Applications of Groebner Bases to Coding Theory

MAT6
M.Sc. Thesis in Mathematics

Supervisors: Diego Ruano and Olav Geil

Nicola Marchetti

Aalborg University
Department of Mathematical Sciences
Fredrik Bajers Vej 7 G • 9220 Aalborg East • Denmark

AALBORG UNIVERSITET

Department of Mathematical Sciences
Fredrik Bajers Vej 7G
9220 Aalborg East
Telephone 99 40 88 04
Fax 98 15 81 29
http://www.math.aau.dk

Title: Applications of Groebner Bases to Coding Theory

Project period: MAT6, spring semester 2010

M.Sc. candidate:

Nicola Marchetti

Supervisors: Diego Ruano and Olav Geil
Pages number: 71
Finished: June 15, 2010.

Summary:

This MSc in Mathematics thesis deals with the application to coding theory of concepts related to Groebner bases theory.
We first recall the concepts of polynomials, ideals, monomial orders and polynomial division. Next, modules are introduces, discussing on the associated monomial orders and Groebner bases.
Thereafter, we recall the theory of cyclic codes, which are then used afterwards to build quasicyclic codes. In particular, we discuss linear codes, and then cyclic codes. For the latter, we focus on the problem of factorizing $x^n - 1$. Later on some basic theory is recalled, and thereafter one introduces the concept of zeros and minimum distance of a cyclic codes. At last, we treat BCH and Reed-Solomon codes.
We then introduce quasicyclic codes and the related algebraic theory. We will especially focus on studying quasicyclic codes's algebraic structure by using the tool of Groebner bases of modules. Finally, we deal with the decoding of quasicyclic codes in Groebner basis form, and with the decoding of restriction-1 1-generator quasicyclic codes. Further, some decoding algorithms for Reed-Solomon codes are discussed, and we then deal with the decoding of quasicyclic codes formed by blocks, constituted in turn by Reed-Solomon codes.

Preface

This thesis is the outcome of a project done during the 2010 spring MAT6 semester at Aalborg University.

The project is inspired by the discussions that took place in occasion of the weekly meetings with the supervisors, and the main reference material is listed in the Bibliography, at the end of the report.

Several examples have been put in the report, in order to better the understanding of the theoretical concepts, and in this respect the computer algebra system *Singular* has been used as a support.

The MSc in Mathematics candidate Nicola Marchetti would like to express his gratitude to the supervisors, Dr. Diego Ruano and Associate Prof. Olav Geil, for their constant support throughout all the duration of the project.

Contents

Chapter 1

Introduction

This MSc in Mathematics thesis deals with the application of concepts related to Groebner bases theory, studied during the former semester MAT5, to coding theory. This report is structured into four main chapters, i.e.:

- Chapter 2, *Modules*: we first recall the concepts of polynomials, ideals, monomial orders and polynomial division. Next, modules are introduces, discussing on the associated monomial orders and Groebner bases;
- Chapter 3, *Cyclic Codes*: this chapter is devoted to recalling the theory of cyclic codes, which are then used in Chapter 4 to build quasicyclic codes. In particular, we discuss linear codes, and then cyclic codes. For the latter, we focus on the problem of factorizing $x^n - 1$. Later on some basic theory is recalled, and thereafter one introduces the concept of zeros and minimum distance of a cyclic codes. At last, we treat BCH and Reed-Solomon codes;
- Chapter 4, *Quasicyclic Codes and their Algebraic Structure*: this chapter introduces quasicyclic codes and the related algebraic theory. The last part of the chapter focuses on studying quasicyclic codes's algebraic structure by using the tool of Groebner bases of modules;
- Chapter 5, *Decoding Issues related to Quasicyclic Codes*: the last chapter first deals with the decoding of quasicyclic codes in Groebner basis form, and with the decoding of restriction-1 1-generator quasicyclic codes. Later on, some decoding algorithms for Reed-Solomon codes are discussed, and the last section deals with the decoding of quasicyclic codes formed by blocks, constituted in turn by Reed-Solomon codes.

The main personal contributions of the thesis consist on:

- Re-elaboration of material from state-of-the-art, including further comments, proofs and examples when it was found appropriate;
- Use of *Singular* computer algebra system [2005 Greuel, Pfister & Schoenemann] throughout the report, presenting some original examples of application of the theoretical concepts which are the subject of this thesis.

Chapter 2

Modules

In this chapter, we will first recall some concepts on polynomials, ideals, monomial orders and polynomial division. We will then discuss modules over rings, and we will conclude the chapter dealing with monomial orders and Groebner bases for modules.

2.1 Polynomials and Ideals

Let us first define **monomials** and **polynomials**:

Definition 2.1. *A* **monomial** *in $x_1, \ldots, x_n$ is a product of the form*

$$x_1^{\alpha_1} \cdot x_2^{\alpha_2} \cdots x_n^{\alpha_n} \tag{2.1}$$

where all of the exponents $\alpha_1, \ldots, \alpha_n$ are nonnegative integers. The **total degree** *of this monomial is the sum $\alpha_1 + \cdots + \alpha_n$.*

We can simplify the notation for monomials as follows: let $\alpha = (\alpha_1, \ldots, \alpha_n)$ be an n-tuple of nonnegative integers. Then we set

$$x^\alpha = x_1^{\alpha_1} \cdot x_2^{\alpha_2} \cdots x_n^{\alpha_n}$$

We let $|\alpha| = \alpha_1 + \cdots + \alpha_n$ denote the total degree of the monomial x^α.

Definition 2.2. *A* **polynomial** *f in $x_1, \ldots, x_n$ with coefficients in a field k is a finite linear combination (with coefficients in k) of monomials. We will write a polynomial in the form*

$$f = \sum_\alpha a_\alpha x^\alpha, \quad a_\alpha \in k \tag{2.2}$$

where the sum is over a finite number of n-tuples $\alpha = (\alpha_1, \ldots, \alpha_n) \in \mathbb{Z}_{\geq 0}^n$. The set of all polynomials in $x_1, \ldots, x_n$ with coefficients in k is denoted $k[x_1, \ldots, x_n]$.

In the following definition, we introduce a terminology which we use when dealing with polynomials:

Definition 2.3. *Let $f = \sum_\alpha a_\alpha x^\alpha$ be a polynomial in $k[x_1, \ldots, x_n]$.*

i. We call a_α the **coefficient** *of the monomial x^α.*

ii. If $a_\alpha \neq 0$, then we call $a_\alpha x^\alpha$ a **term** *of f.*

iii. The **total degree** *of f, denoted $\deg(f)$, is the maximum $|\alpha|$ such that the coefficient a_α is nonzero.*

For example, $x^3y + x^2 + y$ is a polynomial in $\mathbb{F}_2[x, y]$, with three terms and total degree four.

One can show that, under addition and multiplication, $k[x_1, \ldots, x_n]$ satisfies all the field axioms, except for the existence of multiplicative inverses, e.g. there is no multiplicative inverse for x_1, as $1/x_1$ is not a polynomial. Such a mathematical structure is called a commutative ring, and that is why we usually refer to $k[x_1, \ldots, x_n]$ as a *polynomial ring*.

