we have, for $k = 0, \ldots, e-1$,

$$\sum_{i=0}^{k} \sigma_i s_{k-i+1} + \sigma_{k+1} = 0 \qquad \text{if } k \text{ is even},$$

$$\sum_{i=0}^{k} \sigma_i s_{k-i+1} = 0 \qquad \text{if } k \text{ is odd},$$

and, for $k = e, \ldots, 2t-1$

$$\sum_{i=0}^{k} \sigma_i s_{k-i+1} = 0,$$

where $\sigma_k := 0$ if $k > e$.

Then we obtain the linear system

$$\widetilde{S}_e \sigma = 0$$

where $\widetilde{S}_e \in M(2t, e+1, \mathbb{F}_{2^m})$ is the matrix defined as follows:

$$\widetilde{S}_e = \begin{bmatrix} 0 & \cdots & & & 0 & 1 & s_1 \\ 0 & \cdots & & & 0 & s_1 & s_2 \\ 0 & \cdots & & 1 & s_1 & s_2 & s_3 \\ 0 & \cdots & 0 & s_1 & s_2 & s_2 & s_4 \\ \vdots & \iddots & \iddots & & & & \vdots \\ \delta & s_1 & & \cdots & & & s_e \\ s_1 & s_2 & & \cdots & & & s_{e+1} \\ s_2 & s_3 & & \cdots & & & s_{e+2} \\ \vdots & \vdots & & & & & \vdots \\ s_{2t-e} & s_{2t-e+1} & & \cdots & & & s_{2t} \end{bmatrix}, \qquad \delta = \begin{cases} 0 & \text{if } e \text{ is even} \\ 1 & \text{if } e \text{ is odd} \end{cases}$$

Definition 3.5.1. The matrix $\widetilde{S}_e \in M(2t, e+1, \mathbb{F}_{2^m})$ defined above is called the *e-th extended syndrome matrix* of the word r.

We are intereseted to understand what happens when $t < e < 2t$. Are those new conditions useful, or they are dependent by the equations in S_e? Obviously, since S_e is a submatrix of $\widetilde{S}_e$, $\dim \ker(\widetilde{S}_e) \leq \dim \ker(S_e) = 2s+1$. Then we would like to know if $\dim \ker(\widetilde{S}_e) < \dim \ker(S_e)$ or if they are equal.

Proposition 3.5.2. $\dim\ker(\widetilde{S}_e) \leq s+1$

Proof. Consider the submatrix $(t-s)\times(t-s)$ obtained taking the first $t-s$ columns and the last $t-s$ rows. This submatrix is exactly the principal square submatrix S'_e of S_e. Then we add the rows L_{2j-1} and the columns C_{t+s-2j} for $j = 0, 1, \ldots, s-1$. We obtain the following submatrix:

$$T = \left[\begin{array}{c|cccc} & & & & 1 \\ & 0 & 1 & & s_2 \\ 0 & & \ddots & \ddots & \\ & 1 & s_2 & & \\ & 1\ s_2 & s_4 & & s_{2s} \\ \hline S'_e & & \Lambda & & \end{array}\right].$$

It is clear that $\det(T) = \pm\det(S'_e) \neq 0$, so $rk(\widetilde{S}_e) \geq t$. Hence $\dim\ker(\widetilde{S}_e) \leq s+1$. □

For a binary BCH code ther is a nice property that involves the syndromes, and it is expressed by the following proposition.

Proposition 3.5.3. *In a binary BCH code C the syndromes satisfy the relation*

$$s_{2j} = s_j^2 \quad \forall\, j = 1, \ldots, t.$$

Proof. Since C is a binary BCH code the errors E_i are all 1s. So the syndromes can be rewritten as

$$s_j = \sum_{i=1}^{e} E_i a_i^j = \sum_{i=1}^{e} a_i^j.$$

Then

$$s_{2j} = \sum_{i=1}^{e} a_i^{2j} \overset{(*)}{=} \left(\sum_{i=1}^{e} a_i^j\right)^2 = s_j^2$$

where the identity $(*)$ holds because we are in characteristic 2. □

Theorem 3.5.4. *Let $T_k \in M(2k, 2k, \mathbb{F}_{2^m})$ be the matrix defined as follows:*

$$T_k = \begin{bmatrix} 0 & \cdots & & & 0 & 1 & s_1 \\ 0 & \cdots & & & 0 & s_1 & s_2 \\ 0 & \cdots & & 1 & s_1 & s_2 & s_3 \\ 0 & \cdots & 0 & s_1 & s_2 & s_3 & s_4 \\ \vdots & & & & & & \vdots \\ 1 & s_1 & & \cdots & & & s_{2k-1} \\ s_1 & s_2 & & \cdots & & & s_{2k} \end{bmatrix},$$

with the properties $s_{2j} = s_j^2 \quad \forall\, j = 1, \ldots, k$. Then the rank of T_k is exactly k.

Proof. We proceed by induction on k.

For $k = 1$ we have the matrix

$$T_1 = \begin{bmatrix} 1 & s_1 \\ s_1 & s_2 \end{bmatrix},$$

that has determinant $s_1^2 - s_2 = 0$, and so its rank is 1.

Now suppose that $rk(T_{k-1}) = k - 1$; let us transform the matrix through the following series of row and column operations:

$$C_{2k} \leftarrow C_{2k} + s_1 C_{2k-1},$$

and, for all odd j such that $3 \leq j \leq 2k - 3$,

$$L_h \leftarrow L_h + s_{h-j} L_j \qquad \forall\, h \in \{j+1, j+2, \ldots, 2k\}.$$

After those operations the matrix obtained from T_k is the following:

$$T'_k = \begin{bmatrix} 0 & D \\ M & ? \end{bmatrix},$$

where $D = \begin{bmatrix} 1 & 0 \\ s_1 & 0 \end{bmatrix}$, M is equivalent to T_{k-1} and its odd columns are

$$\begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \ldots, \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}.$$

By the induction hypothesis M has rank $k-1$ and hence its column space is spanned by the above vectors. To conclude that T_k has rank k we then need to prove that the even-labelled entries in the last column of T'_k are all zero.

Let $p \in \{2, 3, \ldots, k\}$. By careful examination we find that the $(2p, 2k)$ entry of T'_k equals

$$s_{2p} + \sum_{(i_1,\ldots,i_\ell)\in A_p} s_{i_1} s_{i_2} \cdot \ldots \cdot s_{i_\ell},$$

where A_p is the set of all sequences $(i_1, \ldots, i_l)$ of positive integers whose sum is $2p$ and in which i_1 and i_l are odd and the other i_h are even. Our goal is to show that this sum equals s_{2p}

Observe that the map $\sigma : (i_1, i_2 \ldots, i_\ell) \mapsto (i_\ell, i_{\ell-1}, \ldots, i_1)$ is an involution on A_p and $s_{i_1} s_{i_2} \cdot \ldots \cdot s_{i_\ell} = s_{i_\ell} s_{i_{\ell-1}} \cdot \ldots \cdot s_{i_1}$, so the sum is only on the symmetric sequences in A_p, i.e.

$$\sum_{(i_1,\ldots,i_\ell)\in A_p} s_{i_1} s_{i_2} \cdot \ldots \cdot s_{i_\ell} = \sum_{(i_1,\ldots,i_\ell)\in A_p (i_1,\ldots,i_\ell)=(i_\ell,\ldots,i_1)} s_{i_1} s_{i_2} \cdot \ldots \cdot s_{i_\ell}$$

Next, if we have a symmetric sequence $(i_1, \ldots, i_{2j}) = (i_{2j}, \ldots, i_1)$ in A_p with an even number of entries and $j \geq 2$ we use the equality

$$s_{i_j} s_{i_{j+1}} = s_{i_j}^2 = s_{2i_j}$$

to see that it defines the same product as the symmetric sequence with $2j-1$ entries $(i_1, \ldots, i_{j-1}, 2i_j, i_{j+2}, \ldots i_{2j})$. Pairing sequences in this manner and, if p is odd, noting that $s_p^2 = s_{2p}$, we find that

$$\sum_{(i_1,\ldots,i_\ell)\in A_p} s_{i_1} s_{i_2} \cdot \ldots \cdot s_{i_\ell} = \begin{cases} s_{2p} + \sum_{(i_1,\ldots,i_\ell)\in B_p} s_{i_1} s_{i_2} \cdot \ldots \cdot s_{i_\ell} & \text{if } p \text{ is odd} \\ \sum_{(i_1,\ldots,i_\ell)\in B_p} s_{i_1} s_{i_2} \cdot \ldots \cdot s_{i_\ell} & \text{if } p \text{ is even,} \end{cases}$$

where B_p denotes the subset of A_p consisting of the symmetric sequences $(i_1, \ldots, i_{2j+1})$ in which $i_{j+1} = 2t$ for some odd integer t. If p is odd then B_p is empty and we are done.

Assume now that $p = 2q$ for some integer q. Then,

$$\theta : (i_1, ..., i_{2j+1}) \mapsto (i_1, ..., i_{j-1}, i_j, (i_{j+1})/2)$$

maps B_p bijectively onto A_q, and using $s_{i_{j+1}} = s^2_{(i_{j+1}/2)}$ we deduce that

$$\sum_{(i_1,\ldots,i_\ell)\in B_p} s_{i_1} s_{i_2} \cdot \ldots \cdot s_{i_\ell} = \left(\sum_{(i_1,\ldots,i_\ell)\in A_q} s_{i_1} s_{i_2} \cdot \ldots \cdot s_{i_\ell} \right)^2 .$$

By induction, we conclude that

$$\sum_{(i_1,\ldots,i_\ell)\in B_p} s_{i_1}s_{i_2}\cdot\ldots\cdot s_{i_\ell} = (s_{2q})^2 = s_{2p}.$$

□

Corollary 3.5.5. $\dim\ker(\widetilde{S}_e) = s+1$

Proof. We have that $\dim\ker(\widetilde{S}_e) \leq s+1$, and we know by the previous theorem that all the columns of $\widetilde{S}_e$ are linear combinations of the t vectors

$$\begin{bmatrix} 0\\0\\\vdots\\0\\0\\1\\s_1 \end{bmatrix}, \begin{bmatrix} 0\\0\\\vdots\\1\\s_1\\s_2\\s_3 \end{bmatrix}, \ldots, \begin{bmatrix} 1\\s_1\\\vdots\\s_{2t-4}\\s_{2t-3}\\s_{2t-2}\\s_{2t-1} \end{bmatrix}.$$

So $rk(\widetilde{S}_e) \leq t$. Then $\dim\ker(\widetilde{S}_e) = t+s+1-rk(\widetilde{S}_e) \geq t+s+1-t = s+1$, and we conclude. □

Now we know that in binary BCH codes the kernel of the extended syndrome matrix $\widetilde{S}_e$ is $s+1$-dimensional. So there exist only $s+1$ polynomials $f_0, f_1, \ldots, f_s \in \mathbb{F}_{2^m}[z]$ such that $\{f_0, f_1, \ldots, f_s\}$ is an escalier basis for $\ker(\widetilde{S}_e)$, and we put

$$\widetilde{\mathcal{F}} := \{f_0, f_1, \ldots, f_s\}$$

Definition 3.5.6. In this case the divisibility set of $H(z) = z^{q^m-1} - 1$ by $\widetilde{\mathcal{F}}$ is called *extended error-locator set of degree e of the received word r* and is denoted by

$$\widetilde{\Sigma}(r,e) := \left\{\sigma(\underline{\alpha}, z) \mid \underline{\alpha} \in V(\mathcal{I}_{\widetilde{\mathcal{F}},H})\right\}.$$

Also the decoding function is naturally extended as the function

$$\widetilde{\mathcal{D}}_{r,e} : \widetilde{\Sigma}(r,e) \longrightarrow \mathbb{F}_2^n,$$

where $\widetilde{\mathcal{D}}_{r,e}(\sigma) = r - \varepsilon$, and the vector ε is obtained by Forney's equations or by solving the linear system (3.1).

We can state now a stronger form of the Fundamental Theorem for BCH list decoding, that is valid only for binary BCH codes.

Theorem 3.5.7 (Fundamental Theorem for binary BCH list decoding)**.** *Let C be a binary BCH code of length $n = 2^m - 1$ and designed distance $\delta = 2t+1$ over $\mathbb{F}_2$. Let $r \in \mathbb{F}_2^n$ be a received word and suppose that in r occur $e > t$ errors, with $e = t + s$ and e-th extended syndrome matrix $\widetilde{S}_e \in M(2t, t+s+1, \mathbb{F}_2^m)$. Let $\widetilde{\mathcal{F}} = \{f_0, \dots, f_s\}$ be an escalier basis for $\ker(\widetilde{S}_e)$ and let $\widetilde{\Sigma}(r, e)$ be the extended error-locator set of degree e for r. Then, for every $\sigma \in \widetilde{\Sigma}(r, e)$, $\widetilde{\mathcal{D}}_{r,e}(\sigma) \in C$.*

Proof. It follows by Fundamental Theorem for BCH list decoding. In fact the extended syndrome matrix $\widetilde{S}_e$ contains the conditions that $\sigma'(z) = \omega(z)$, and therefore that

$$\frac{\omega(a_i^{-1})}{\sigma'(a_i^{-1})} \in \mathbb{F}_2 \qquad \forall\, i = 1, \dots, e,$$

□

3.6 Decoding up to a certain error-threshold

Let C be a primitive BCH code of length $q^m - 1$ and designed distance $\delta = 2t + 1$ over $\mathbb{F}_q$, and let r be a received word.

Everything we have discussed before is true with the assumption that we have *exactly* e errors in r. Now suppose that we don't know how many errors occur in r. We would like to correct up to κ errors and we just suppose that $t < \kappa < 2t$.

Definition 3.6.1. We call such integer κ the *error-threshold*.

Definition 3.6.2. We define, for $k = 1, \dots, n$, the set

$$T_k := \{p(z) \in \mathbb{F}_q[z] \mid p(z) | z^n - 1, \deg p = k\}$$

the set of all polynomials of a fixed degree that divide $z^{q-1} - 1$, and

$$T := \bigcup_{k=1}^{n} T_k = \{p(z) \in \mathbb{F}_q[z] \mid p(z) | z^n - 1, \}\,.$$

We would like to understand what happens when we try to correct up to κ errors. In particular we want to understand if in $\Sigma(r, \kappa)$ we find the error-locator polynomials of *all* the codewords with distance *at most* κ from r, and not only the codewords with distance *exactly* κ.

Definition 3.6.3. Given an error threshold κ, we denote by

$$L(r, \kappa) = \{c \in C \mid d(c, r) \leq \kappa\} = B(r, \kappa) \cap C$$

the set of all codewords $c \in C$ with distance at most κ from r.

We first prove the following important result.

Lemma 3.6.4. *Let $\sigma(z) \in \mathbb{F}_{q^m}[z]_{\leq \kappa}$ be a polynomial. Then*

$$\sigma \in \ker(S_\kappa) \iff \sigma(z)S(z) \equiv \omega(z) \mod (z^{2t}), \text{ with } \deg \omega < \kappa$$

Proof. Let $\omega(z) \in \mathbb{F}_{q^m}[z]_{<2t}$ be the polynomial such that

$$\sigma(z)S(z) \equiv \omega(z) \mod (z^{2t}).$$

Then $\deg \omega < \kappa$ if and only if $\omega_\ell = 0$ for $\ell = \kappa, \ldots, 2t-1$. But

$$\omega_{\kappa+j-1} = \sum_{i=0}^{\kappa} \sigma_{\kappa-i} S_{i+j}.$$

Hence $\deg \omega < \kappa$ if and only if σ satisfies the linear system $S_\kappa \sigma = 0$, i.e. $\sigma \in \ker(S_\kappa)$.

□

Now we prove that every codeword $c \in L(r, \kappa)$ has its error-locator polynomial in $\ker(S_\kappa)$.

Theorem 3.6.5. *Let $c \in C$ be a codeword such that $d(c, r) = e \leq \kappa$. Then there is a polynomial $\sigma_c \in \ker(S_\kappa) \cap T_e$ such that $\mathcal{D}_{r,\kappa}(\sigma_c) = c$.*

Proof. Since $d(c, r) = e$, there exist $a_1, \ldots, a_e \in \mathbb{F}_{q^m}^*$ such that

$$\sigma_c(z) = \prod_{i=1}^{e} (1 - a_i z)$$

is the error-locator polynomial, and

$$\omega_c(z) = \sum_{i=1}^{e} E_i a_i \prod_{j \neq i} (1 - a_j z)$$

is the error-evaluator polynomial. The polynomials σ_c and ω_c satisfy the key equation

$$\sigma_c(z)S(z) \equiv \omega_c(z) \mod (z^{2t}),$$

so, by Lemma 3.6.4, $\sigma_c \in \ker(S_\kappa)$. Moreover by definition $\sigma_c \in T_e$ and decoding r by σ_c we obtain c. □

Theorem 3.6.6. *Let $c \in C$ be a codeword such that $d(c, r) = e \leq \kappa$. Then there exist exactly $\binom{n-e}{\kappa-e}$ polynomials σ_i for $i = 1, \ldots, \binom{n-e}{\kappa-e}$ such that $\sigma_i \in \ker(S_\kappa) \cap T_\kappa$ and $\mathcal{D}_{r,\kappa}(\sigma_i) = c$.*

Moreover the subspace $W \subset \mathbb{F}_{q^m}[z]_{\leq \kappa}$ generated by the σ_is is equal to $\langle \sigma_c, z\sigma_c, \ldots z^{\kappa-e}\sigma_c \rangle$. and it contains all the polynomials $\sigma \in T$ such that $\mathcal{D}_{r,\kappa}(\sigma) = c$.

Proof. Let $\tilde{\sigma}$ as in the previous theorem, i.e. such that $\deg \tilde{\sigma} = e$ and it is the error-locator polynomial of r. Put $\ell := \kappa - e$, and let be $a_1, \ldots, a_e$ the positions of the errors and $E_1, \ldots, E_e$ the corresponding errors. Then the polynomial

$$\tilde{\omega}(z) := \sum_{i=1}^{e} E_i a_i \prod_{j \neq i} (1 - a_j z)$$

is such that

$$\tilde{\omega}(z) \equiv S(z)\tilde{\sigma}(z) \qquad \mod (z^{2t}), \tag{3.5}$$

and $\deg \tilde{\omega} < e$.

Then we can choose $a_{e+1}, \ldots, a_{e+\ell} \in \mathbb{F}_{q^m}^* \setminus \{a_1, \ldots, a_e\}$ all distinct, and we put

$$\bar{\sigma}(z) := \tilde{\sigma}(z) \prod_{i=e+1}^{e+\ell} (1 - a_i z).$$

Moreover we put $E_i := 0$ for $i = e+1, \ldots, e+\ell$. Then, by (3.5), multiplying both sides by $\prod_{i=e+1}^{e+\ell}(1 - a_i z)$, we obtain

$$\tilde{\omega}(z) \prod_{i=e+1}^{e+\ell} (1 - a_i z) \equiv S(z)\bar{\sigma}(z)$$

But

$$\begin{aligned}
\tilde{\omega}(z) \prod_{i=e+1}^{e+\ell} (1 - a_i z) &= \sum_{i=1}^{e} E_i a_i \prod_{j \neq i} (1 - a_j z) \prod_{i=e+1}^{e+\ell} (1 - a_i z) = \\
&= \sum_{i=1}^{e} E_i a_i \prod_{\substack{j=1,\ldots,\kappa \\ j \neq i}} (1 - a_j z) = \\
&\overset{(*)}{=} \sum_{i=1}^{\kappa} E_i a_i \prod_{\substack{j=1,\ldots,\kappa \\ j \neq i}} (1 - a_j z) \\
&= \bar{\omega}(z)
\end{aligned}$$

where the equality $(*)$ holds because

$$\sum_{i=e+1}^{\kappa} E_i a_i \prod_{\substack{j=1,\dots\kappa \\ j\neq i}} (1 - a_j z) = 0.$$

The polynomials $(\bar{\sigma}, \bar{\omega})$ satisfy the key equation, with $\deg \bar{\sigma} = \kappa$ and $\deg \bar{\omega} < \kappa$, so $\bar{\sigma} \in \ker(S_\kappa)$.

Using Forney's equations we obtain the error vector with entries E_i, $i = 1, \dots, \kappa$ in the positions corresponding to $a_1, \dots, a_\kappa$, and hence $\mathcal{D}_{r,\kappa}(\bar{\sigma}) = c$. □

Remark 3.6.7. This theorem is a kind of inverse of the *Fundamental Theorem for RS List Decoding*. It ensures that in the set $\Sigma(r, \kappa)$ we find the error-locator polynomials of all the codewords in the set $L(r, \kappa)$. In particular, if we fix the degree of the error-locator polynomial, we are not fixing the number of errors that can occur, but only the maximum of such errors.

Formally, recall that the *decoding function* is defined as the function

$$\mathcal{D}_{r,\kappa} : \Sigma(r, \kappa) \longrightarrow \mathbb{F}_q^n$$

with $\mathcal{D}_{r,\kappa}(\sigma) = c$, where c is the vector obtained decoding r by σ. The *Fundamental Theorem for RS List Decoding* states that c is a codeword and the image of $\mathcal{D}_{r,\kappa}$ is contained in $L(r, \kappa)$. Theorem (3.6.6) states that such a function is surjective onto $L(r, \kappa)$, i.e.

$$\mathcal{D}_{r,\kappa} : \Sigma(r, \kappa) \twoheadrightarrow L(r, \kappa).$$

Remark 3.6.8. For binary BCH codes we can similarly prove that in $\ker(\widetilde{S}_e)$ we can find the error-locator polynomials of all the codewords with distance *exactly* κ. Hence the extended decoding function

$$\widetilde{\mathcal{D}}_{r,\kappa} : \widetilde{\Sigma}(r, \kappa) \twoheadrightarrow S(r, \kappa)$$

is surjective onto the set

$$S(r, \kappa) = \{c \in C \mid d(c, r) = \kappa\}.$$

Observe that in this situation we don't know how many errors occur in r so we don't know a priori the rank of the syndrome matrix S_κ. In fact, if r is itself a codeword, the κ-th syndrome matrix S_κ is the zero matrix.

Definition 3.6.9. We denote by

$$e := \min\{d(c, r) \mid c \in C\}$$

the minimum weight of an error pattern.

The following proposition tells us something about the rank of the syndrome matrix.

Proposition 3.6.10. *Let r be a received word, and let e be the minimum weight of an error pattern. Given an error threshold $\kappa = t+s$ with $0 < s < t$. Then*

$$rk(S_\kappa) = \min\{t-s, e\}.$$

Proof. Let e be the minimum weight of an error pattern. Hence

$$s_i = r(\beta^i) = \varepsilon(\beta^i) = \sum_{j=1}^{e} E_j a_j^i$$

We can rearrange the a_i such that $E_i \neq 0$ for $i = 1, \ldots, e$. As in the proof of Proposition 3.3.1 we take the matrices

$$\widetilde{V} = \begin{bmatrix} 1 & a_1 & \cdots & a_1^{t-s-1} \\ 1 & a_2 & \cdots & a_2^{t-s-1} \\ \vdots & & \ddots & \vdots \\ 1 & a_{t-s} & \cdots & a_{t-s}^{t-s-1} \end{bmatrix}, \quad \widetilde{D} = \begin{bmatrix} E_1 a_1 & & & 0 \\ & E_2 a_2 & & \\ & & \ddots & \\ 0 & & & E_{t-s} a_{t-s} \end{bmatrix},$$

and we observe that the principal minor S'_κ of S_κ with dimension $(t-s)\times(t-s)$ satisfies

$$S'_e = \widetilde{V}^T \widetilde{D} \widetilde{V}.$$

The matrix $\widetilde{V}$ is invertible, so it has maximum rank. Since the rank of the matrix $\widetilde{D}$ is exactly $\min\{t-s, e\}$ we have that $rk(S'_\kappa) = \min\{t-s, e\}$, and hence $rk(S_\kappa) \geq \min\{t-s, e\}$.

Now we show the other inequality, i.e. $rk(S_\kappa) \leq \min\{t-s, e\}$. Since $S_\kappa \in M(t-s, t+s+1, \mathbb{F}_{q^m})$ we have trivially that $rk(S_\kappa) \leq t-s$. Moreover, if $\bar{c}$ is a codeword with distance e from r with error-locator polynomial $\sigma_{\bar{c}}$, by Theorem 3.6.6 we have that the subspace $\langle \sigma_{\bar{c}}, z\sigma_{\bar{c}}, \ldots z^{\kappa-e}\sigma_{\bar{c}} \rangle$ is contained in $\ker(S_\kappa)$. So $\dim \ker(S_\kappa) \geq \kappa - e + 1 = t + s - e + 1$. Therefore $rk(S_\kappa) \leq e$. □

Chapter 4

Bounds on list decoding

In this chapter, we state some combinatorial result concerning list decoding. In particular we give some bounds on the size of list decoding for a general code first, and then for the special cases of BCH and Reed-Solomon codes. This is because in order to perform list decoding up to a certain error treshold κ efficiently, we need the guarantee that every ball of radius κ has a "small"number of codewords. The motivation is that the list decoding algorithm will have a runtime that obviously is at least the size of the list of codewords it outputs, and we want the algorithm to be efficient even in the worst case.

For a code C of length n and minimum distance d any Hamming ball of radius less than $\frac{d}{2}$ can have at most one codeword, and we have unique decoding. For list decoding we would like to have some "small "upper bounds on the number of codewords in a ball of radius κ for κ greater than $\frac{d}{2}$.

We first state some results about bounds for a general code C over an alphabet $\mathcal{Q}$ of q elements. One of this is based on a classical bound in coding theory, called the Johnson bound [8]. Then we focus on BCH and Reed-Solomon codes and we use their code structure to develope some other bounds. For these kinds of codes we give different bounds by studing the problem from different points of view. One is based on the study of the cardinality of a variety in a polynomial ring over $\mathbb{F}_q$, as we have seen in the second Chapter. Another one is derived from a purely combinatorial problem, that we observed is a general case of our problem: the uniform packing set. The last one, proved by Madhu Sudan in [11] and in [5] is more particular, and it works only for Reed-Solomon codes with rate $\simeq \frac{1}{3}$ or less.

4.1 General bounds

For a code C of length n over an alphabet $\mathcal{Q}$ of q elements, the classical bound on the number of codeword in an Hamming ball is the Johnson Bound. There are many different proofs of the Johnson bound in literature. The original proof and some of its derivatives follow a linear algebra based argument [8], while more recent proofsare more geometric.

In the following we work with a code C of length n over the alphabet $[q] = \{1, \ldots, q\}$. We use

$$B_q(r, e) = \{x \in [q]^n \mid d(r, x) \leq e\}$$

to denote the Hamming ball of radius e, where e is a positive integer. Most of the results and the proofs of this section can be found in [6].

Definition 4.1.1. Let A be a finite set, and let j be a positive integer. We denote by

$$\binom{A}{j} := \{B \subseteq A \mid |B| = j\}$$

the set of all subsets of A with cardinality j.

Theorem 4.1.2. *(Johnson Bound) Suppose $r \in [q]^n$, and $B \subseteq [q]^n$. Let*

$$d = \mathop{\mathbb{E}}_{\{x,y\} \in \binom{B}{2}} [d(x, y)],$$
$$e = \mathop{\mathbb{E}}_{x \in B} [d(r, x)].$$

Then

$$|B| \leq \frac{\frac{q}{q-1} \cdot \frac{d}{n}}{\left(1 - \frac{q}{q-1} \cdot \frac{e}{n}\right)^2 - \left(1 - \frac{q}{q-1} \cdot \frac{d}{n}\right)},$$

provided the denominator is positive.