Let us define the following algebraic object:

Definition 2.4. *A subset $I \subset k[x_1, \ldots, x_n]$ is an* **ideal** *if it satisfies:*

i. $0 \in I$

ii. if $f,g \in I$, then $f+g \in I$

iii. if $f \in I$ and $h \in k[x_1, \ldots, x_n]$, then $hf \in I$

Let us introduce a first example of an ideal:

Definition 2.5. *Let $f_1, \ldots, f_s$ be polynomials in $k[x_1, \ldots, x_n]$. We set:*

$$< f_1, \ldots, f_s >= \left\{ \sum_{i=1}^{s} h_i f_i : \ h_1, \ldots, h_s \in \ k[x_1, \ldots, x_n] \right\} \tag{2.3}$$

Let us introduce the concept of *ideal generated by $f_1, \ldots, f_s$*:

Lemma 2.1. *If $f_1, \ldots, f_s \in k[x_1, \ldots, x_n]$, then $\langle f_1, \ldots, f_s \rangle$ is an ideal of $k[x_1, \ldots, x_n]$. One will call $\langle f_1, \ldots, f_s \rangle$ the* **ideal generated by** *$f_1, \ldots, f_s$.*

2.2 Monomial Orders and Polynomial Division

Let us first note that we can reconstruct the monomial $x^\alpha = x_1^{\alpha_1} \cdots x_n^{\alpha_n}$ from the n-tuple of exponents $\alpha = (\alpha_1, \ldots, \alpha_n) \in \mathbb{Z}_{\geq 0}^n$. We then have a one-to-one correspondence between the monomials in $k[x_1, \ldots, x_n]$ and $\mathbb{Z}_{\geq 0}^n$. Moreover, any ordering $>$ we establish on the space $\mathbb{Z}_{\geq 0}^n$

will give us an ordering on monomials: if $\alpha > \beta$ according to this ordering, we will also say that $x^\alpha > x^\beta$.

Since a polynomial is a sum of monomials, we would like to be able to arrange the terms in a polynomial in a certain order, either ascending or descending, in an unambiguous way. In order to be able to do this, we must be able to compare every pair of monomials to establish their proper relative positions; by doing so, we say we require our orderings to be *linear* or *total* orderings, i.e. for every pair of monomials x^α and x^β, just one of the following statements should be true:

$$x^\alpha > x^\beta, \quad x^\alpha = x^\beta, \quad x^\alpha < x^\beta \tag{2.4}$$

Concerning the ordering, the sum operation on polynomials presents no problems, while the multiplication is a bit more tricky, as the leading term in the product could be different from the product of the monomial and the leading term of the original polynomial (*remark*: we will introduce formally the concept of leading term in a while, basically we will define it as the biggest monomial, with associated coefficient, according to a certain ordering we assume). We will then require that all monomial orderings have the following additional property: if $x^\alpha > x^\beta$ and x^γ is any monomial, then we require that $x^\alpha x^\gamma > x^\beta x^\gamma$. In terms of the exponent vectors, this means that if $\alpha > \beta$ in our ordering on $\mathbb{Z}^n_{\geq 0}$, then, for all $\gamma \in \mathbb{Z}^n_{\geq 0}$, we have that $\alpha + \gamma > \beta + \gamma$.

With the above in mind, we introduce the following

Definition 2.6. *A* **monomial ordering** $>$ *on* $k[x_1, \ldots, x_n]$ *is any relation* $>$ *on* $\mathbb{Z}^n_{\geq 0}$*, or equivalently, any relation on the set of monomials* x^α*,* $\alpha \in \mathbb{Z}^n_{\geq 0}$*, satisfying:*

i. $>$ *is a total ordering on* $\mathbb{Z}^n_{\geq 0}$*.*

ii. If $\alpha > \beta$ *and* $\gamma \in \mathbb{Z}^n_{\geq 0}$*, then* $\alpha + \gamma > \beta + \gamma$*.*

iii. $>$ *is a well-ordering on* $\mathbb{Z}^n_{\geq 0}$*. This means that every nonempty subset of* $\mathbb{Z}^n_{\geq 0}$ *has a smallest element under* $>$*.*

We will now introduce three examples of monomial orderings, namely the *lexicographic order*, the *graded lex order*, and the *graded reverse lex order*.

Definition 2.7. (**Lexicographic Order**) *Let* $\alpha = (\alpha_1, \ldots, \alpha_n), \beta = (\beta_1, \ldots, \beta_n) \in \mathbb{Z}^n_{\geq 0}$*. We say* $\alpha >_{lex} \beta$ *if, in the vector difference* $\alpha - \beta \in \mathbb{Z}^n$*, the leftmost nonzero entry is positive. We will write* $x^\alpha >_{lex} x^\beta$ *if* $\alpha >_{lex} \beta$*.*

In practice, when we work with polynomials in up to three variables, we call the variables x, y, z instead of x_1, x_2, x_3. We once again assume, unless differently stated, that the alphabetical order $x > y > z$ on the variables is used to define the lexicographic ordering. As a consequence, when dealing with n variables, we have that $x_1 > \cdots > x_n$.

Definition 2.8. (**Graded Lex Order**) *Let* $\alpha, \beta \in \mathbb{Z}^n_{\geq 0}$*. We say* $\alpha >_{grlex} \beta$ *if*

$$|\alpha| = \sum_{i=1}^{n} \alpha_i > |\beta| = \sum_{i=1}^{n} \beta_i, \quad \text{or } |\alpha| = |\beta| \text{ and } \alpha >_{lex} \beta \tag{2.5}$$

Definition 2.9. (**Graded Reverse Lex Order**) *Let $\alpha, \beta \in \mathbb{Z}^n_{\geq 0}$. We say $\alpha >_{grevlex} \beta$ if*

$$|\alpha| = \sum_{i=1}^{n} \alpha_i > |\beta| = \sum_{i=1}^{n} \beta_i, \quad \text{or } |\alpha| = |\beta| \text{ and the rightmost nonzero entry of } \alpha - \beta \in \mathbb{Z}^n \text{ is negative} \tag{2.6}$$

Let us introduce the following definition:

Definition 2.10. *Let $f = \sum_\alpha a_\alpha x^\alpha$ be a nonzero polynomial in $k[x_1, \ldots, x_n]$ and let $>$ be a monomial order.*

i. The **multidegree** *of f is*

$$\text{multideg}(f) = \max(\alpha \in \mathbb{Z}^n_{\geq 0} : a_\alpha \neq 0) \tag{2.7}$$

(the maximum is taken with respect to $>$).

ii. The **leading coefficient** *of f is*

$$\text{LC}(f) = a_{\text{multideg}(f)} \in k \tag{2.8}$$

iii. The **leading monomial** *of f is*

$$\text{LM}(f) = x^{\text{multideg}(f)} \tag{2.9}$$

(with coefficient 1).

iv. The **leading term** *of f is*

$$\text{LT}(f) = \text{LC}(f) \cdot \text{LM}(f) \tag{2.10}$$

2.3 Modules over Rings

Modules are to rings what vector spaces are to fields. A geometric structure on a variety often corresponds algebraically to a module or an element of a module over the coordinate ring of the variety.

If R is a commutative ring with identity, an R-module is defined as follows.

Definition 2.11. *A* **module over a ring R** *(or* **R-module***) is a set M together with a binary operation, usually written as addition, and an operation of R on M, called (scalar) multiplication, satisfying the following properties:*

i. M is an abelian group under addition. That is, addition in M is associative and commutative, there is an additive identity element $\mathbf{0} \in M$, and each element $\mathbf{f} \in M$ has an additive inverse $-\mathbf{f}$ satisfying $\mathbf{f} + (-\mathbf{f}) = \mathbf{0}$.

ii. For all $a \in R$ and all $\mathbf{f}, \mathbf{g} \in M$, $a(\mathbf{f} + \mathbf{g}) = a\mathbf{f} + a\mathbf{g}$.

iii. For all $a, b \in R$ and all $\mathbf{f} \in M$, $(a+b)\mathbf{f} = a\mathbf{f} + b\mathbf{f}$.

iv. For all $a, b \in R$ and all $\mathbf{f} \in M$, $(ab)\mathbf{f} = a(b\mathbf{f})$.

v. If 1 is the multiplicative identity in R, $1\mathbf{f} = \mathbf{f}$ for all $\mathbf{f} \in M$.

The simplest modules are those consisting of all $m \times 1$ columns of elements of R with componentwise addition and scalar multiplication:

$$\begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_m \end{pmatrix} + \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{pmatrix} = \begin{pmatrix} a_1 + b_1 \\ a_2 + b_2 \\ \vdots \\ a_m + b_m \end{pmatrix} \tag{2.11}$$

$$c \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_m \end{pmatrix} = \begin{pmatrix} ca_1 \\ ca_2 \\ \vdots \\ ca_m \end{pmatrix} \tag{2.12}$$

for any $a_1, \ldots, a_m, b_1, \ldots, b_m, c \in R$. We call such column a *vector* and the set of all such R^m.