Corollary 4.1.3. *Let C be any code of length n and minimum distance d over the alphabet $[q]$, with*

$$d = \left(1 - \frac{1}{q}\right)(1 - \delta)\, n$$

for some $\delta \in (0, 1)$. Let

$$e = \left(1 - \frac{1}{q}\right)(1 - \gamma)\, n$$

for some $\gamma \in (0,1)$ be an integer. Suppse that $\gamma^2 > \delta$. Then, for all $r \in [q]^n$,

$$|B_q(r,e) \cap C| \leq \frac{1-\delta}{\gamma^2 - \delta}.$$

Proof. Let $B = B_q(r,e) \cap C$. Let

$$\mathop{\mathbb{E}}_{\{x,y\} \in \binom{B}{2}}[d(x,y)] = \left(1 - \frac{1}{q}\right)(1-\delta')\,n,$$
$$\mathop{\mathbb{E}}_{x \in B}[d(r,x)] = \left(1 - \frac{1}{q}\right)(1-\gamma')\,n.$$

Then we have $\delta' \leq \delta < \gamma^2 \leq \gamma'^2$, and by Theorem 4.1.2,

$$|B_q(r,e) \cap C| \leq \frac{1-\delta'}{\gamma'^2 - \delta'} = 1 + \frac{1-\gamma'^2}{\gamma'^2 - \delta'} \leq 1 + \frac{1-\gamma^2}{\gamma^2 - \delta} = \frac{1-\delta}{\gamma^2 - \delta}.$$

□

We can also state an alphabet independent version of the Johnson Bound, that is valide for large alphabet.

Corollary 4.1.4. *Let C be a code of length n and minimum distance d over the alphabet $[q]$. Suppose $r \in [q]^n$ and*

$$(n-e)^2 > n(n-d).$$

Then

$$|B_q(r,e) \cap C| \leq \frac{nd}{(n-e)^2 - n(n-d)}.$$

In particular, $|B_q(r,e) \cap C| \leq n^2$.

Proof. The denominator in the upper bound in Theorem 4.1.2 equals

$$\frac{q}{q-1}\left(\frac{q}{q-1}\frac{e^2}{n^2} - \frac{2e}{n} + \frac{d}{n}\right) \geq \frac{q}{q-1}\left(\left(1 - \frac{e}{n}\right)^2 - \left(1 - \frac{d}{n}\right)\right).$$

Therefore it follows that

$$|B_q(r,e) \cap C| \leq \frac{nd}{(n-e)^2 - n(n-d)}.$$

□

One may wonder whether the Johnson Bound is tight, or whether it may be possible to improve it and show that for every code with fixed distance and length, Hamming balls of radius greater than the Johnson radius still have polinomially many codewords, where the *Johnson radius* is defined to be

$$e_J(n,d,q) := \left(1-\frac{1}{q}\right)\left(1-\sqrt{1-\frac{q}{q-1}\cdot\frac{d}{n}}\right)n.$$

It turns out that the Johnson Bound is the best possible, that is there exist codes which have super-polinomially many codewords in an Hamming ball of radius slightly bigger than the Johnson radius. However, for most codes with some additional structure, the Johnson Bound is not tight.

4.2 Madhu Sudan bound

With the approach used by Madhu Sudan that we will describe in the next chapter, we can give an upper bound on the cardinality of the list of codewords that have distance not greater than $n-\ell$ from a received word r. This bound is based on Madhu Sudan work [11], and it works only for Reed-Solomon codes, because it takes advantage of the structure of this kind of codes given in Theorem 1.6.3. So, as usual suppose we have an $[n, k+1, n-k]$ Reed Solomon code over $\mathbb{F}_q$, where $n-k = 2t+1$ is the minimum distance. The result, proved by Madhu Sudan in [11] and in [5], is the following.

Lemma 4.2.1. *Let r be a received word, and α be a primitive element of $\mathbb{F}_q$. If*

$$\frac{\ell}{n} \geq \left(\sqrt{2+\frac{k}{4n}}\cdot\sqrt{\frac{k}{n}}\right)-\frac{k}{2n}, \tag{4.1}$$

then the number of polynomials f of degree at most k such that $|\{i \mid f(\alpha^i) = r_i\}| \geq \ell$ is at most

$$\left\lfloor \frac{\ell}{k}+\frac{1}{2}-\sqrt{\left(\frac{\ell}{k}+\frac{1}{2}\right)^2-\frac{2n}{k}}\right\rfloor \leq \frac{2n}{\ell+\frac{k}{2}}$$

This is a good bound for list decoding of a Reed-Solomon code, but it works only with the condition 4.1, while the bounds given in the previous section work without any condition on the error-treshold, and they work for every kind of code, in particular also for BCH codes.

The following proposition gives us precise conditions on the rate of the code in order to make the bound of Lemma 4.2.1 work.

Proposition 4.2.2. *If*

$$k > \frac{2n+2-\sqrt{n^2+14n+1}}{3},$$

then condition 4.1 does not hold for more than the error-correcting radius,

Proof. Observe that for all ℓ such that $n - \ell \geq t + 1$, we have

$$\frac{\ell}{n} \leq \frac{n-t-1}{n}.$$

Since $t = \dfrac{n-k-1}{2}$, we can rewrite it as

$$\frac{\ell}{n} \leq \frac{n+k-1}{n}.$$

Now an easy calculation shows that the system of inequalities

$$\begin{cases} n \geq k+1 \geq 1 \\ \dfrac{n+k-1}{n} \geq \left(\sqrt{2+\dfrac{k}{4n}} \cdot \sqrt{\dfrac{k}{n}}\right) - \dfrac{k}{2n} \end{cases} \tag{4.2}$$

is equivalent to the system

$$\begin{cases} n \geq 1 \\ 0 \leq k \leq \dfrac{2n+2-\sqrt{n^2+14n+1}}{3}. \end{cases} \tag{4.3}$$

By hypothesis the system 4.3 is not satisfied, then also the system 4.2 is not satisfied.

Hence for all ℓ such that $n - \ell \geq t + 1$, we have

$$\frac{\ell}{n} \leq \frac{n-t-1}{n} < \left(\sqrt{2+\frac{k}{4n}} \cdot \sqrt{\frac{k}{n}}\right) - \frac{k}{2n},$$

i.e. the condition 4.1 does not hold. □

Remark 4.2.3. This proposition gives us a strong condition on the rate such that the bound given in 4.2.1 holds. In fact

$$\frac{k+1}{n} \leq \frac{2n+2-\sqrt{n^2+14n+1}+3}{3n} \underset{n\to+\infty}{\longrightarrow} \frac{1}{3}.$$

This shows that Madhu Sudan bound does not work for Reed Solomon codes with rate greater than $\frac{1}{3}$.

4.3 Reduction to a packing set problem

As we have seen at the end of the previous section, the bound given in Lemma 4.2.1 works only for low-rate codes. In order to find other bounds that works also for high-rate codes, we try to turn our problem into a different combinatoric problem. First we prove the following lemma

Lemma 4.3.1. *Let C be a BCH code of length $n = q^m - 1$ over $\mathbb{F}_q$ with designed distance $\delta = 2t + 1$, and let r be a received word. Let $e = t + s$ be the error-correcting threshold and suppose we have*

$$\sigma_{c_1}, \sigma_{c_2} \in \Sigma(r, e).$$

If $c_1 \neq c_2$ then $\deg \gcd\{\sigma_{c_1}, \sigma_{c_2}\} \leq 2s - 1$.

Proof. Let e_{c_i} be the degree of $\sigma_{c_i}(z)$ for $i = 1, 2$, and let $p(z) := \gcd(\sigma_{c_1}, \sigma_{c_2})$, with $\deg p = l$. Then, for $h = 1, \ldots, l$ there exist distinct $j_h \in \{1, \ldots, q^{m-1}\}$, such that

$$p(z) = \prod_{h=1}^{l} (1 - \beta^{j_h} z).$$

So $\sigma_{c_i}(z) = p(z) g_i(z)$ for $i = 1, 2$, for some polynomial $g_i(z) \in \mathbb{F}_{q^m}[z]$ such that

$$g_i(z) = \prod_{h=l+1}^{e_{c_1}} (1 - \beta^{j_h^{(i)}} z).$$

Hence c_1 and r differ in the positions $j_1, j_2, \ldots, j_l, j_{l+1}^{(1)}, \ldots, j_{e_{c_1}}^{(1)}$. Similarly, c_2 and r differ in the positions $j_1, j_2, \ldots, j_l, j_{l+1}^{(2)}, \ldots, j_{e_{c_2}}^{(2)}$. Therefore c_1 and c_2 can differ only in the positions $j_1, j_2, \ldots, j_l, j_{l+1}^{(1)}, \ldots, j_{e_{c_1}}^{(1)}, j_{l+1}^{(2)}, \ldots, j_{e_{c_2}}^{(2)}$, which are $e_{c_1} + e_{c_2} - \deg \gcd(\sigma_{c_1}, \sigma_{c_2})$. Since the minimum distance of the code C is $2t + 1$ we have

$$2t + 1 \leq d(c_1, c_2) \leq e_{c_1} + e_{c_2} - \deg p \leq e + e - l = 2t + 2s - l.$$

Hence it must be $l \leq 2s - 1$. □

Remark 4.3.2. Recall that the set

$$L(r, e) = \{c \in C \mid d(c, r) \leq e\}$$

is defined to be the set of all codewords that have distance not greater than e from r.
By Theorem 3.6.6 for every codeword $c \in L(r, e)$ there exist a polynomial $\tau \in \Sigma(r, e)$ such that decoding r by τ we obtain the codeword c.

So for every codeword $c \in L(r,e)$ we can fix a polynomial $\tau_c \in \Sigma(r,e)$ such that decoding r by τ we obtain c, and we define the maps

$$\begin{array}{rcl} i : L(r,e) & \longrightarrow & \Sigma(r,e) \\ c & \longmapsto & \tau_c \end{array} \tag{4.4}$$

that is obviously injective.

Now, given a polynomial in $\Sigma(r,e)$ we can identify it by its set of roots. Formally we define the map

$$\begin{array}{rcl} \nu : \Sigma(r,e) & \longrightarrow & \binom{\mathbb{F}_{q^m}^*}{e} \\ \tau & \longmapsto & A_\tau \end{array} \tag{4.5}$$

where

$$A_\tau := \left\{ \alpha \in \mathbb{F}_{q^m}^* \mid \tau(\alpha) = 0 \right\}$$

is the set of all roots of the polynomial τ.

Now we introduce a new combinatoric problem, the uniform packing problem.

Definition 4.3.3. Given a finite set A of n elements, and two positive integers $0 < r \leq j \leq n$, a $j-(A,r)$ *uniform packing* is a collection $\mathcal{A} \subset \binom{A}{j}$ such that for each r-subset $T \in \binom{A}{r}$ there exist at most one $B \in \mathcal{A}$ such that $T \subset B$.

The following result permits to turn the problem of bounding the number of codewords $c \in C$ with distance at most e from a received word r into the problem of finding the maximal cardinality of a particular uniform packing of the set $\mathbb{F}_{q^m}^*$.

Theorem 4.3.4. *Let C be a BCH code of length $n = q^m - 1$ over $\mathbb{F}_q$ with designed distance $\delta = 2t+1$, and let r be a received word. Let $e = t+s$ be the error-correcting threshold and let i and ν be the two maps defined in (4.4) and (4.5). Then*

$$\nu \circ i \left(L(r,e) \right)$$

is a $e-(\mathbb{F}_{q^m}^*, 2s)$ uniform packing.

Proof. Let c_1 and c_2 two distinct elements in $L(r,e)$ and put $\tau_1 := i(c_1)$, $\tau_2 := i(c_2)$. By Lemma 4.3.1 we have

$$\deg \gcd\{\tau_1, \tau_2\} \leq 2s - 1.$$

This implies that they have at most $2s-1$ common roots. So every $2s$-subset of $\mathbb{F}_{q^m}^*$ is contained in at most one A_τ for $\tau \in i(L(r,e))$. Hence

$$\nu \circ i(L(r,e))$$

is a $e-(\mathbb{F}_{q^m}^*, 2s)$ uniform packing. □

So, by this theorem, a bound for the number of polynomials can be obtained by finding an upper bound on the maximal cardinality of a e-uniform packing for $\binom{\mathbb{F}_{q^m}^*}{2s}$. Formally,

Definition 4.3.5. Let $0 < r \leq j \leq n$ be three positive integers. We denote by $D(n,j,r)$ the maximum cardinality of a $j-([n],r)$ uniform packing, where $[n] = \{1,\ldots,n\}$.

Corollary 4.3.6. *Let C be a BCH code of length $n = q^m - 1$ over $\mathbb{F}_q$ with designed distance $\delta = 2t+1$, and let r be a received word. Let $e = t+s$ be the error-correcting threshold. Then*

$$|L(r,e)| \leq D(q^m - 1, e, 2s)$$

□

So we can try to find an upper bound for $D(n,j,r)$ and use it for bounding the cardinality of $L(r,e)$.
There are two well-known bounds on the cardinality of a $j-([n],r)$ uniform packing.

Theorem 4.3.7 (First Johnson Bound for set packing [8])**.** *Every $j-([n],r)$ uniform packing can not have more than $U(n,j,r)$ elements, where $U(n,j,r)$ is the* floor function *defined by*

$$U(n,j,r) := \left\lfloor \frac{n}{j} \left\lfloor \frac{n-1}{j-1} \left\lfloor \cdots \left\lfloor \frac{n-r+1}{j-r+1} \right\rfloor \cdots \right\rfloor \right\rfloor \right\rfloor.$$

□

The First Johnson Bound is a good bound when n is big. But when n is not so big it is less effective. In this case we have another bound

Theorem 4.3.8 (Second Johnson Bound for set packing [8])**.** *If $j^2 > (r-1)n$ then*

$$D(n,j,r) \leq \left\lfloor \frac{n(j-r+1)}{j^2-(r-1)n} \right\rfloor. \tag{4.6}$$

□

4.4 Bounds for some special cases

We want to understand how the bounds given till now work when we try to correct just one more error.

First we analize what Lemma 4.2.1 says when we try to find all the codewords that have distance not greater than $t+1$, where t is the error-correcting radius of C.

So, with the notations of Lemma 4.2.1, $n-\ell = t+1$, and $n-k = d = 2t+1$. From this two equations we can find t and ℓ in function of n and k, i.e.

$$\begin{cases} t = \frac{n-k-1}{2} \\ \ell = \frac{n+k-1}{2}. \end{cases} \tag{4.7}$$

Theorem 4.4.1. *Let C be an $[n, k+1, n-k]$ Reed Solomon code over $\mathbb{F}_q$ that can correct up to t errors and let r be a received word. If*

$$k \leq \frac{2n+2-\sqrt{n^2+14n+1}}{3}$$

then the number of codewords with distance at most $t+1$ from r is at most 3.
If moreover

$$k > \frac{n+3}{6}$$

then the number of such codewords is at most 2.

Proof. We need just to apply Lemma 4.2.1 when $n-\ell = t+1$. Substituting the equations 4.7 in 4.1 we obtain

$$\frac{n+k-1}{2n} \geq \left(\sqrt{2+\frac{k}{4n}} \cdot \sqrt{\frac{k}{n}}\right) - \frac{k}{2n}$$

As we have seen in the proof of Proposition 4.2.2, the system

$$\begin{cases} n \geq k+1 \geq 1 \\ \frac{n+k-1}{n} \geq \left(\sqrt{2+\frac{k}{4n}} \cdot \sqrt{\frac{k}{n}}\right) - \frac{k}{2n} \end{cases} \tag{4.8}$$

is equivalent to the system

$$\begin{cases} n \geq 1 \\ 0 \leq k \leq \frac{2n+2-\sqrt{n^2+14n+1}}{3}. \end{cases} \tag{4.9}$$

So, in this case Lemma 4.2.1 states that there exist at most

$$\frac{2n}{\ell + \frac{k}{2}} = \frac{4n}{n + 2k - 1} < 4$$

codewords with distance at most $t + 1$ from r.

If moreover

$$k > \frac{n+3}{6}$$

then

$$\frac{4n}{n + 2k - 1} < 3$$

and we conclude. □

Remark 4.4.2. Observe that the theorem above is true only for low-rate codes. In fact

$$\frac{k}{n} \leq \frac{2n + 2 - \sqrt{n^2 + 14n + 1}}{3n} \underset{n \to +\infty}{\longrightarrow} \frac{1}{3}.$$

Now consider the more general case in which we are working with a BCH code C, and we are trying to decoding up to $t + 1$ errors, where t is the error-correcting radius. We are going to apply the two Johnson Bounds to give new bounds on the cardinality of $L(r, t + 1)$.

Corollary 4.4.3. *Let C be a BCH code of length n and designed distance $\delta = 2t + 1$ over $\mathbb{F}_q$. Then, for every possible received word $r \in \mathbb{F}_q^n$ we have*

$$|L(r, t + 1)| \leq \left\lfloor \frac{n}{t+1} \left\lfloor \frac{n-1}{t} \right\rfloor \right\rfloor.$$

□

Corollary 4.4.4. *Let C be a BCH code of length n and designed distance $\delta = 2t + 1$ over $\mathbb{F}_q$. Then, for every possible received word $r \in \mathbb{F}_q^n$ we have*

$$|L(r, t + 1)| \leq \left\lfloor \frac{tn}{(t+1)^2 - n} \right\rfloor,$$

provided that $(t + 1)^2 > n$.

□

If we are in the situation that $(t + 1)^2 > n$, it can be shown that the Second Johnson Bound is almost always a stronger result than the First. Now we prove a corollary of the Second Johnson Bound. We will see later that it permits to show the existence of BCH code of arbitrary rate with a "small"number of solution to the list decoding problem.

Corollary 4.4.5. *If* $n < \frac{(t+1)(t+2)}{2}$ *then*

$$|L(r, t+1)| \leq t$$

Proof. It trivially follows by substituting n by $\frac{(t+1)(t+2)}{2} - 1$ in the Second Johnson Bound. □

Remark 4.4.6. If C is a Reed Solomon code, according to 4.7, the condition

$$n < \frac{(t+1)(t+2)}{2}$$

can be rewritten as

$$(n-k+1)(n-k+3) > 8n,$$

from which we obtain the quadratic inequality in k given by

$$k^2 - (2n+4)k + (n^2 - 4n + 3) > 0.$$

Now It is an easy calculation showing that the system

$$\begin{cases} n \geq k+1 \geq 1 \\ k^2 - (2n+4)k + (n^2 - 4n + 3) > 0 \end{cases} \tag{4.10}$$

is equivalent to the system

$$\begin{cases} n \geq 1 \\ 0 \leq k < n + 2 - \sqrt{8n+1}. \end{cases} \tag{4.11}$$

Hence, this bound is true also for some high rate code. In fact

$$\frac{k}{n} \leq \frac{n + 2 - \sqrt{8n+1}}{n} \underset{n \to +\infty}{\longrightarrow} 1$$

From this remark we can observe that for every rate $R \in (0,1)$, there exist a BCH code C with rate R such that the size of the list decoding problem for one more error than the error-correcting radius is at most the error correcting radius. Formally we can state the following result.

Theorem 4.4.7. *For every* $\varepsilon > 0$ *there exist a BCH code* C *of rate* $R > 1 - \varepsilon$ *and length* n_ε *such that for every* $r \in \mathbb{F}_{q^m}^{n_\varepsilon}$

$$\left| L\left(r, \frac{1-R}{2} n_\varepsilon + 1\right) \right| \leq \frac{1-R}{2} n_\varepsilon < \frac{\varepsilon}{2} n_\varepsilon.$$

Chapter 5

Computational aspects of list decoding

The Euclidean division approach proposed in the previous chapter has the advantage of working for every Reed Solomon code and for every binary BCH code. However in general the computational cost is very high because it involves the use of Grobner Basis.

In this chapter we analize some computational aspects of the algorithm for list decoding presented in Chapter 3. Moreover, we show that for some special cases we can avoid the use of Grobner basis, reducing significantly the cost of our approach.

At the end of the chapter we describe the most important algorithms for list decoding of Reed-Solomon codes. The first one has been developed in 1995 by Madhu Sudan [11], and the second one is an improvement of such an algorithm elaborated by Guruswami [7].

5.1 Calculation of the divisibility ideal

To seek irreducible polynomials or to factor reducible ones, it is useful to have a criteria for the divisibility over a field $\mathbb{F}$ of a polynomial $f(x)$ by a polynomial $g(x)$ of degree less than that of $f(x)$. In particular, as we have seen in the previous chapters, we are interested in such a criteria in the case $\mathbb{F} = \mathbb{F}_q(\underline{a})$, where q is a power of a prime p, and $\mathbb{F}_q(\underline{a})$ denotes the fraction field of the polynomial ring $\mathbb{F}_q[\underline{a}] := \mathbb{F}_q[a_1, \ldots, a_k]$.

Let $f(x) \in \mathbb{F}[x]$ and $g(x) \in \mathbb{F}[x]$ be two polynomials of arbitrary degree n and e ($e < n$), respectively,

$$\begin{aligned} f(x) &= \sum_{i=0}^{n} a_i x^i, \qquad a_n \neq 0, a_i \in \mathbb{F}, \\ g(x) &= x^e - \sum_{i=0}^{e-1} b_i x^i \qquad b_i \in \mathbb{F}. \end{aligned} \tag{5.1}$$

Obviously, the polynomial $f(x)$ is divisible by $g(x)$ if and only if the remainder of the division of $f(x)$ by $g(x)$ is identically zero.

In order to determine the criteria we are looking for, we introduce a definition.

Definition 5.1.1. We call *divisibility sequence* of the two polynomials $f(x)$, $g(x) \in \mathbb{F}[x]$ defined as in (5.1), the sequence defined recursively as follows

$$T_{h+1} = b_{e-1}T_h + b_{e-2}T_{h-1} + \ldots + b_0 T_{h-e+1}, \tag{5.2}$$

with initial conditions

$$T_0 = 1, \qquad T_h = 0 \quad \text{for } -e+1 \leq h \leq -1.$$

or, equivalently,

$$\begin{cases} T_0 = 1, \\ T_1 = b_{e-1}T_0, \\ T_2 = b_{e-1}T_1 + b_{e-2}T_0, \\ \vdots \\ T_{e-1} = b_{e-1}T_{e-2} + b_{e-2}T_{e-3} + \ldots + b_1 T_0. \end{cases}$$

Proposition 5.1.2 ([1])**.** *The polynomial $f(x)$ is divisible by the polynomial $g(x)$ if and only if their divisibility sequence satisfies the relations*

$$\sum_{h=e-j-1}^{n} a_h T_{h-e+j+1} = 0 \quad \textit{for } j = 0, 1, \ldots e-1. \tag{5.3}$$

□

From this result, choosing $f(x) = x^n - 1$, we easily obtain the following result.

Corollary 5.1.3. *$x^n - 1$ is divisible by $g(x)$ if and only if*

$$\begin{cases} T_{n-e+1+j} = 0 \quad \textit{for } j = 0, 1, \ldots, e-2, \\ T_n - 1 = 0. \end{cases} \tag{5.4}$$

□

We come back now to our problem. We would like to compute the ideal $\mathcal{I}_{\mathcal{F},H}$, where $H(x) = x^{q-1} - 1$. In fact this problem, as seen in the prevoius chapter, is strictly related to our approach to the list decoding problem.

Suppose we have an escalier basis $\mathcal{F} = \{f_0, \ldots, f_k\}$ for a subspace W of degree e of $\mathbb{F}_q[x]_{\leq e}$. We would like to know if the polynomial

$$F(\underline{a}, x) = f_0(x) + \sum_{i=1}^{k} a_i f_i(x)$$

divides $x^{q-1} - 1$. We can write

$$F(\underline{a}, x) = x^e - \sum_{i=0}^{e-1} c_i(\underline{a}) x^i,$$

where $c_i(\underline{a})$ is a polynomial in the variables $\underline{a} = (a_1, \ldots, a_k)$ with $\deg_{\underline{a}} c_i \leq 1$.

Consider the polynomial sequence $(T_h(\underline{a}))_h$ with coefficients in $\mathbb{F}_q$ defined, as in (5.2), by

$$\begin{cases} T_{h+1}(\underline{a}) = c_{e-1}(\underline{a}) T_h(\underline{a}) + \ldots + c_0(\underline{a}) T_{h-e+1}(\underline{a}), \\ T_0(\underline{a}) = 1, \\ T_\ell(\underline{a}) = 0, \qquad \text{for } -e+1 \leq \ell \leq -1. \end{cases}$$

By Corollary 5.1.3, the polynomial $F(\underline{\alpha}, x)$ divides $x^{q-1} - 1$ if and only if

$$\begin{cases} T_{n-e+1+j}(\underline{\alpha}) = 0 \quad \text{for } j = 0, 1, \ldots, e-2, \\ T_n(\underline{\alpha}) - 1 = 0. \end{cases} \tag{5.5}$$

So, if we define the ideal

$$\mathcal{J} := \left(T_n(\underline{a}) - 1, T_{n-1}(\underline{a}), T_{n-2}(\underline{a}), \ldots, T_{n-e+1}(\underline{a})\right),$$

we obtain that

$$\mathcal{V}(\mathcal{J}) = \mathcal{V}(\mathcal{I}_{\mathcal{F},H}).$$

Moreover, with an argument similar to that used in [1], it is easy to prove that

$$\mathcal{J} = \mathcal{I}_{\mathcal{F},H}.$$

5.2 Computing the divisibility set

Let $\mathcal{I}_{\mathcal{F},H}$ the computed divisibility ideal. Now our goal is solving the system of multivariate polynomial equations given by

$$\begin{cases} r_0(a_1, \ldots, a_k) = 0, \\ r_1(a_1, \ldots, a_k) = 0, \\ \vdots \\ r_{e-1}(a_1, \ldots, a_k) = 0. \end{cases} \tag{5.6}$$

So our problem has turned into the problem of solving systems of multivariate polynomial equations.

In the general case this problem is NP-complete, even if all the equations are quadratic and the field is $\mathbb{F}_2$. The classical algorithm for solving such a system is Buchberger's algorithm for constructing Grobner bases, and its many variants. The algorithm orders the monomials, typically in lexicographic order, and eliminates the top monomial by combining two equations with appropriate polynomial coefficients. This process is repeated until all but one of the variables are eliminated, and then solves the remaining univariate polynomial equation. Unfortunately, the degrees of the remaining monomials increase rapidly during the elimination process, and thus the time complexity of the algorithm makes it often impractical even for a modest number of variables. In the worst case Buchberger's algorithm is known to run in double exponential time, and on average its running time seems to be single exponential.

However, in some special cases we can avoid the use of Grobner bases, and we can use other simple techniques for solving a system of polynomial equation. For example when the number of variables is 1 or 2,

5.3 Some special cases

5.3.1 Binary BCH codes with $e = t + 1$

We come back to the list decoding problem. we observe that, when we try to correct one more error than the error-correcting radius in a binary BCH code, i.e. when the number of errors is $e = t + 1$, by Theorem 3.5.7 we have that $\dim \ker \widetilde{S}_e = 2$. So, given an escalier basis $\widetilde{\mathcal{F}} = \{f_0, f_1\}$, the divisibility ideal $\mathcal{I}_{\widetilde{\mathcal{F}},x^{2^m-1}-1}$ is an ideal in the ring $\mathbb{F}_{2^m}[a]$. Hence we have to compute a system of univariate polynomial equations. Let $p(a) = \gcd(r_0(a), \ldots, r_{e-1}(a))$, then $\mathcal{I}_{\widetilde{\mathcal{F}},x^{2^m-1}-1} = (p(a))$, so the time complexity for solving such a problem is

equal to the running time of a gcd plus the time for solving a univariate polynomial equation.

In detail, let $T(n,d)$ denotes the running time of a gcd between n polynomials of degree at most d, then, observing that

$$\gcd\{p_1,\ldots,p_n\} = \gcd\{\gcd\{p_1,p_2\},p_3,\ldots,p_n\},$$

we have that

$$\begin{aligned} T(n,d) &= T(2,d) + T(n-1,d) = T(2,d) + T(2,d) + T(n-2,d) = \\ &= \ldots = (n-1)T(2,d) = O(nT(2,d)). \end{aligned}$$

Since the cost of a polynomial greatest common divisor between two polynomials of degree at most d can be taken as $O(d\log^2(d))$ operations in $\mathbb{F}_q$ using fast methods, we have $T(n,d) = O(nd\log^2(d))$. In our case $n = t+1$ and $d = q-t-1$. So $O((q-t-1)(t+1)\log^2(q-t-1))$ operations occurs for this step. Moreover, the polynomial obtained is a squarefree polynomial with all the roots in $\mathbb{F}_{2^m}^*$, so we can find the roots with an exhaustive search. This can be done in $O((q-1)V(t+1))$ operations, where $V(t+1)$ denotes the number of operations that the evaluation of a polynomial of degree $t+1$ needs. This can be done using $O((t+1)\log^{-1}(t+1))$ additions and $O(2\sqrt{3(t+1)})$ multiplications, since we are in a field of characteristic 2. In total we have

$$O((q-t-1)(t+1)\log^2(q-t-1)) + O((q-1)((t+1)\log^{-1}(t+1)))$$

field operations.