We can obtain other example of R-modules by considering *submodules* of R^m, i.e., subsets of R^m which are closed under addition and scalar multiplication by elements of R and which are then modules in their own right.

As an example, we can consider the set of all column vectors which can be written as an R-linear combination of a finite set of vectors $\mathbf{f}_1, \ldots, \mathbf{f}_s$:

$$\{a_1\mathbf{f}_1 + \cdots + a_s\mathbf{f}_s \in R^m, \text{ where } a_1, \ldots, a_s \in R\} \tag{2.13}$$

This particular R-module is denoted $\langle \mathbf{f}_1, \ldots, \mathbf{f}_s \rangle$. We will now prove that:

Theorem 2.1. *We have that $M = \langle \mathbf{f}_1, \ldots, \mathbf{f}_s \rangle$, as defined in Eq. (2.13), is an R-module.*

Proof. Let us start by proving property (i) of Def. 2.11. The addition is associative, in fact:

$$\left(\sum_{i=1}^{s} a_i\mathbf{f}_i + \sum_{i=1}^{s} b_i\mathbf{f}_i\right) + \sum_{i=1}^{s} c_i\mathbf{f}_i = \sum_{i=1}^{s} a_i\mathbf{f}_i + \left(\sum_{i=1}^{s} b_i\mathbf{f}_i + \sum_{i=1}^{s} c_i\mathbf{f}_i\right)$$

by repeatedly applying component-wise the addition associativity for rings. Similarly, by repeatedly applying component-wise the addition commutativity for rings, we obtain that:

$$\sum_{i=1}^{s} a_i\mathbf{f}_i + \sum_{i=1}^{s} b_i\mathbf{f}_i = \sum_{i=1}^{s} b_i\mathbf{f}_i + \sum_{i=1}^{s} a_i\mathbf{f}_i$$

Moreover, there is an additive element $\mathbf{0} \in M$, being that element $\mathbf{0} = \sum_{i=1}^{s} 0\mathbf{f}_i$, where 0 is additive identity element for R, and each element $\mathbf{f} \in M$ has an additive inverse $-\mathbf{f}$ satisfying $\mathbf{f} + (-\mathbf{f}) = 0$, being that inverse for $\sum_{i=1}^{s} a_i\mathbf{f}_i$, equal to $\sum_{i=1}^{s} (-a_i)\mathbf{f}_i$, where $-a_i$ are the additive inverses of the a_i in R, and we repeatedly apply this component-wise.

As per property (ii) of Def. 2.11, we have that:

$$c(\sum_{i=1}^{s} a_i\mathbf{f}_i + \sum_{i=1}^{s} b_i\mathbf{f}_i) = c\sum_{i=1}^{s} a_i\mathbf{f}_i + c\sum_{i=1}^{s} b_i\mathbf{f}_i$$

by repeatedly applying component-wise the distributivity of multiplication with respect to addition for rings.

Property (iii) of Def. 2.11 follows from repeatedly applying component-wise multiplication commutativity for commutative rings and property (ii) of Def. 2.11.

Property (iv) of Def. 2.11 follows from repeatedly applying component-wise multiplication associativity for rings.

If 1 is the multiplicative identity in R, property (v) of Def. 2.11 follows from Eq. (2.12). □

Submodules of R^m when R is a polynomial ring can exhibit a behavior that is quite different from vector spaces, as it will be shown in the following example.

Example 2.1

Let us consider $R = k[x, y, z]$. Let $M \subset R^3$ be the module $\langle \mathbf{f}_1, \mathbf{f}_2, \mathbf{f}_3 \rangle$ where

$$\mathbf{f}_1 = \begin{pmatrix} y \\ -x \\ 0 \end{pmatrix},\ \mathbf{f}_2 = \begin{pmatrix} z \\ 0 \\ -x \end{pmatrix},\ \mathbf{f}_3 = \begin{pmatrix} 0 \\ z \\ -y \end{pmatrix} \tag{2.14}$$

The set $\{\mathbf{f}_1, \mathbf{f}_2, \mathbf{f}_3\}$ is minimal, in the sense that $M \neq \langle \mathbf{f}_i, \mathbf{f}_j \rangle,\ 1 \leq i < j \leq 3$. Indeed, considering $\langle \mathbf{f}_i, \mathbf{f}_j \rangle = \langle \mathbf{f}_1, \mathbf{f}_2 \rangle$ we have that:

$$\langle \mathbf{f}_1, \mathbf{f}_2, \mathbf{f}_3 \rangle = \langle \mathbf{f}_1, \mathbf{f}_2 \rangle \tag{2.15}$$

$$a_1\mathbf{f}_1 + a_2\mathbf{f}_2 + a_3\mathbf{f}_3 = a_4\mathbf{f}_1 + a_5\mathbf{f}_2 \tag{2.16}$$

$$a_1 \begin{pmatrix} y \\ -x \\ 0 \end{pmatrix} + a_2 \begin{pmatrix} z \\ 0 \\ -x \end{pmatrix} + a_3 \begin{pmatrix} 0 \\ z \\ -y \end{pmatrix} = a_4 \begin{pmatrix} y \\ -x \\ 0 \end{pmatrix} + a_5 \begin{pmatrix} z \\ 0 \\ -x \end{pmatrix} \tag{2.17}$$

$$\begin{pmatrix} a_1y + a_2z \\ -a_1x + a_3z \\ -a_2x - a_3y \end{pmatrix} = \begin{pmatrix} a_4y + a_5z \\ -a_4x \\ -a_5x \end{pmatrix} \tag{2.18}$$

which is verified if

$$a_1 = a_4 \tag{2.19}$$

$$a_2 = a_5 \tag{2.20}$$

$$a_3 = 0 \tag{2.21}$$

from which we see that the only way to get M by $\langle \mathbf{f}_1, \mathbf{f}_2 \rangle$, is to have $a_3 = 0$. Analogously we can prove that $M \neq \langle \mathbf{f}_1, \mathbf{f}_3 \rangle$ and that $M \neq \langle \mathbf{f}_2, \mathbf{f}_3 \rangle$.

The set $\{\mathbf{f}_1, \mathbf{f}_2, \mathbf{f}_3\}$ is R-linearly dependent, i.e. there exist $a_1, a_2, a_3 \in R = k[x, y, z]$, not all zero, such that $a_1\mathbf{f}_1 + a_2\mathbf{f}_2 + a_3\mathbf{f}_3 = \mathbf{0}$, where $\mathbf{0}$ is the zero vector in R^3. Indeed by solving

$$a_1 \begin{pmatrix} y \\ -x \\ 0 \end{pmatrix} + a_2 \begin{pmatrix} z \\ 0 \\ -x \end{pmatrix} + a_3 \begin{pmatrix} 0 \\ z \\ -y \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \tag{2.22}$$

i.e.

$$\begin{aligned} a_1y + a_2z &= 0 \quad (2.23)\\ -a_1x + a_3z &= 0 \quad (2.24)\\ -a_2x - a_3y &= 0 \quad (2.25) \end{aligned}$$

we get that any triple $(\frac{z}{x}a_3, -\frac{y}{x}a_3, a_3)^T$ is solving Eq. (2.22), and not only $(0,0,0)^T$.

The calculations above show that this is an example of a submodule of $k[x,y,z]^3$ in which there is a minimal generating set which is not linearly independent. *This phenomenon cannot occur in any vector space.*

Part of the reason why the concept of a *module* is very useful is that it simultaneously generalize the concepts of *ideal* and *quotient ring*. This is confirmed by the following result:

Theorem 2.2. *An ideal $I \subset R$ is an R-module, and every module $M \subset R$ is an ideal.*

After defining any algebraic structure, one usually defines maps that preserve that structure. Accordingly, we now define module homomorphisms, the analogues of linear mappings between vector spaces.