5.3.2 RS and binary BCH codes with $e = t+1$

As we have seen in Chapter 3, when we try to correct one more error than the error-correcting radius in a Reed-Solomon code over $\mathbb{F}_q$ or when we try to correct two more errors in a binary BCH code, by Theorems 3.4.1 and 3.5.7, we have that $\dim ker S = 3$. So the divisibility ideal is an ideal in the ring $\mathbb{F}[a_1,a_2]$, where $\mathbb{F} = \mathbb{F}_q$ in the first case, and $\mathbb{F} = \mathbb{F}_{2^m}$ in the second. Here we can avoid the computation of a Grobner Basis, by using a technique involving the resultant.

Here we present the general approach (see [2], ch. 3) with k polynomials in n variables for computing the first elimination ideal. We first introduce the following notations

Let $\mathbb{F}$ be a field, and let $I \subseteq \mathbb{F}[x_1,\ldots,x_n]$ be an ideal, with

$$I = (f_0(x_1,\ldots,x_n),\ldots,f_k(x_1,\ldots,x_n)).$$

Let $u_1, \ldots, u_k$ be algebrically indipendent over $\mathbb{F}$. Consider the polynomials

$$G(u_1, \ldots, u_k, x_1, \ldots, x_n) := \sum_{i=1}^{k} u_i f_i(x_1, \ldots, x_n),$$

$$p(u_1, \ldots, u_k, x_2, \ldots, x_n) = Res_{x_1}(f_0(x_1, \ldots, x_n), G(u_1, \ldots, u_k, x_1, \ldots, x_n)).$$

that lie in the ring $\mathbb{F}[u_1, \ldots, u_k, x_1, \ldots, x_n]$.

We can write p as a polynomial in $\mathbb{F}[x_1, \ldots, x_n][u_1, \ldots, u_k]$ as

$$p(u_1, \ldots, u_k, x_2, \ldots, x_n) = \sum_{\gamma \in \mathbb{N}^k} h_\gamma(x_2, \ldots, x_n)\underline{u}^\gamma.$$

Definition 5.3.1. The non zero polynomials $h_\gamma \in \mathbb{F}[x_2, \ldots, x_n]$ are called the *generalized resultants* of the polynomials $f_0(x_1, \ldots, x_n), \ldots, f_k(x_1, \ldots, x_n)$.

Theorem 5.3.2. *Suppose $\mathbb{F}$ is an algebrically closed field. Let $J \subseteq \mathbb{F}[x_2, \ldots, x_n]$ be the ideal generated by the polynomials $\{h_\gamma(x_2, \ldots, x_n)\}_{\gamma \in \mathbb{N}^k}$, and let $I_1 := I \cap \mathbb{F}[x_2, \ldots, x_n]$ be the first elimination ideal. For each $0 \leq i \leq k$, write f_i in the form*

$$f_i(x_1, \ldots, x_n) = g_i(x_2, \ldots, x_n)x_1^{N_i} + \text{ terms in which } x_1 \text{ has degree less than } N_i,$$

where $N_i \geq 0$ and g_i is non zero. If $\mathcal{V}(I_1) \cap \mathcal{V}(g_0, \ldots, g_k) = \emptyset$, then

$$J \subseteq I_1 \subseteq \sqrt{J}.$$

Proof. We first prove $J \subseteq I_1$. Let $h_\gamma \in J$. We know that there exist $A, B \in \mathbb{F}[x_1, \ldots, x_n, u_1, \ldots, u_k]$ such that

$$Af_0 + B(u_1 f_1 + \ldots + u_k f_k) = Res_{x_1}(f_0, G). \tag{5.7}$$

We can write

$$A = \sum_{\alpha} A_\alpha \underline{u}^\alpha$$

and

$$B = \sum_{\beta} B_\beta \underline{u}^\beta,$$

where $A_\alpha, B_\beta \in \mathbb{F}[x_1, \ldots, x_n]$. Now we show that $h_\gamma \in I$.
We put $e_1 = (1, 0, 0, \ldots, 0), e_2 = (0, 1, 0, \ldots, 0), \ldots, e_k = (0, 0, \ldots, 0, 1)$. So we can rewrite

$$G = \sum_{i=1}^{k} f_i \underline{u}^{e_i}.$$

Hence the equation (5.7) can be written as

$$\sum_{\gamma\in\mathbb{N}^k} h_\gamma(x)\underline{u}^\gamma = \left(\sum_\alpha A_\alpha \underline{u}^\alpha\right) f_1 + \left(\sum_\beta B_\beta \underline{u}^\beta\right)\left(\sum_{i=1}^k f_i \underline{u}^{e_i}\right)$$
$$= \sum_{\gamma\in\mathbb{N}^k}\left(A_\gamma f_0 + \sum_{i:\beta+e_i=\gamma} B_\beta f_i\right)\underline{u}^\gamma.$$

So $h_\gamma = A_\gamma f_0 + \sum_{i:\beta+e_i=\gamma} B_\beta f_i$, and $h_\gamma \in I$. Moreover $h_\gamma \in \mathbb{F}[x_2,\ldots,x_n]$ and we conclude that $h_\gamma \in I_1$.

Conversely, in order to show $I_1 \subseteq \sqrt{J}$, we prove that $\mathcal{V}(J) \subseteq \mathcal{V}(I_1)$ and then by Nullstellensatz we conclude. Suppose $\alpha = (\alpha_2,\ldots,\alpha_n) \in \mathcal{V}(J)$. Then

$$0 = \sum_{\gamma\in\mathbb{N}^k} h_\gamma(\alpha)\underline{u}^\gamma = p(\underline{u},\alpha) \overset{(*)}{=} Res_y(f_0(x_1,\alpha), G(\underline{u},x_1,\alpha)),$$

where the identity $(*)$ holds because $\alpha \notin \mathcal{V}(g_1,\ldots,g_k)$. Hence there exists a polynomial $q \in \mathbb{F}[\underline{u}][x_1]$ such that $q(\underline{u})(x_1)$ divides $f_0(x_1,\alpha)$ and $G(\underline{u},x_1,\alpha)$, with $\deg_{x_1} q > 0$. Since $q(\underline{u})(x_1)$ divides $f_0(x_1,\alpha)$, and $f_0(x_1,\alpha)$ belongs to $\mathbb{F}[x_1]$, then also $q(\underline{u})(x_1)$ is in $\mathbb{F}[x_1]$ and we can write $q(\underline{u})(x_1) = q(x_1)$.
Moreover $q(x_1)$ divides $G(\underline{u},x_1,\alpha) = \sum u_i f_i(x_1,\alpha)$, so $q(x_1)$ divides each $f_i(x_1,\alpha)$ for $i = 1,\ldots,k$. Therefore, for every root δ of $q(x_1)$, we have that $f_i(\delta,\alpha) = 0$ for $i = 0,1,\ldots,k$, i.e. $(\delta,\alpha) \in \mathcal{V}(I)$. This implies that

$$\mathcal{V}((x_1-\delta, x_i-\alpha_i)) \subseteq \mathcal{V}(I).$$

Hence we have the following chain of inclusions

$$\sqrt{I} = \mathcal{I}(\mathcal{V}(I)) \subseteq \mathcal{I}(\mathcal{V}((x_1-\delta,x_i-\alpha_i))) = (x_1-\delta,x_i-\alpha_i).$$

Therefore

$$\sqrt{I_1} \subseteq (x_1-\delta,x_i-\alpha_i) \cap \mathbb{F}[x_2,\ldots,x_n] = (x_i-\alpha_i),$$

and we obtain

$$\alpha \in \mathcal{V}(I_1)$$

□

Corollary 5.3.3. *With the hypothesis of Theorem (5.3.2) we have*

$$\mathcal{V}(I_1) = \mathcal{V}(J)$$

even if $\mathbb{F}$ *is not algebrically closed, where* $\mathcal{V}(I)$ *is defined by*

$$\mathcal{V}(I) = \left\{\alpha \in \bar{\mathbb{F}} \mid f(\alpha) = 0 \;\forall\; f \in I\right\}.$$

We can use Corollary 5.3.3 to compute the variety of the first elimination ideal of the divisibility ideal $\mathcal{I}_{\widetilde{\mathcal{F}},x^n-1} = (r_0(a_1,a_2),\ldots,r_{e-1}(a_1,a_2))$. As in Theorem 5.3.2 we call

$$G(u_1,\ldots,u_{e-1},a_1,a_2) := \sum_{i=1}^{e-1} u_i r_i(a_1,a_2),$$

and

$$p(u_1,\ldots,u_{e-1},a_2) = Res_{a_1}(f_0(a_1,a_2), G(u_1,\ldots,u_{e-1},a_1,a_2)).$$

By Corollary 5.3.3 we have that $\mathcal{V}(J) = \mathcal{V}(\mathcal{I}_{\widetilde{\mathcal{F}},x^n-1} \cap \mathbb{F}[a_2])$. However the computation of the ideal J is quite expensive. We can avoid this computation finding a different ideal $\widetilde{J}$ such that "with high probability does not differ so much from J". The idea is computing this resultant specializing the polynomial G in the variables $u_1,\ldots,u_{e-1}$. Formally, if we write

$$G = g(u_1\ldots,u_{e-1},a_2)a_1^N + \text{ terms in which } a_1 \text{ has degree less than } N,$$

then, for all $(\lambda_1,\ldots,\lambda_{e-1}) \in \mathbb{F}^{e-1}$ such that $g(\lambda_1,\ldots,\lambda_{e-1},a_2) \neq 0$, we have

$$p(\lambda_1,\ldots,\lambda_{e-1},a_2) = Res_{a_1}(f_0(a_1,a_2), G(\lambda_1,\ldots,\lambda_{e-1},a_1,a_2)),$$

and

$$(p(\lambda_1,\ldots,\lambda_{e-1},a_2)) \subseteq J.$$

If we pick randomly a point $\lambda \in \mathbb{F}^{e-1}$ we can not expect that

$$p_\lambda(a_2) := p(\lambda,a_2)$$

is exactly the generator of the ideal J. However we can repeat this computation. Choose another point $\gamma = (\gamma_1,\ldots,\gamma_{e-1}) \in \mathbb{F}^{e-1}$ such that $g(\gamma) \neq 0$ and γ is not proportional to λ. If we call

$$p_\gamma(a_2) := p(\gamma,a_2),$$

again we have that

$$(p_\gamma) \subseteq J.$$

Now we denote by

$$\tilde{h}(a_2) := \gcd\{p_\gamma(a_2), p_\lambda(a_2)\}.$$

We would like to understand if $(\tilde{h}) = J$.

Let $h(a_2) \in \mathbb{F}[a_2]$ be the generator of the ideal J. Since $(\tilde{h}(a_2)) \subseteq (h(a_2))$, the polynomial $\tilde{h}(a_2)$ is divisible by $h(a_2)$. Moreover $(\tilde{h}(a_2)) = (h(a_2))$ if and only if

$$\left(\frac{\tilde{h}(a_2)}{h(a_2)}\right) = (1).$$

This is equivalent to say that

$$\gcd\left\{\frac{p_\lambda(a_2)}{h(a_2)}, \frac{p_\gamma(a_2)}{h(a_2)}\right\} = 1.$$

Now we can rewrite this condition as a condition on the resultant, that is

$$Res_{a_2}\left(\frac{p_\lambda(a_2)}{h(a_2)}, \frac{p_\gamma(a_2)}{h(a_2)}\right) \neq 0.$$

Now consider $u_1, \ldots, u_{e-1}$, $v_1, \ldots, v_{e-1}$, two sets of algebrically indipendent elements over $\mathbb{F}$, and let $\tilde{p}$ be the polynomial defined by

$$\tilde{p}(u_1, \ldots, u_{e-1}, a_2) = \frac{p(u_1, \ldots, u_{e-1}, a_2)}{h(a_2)}.$$

Hence $(\tilde{h}(a_2)) = J$ if and only if

$$Res_{a_2}(\tilde{p}(\lambda_1, \ldots, \lambda_{e-1}, a_2), \tilde{p}(\gamma_1, \ldots, \gamma_{e-1}, a_2)) \neq 0.$$

Let $H(u, v)$ be the polynomial defined by

$$H(u, v) := Res_{a_2}(\tilde{p}(u, a_2), \tilde{p}(v, a_2)), \tag{5.8}$$

thus

$$(\tilde{h}) = J \iff H(\lambda, \gamma) \neq 0. \tag{5.9}$$

Here the following question naturally arises: choosing randomly λ and γ what is the probability that $H(\lambda, \gamma) \neq 0$? The answer is given by the following well-known result.

Theorem 5.3.4 (Schwartz-Zippel Lemma)**.** *Let $H \in \mathbb{F}[z_1, \ldots, z_n]$ be a non-zero polynomial of total degree $d \geq 0$ over a field $\mathbb{F}$. Let $S \subset \mathbb{F}$ be a finite subset and let $r_1, \ldots, r_n$ be selected at random indipendently and uniformly from S. Then*

$$\mathbb{P}[H(r_1, \ldots, r_n) = 0] \leq \frac{d}{|S|} \tag{5.10}$$

Proof. The proof is by induction on n. For $n = 1$, H can have at most d roots. This gives us the base case.
Now suppose that the theorem holds for all polynomials in $n-1$ variables. We consider H as a polynomial in z_1, and we can write it as

$$H(z_1, \ldots, z_n) = \sum_{i=0}^{d} H_i(z_2, \ldots, z_n) z_1^i.$$

Since H is not identically 0, there exists some i such that H_i is not identically 0. If we take the largest such i, then $\deg H_i \leq d - i$, since the degree of $z_1^i H_i$ is at most d. Now we randomly pick $r_2, \ldots, r_n$ from S. By the induction hypothesis,

$$\mathbb{P}[H_i(r_2, \ldots, r_n) = 0] \leq \frac{d-i}{|S|}.$$

If $H_i(r_2, \ldots, r_n) \neq 0$, then $H(z_1, r_2, \ldots, r_n)$ is of degree i, and so

$$\mathbb{P}\big[H(r_1, r_2, \ldots, r_n) = 0 \mid H_i(r_2, \ldots, r_n) \neq 0\big] \leq \frac{i}{|S|}.$$

If we denote the event $H(r_1, r_2, \ldots, r_n) = 0$ by A and the event $H_i(r_2, \ldots, r_n) = 0$ by B, we have

$$\begin{aligned}
\mathbb{P}[A] &= \mathbb{P}[A \cap B] + \mathbb{P}[A \cap B^c] \\
&= \mathbb{P}[B]\mathbb{P}[A|B] + \mathbb{P}[B^c]\mathbb{P}[A|B^c] \\
&\leq \mathbb{P}[B] + \mathbb{P}[A|B^c] \\
&\leq \frac{d-i}{|S|} + \frac{i}{|S|} = \frac{d}{|S|},
\end{aligned}$$

and this complete the proof. □

Now we figure out a bound on the total degree of the polynomial H in 5.8.

Lemma 5.3.5. *Let $H \in \mathbb{F}[u, v]$ be the polynomial defined in (5.8). Then*

$$\deg_{(u,v)} H \leq 4(q-e)^3.$$

Proof. By definition of the polynomial H, we have

$$\begin{aligned}
\deg_{(u,v)} H &= \deg_{(u,v)} Res_{a_2}(\tilde{p}(u, a_2), \tilde{p}(v, a_2)) \leq \\
&\leq (\deg_{(u,v)} \tilde{p})(\deg_{a_2} \tilde{p}) + (\deg_{(u,v)} \tilde{p})(\deg_{a_2} \tilde{p}) = \\
&= 2(\deg_{(u,v)} \tilde{p})(\deg_{a_2} \tilde{p}) = \\
&= 2(\deg_{(u,v)} p)(\deg_{a_2} p - \deg_{a_2} h) \leq \\
&\leq 2(\deg_{(u,v)} p)(\deg_{a_2} p).
\end{aligned}$$

Now we have to estimate $\deg_{(u,v)} p$ and $\deg_{a_2} p$.

For the first one we have

$$\begin{aligned}\deg_u p = \deg_u Res_{a_1}(r_0, G) \leq \\ \leq (\deg_u r_0)(\deg_{a_1} G) + (\deg_u G)(\deg_{a_1} r_0) = \\ \overset{(*)}{=} \deg_{a_1} r_0,\end{aligned}$$

where the identity $(*)$ holds because $\deg_u r_0 = 0$ and $\deg_u G = 1$.

For the second term we obtain

$$\begin{aligned}\deg_{a_2} p = \deg_{a_2} Res_{a_1}(r_0, G) \leq \\ \leq (\deg_{a_2} r_0)(\deg_{a_1} G) + (\deg_{a_2} G)(\deg_{a_1} r_0).\end{aligned}$$

Moreover, by Lemma 2.1.7 we have

$$\begin{cases}\deg_{a_1} r_0 \leq q - e \\ \deg_{a_1} G \leq \max\left\{\deg_{a_1} r_i \mid i = 1, \ldots, e-1\right\} \leq q - e \\ \deg_{a_2} r_0 \leq q - e \\ \deg_{a_2} G \leq \max\left\{\deg_{a_2} r_i \mid i = 1, \ldots, e-1\right\} \leq q - e.\end{cases} \tag{5.11}$$

So, remounting all the inequality obtained, we have

$$\begin{aligned}\deg_{(u,v)} H \leq 2(\deg_{(u,v)} p)(\deg_{a_2} p) \leq \\ \leq 2(\deg_{a_1} r_0)\left[(\deg_{a_2} r_0)(\deg_{a_1} G) + (\deg_{a_2} G)(\deg_{a_1} r_0)\right] \leq \\ \leq 2(q-e)\left[2(q-e)^2\right] = 4(q-e)^3.\end{aligned}$$

□

Howevere here we are working in a finite field $\mathbb{F}$ with q elements and so at a first look the Schwartz-Zippel Lemma does not help us. In fact, using the bound given in Lemma 5.3.5, we would obtain

$$\mathbb{P}[H(\lambda, \gamma) \neq 0] = 1 - \mathbb{P}[H(\lambda, \gamma) = 0] \geq 1 - \frac{4(q-e)^3}{q},$$

that is almost always negative and it does not say anything about the probability.

We can avoid this problem with a little trick. We can embed the field $\mathbb{F}$ into a finite extension $\mathbb{K}$ and use the Schwartz-Zippel Lemma here. This

embedding does not modify the structure of the ideal we are working on. In fact consider the natural embedding

$$i : \mathbb{F}[a_2] \hookrightarrow \mathbb{K}[a_2].$$

Here the inclusion i maps the ideal $J = (h(a_2))$ into the ideal of $J^e = (h(a_2)) \subset \mathbb{K}[a_2]$. In particular

$$\mathcal{V}(J) = \mathcal{V}(J^e).$$

So we can work directly on $\mathbb{K}[a_2]$.

In particular $\mathbb{F} = \mathbb{F}_q$ and so $\mathbb{K} = \mathbb{F}_{q^s}$ for a certain positive integer $s > 0$. If we choose the integer s quite large such that the quantity

$$\frac{4(q-e)^3}{q^s}$$

is small enough to make the probability that $H(\lambda, \gamma) \neq 0$ close to 1.

Formally we can sumarize everything said above in the following results.

Theorem 5.3.6. *Let C be an $[n, k, n-k+1]$ Reed-Solomon code over $\mathbb{F}_q$, where $n = q-1$ and $n-k+1 = 2t+1$ is the minimum distance of C. Let $r \in \mathbb{F}_q^n$ be a received word and suppose that the minimum error weight e is greater or equal to $t-1$. Consider an escalier basis $\mathcal{F} = \{f_0, f_1, f_2\}$ for $\ker S_{t+1}$, where S_{t+1} is the $(t+1)$th syndrome matrix of r, and let $\mathcal{I}_{\mathcal{F}, x^{q-1}-1} = (r_0, \ldots, r_t) \subset \mathbb{F}_q[a_1, a_2]$ be the divisibility ideal.*

Then $\forall\ \varepsilon > 0\ \ \exists s \in \mathbb{N}$ such that, choosing at random indipendently and uniformly $\lambda_1, \ldots, \lambda_t$, $\gamma_1, \ldots, \gamma_t \in \mathbb{F}_{q^s}$, and defining

$$p_\lambda = Res_{a_1}\left(r_0, \sum \lambda_i r_i\right)$$

$$p_\gamma = Res_{a_1}\left(r_0, \sum \gamma_i r_i\right),$$

we have

$$\mathbb{P}[J^e = (p_\lambda, p_\gamma)] \geq 1 - \varepsilon.$$

□

An equivalent version of this result can be reformulate also for binary BCH codes when we try to correct *exactly* two more errors. In fact we have shown in Theorem 3.5.5 that, for a binary BCH code with minumum error weight greater or equal to $t-2$, $dim \ker S_{t+2} = 3$.

Theorem 5.3.7. *Let C be a binary BCH code of length $n = 2^m - 1$ and designed distance $\delta = 2t + 1$ over $\mathbb{F}_2$. Let $r \in \mathbb{F}_2^n$ be a received word and suppose that the minimum error weight e is greater or equal to $t-2$. Consider an escalier basis $\widetilde{\mathcal{F}} = \{f_0, f_1, f_2\}$ for $\ker \widetilde{S}_{t+2}$, where $\widetilde{S}_{t+2}$ is the extended $(t+2)$th syndrome matrix of r, and let $\mathcal{I}_{\widetilde{\mathcal{F}}, x^{2^m-1}-1} = (r_0, \ldots, r_{t+1}) \subset \mathbb{F}_q[a_1, a_2]$ be the divisibility ideal.*

Then $\forall\ \varepsilon > 0\ \ \exists s \in \mathbb{N}$ such that, choosing at random indipendently and uniformly $\lambda_1, \ldots, \lambda_{t+1}$, $\gamma_1, \ldots, \gamma_{t+1} \in \mathbb{F}_{2^s}$, and defining

$$p_\lambda = Res_{a_1}\left(r_0, \sum \lambda_i r_i\right)$$

$$p_\gamma = Res_{a_1}\left(r_0, \sum \gamma_i r_i\right),$$

we have

$$\mathbb{P}[J^e = (p_\lambda, p_\gamma)] \geq 1 - \varepsilon.$$

□

Now we show that the unlucky case in which

$$(\tilde{h}) = (p_\lambda, p_\gamma) \subsetneq J$$

is not so "unlucky ". In fact, if $(\tilde{h}) \subsetneq (h)$ it means that

$$\tilde{h} = hg$$

with $\deg r \geq 1$. Consider a root β of g. We have to distinguish two cases:

- β is a root of h. So it does not modify the computation of $\mathcal{V}(J)$.

- β is not a root of h. This means that we can not extend β to a point in $\mathcal{V}(\mathcal{I}_{\mathcal{F},H})$ with second coordinate equal to β, and it does not affect the computation of $\mathcal{V}(\mathcal{I}_{\mathcal{F},H})$.

However, since we know that $\mathcal{I}_{\mathcal{F},H}$ is radical and $\mathcal{V}(J) \subseteq \mathbb{F}_q$, we can increase the accuracy of the computation by taking the ideal

$$\widetilde{J} = (p_\lambda(a_2), p_\gamma(a_2), a_2^q - a_2) = \left(\gcd\left\{p_\lambda(a_2), p_\gamma(a_2), a_2^q - a_2\right\}\right).$$

Hence we are able to remove all the exceeding roots that are not in $\mathbb{F}_q$ and all the multiple roots (but we are not able to remove the exceeding roots that lie in $\mathbb{F}_q$). Moreover

$$\mathbb{P}\left[\widetilde{J} = \mathcal{I}_{\mathcal{F},H} \cap \mathbb{F}_q[a_2]\right] \geq \mathbb{P}\left[(p_\lambda, p_\gamma) = J^e\right].$$

The procedure described in this section let us significantly reduce the cost of the computation of $\mathcal{V}(\mathcal{I}_{\mathcal{F},H})$ and it avoids the use of Grobner basis, whose computational cost is exponential.

5.4 Sudan and Guruswami-Sudan algorithms

Here we briefly present the Madhu Sudan's work on list decoding for Reed-Solomon codes. This approach is based on the representation of an RS code given by Theorem 1.6.3. The work is focused on the solution of the following problem:

Problem 1. *Given a finite field $\mathbb{F}$, n distinct pairs of element $\{(x_i, y_i))\}_{i=0}^{n-1}$ of $\mathbb{F}\times\mathbb{F}$, and two integers k and ℓ, determine a list of all functions $f : \mathbb{F} \longrightarrow \mathbb{F}$ satisfying the conditions*

$$\begin{aligned} f(x) &\in \mathbb{F}[x]_{\leq k}, \\ |\{i \mid f(x_i) = y_i\}| &\geq \ell. \end{aligned} \tag{5.12}$$

Suppose we are working with an RS code C of length $n = q - 1$ and designed distance d over $\mathbb{F}_q$. The dimension of the code is $k + 1$, where $k = n - d$. Suppose we have a received word $r = (r_0, \ldots, r_{n-1})$. Replacing x_i with α^i and y_i with r_i, by Theorem 1.6.3, solving the problem 1 equals to finding all the codewords $c = (c_0, \ldots, c_{n-1})$ with $c_i = f(\alpha^i)$, such that $d(c, r) \leq n - \ell$, i.e. it equals to the list decoding problem with error bound $e = n - \ell$.

Definition 5.4.1. For weight ω_x, $\omega_y \in \mathbb{N}^+$, the $(\omega_x, \omega_y)-$*weighted degree* of a monomial $q_{ij}x^iy^j$ is $i\omega_x + j\omega_y$. The $(\omega_x, \omega_y)-$*weighted degree* of a polynomial $Q(x, y) \in \mathbb{F}[x, y]$ is the maximum, over the monomials with non-zero coefficients, of the $(\omega_x, \omega_y)-$weighted degree of the monomials.

The algorithm is the following.

Algorithm 1. *1. Input: $n, k, \ell : \{(x_0, y_0), \ldots, (x_{n-1}, y_{n-1})\}$.*

2. Parameters $s = \lceil \sqrt{2(n + 1)/k} \rceil - 1, m = \lceil k/2 \rceil - 1$.

3. Find any polynomial function $Q : \mathbb{F} \times \mathbb{F} \longrightarrow \mathbb{F}$ satisfying

$$\begin{cases} Q(x, y) \text{ has } (1, k) - \text{weighted degree at most } m + sk, \\ \forall\, i = 0, \ldots, n - 1, Q(x_i, y_i) = 0, \\ Q \text{ is not identically zero.} \end{cases} \tag{5.13}$$

4. Factor the polynomial Q into irreducible factors.

5. Output all the polynomials $f \in \mathbb{F}_q[x]$ of degree at most k such that $(y - f(x))$ is a factor of Q and $f(x_i) = y_i$ for at least ℓ values of $i \in \{0, \ldots, n - 1\}$.

The correctnes of the algorithm stated above is guaranteed by the following results. For the proofs see [11].

Proposition 5.4.2. *If a polynomial function $Q : \mathbb{F} \times \mathbb{F} \longrightarrow \mathbb{F}$ satisfying (5.13) exists, then one can be found in polynomial time in n.*

Proposition 5.4.3. *If*

$$(m+1)(s+1) + k\binom{s+1}{2} > n$$

then there exists a polynomial function $Q(x, y)$ satisfying (5.13).

Proposition 5.4.4. *If $Q(x, y)$ is a polynomial function satisfying (5.13) and $f(x)$ is a polynomial function satisfying (5.12) and $\ell > m+sk$, then $(y-f(x))$ divides $Q(x, y)$.*

Theorem 5.4.5. *Given a sequence of n distinct pairs $\{(x_i, y_i)\}_{i=0}^{n-1}$, where the x_is and the y_is are elements of a field $\mathbb{F}$, and integer parameters ℓ and k such that $\ell \geq k\lceil\sqrt{2(n+1)/k}\rceil - \lfloor k/2\rfloor$, there exists an algorithm, which runs in polynomial time in n, that can find all the polynomial f of degree at most k such that the number of points (x_i, y_i) that satisfy $y_i = f(x_i)$ is at least ℓ.*

This algorithm can correct (in the sense of list-decoding) up to $n - \sqrt{2nk}$ errors.