Definition 2.12. *An* **R-module homomorphism** *between two R-modules M and N is an R-linear map between M and N, i.e. a map $\varphi: M \to N$ is an R-module homomorphism if for all $a \in R$ and all $\mathbf{f}, \mathbf{g} \in M$, we have*

$$\varphi(a\mathbf{f} + \mathbf{g}) = a\varphi(\mathbf{f}) + \varphi(\mathbf{g})$$

The definitions of kernel and image carry over homomorphisms from rings to modules:

Definition 2.13. *If $\varphi: M \to N$ is an R-module homomorphism between two R-modules M and N, define the* **kernel** *of φ, denoted* $\ker(\varphi)$*, to be the set*

$$\ker(\varphi) = \{\mathbf{f} \in M : \ \varphi(\mathbf{f}) = 0\}$$

and the **image** *of φ, denoted* $\mathrm{im}(\varphi)$*, to be the set*

$$\mathrm{im}(\varphi) = \{\mathbf{g} \in N : \ \text{there exists } \mathbf{f} \in M \text{ with } \varphi(\mathbf{f}) = \mathbf{g}\}$$

The homomorphism φ is said to be an **isomorphism** *if it is both one-to-one and onto, and two R-modules M, N are called* **isomorphic**, *written $M \cong N$ if there is some isomorphism $\varphi: M \to N$.*

When we introduce the notions of linear combinations and linear independence and R is not a field, e.g. $R = k[x_1, \ldots, x_n]$, the theory of modules begins to develop a different flavor from the theory of vector spaces. As in linear algebra, we have the following definition:

Definition 2.14. *A subset $F = \{\mathbf{f}_1, \ldots, \mathbf{f}_n\}$ of a module M is* **linearly independent** *over R (or R-linearly independent) if the only linear combination $a_1\mathbf{f}_1 + \cdots + a_n\mathbf{f}_n$ with $a_i \in R$ and $\mathbf{f}_i \in F$ which equals $\mathbf{0} \in M$ is the trivial one in which $a_1 = \cdots = a_n = 0$. A set $F \subset M$ which is R-linearly independent and which spans M is said to be a* **basis** *for M.*

While every vector space over a field has a basis, not every module has one.

Example 2.2

Let us denote by $\langle F \rangle$ the submodule generated by a set F. If $\langle F \rangle = M$, we say that F *spans* (or *generates*) M. Let us consider the ideal $M = \langle x, y \rangle \subset R = k[x, y]$, which is the same as the R-module generated by x and y in R. The set $\{x, y\}$ is not a basis for M as a module because x and y are not linearly independent. For example, there is a linear dependence relation $y \cdot x - x \cdot y = 0$, but the coefficients y and x are not 0. On the other hand, since $\{x, y\}$ spans M, it is a basis for M as an ideal. Thus the meaning of the word "basis" depends on the context.

The following proposition gives a characterization of module bases:

Proposition 2.1. *Let M be a module over a ring R. A set $F \subset M$ is a module basis for M if and only if every $\mathbf{f} \in M$ can be written in one and only one way as a linear combination*

$$\mathbf{f} = a_1\mathbf{f}_1 + \cdots + a_n\mathbf{f}_n$$

where $a_i \in R$ and $\mathbf{f}_i \in F$.

Unlike vector spaces, modules need not have any generating set which is linearly independent. Those that do are given a special name:

Definition 2.15. *Let M be a module over a ring R. M is said to be a* **free module** *if M has a module basis (i.e., a generating set that is R-linearly independent).*

Example 2.3

The R-module $M = R^m$ is a free module. The standard basis elements

$$\mathbf{e}_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \ \mathbf{e}_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \ \ldots, \ \mathbf{e}_m = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 1 \end{pmatrix}$$

form one basis for M.

On the other hand, an example on non-free module is given by any R-module (ideal) $M \subset R = k[x_1, \ldots, x_n]$ which requires more than a single polynomial to generate it. Indeed this kind of modules cannot be generated by an R-linearly independent set, since any pair of polynomials $f_1, f_2 \in R$ that might appear in a generating set satisfies a non-trivial linear dependence relation $f_2 f_1 - f_1 f_2 = 0$ with nonzero coefficients $f_2, \ -f_1$ in R.

The fact that some modules do not have bases raises the issue of how to explicitly handle the computations in modules. Let us first note that one not only needs a generating set, but also the set of all relations satisfied by the generators, otherwise we have no way of knowing in general if two elements expressed in terms of the generators are equal or not.

Let us spend a few more words about relations. Suppose that $F = (\mathbf{f}_1, \ldots, \mathbf{f}_t)$ is an ordered t-tuple of elements of some R-module M, so that $\mathbf{f}_1, \ldots, \mathbf{f}_t \in M$. Then a relation on F is an R-linear combination of the $\mathbf{f}_i$ which is equal to $\mathbf{0}$:

$$a_1\mathbf{f}_1 + \cdots + a_t\mathbf{f}_t = \mathbf{0} \in M$$

We think of a relation on F as a t-tuple $(a_1, \ldots, a_t)$ of elements of R, or equivalently as an element of R^t. Such relations are called *syzygies*. We have the following result:

Proposition 2.2. *Let $(\mathbf{f}_1, \ldots, \mathbf{f}_t)$ be an ordered t-tuple of elements $f_i \in M$. The set of all $(a_1, \ldots, a_t)^T \in R^t$ such that $a_1\mathbf{f}_1 + \cdots + a_t\mathbf{f}_t = 0$ is an R-submodule of R^t, called the* **(first) syzygy module** *of $(\mathbf{f}_1, \ldots, \mathbf{f}_t)$, and denoted $\mathrm{Syz}(\mathbf{f}_1, \ldots, \mathbf{f}_t)$.*

Prop. 2.2 allows us to be precise about what it means to know all relations on a fixed set of generators of a module. If there are t generators, then the set of relations is just a submodule of R^t. Hence, we know all relations on a set of generators of a module if we can find a set of generators for the first syzygy module.

Since we think of elements of R^t as column vectors, we can think of a finite collection of syzygies as columns of a matrix. If M is a module spanned by the t generators $\mathbf{f}_1, \ldots, \mathbf{f}_t$, then a *presentation matrix* for M is any matrix whose columns generate $\mathrm{Syz}(\mathbf{f}_1, \ldots, \mathbf{f}_t) \subset R^t$.

Example 2.4

The fact that some modules do not have bases and the fact that even when they do have, we may not be able to find them, raised the question of how to explicitly handle computations in modules. We already mentioned that we do not only need a generating set, but also the set of all relations satisfied by the generators.

For instance, suppose we know that M is a $\mathbb{Q}[x, y]$-module and that $\mathbf{f}_1,\ \mathbf{f}_2,\ \mathbf{f}_3$ is a generating set. If we want to know whether $2\mathbf{f}_1 + 3\mathbf{f}_2 + 3y\mathbf{f}_3$ and $\mathbf{f}_1 + 2\mathbf{f}_2 + 4y\mathbf{f}_3$ represent the same element, then we have to verify if the difference, $\mathbf{f}_1 + \mathbf{f}_2 - y\mathbf{f}_3$, equals zero in M.

If we knew that every relation on the $\mathbf{f}_1,\ \mathbf{f}_2,\ \mathbf{f}_3$ was a $\mathbb{Q}[x, y]$-linear combination of the relations $3\mathbf{f}_1 + (x-1)\mathbf{f}_2 = 0$, $2\mathbf{f}_1 + (x + 2y - 2)\mathbf{f}_2 - y\mathbf{f}_3 = 0$, and $\mathbf{f}_2 - \mathbf{f}_3 = 0$, then we can settle the problem provided we can decide whether $\mathbf{f}_1 + \mathbf{f}_2 - y\mathbf{f}_3 = 0$ is a $\mathbb{Q}[x, y]$-linear combination of the given relations, which it is, indeed, since:

$$3\mathbf{f}_1 + (x-1)\mathbf{f}_2 = 0 \tag{2.26}$$
$$2\mathbf{f}_1 + (x + 2y - 2)\mathbf{f}_2 - y\mathbf{f}_3 = 0 \tag{2.27}$$
$$\mathbf{f}_2 - \mathbf{f}_3 = 0 \tag{2.28}$$

if we sum together Eq. (2.26), Eq. (2.27) multiplied by -1, and Eq. (2.28) multiplied by $2y$, we obtain:

$$\mathbf{f}_1 + \mathbf{f}_2 - y\mathbf{f}_3 = 0$$

If A is a presentation matrix for a module M with respect to some generating set of M, then we say that A *presents* the module M. Note that the number of rows of A is equal to the number of generators in the generating set of M.

The presentation matrix of a module M is not unique. It depends on the set of generators $\mathbf{f}_1, \ldots, \mathbf{f}_t$ that one chooses for M, and the set of elements $a_1, \ldots, a_t$ that one chooses to span the module of syzygies on the chosen set of generators of M.

The importance of presentation matrices is that once we have a presentation matrix A for a module M, we have a concrete set of generators and relations for M (actually for an isomorphic copy of M), and so we can work concretely with M.

We conclude this section by making some considerations about *finitely generated* and *non-finitely generated* modules. We say that M is finitely generated, if there is a finite set that generates M.

Example 2.5

$M = k[x]$ as a module over $R = k$ is not finitely generated, in fact one generating set to generate a polynomial of any degree is e.g. $\{1, x, x^2, x^3, \ldots\}$. As an example, to generate $1 + x + x^2 + x^3$, we need to do $1 \cdot 1 + 1 \cdot x + 1 \cdot x^2 + 1 \cdot x^3$, where $1 \in R = k$.

Instead, $M = k[x]$ as a module over $R = k[x]$ is finitely generated, indeed one generating set to generate a polynomial of any degree is e.g. $\{1\}$. Considering the same polynomial as before, i.e. $1 + x + x^2 + x^3$, we need to do $(1 + x + x^2 + x^3) \cdot 1$, where $1 + x + x^2 + x^3 \in R = k[x]$.

2.4 Monomial Orders and Groebner Bases for Modules

In this section, R will stand for a polynomial ring $k[x_1, \ldots, x_n]$. Here we will shortly introduce a theory of monomial orders in the free modules R^m and introduce Groebner bases for submodules $M \subset R^m$. We will see in a while that, once we introduce the terminology needed to extend the notion of monomial orders to the free modules R^m, the module case follows the ideal case almost exactly.

Let $\mathbf{e}_i$ be the standard basis vector in R^m, i.e. the column vector with a 1 in the i-th row and a 0 in all other rows. A *monomial* $\mathbf{m}$ in R^m is an element of the form $x^\alpha \mathbf{e}_i$ for some i. Every element $\mathbf{f} \in R^m$ can be written as a k-linear combination of monomials $\mathbf{m}_i$

$$\mathbf{f} = \sum_{i=1}^{m} c_i \mathbf{m}_i$$

where $c_i \in k,\ c_i \neq 0$.

Example 2.6

For example, in $k[x, y]^3$

$$\begin{aligned}
\mathbf{f} &= \begin{pmatrix} -2x^3y + 1 \\ x^2 + 2xy \\ -y \end{pmatrix} \\
&= (-2)\begin{pmatrix} x^3y \\ 0 \\ 0 \end{pmatrix} + 1\begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} + 1\begin{pmatrix} 0 \\ x^2 \\ 0 \end{pmatrix} + 2\begin{pmatrix} 0 \\ xy \\ 0 \end{pmatrix} + (-1)\begin{pmatrix} 0 \\ 0 \\ y \end{pmatrix} \\
&= -2x^3y\mathbf{e}_1 + \mathbf{e}_1 + x^2\mathbf{e}_2 + 2xy\mathbf{e}_2 - y\mathbf{e}_3
\end{aligned}$$

If $\mathbf{m}$, $\mathbf{n}$ are monomials in R^m, $\mathbf{m} = x^\alpha \mathbf{e}_i$, $\mathbf{n} = x^\beta \mathbf{e}_j$, then we say that $\mathbf{n}$ divides $\mathbf{m}$ if and only if $i = j$ and x^β divides x^α.

If $\mathbf{m}$ and $\mathbf{n}$ are monomials containing the same basis element $\mathbf{e}_i$, we define the greatest common divisor, $\mathrm{GCD}(\mathbf{m}, \mathbf{n})$, and least common multiple, $\mathrm{LCM}(\mathbf{m}, \mathbf{n})$ to be the greatest common divisor and least common multiple, respectively, of x^α and x^β, times $\mathbf{e}_i$. On the other hand, if $\mathbf{m}$, $\mathbf{n}$ contain different standard basis vectors, we define $\mathrm{LCM}(\mathbf{m}, \mathbf{n}) = \mathbf{0}$.

We say that a submodule $M \subset R^m$ is a *monomial submodule* if M can be generated by a collection of monomials. As for monomial ideals, $\mathbf{f}$ is in a monomial submodule M if and only if every term belonging to $\mathbf{f}$ is in M. Monomial submodules have properties that are analogous to those of monomial ideals, as we can see from the following proposition:

Proposition 2.3. *The following holds:*

i. Every monomial submodule of R^m is generated by a finite collection of monomials.

ii. **Ascending Chain Condition (ACC)**. *Every infinite ascending chain $M_1 \subset M_2 \subset \cdots$ of monomial submodules of R^m stabilizes. That is, there exists N such that $M_N = M_{N+1} = \cdots = M_{N+l} = \cdots$ for all $l \geq 0$.*

iii. Let $\{\mathbf{m}_1, \ldots, \mathbf{m}_t\}$ be a set of monomial generators for a monomial submodule of R^m, and let $\epsilon_1, \ldots, \epsilon_t$ denote the standard basis vectors in R^t. Let $\mathbf{m}_{ij} = \mathrm{LCM}(\mathbf{m}_i, \mathbf{m}_j)$. The syzygy module $\mathrm{Syz}(\mathbf{m}_i, \ldots, \mathbf{m}_t)$ is generated by the syzygies $\sigma_{ij} = (\mathbf{m}_{ij}/\mathbf{m}_i)\epsilon_i - (\mathbf{m}_{ij}/\mathbf{m}_j)\epsilon_j$, for all $1 \leq i < j \leq t$.

Proof. (ii): We know from Th. 2.2 that if a subset $M \subset R$ is a module over R, then M is an ideal in R, and we can therefore prove the ACC considering ideals.

Given the ascending chain $I_1 \subset I_2 \subset I_2 \subset I_3 \subset \cdots$, consider the set $I = \bigcup_{i=1}^{\infty} I_i$. We first show that I is also an ideal in R (we have to prove that the infinite sum of ideals is still an ideal, as the fact that a property is valid for a finite number of elements does not imply in general that it is still true for an infinite number of elements).

We have that $0 \in I$, as $0 \in I_i$ for all i (from the definition of ideal). Moreover, if $f, g \in I$, by definition we have that $f \in I_i$, $g \in I_j$ for some i and j, possibly different. However, since the ideals I_i form an ascending chain, relabeling such that $i \leq j$, we will have that both $f, g \in I_j$, but since I_j is an ideal, then the sum $f + g \in I_j$ too, and hence $f + g \in I$.

Finally, if $f \in I$ and $r \in R$, then by the definition of I, $f \in I_i$ for some i, and $r \cdot f \in I_i$, by one of the properties of an ideal, as I_i is an ideal. But as $I_i \subset I$, then we have $r \cdot f \in I_i \subset I$. We have then proved that I is an ideal.