In 1998 Sudan and his then doctoral student Guruswami presented an improvement on the above algorithm for list decoding Reed–Solomon codes, and it corrects up to $n - \sqrt{kn}$ errors.

Algorithm 2. *1. Input: $n, k, \ell : \{(x_0, y_0), \ldots, (x_{n-1}, y_{n-1})\}$.*

2. Parameters r, s such that

$$r\ell > s, \quad \text{and} \quad n\binom{r+1}{2} < \frac{s(s+2)}{2k}.$$

In particular set

$$r = 1 + \left\lfloor \frac{kn + \sqrt{k^2n^2 + 4(\ell^2 - kn)}}{2(\ell^2 - kn)} \right\rfloor,$$
$$s = r\ell - 1.$$

3. Find a polynomial function $Q : \mathbb{F}\times\mathbb{F} \longrightarrow \mathbb{F}$ whose $(1, k)$-weighted degree is at most s, i.e. find the values for its coefficients $\{q_{j_1,j_2} \mid j_1, j_2 \geq 0, j_1 + kj_2 \leq s\}$ such that the following conditions hold:

- *At least one* q_{j_1,j_2} *is non zero;*
- *For every* $i \in \{0,\ldots,n-1\}$ *if* $Q^{(i)}$ *denotes the shift of* Q *to* (x_i, y_i)*, then all coefficients of* $Q^{(i)}$ *of total degree less than* r *are* 0*. Formally,*

$$\forall\, i \in \{0,\ldots,n-1\},\ \forall\, j_1, j_2 \geq 0 \ \textit{s.t.}\ j_1 + j_2 < r,$$

$$q^{(i)}_{j_1,j_2} := \sum_{j_1' \geq j_1} \sum_{j_2' \geq j_2} \binom{j_1'}{j_1} \binom{j_2'}{j_2} q_{j_1',j_2'} x_i^{j_1'-j_1} y_i^{j_2' j_2} = 0.$$

4. *Output all the polynomials* $f \in \mathbb{F}_q[x]$ *of degree at most* k *such that* $(y - f(x))$ *is a factor of* Q *and* $f(x_i) = y_i$ *for at least* ℓ *values of* $i \in \{0,\ldots,n-1\}$*.*

The correctness of this algorithm is guaranteed by the following results (see [7]).

Lemma 5.4.6. *If* $f(x)$ *is a polynomial of degree at most* k *such that* $y_i = f(x_i)$ *for at least* ℓ *values of* $i \in \{0,\ldots,n-1\}$ *and* $r\ell > s$*, then* $y - f(x)$ *divides* $Q(x,y)$*.*

□

Lemma 5.4.7. *If*

$$n\binom{r+1}{2} < \frac{s(s+2)}{2k},$$

then a polynomial Q *with the properties sought in step 3 of algorithm 2 does exist, and it can be found in polynomial time by solving a linear system.*

□

Lemma 5.4.8. *If* n, k, ℓ *satisfy* $\ell^2 > kn$*, then for the choice of* r *and* s *made in step 2 of algorithm 2,*

$$n\binom{r+1}{2} < \frac{s(s+2)}{2k} \qquad \textit{and} \qquad r\ell > s$$

both hold.

□

Theorem 5.4.9. *Algorithm 2 on inputs* n, k, ℓ *and the points* $\{(x_i, y_i)\}_{i=0}^{n-1}$*, correctly solves the polynomial reconstruction problem, provided* $\ell > \sqrt{kn}$*.*

□

5.5 Comparison between the algorithms

Here we discuss pros and cons of our algorithm in comparison with Guruswami-Sudan algorithm.

The Guruswami-Sudan algorithm is the most important list-decoding algorithm developed for Reed-Solomon codes. As we have seen in the previous section, it has the advantage of solving the list decoding problem in polynomial time in the length of the code C. However it effectively works only for low-rate codes, In fact the GS algorithms can correct up to $n - \sqrt{nk}$ errors, that in case of high-rate codes can be a value arbitrariarily small.

Consider the algorithm that we have discussed in this thesis. In comparison of GS algorithm, it has the advantage of working for every kind of Reed Solomon code, without any restriction on its rate. Moreover it works also for binary BCH codes, while GS does not. The disadvantage is that in the general case its running time is not polynomial in the length of C, because it involves the computation of a Gröbner basis. However, as we have seen in Section 5.3, in some special cases we can reduce the computationl cost of the algorithm by making it polynomial in the length of C. In these cases our algorithm is really competitive, especially because GS algorithm does not always work.

Chapter 6

Examples

In this section we illustrate some examples in order to show how the algorithm works step by step. Some of this examples follows by introducing random errors, while some others are constructed ad hoc, in order to find more codewords as possible in a Hamming ball of radius e, as shown in [9].

6.1 Binary BCH code with $n = 31$ and $\delta = 11$

Here we show an example of how the algorithm works for binary BCH codes. Let C be a primitive binary BCH code of length $n = 31$ and designed distance $\delta = 11$. Consider γ as a primitive element of $\mathbb{F}_{32}$ over $\mathbb{F}_2$, with γ satisfying

$$\gamma^5 + \gamma^2 + 1 = 0.$$

Suppose that the codeword $c = (0, 0, \ldots, 0)$ is transmitted, and 6 random errors occur during the transmission, so that the received word, seen as a polynomial, is

$$r = e_8 + e_{14} + e_{21} + e_{24} + e_{27} + e_{30},$$

where we denoted by e_i the vector whose $i + 1$-th entry equals 1.

The syndrome vector is given by

$$(s_1, s_2, \ldots, s_{10}) = \left(\gamma^{23}, \gamma^{15}, \gamma^9, \gamma^{30}, \gamma, \gamma^{18}, \gamma^{25}, \gamma^{29}, \gamma^8, \gamma^2\right).$$

and we can remove the linearly dependent equations from the 6-th extended syndrome matrix, obtaining the matrix

$$\widetilde{S} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & \gamma^{23} \\ \gamma^{23} & \gamma^{15} & \gamma^9 & \gamma^{30} & \gamma & \gamma^{18} & \gamma^{25} \\ \gamma^{15} & \gamma^9 & \gamma^{30} & \gamma & \gamma^{18} & \gamma^{25} & \gamma^{29} \\ \gamma^9 & \gamma^{30} & \gamma & \gamma^{18} & \gamma^{25} & \gamma^{29} & \gamma^8 \\ \gamma^{30} & \gamma & \gamma^{18} & \gamma^{25} & \gamma^{29} & \gamma^8 & \gamma^2 \end{bmatrix},$$

with $\ker(\widetilde{S}) = \ker(\widetilde{S}_6)$.
Here the kernel of the matrix $\widetilde{S}$ is generated by the polynomials

$$f_0(x) = x^6 + \gamma x^5 + \gamma^{23}x^4 + \gamma^{25}x^3 + \gamma^2 x^2,$$
$$f_1(x) = x^5 + \gamma^{26}x^4 + \gamma^2 x^3 + \gamma^2 x^2 + \gamma^{25}x + \gamma^2.$$

The Euclidean division between $H(x) = x^{31} - 1$ and

$$\begin{aligned} F(a,x) =& x^6 + (a+z)x^5 + (\gamma^{26}a + \gamma^{23})x^4 + (\gamma^2 a + \gamma^{25})x^3 + \\ & (\gamma^2 a + \gamma^2)x^2 + \gamma^{25}ax + \gamma^2 a \end{aligned}$$

produces the divisibility ideal

$$\mathcal{I} = (a^2 + \gamma^2 a + \gamma^6),$$

whose variety is

$$\mathcal{V}(\mathcal{I}) = \left\{\gamma^8, \gamma^{29}\right\}.$$

So the divisibility set is $\Sigma(r,6) = \{\sigma_1(x), \sigma_2(x)\}$, where

$$\sigma_1(x) = x^6 + \gamma^{23}x^5 + \gamma^{11}x^4 + \gamma^3 x^3 + \gamma^{22}x^2 + \gamma^2 x + \gamma^{10},$$
$$\sigma_2(x) = x^6 + \gamma^{27}x^5 + \gamma^{10}x^4 + \gamma^{21}x^3 + \gamma^5 x^2 + \gamma^{23}x + 1.$$

From this divisibility set, by applying the decoding function on it, we have

$$\mathcal{D}_{r,6}(\sigma_1(x)) = r + e_7 + e_{16} + e_{17} + e_{22} + e_{23} + e_{29},$$
$$\mathcal{D}_{r,6}(\sigma_2(x)) = r + e_8 + e_{14} + e_{21} + e_{24} + e_{27} + e_{30} = (0,0,\ldots,0).$$

So there are two different codewords at distance 6 from r.

6.2 $[15,7,9]$ RS code

Consider the finite field $\mathbb{F}_{16}$ with primitive element γ satisfying $\gamma^4 + \gamma + 1$. Let C be the $[15,7,9]$ Reed Solomon code over $\mathbb{F}_{16}$. Here the error-correction radius is $t = 4$, and we would like to correct up to $e = 5$ errors. Suppose that $c = (0,\ldots,0)$ is the transmitted word, and we generate 5 random errors, so the received word is

$$r = \left(0,0,\gamma^{11},0,\gamma^{12},\gamma^{11},0,0,0,0,0,0,\gamma^3,0,\gamma^7\right).$$

The vector of the syndromes is given by

$$(s_1,\ldots,s_8) = \left(0,\gamma,\gamma^{14},\gamma^3,\gamma^3,\gamma^2,\gamma^6,\gamma^{14}\right),$$

and the 5-th syndrome matrix is

$$S_5 = \begin{bmatrix} 0 & \gamma & \gamma^{14} & \gamma^3 & \gamma^3 & \gamma^2 \\ \gamma & \gamma^{14} & \gamma^3 & \gamma^3 & \gamma^2 & \gamma^6 \\ \gamma^{14} & \gamma^3 & \gamma^3 & \gamma^2 & \gamma^6 & \gamma^{14} \end{bmatrix}.$$

Computing the kernel of S_5 we obtain

$$\ker(S_5) = \langle f_0, f_1, f_2 \rangle,$$

where

$$\begin{aligned} f_0(x) &= x^5 + \gamma x^4 + \gamma^{10} x^3 + \gamma^8 x^2, \\ f_1(x) &= x^4 + x^3 + \gamma^7 x^2 + \gamma^3 x, \\ f_2(x) &= x^3 + \gamma x^2 + \gamma^8 x + 1. \end{aligned}$$

After the Euclidean division of the polynomial $H(x) = x^{15} - 1$ by $F(a, b, x) = f_0(x) + a f_1(x) + b f_2(x)$, we obtain that the divisibility ideal is given by

$$\mathcal{I} = (r_0(a, b), r_1(a, b), r_2(a, b), r_3(a, b), r_4(a, b)),$$

where the polynomial r_i are defined by

$$
\begin{aligned}
r_0(a,b) = {} & a^{10}b + a^9b + a^8b^2 + \gamma^4a^8b + \gamma^{14}a^7b + \gamma^2a^6b^2 + \gamma^{12}a^6b + a^5b^3 + \\
& a^5b + a^4b^4 + \gamma^4a^4b^3 + a^2b^5 + \gamma^7a^2b^2 + a^2b + ab^5 + \gamma^{10}ab^3 + \\
& ab + b^6 + \gamma^4b^5 + \gamma^5b^4 + \gamma^9b^3 + \gamma^7b^2 + \gamma^4b + 1, \\
r_1(a,b) = {} & \gamma^3a^{11} + \gamma^8a^{10}b + \gamma^3a^{10} + \gamma^6a^9b + \gamma^7a^9 + \gamma^8a^8b^2 + \gamma^{13}a^8b + \\
& \gamma^2a^8 + \gamma^{10}a^7b + a^7 + \gamma^{13}a^6b^2 + a^6b + \gamma^3a^6 + \gamma^6a^5b^3 + \gamma^7a^5b^2 + \\
& \gamma^3a^5b + \gamma^8a^4b^4 + \gamma^{13}a^4b^3 + \gamma^{14}a^4b^2 + \gamma^{14}a^4b + \gamma^3a^3b^4 + \gamma^9a^3b + \\
& \gamma^3a^3 + \gamma^8a^2b^5 + \gamma^3a^2b^4 + \gamma^9a^2b^2 + \gamma^6a^2b + \gamma^3a^2 + \gamma^6ab^5 + \\
& \gamma^7ab^4 + \gamma^5ab^3 + \gamma^{12}ab^2 + \gamma ab + \gamma^7a + \gamma^8b^6 + \gamma^{13}b^5 + \gamma^2b^4 + \\
& b^3 + \gamma^7b^2 + \gamma^{12}b, \\
r_2(a,b) = {} & \gamma^7a^{11} + \gamma a^{10}b + \gamma^5a^{10} + \gamma^6a^9b + \gamma^3a^9 + \gamma a^8b^2 + \gamma^3a^8b + \\
& \gamma^6a^8 + \gamma^{13}a^7b + \gamma^8a^7 + \gamma^2a^6b + \gamma^{12}a^6 + \gamma^6a^5b^3 + \gamma^3a^5b^2 + \\
& \gamma^5a^5b + a^5 + \gamma a^4b^4 + \gamma^3a^4b^3 + \gamma^8a^4b^2 + \gamma^4a^4b + \gamma a^4 + \\
& \gamma^7a^3b^4 + \gamma a^3b + \gamma^{12}a^3 + \gamma a^2b^5 + \gamma^5a^2b^4 + \gamma^2a^2b^2 + \gamma^2a^2b + \\
& \gamma^{11}a^2 + \gamma^6ab^5 + \gamma^3ab^4 + \gamma^3ab^3 + \gamma^2ab^2 + \gamma^2ab + \gamma^7a + \gamma b^6 + \\
& \gamma^3b^5 + \gamma^{14}b^4 + \gamma^5b^3 + \gamma^{14}b^2 + \gamma^4b + \gamma^{12}, \\
r_3(a,b) = {} & a^{11} + a^{10}b + \gamma^{13}a^{10} + \gamma a^9b + \gamma^6a^9 + a^8b^2 + \gamma^8a^8b + \gamma^4a^7b + \\
& \gamma^{12}a^6b^2 + \gamma^7a^6b + \gamma^{11}a^6 + \gamma a^5b^3 + \gamma^6a^5b^2 + \gamma^9a^5b + \gamma^7a^5 + \\
& a^4b^4 + \gamma^8a^4b^3 + \gamma^8a^4b^2 + \gamma^{14}a^4b + \gamma^{11}a^4 + a^3b^4 + \gamma^{10}a^3b + \\
& \gamma^7a^3 + a^2b^5 + \gamma^{13}a^2b^4 + \gamma^5a^2b^2 + \gamma a^2b + \gamma^{13}a^2 + \gamma ab^5 + \\
& \gamma^6ab^4 + \gamma^5ab^3 + \gamma^8ab^2 + \gamma^{13}ab + \gamma a + b^6 + \gamma^8b^5 + \gamma^4b^4 + \\
& \gamma^2b^3 + \gamma^9b^2 + \gamma^{14}b + \gamma^{14}, \\
r_4(a,b) = {} & a^{11} + \gamma a^{10} + \gamma^{12}a^9 + \gamma a^8b + \gamma^{13}a^8 + \gamma^4a^7 + \gamma^4a^6b + \gamma^5a^6 + \\
& \gamma^{12}a^5b^2 + \gamma a^5 + \gamma a^4b^3 + \gamma^{13}a^4b^2 + \gamma^9a^4b + \gamma^2a^4 + a^3b^4 + \\
& \gamma^{14}a^3 + \gamma a^2b^4 + \gamma^{10}a^2b + a^2 + \gamma^{12}ab^4 + \gamma^5ab^2 + \gamma^{14}a + \gamma b^5 + \\
& \gamma^{13}b^4 + \gamma^5b^3 + \gamma^6b^2 + \gamma^{13}b + 1.
\end{aligned}
$$