By the Hilbert Basis Theorem, the ideal I must have a finite generating set: $I = \langle f_1, \ldots, f_s \rangle$. But each of the generators is contained in some one of the I_j, say $f_i \in I_{j_i}$, for some j_i, $i = 1, \ldots, s$. Let us take N to be the maximum of the j_i. Then by definition of ascending chain, we have that $f_i \in I_N$ for all i, but then

$$I = \langle f_1, \ldots, f_s \rangle \subset I_N \subset I_{N+1} \subset \cdots \subset I$$

where the ascending chain condition stabilizes with I_N, since all the subsequent ideals in the chain are equal, as by having $I \subset I_N \subset I_{N+1} \subset \cdots \subset I$, then $I_N = I_{N+1} = \cdots$. □

Extending the theory of Groebner bases to modules will involve three things:

- defining orders on the monomials of R^m;
- constructing a division algorithm on elements of R^m;
- extending the Buchberger's algorithm to R^m.

The definition of a monomial order on R^m is the same as the definition in R, namely:

Definition 2.16. *An ordering relation $>$ on the monomials of R^m is a* **monomial ordering** *if:*

i. $>$ *is a total order;*

ii. for every pair of monomials $\mathbf{m},\ \mathbf{n} \in R^m$ *with* $\mathbf{m} > \mathbf{n}$*, we have* $x^\alpha \mathbf{m} > x^\alpha \mathbf{n}$ *for every monomial* $x^\alpha \in R$*;*

iii. $>$ *is a well-ordering.*

Some of the most useful monomial orders on R^m come by extending monomial orders on R itself. We will introduce in a while two particularly natural ways to do this. First we have to choose an ordering on the standard basis vectors, and we choose to use the "downward" ordering on the entries in a column (although we could have chosen any other ordering):

$$\mathbf{e}_1 > \mathbf{e}_2 > \cdots > \mathbf{e}_m$$

Let us now introduce the following definition:

Definition 2.17. *Let $>$ be any monomial order on R.*

i. (TOP extension of $>$). We say $x^\alpha \mathbf{e}_i >_{TOP} x^\beta \mathbf{e}_j$ if $x^\alpha > x^\beta$, or if $x^\alpha = x^\beta$ and $i < j$.

ii. (POT extension of $>$). We say $x^\alpha \mathbf{e}_i >_{POT} x^\beta \mathbf{e}_j$ if $i < j$, or if $i = j$ and $x^\alpha > x^\beta$.

In Def. 2.17, TOP stands for "Term Over Position", meaning that a TOP order sorts monomials first by term order on R, then breaks ties using the position within the vector in R^m. POT order works the other way around, from which the name "Position Over Term".

Example 2.7

Let us now use Singular to make some considerations on the vector $\mathbf{f}$ from Example 2.6 and on monomial orders. We consider the field $k = \mathbb{Q}$. Let us consider an example with the TOP extension of lexicographic order, and the "upward" ordering $\mathbf{e}_1 < \mathbf{e}_2 < \mathbf{e}_3$ on the standard basis elements of the module $k[x, y]^3$:

```
> ring R = 0,(x,y),(lp,C);
% 0 indicates the fields of rationals; (lp,C) indicates the TOP extension of lex order
% C indicates upward ordering on the standard basis elements
```

```
> vector f=[-2x3y+1,x2+2xy,-y];
> f;
-2x3y*gen(1)+x2*gen(2)+2xy*gen(2)-y*gen(3)+gen(1)
% gen(i) is the i-th vector of the standard basis
```

and let us see how to order the same vector, but accordingly to POT order:

```
> ring R = 0,(x,y),(C,lp);
% (C,lp) indicates indicates the POT extension of lex order
> vector f=[-2x3y+1,x2+2xy,-y];
> f;
-y*gen(3)+x2*gen(2)+2xy*gen(2)-2x3y*gen(1)+gen(1)
```

Once we have an ordering $>$ on monomials, we can write any element $\mathbf{f} \in R^m$ as a sum of terms

$$\mathbf{f} = \sum_{i=1}^{t} c_i \mathbf{m}_i$$

with $c_i \neq 0$ and $\mathbf{m}_1 > \mathbf{m}_2 > \cdots > \mathbf{m}_t$. We define the *leading coefficient*, *leading monomial*, and *leading term* of f as in the ring case:

$$\begin{aligned} LC_>(\mathbf{f}) &= c_1 \\ LM_>(\mathbf{f}) &= \mathbf{m}_1 \\ LT_>(\mathbf{f}) &= c_1\mathbf{m}_1 \end{aligned}$$

Now that we have a monomial ordering in R^m we can divide by a set $F \subset R^m$ in the same way we did in R:

Theorem 2.3. *(***Division Algorithm in** R^m*).* *Fix any monomial ordering on R^m and let $F = (\mathbf{f}_1, \ldots, \mathbf{f}_s)$ be an ordered s-tuple of elements of R^m. Then every $\mathbf{f} \in R^m$ can be written as*

$$\mathbf{f} = a_1\mathbf{f}_1 + \cdots + a_s\mathbf{f}_s + \mathbf{r}$$

where $a_i \in R$, $\mathbf{r} \in R^m$, $\mathrm{LT}(a_i\mathbf{f}_i) \leq \mathrm{LT}(\mathbf{f})$ for all i, and either $\mathbf{r} = 0$ or $\mathbf{r}$ is a k-linear combination of monomials none of which is divisible by any of $\mathrm{LM}(\mathbf{f}_1), \ldots, \mathrm{LM}(\mathbf{f}_s)$. We call $\mathbf{r}$ the remainder on division by F.

As we know, the division algorithm behaves best when the set of divisors has the defining property of a Groebner basis; let us then introduce the definition of Groebner bases for submodules:

Definition 2.18. *Let M be a submodule of R^m, and let $>$ be a monomial order.*

i. We will denote by $\langle \mathrm{LT}(M) \rangle$ the monomial submodule generated by the leading terms of all $\mathbf{f} \in M$ with respect to $>$.

ii. A finite collection $\mathcal{G} = \{\mathbf{g}_1, \ldots, \mathbf{g}_s\} \subset M$ is called a **Groebner basis** *for M if $\langle \mathrm{LT}(M) \rangle = \langle \mathrm{LT}(\mathbf{g}_1), \ldots, \mathrm{LT}(\mathbf{g}_s) \rangle$.*

The properties of Groebner bases for ideals of R with respect to division, extend immediately (and with the same proofs) to submodules of R^m, e.g.

Proposition 2.4. *Let $\mathcal{G}$ be a Groebner basis for a submodule $M \subset R^m$, and let $\mathbf{f} \in R^m$. Then:*

i. $\mathbf{f} \in M$ *if and only if the remainder on division by $\mathcal{G}$ is zero.*

ii. A Groebner basis for M generates M as a module: $M = \langle \mathcal{G} \rangle$.

Some care must be put when dealing with point (ii) of Prop. 2.4. In general, it is not true that a Groebner basis is a *basis* for M as an R-module, indeed a Groebner basis is a set of generators for M, but it need not be linearly independent over R.

(Monic) reduced Groebner bases may be defined as for ideals [2006 Cox, Little & O'Shea]:

Definition 2.19. *A* **monic reduced Groebner basis** *for a polynomial module M is a Groebner basis G for M such that:*

i. $\mathrm{LC}(\mathbf{p}) = 1$ *for all* $\mathbf{p} \in G$.

ii. For all $\mathbf{p} \in G$, *no monomial of* $\mathbf{p}$ *lies in* $\langle \mathrm{LT}(G - \{\mathbf{p}\}) \rangle$.

and there is a unique (monic) reduced Groebner basis for each submodule in R^m once we choose a monomial order.

As we showed in Prop. 2.3 (ii), another property that extends from ideals of R to submodules of R^m is the *Ascending Chain Condition (ACC).*

We now focus on the extension of Buchberger's algorithm to the module case. Let us first introduce the following definition:

Definition 2.20. *Fix a monomial order on R^m, and let $\mathbf{f}, \mathbf{g} \in R^m$. The* **S-vector** *of $\mathbf{f}$ and $\mathbf{g}$, denoted $S(\mathbf{f}, \mathbf{g})$, is the following element of R^m:*

$$S(\mathbf{f}, \mathbf{g}) = \frac{\mathbf{m}}{\mathrm{LT}(\mathbf{f})}\mathbf{f} - \frac{\mathbf{m}}{\mathrm{LT}(\mathbf{g})}\mathbf{g}$$

where $\mathbf{m} = \mathrm{LCM}(\mathrm{LT}(\mathbf{f}), \mathrm{LT}(\mathbf{g}))$.

The foundation for an algorithm that allows to compute Groebner bases for submodules of R^m is the following generalization of Buchberger's Criterion:

Theorem 2.4. *(***Buchberger's Criterion for Submodules***). A set $\mathcal{G} = \{\mathbf{g}_1, \ldots, \mathbf{g}_s\} \subset R^m$ is a Groebner basis for the module it generates if and only if the remainder on division by $\mathcal{G}$ of $S(\mathbf{g}_i, \mathbf{g}_j)$ is $\mathbf{0}$ for all i, j.*

To compute Groebner bases, we need a version of Buchberger's Algorithm. Using Th. 2.4, the Buchberger's algorithm extends immediately to the module case:

Theorem 2.5. *Let $F = (\mathbf{f}_1, \ldots, \mathbf{f}_t)$ where $\mathbf{f}_i \in R^m$, and fix a monomial order on R^m. The following algorithm computes a Groebner basis $\mathcal{G}$ for $M = \langle F \rangle \subset R^m$, where $\overline{S(\mathbf{f}, \mathbf{g})}^{\mathcal{G}'}$ denotes the remainder on division by $\mathcal{G}'$, using Th. 2.3*

Input: $F = (\mathbf{f}_1, \ldots, \mathbf{f}_t) \subset R^m$, an order $>$.
Output: a Groebner basis $\mathcal{G}$ for $M = \langle F \rangle$, with respect to $>$.

$\mathcal{G} := F$
repeat
  $\mathcal{G} = \mathcal{G}'$
  for *each pair* $\mathbf{f} \neq \mathbf{g}$ *in* $\mathcal{G}'$ ***do***
    $S := \overline{S(\mathbf{f}, \mathbf{g})}^{\mathcal{G}'}$
    if $S \neq 0$ ***then***
      $\mathcal{G} := \mathcal{G} \cup \{S\}$
    end if
  end for
until $\mathcal{G} = \mathcal{G}'$

Chapter 3

Cyclic Codes

This chapter is devoted to recalling the theory of cyclic codes, which are then used in Chapter 4 to build quasicyclic codes. In particular, we discuss linear codes, and then cyclic codes. For the latter, we focus on the problem of factorizing $x^n - 1$. Later on some basic theory is recalled, and thereafter one introduces the concept of zeros and minimum distance of a cyclic codes. At last, we treat BCH and Reed-Solomon codes.

3.1 Linear Codes

Among all types of codes, linear codes are studied the most, since because of their algebraic structure they are easier to describe, encode and decode with respect to nonlinear codes.

Let $\mathbb{F}_q^n$ denote the vector space of all n-tuples over the finite field $\mathbb{F}_q$. An (n, M) *code* $\mathcal{C}$ over $\mathbb{F}_q$ is a subset of $\mathbb{F}_q^n$ of cardinality M. We usually write the vectors in $\mathbb{F}_q^n$ in the form $(a_1, a_2, \ldots, a_n)$ and call the vectors in $\mathcal{C}$ *codewords*. Codewords are sometimes specified in other ways, e.g. by using the polynomial representation used for codewords in cyclic codes, which we will discuss in Section 3.2.

If we do not impose further structure on a code, its usefulness would be limited. The most useful additional structure to impose is that of linearity; in this respect, if $\mathcal{C}$ is a k-dimensional subspace of $\mathbb{F}_q^n$, then $\mathcal{C}$ will be called an $[n, k]$ *linear code* over $\mathbb{F}_q$. The linear code $\mathcal{C}$ has q^k codewords.

The two most common ways to present a linear code are with either a generator matrix of a parity check matrix. A *generator matrix* for an $[n, k]$ code $\mathcal{C}$ is any $k \times n$ matrix G whose rows form a basis for $\mathcal{C}$. In general there are many generator matrices for a code. For any set of k independent columns of a generator matrix G, the corresponding set of coordinates forms an *information set* for $\mathcal{C}$, and the remaining $r = n - k$ coordinates are named a *redundancy set* and r is called the *redundancy* of $\mathcal{C}$.

Since a linear code is a subspace of a vector space, it is the kernel of some linear transformation. In particular, there is an $(n - k) \times k$ matrix H, called a *parity check matrix* for the $[n, k]$ code $\mathcal{C}$,

defined by

$$\mathcal{C} = \left\{ \mathbf{x} \in \mathbb{F}_q^n \mid H\mathbf{x}^T = \mathbf{0} \right\}$$

An important invariant of a code is the minimum distance between codewords. The *(Hamming) distance* $d(\mathbf{x}, \mathbf{y})$ between two vectors $\mathbf{x}, \mathbf{y} \in \mathbb{F}_q^n$ is defined to be the number of coordinates in which $\mathbf{x}$ and $\mathbf{y}$ differ. The *(Hamming) weight* $\mathrm{wt}(\mathbf{x})$ of a vector $\mathbf{x} \in \mathbb{F}_q^n$ is the number of nonzero coordinates in $\mathbf{x}$.

We will prove the following properties of the distance function $d(\mathbf{x}, \mathbf{y})$:

Theorem 3.1. *The distance function $d(\mathbf{x}, \mathbf{y})$ is a metric over the vector space $\mathbb{F}_q^n$, i.e. $d(\mathbf{x}, \mathbf{y})$ satisfies the following four properties:*

i. (non-negativity) $d(\mathbf{x}, \mathbf{y}) \geq 0$ for all $\mathbf{x}, \mathbf{y} \in \mathbb{F}_q^n$

ii. $d(\mathbf{x}, \mathbf{y}) = 0$ if and only if $\mathbf{x} = \mathbf{y}$

iii. (symmetry) $d(\mathbf{x}, \mathbf{y}) = d(\mathbf{y}, \mathbf{x})$ for all $\mathbf{x}, \mathbf{y} \in \mathbb{F}_q^n$

iv. (triangle inequality) $d(\mathbf{x}, \mathbf{z}) \leq d(\mathbf{x}, \mathbf{y}) + d(\mathbf{y}, \mathbf{z})$ for all $\mathbf{x}, \mathbf{y}, \mathbf{z} \in \mathbb{F}_q^n$

Proof. (i) We have that $d(\mathbf{x}, \mathbf{y}) \geq 0$ for all $\mathbf{x}, \mathbf{y} \in \mathbb{F}_q^n$, since the weight cannot be negative.

(ii) Also, $d(\mathbf{x}, \mathbf{y}) = 0$ if and only if $\mathbf{x} = \mathbf{y}$, since $\mathrm{wt}(\mathbf{x} - \mathbf{x}) = \mathrm{wt}(\mathbf{0}) = 0$, and only the null codeword has weight 0.

(iii) We have that $d(\mathbf{x}, \mathbf{y}) = d(\mathbf{y}, \mathbf{x})$ for all $\mathbf{x}, \mathbf{y} \in \mathbb{F}_q^n$, since $\mathbf{x} - \mathbf{y}$ has 0's in exactly the same coordinates as $\mathbf{y} - \mathbf{x}$, and therefore $\mathrm{wt}(\mathbf{x} - \mathbf{y}) = \mathrm{wt}(\mathbf{y} - \mathbf{x})$.