In order to compute the variety of the ideal $\mathcal{I}$, we use the probabilistic method described in Section 5.3.2. Consider the extension field $\mathbb{F}_{4096}$, with δ primitive element which satisfies the relation

$$\delta^{12} + \delta^7 + \delta^6 + \delta^5 + \delta^3 + \delta + 1$$

over $\mathbb{F}_2$. Here we have that $\gamma = \delta^{273} = \delta^{10} + \delta^9 + \delta^8 + \delta^4 + \delta^3 + \delta^2$. We can choose randomly two 4-dimensional vector in $\mathbb{F}_{4096}$, e.g.

$$\left(\delta^{3932}, \delta^{1964}, \delta^{1120}, \delta^{69}\right),$$

$$\left(\delta^{3435}, \delta^{1242}, \delta^{790}, \delta^{137}\right).$$

Now, we take

$$G_1(a,b) = \delta^{3932} r_1(a,b) + \delta^{1964} r_2(a,b) + \delta^{1120} r_3(a,b) + \delta^{69} r_4(a,b),$$

$$G_2(a,b) = \delta^{3435} r_1(a,b) + \delta^{1242} r_2(a,b) + \delta^{790} r_3(a,b) + \delta^{137} r_4(a,b).$$

If $p(b)$ is the polynomial defined by

$$\tilde{h}(b) = \gcd\{Res_a(r_0(a,b), G_1(a,b)), Res_a(r_0(a,b), G_2(a,b)\},$$

then we have

$$\tilde{h}(b) = b - \left(\delta^{11} + \delta^9 + \delta^5 + \delta^4 + \delta^3 + \delta^2 + \delta\right) = b - \gamma^8.$$

Now, substituting $b = \gamma^8$, we obtain that the variety of $\mathcal{I}$ is

$$\mathcal{V}(\mathcal{I}) = \left\{(\gamma^6, \gamma^8)\right\},$$

and so

$$\Sigma(r,5) = \left\{x^5 + \gamma^{11}x^4 + \gamma^{11}x^3 + \gamma x^2 + \gamma^3 x + \gamma^8\right\}.$$

Therefore, by using the decoding function on the set $\Sigma(r,5)$,we have that the only codeword with distance at most 5 from the received word r is

$$\mathcal{D}_{r,5}\left(x^5 + \gamma^{11}x^4 + \gamma^{11}x^3 + \gamma x^2 + \gamma^3 x + \gamma^8\right) = (0,0,\ldots,0).$$

6.3 $[15,7,9]$ RS code

Here we present another example, suggested by J. Justesen [9]. The code C is the same as before, i.e. the $[15,7,9]$ Reed Solomon code over $\mathbb{F}_{16}$. As a primitive element we consider again γ satisfying $\gamma^4 + \gamma + 1$.

Suppose that $c = (0,0,\ldots,0)$ is the transmitted word and 5 errors occur during the transmission, producing the received word

$$r = (1,0,0,1,0,0,1,0,0,1,0,0,1,0,0).$$

The vector of the syndromes is given by

$$(s_1,\ldots,s_8) = (0,0,0,0,1,0,0,0),$$

and the 5-th syndrome matrix is

$$S_5 = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}.$$

The kernel of S_5 is given by

$$\ker(S_5) = \langle f_0, f_1, f_2 \rangle$$

where

$$\begin{aligned} f_0(x) &= x^5, \\ f_1(x) &= x^4, \\ f_2(x) &= 1. \end{aligned}$$

The Euclidean division between $H(x) = x^{15} - 1$ and $F(a, b, x) = x^5 + ax^4 + b$ produces the divisibility ideal $\mathcal{I}$ generated by

$$\begin{aligned} r_0(a, b) &= a^{10}b + b^3 + 1, \\ r_1(a, b) &= a^9 b + a^4 b^2, \\ r_2(a, b) &= a^8 b, \\ r_3(a, b) &= a^7 b + a^2 b^2, \\ r_4(a, b) &= a^{11} + a^6 b + a b^2. \end{aligned}$$

Using the probabilistic method shown in Section 5.3.2, we obtain that the first elimination ideal is given by

$$\mathcal{I} \cap \mathbb{F}_{16}[b] = \left(b^3 - 1\right),$$

and the variety of $\mathcal{I}$ is

$$\mathcal{V}(\mathcal{I}) = \left\{(0, 1), (0, \gamma^5), (0, \gamma^{10})\right\}.$$

Thus the divisibility set is

$$\Sigma(r, e) = \left\{x^5 + 1, x^5 + \gamma^5, x^5 + \gamma^{10}\right\}.$$

From this we obtain that

$$\begin{aligned} \mathcal{D}_{r,5}\left(x^5 + 1\right) &= (0, 0, \ldots, 0) \\ \mathcal{D}_{r,5}\left(x^5 + \gamma^5\right) &= \left(1, 0, \gamma^5, 1, 0, \gamma^5, 1, 0, \gamma^5, 1, 0, \gamma^5, 1, 0, \gamma^5, \right), \\ \mathcal{D}_{r,5}\left(x^5 + \gamma^{10}\right) &= \left(1, \gamma^{10}, 0, 1, \gamma^{10}, 0, 1, \gamma^{10}, 0, 1, \gamma^{10}, 0, 1, \gamma^{10}, 0\right). \end{aligned}$$

Therefore the algorithm outputs the list

$$\begin{aligned} L(r, 5) = \big\{&(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0), \\ &\left(1, 0, \gamma^5, 1, 0, \gamma^5, 1, 0, \gamma^5, 1, 0, \gamma^5, 1, 0, \gamma^5, \right), \\ &\left(1, \gamma^{10}, 0, 1, \gamma^{10}, 0, 1, \gamma^{10}, 0, 1, \gamma^{10}, 0, 1, \gamma^{10}, 0\right)\big\}. \end{aligned}$$

6.4 $[12, 7, 6]$ RS code

Another example that we can found in [9] is the following. Let C be the $[12, 7, 6]$ Reed Solomon code over $\mathbb{F}_{13}$. Consider 2 as a primitve element. Here the minimum distance is 6 and the error-correcting radius is 2. Suppose that we try to transmit the codeword $c = (0, 0, \ldots, 0)$, and the received word is

$$r = (1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0)\,.$$

In this work we have always dealt with codes that have odd minimum distance, but the procedure is the same. The syndromes vector is given by

$$(s_1, s_2, s_3, s_4, s_5) = (0, 0, 3, 0, 0),$$

and the 3-rd syndrome matrix is

$$S_3 = \begin{bmatrix} 0 & 0 & 3 & 0 \\ 0 & 3 & 0 & 0 \end{bmatrix}.$$

A basis for $\ker(S_3)$ is given by

$$\ker(S_3) = \langle x^3, 1\rangle.$$

By Euclidean division between $H(x) = x^{12} - 1$ and $F(a, x) = x^3 + a$ we obtain that the polynomials r_2 and r_1 of the divisibility ideal $\mathcal{I}$ equal zero, while $r_0(a) = a^4 - 1$. Hence

$$\mathcal{V}(\mathcal{I}) = \{1, 5, 8, 12\}$$

and

$$\Sigma(r, 3) = \left\{x^3 - 1, x^3 - 5, x^3 - 8, x^3 - 12\right\}.$$

Therefore

$$\begin{aligned} L(r, 3) =& \mathcal{D}_{r,3}\left(\Sigma(r, 3)\right) = \\ =& \{(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)\,, \\ & (1, 8, 0, 0, 1, 8, 0, 0, 1, 8, 0, 0)\,, \\ & (1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0)\,, \\ & (1, 0, 0, 5, 1, 0, 0, 5, 1, 0, 0, 5)\}\,. \end{aligned}$$

6.5 $[10, 2, 9]$ RS code

Here we present the third example that we can find in [9]. Let C be the $[10, 2, 9]$ Reed-Solomon code over $\mathbb{F}_{11}$, where we consider 2 as a primitive element. Here the error-correcting radius is 4. Suppose that a codeword c is transmitted and the received word is

$$r = (5, 3, 3, 4, 4, 9, 9, 1, 1, 5).$$

Suppose moreover that we don't know how many errors occur in r and we try to list all the codewords with distance at most 5 from it. Our first try is to find all the codewords with distance at most 5 from r. The syndrome vector of r is

$$(s_1, s_2, \ldots, s_8) = (0, 0, 2, 0, 0, 0, 0, 4)\,,$$

and the 5-th syndrome matrix is

$$S_5 = \begin{bmatrix} 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 & 0 & 4 \end{bmatrix}.$$

A leading basis for $\ker(S_5)$ is given by

$$\ker(S_5) = \langle x^5 + 5, x^2, x \rangle,$$

and the Euclidean division between $x^{10} - 1$ and $x^5 + ax^2 + bx + 5$ produces the divisibility ideal

$$\mathcal{I} = (2, -b, -a + b2, 2ab, a^2) = (1).$$

Hence the variety $\mathcal{V}(\mathcal{I})$ is empty and there are no codewords whose distance is at most 5 from r.

Our second attempt is to find all the codewords with distance at most 6 from r. The 6-th syndrome matrix is

$$S_6 = \begin{bmatrix} 0 & 0 & 2 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 & 4 \end{bmatrix},$$

and its kernel is given by

$$\ker(S_6) = \langle x^6, x^5 + 5, x^3, x^2, x \rangle.$$

Now we perform the Euclidean division between $x^{10} - 1$ and $x^6 + ax^5 + bx^3 + cx^2 + dx + 5a$. We obtain that the divisibility ideal is given by

$$\mathcal{I} = (d + 3, c^2 + 2bd, ac - 1, b^2 + 4a, ab - 2c, a^2 - 5bc),$$

whose variety is

$$\mathcal{V}(\mathcal{I}) = \{(8,2,8,6),(8,10,2,10),(8,8,7,7),(8,7,10,8),(8,6,6,2)\}\,.$$

Since we are working in the polynmial ring $\mathbb{F}_{11}[a,b,c,d]$, in this case we computed a Gröbner basis for $\mathcal{I}$ in order to find its variety. This variety yelds the following list of codewords

$$\begin{aligned} L(r,6) = \big\{&(10,3,0,5,4,2,9,1,7,8)\\ &(5,3,10,2,8,9,0,4,1,6)\\ &(2,7,6,4,0,3,9,10,1,5)\\ &(5,8,3,4,6,10,7,1,0,9)\\ &(0,1,3,7,4,9,8,6,2,5)\big\}, \end{aligned}$$

that are all the codewords with distance at most 6 from r.

Bibliography

[1] O. Brugia, and P. Filipponi, "*Polynomial Divisibility in Finite Fields and Recurring Sequences*", The Fibonacci Quarterly, Vol. 33:459–463, 1995.

[2] D. Cox, J. Little, and D. O'Shea, *Ideals, Varieties, and Algorithms: An Introduction to Computational Algebraic Geometry and Commutative Algebra*, Springer, New York, 1992.

[3] L. Caniglia, A. Galligo, and J. Heintz, "*Some new effectivity bounds in computational geometry*", Applied Algebra, Algebraic Algorithms and Error-Correcting Codes, 6:131–151, 1989.

[4] P. Elias. "*List decoding for noisy channels*", In 1957-IRE WESCON Convention Record, 2:94-104, 1957.

[5] O. Goldreich, R. Rubinfeld, and M. Sudan, "*Learning polynomials with queries: The highly noisy case*", in Proc. 36th Annu. IEEE Symp. Foundations of Computer Science, 294–303, 1995.

[6] V. Guruswami, *List Decoding of Error-Correcting Codes*, Lecture Notes in Computer Science 3282, Springer, 2004.

[7] V. Guruswami and M. Sudan, "*Improved decoding of Reed-Solomon codes and algebraic geometry codes*", IEEE Trans. Inform. Theory vol. 45, 6: 1757–1767, 1999.

[8] S. M. Johnson, "*A new upper bound for error-correcting codes*", IRE Trans. Inform. Theory, IT-8:203-207, 1962.

[9] J. Justesen and T. Høholdt, "*Bounds on list decoding of MDS codes*", IEEE Trans. Inform. Theory, vol. 47:1604–1609, 2001.

[10] J. Schwartz, "*Fast probabilistic algorithms for verification of polynomial identities*", J. ACM, 27(4):701-717, 1980.

[11] M. Sudan, "*Decoding of Reed-Solomon codes beyond the error-correction bound*", J. Compl., vol. 13:180–193, 1997.

[12] R. Zippel, "*Probabilistic algorithms for sparse polynomials*", Proc. EUROSAM '79, Springer Lec. Notes Comp. Sci. 72:216–226, 1979.

[13] J.H. van Lint, *Introduction to Coding Theory*, Springer, New York, 1982.