(iv) We prove the triangle inequality considering the i-th coordinate in the two cases $y_i = x_i \vee y_i = z_i$ and $y_i \neq x_i \ \wedge \ y_i \neq z_i$. Assume first that $y_i = x_i$, then we have that

$$d(x_i, y_i) + d(y_i, z_i) = d(x_i, x_i) + d(x_i, z_i) = d(x_i, z_i)$$

since $d(x_i, x_i) = 0$. Assume then that $y_i \neq x_i \ \wedge \ y_i \neq z_i$; therefore

$$d(x_i, y_i) + d(y_i, z_i) = 2 \tag{3.1}$$

Since $d(x_i, z_i) \leq 1$, as either $x_i = z_i$ or $x_i \neq z_i$, considering also Eq.(3.1) we have that

$$d(x_i, y_i) + d(y_i, z_i) > d(x_i, z_i)$$

Finally, we have proved that

$$d(x_i, z_i) \leq d(x_i, y_i) + d(y_i, z_i)$$

□

3.2 Cyclic Codes

The aim of this section is to introduce the important class of *cyclic codes* and their properties. The main reference of this section is [2003 Huffman & Pless]. Many families of codes including the

Golay codes, the binary Hamming codes, and codes equivalent to the Reed-Muller codes are either cyclic or extended cyclic codes.

While studying cyclic codes of length n, it is convenient to label the coordinate positions as $0, 1, \ldots, n-1$ and think of these as the integers modulo n. A linear code $\mathcal{C}$ of length n over $\mathbb{F}_q$ is *cyclic* provided that for each vector $\mathbf{c} = (c_0, \ldots, c_{n-2}, c_{n-1})$ in $\mathcal{C}$ the vector $(c_{n-1}, c_0, \ldots, c_{n-2})$, obtained from $\mathbf{c}$ by the *cyclic shift* of coordinates $i \mapsto i+1 \pmod{n}$, is also in $\mathcal{C}$.

When examining cyclic codes over $\mathbb{F}_q$, it is usual to represent the codewords in polynomial form; indeed, there is a bijective correspondence between the vectors $\mathbf{c} = (c_0, \ldots, c_{n-2}, c_{n-1})$ in $\mathbb{F}_q^n$ and the polynomials $c(x) = c_0 + c_1 x + \cdots + c_{n-1}x^{n-1}$ in $\mathbb{F}_q[x]$ of degree at most $n-1$. We order the terms of the polynomials from the smallest to the largest degree. From notation's point of view, we will use interchangeably the vector notation $\mathbf{c}$ and the polynomial notation $c(x)$.

Notice that if $c(x) = c_0 + c_1 x + \cdots + c_{n-1}x^{n-1}$, then $xc(x) = c_{n-1}x^n + c_0 x + c_1 x^2 + \cdots + c_{n-2}x^{n-1}$, which, in case x^n were set equal to 1, would represent the codeword $\mathbf{c}$ cyclically shifted of one position to the right. More formally, the fact that a cyclic code is invariant under a cyclic shift implies that if $c(x)$ is in $\mathcal{C}$, then so is $xc(x)$ provided that we multiply modulo $x^n - 1$.

Indeed, let us write the shifted codeword's associated code polynomial

$$\tilde{c} = c_{n-1} + c_0 x + c_1 x^2 + c_{n-2}x^{n-1}$$

Thus we have that

$$\tilde{c} = xc(x) - c_{n-1}(x^n - 1)$$

from which we see that

$$\tilde{c} \equiv xc(x) \pmod{x^n - 1}$$

i.e., $\tilde{c}$ and $xc(x)$ are congruent in the ring of polynomials $\mathbb{F}_q[x] \pmod{x^n - 1}$.

The above discussion suggests that the proper context for studying cyclic codes is the residue class ring

$$\mathcal{R}_n = \mathbb{F}_q[x]/(x^n - 1) \tag{3.2}$$

Under the correspondence of vectors with polynomials, cyclic codes are ideals of $\mathcal{R}_n$ and ideals of $\mathcal{R}_n$ are cyclic codes. We therefore have that *the study of cyclic codes in $\mathbb{F}_q^n$ is equivalent to the study of ideals in $\mathcal{R}_n$*. Let us prove this [lecture cyclic]:

Proposition 3.1. *The polynomial rendering of a cyclic code is an ideal of the ring* $\mathbb{F}_q[x] \pmod{x^n - 1}$.

Proof. We know that $c(x) \in \mathcal{C} \pmod{x^n - 1}$ if and only if $xc(x) \in \mathcal{C} \pmod{x^n - 1}$. Since additional shifts do not take us out of the cyclic code $\mathcal{C}$, we have

$$x^i c(x) \in \mathcal{C} \pmod{x^n - 1}$$

for all i. By linearity, for any $a_i \in \mathbb{F}_q$

$$a_i x^i c(x) \in \mathcal{C} \pmod{x^n - 1}$$

and moreover

$$\sum_{i=0}^{d} a_i x^i c(x) \in \mathcal{C} \pmod{x^n - 1}$$

i.e., for every polynomial $a(x) = \sum_{i=0}^{d} a_i x^i \in \mathbb{F}_q[x]$, the product $a(x)c(x) \pmod{x^n - 1}$ still belongs to $\mathcal{C}$.

As $\mathcal{C}$, being linear, is closed under polynomial addition, we have that the polynomial rendering of a cyclic code is precisely an ideal of the ring $\mathbb{F}_q[x] \pmod{x^n - 1}$. □

The correspondence of Prop. 3.1 was first noted by Prange, and opened the way for the application of algebra to cyclic codes.

We will see later on (see Section 3.2.2) that the study of ideals in $\mathcal{R}_n$ depends on factoring $x^n - 1$, the latter being the subject of Section 3.2.1.

3.2.1 Factoring $x^n - 1$

Let us first introduce some important concepts that we will use later on.

Definition 3.1. *A* **cyclic group** *is a group $\mathcal{G}$ containing an element g such that $\mathcal{G} = \{g^n : n \in \mathbb{Z}\}$. The element g is called a* **generator** *of G, and we say that G is* **generated** *by g.*

Let us define by $\mathbb{F}_q^*$ the group of nonzero elements in $\mathbb{F}_q$. From [2003 Huffman & Pless], Th. 3.3.1, pg. 104, we have that:

Theorem 3.2. *It holds what follows:*

i. The group $\mathbb{F}_q^*$ is cyclic of order $q - 1$ under the multiplication of $\mathbb{F}_q$.

ii. If γ is a generator of this cyclic group, then $\mathbb{F}_q = \{0, 1 = \gamma^0, \gamma, \gamma^2, \ldots, \gamma^{q-2}\}$, and $\gamma^i = 1$ if and only if $(q - 1)|i$.

Each generator γ of $\mathbb{F}_q$ is called a *primitive element* of $\mathbb{F}_q$.

When analyzing the field structure, it is useful to know the number of primitive elements in $\mathbb{F}_q$ and how to find them all once one primitive element has been found. Since $\mathbb{F}_q^*$ is cyclic, let us recall a few things about finite cyclic groups. In any finite cyclic group $\mathcal{G}$ of order n with generator g, the generators of $\mathcal{G}$ are precisely the elements g^i where $\gcd(i, n) = 1$. We let $\phi(n)$ be the number of integers i with $1 \leq i \leq n$ such that $\gcd(i, n) = 1$; ϕ is called the *Euler totient* ot the *Euler ϕ-function*. So there are $\phi(n)$ generators of $\mathcal{G}$. The *order* of an element $\alpha \in \mathcal{G}$ is the smallest positive integer i such that $\alpha^i = 1$.

Let us also introduce the following Theorem ([2003 Huffman & Pless], Th. 3.3.3, pg. 105):

Theorem 3.3. *Let γ be a primitive element of $\mathbb{F}_q$.*

i. There are $\phi(q-1)$ primitive elements in $\mathbb{F}_q$; these are the elements γ^i where $\gcd(i, q-1) = 1$.

ii. For any d where $d|(q - 1)$, there are $\phi(d)$ elements in $\mathbb{F}_q$ of order d; these are the elements $\gamma^{(q-1)i/d}$ where $\gcd(i, d) = 1$.

An element $\xi \in \mathbb{F}_q$ is an n-th *root of unity* provided $\xi^n = 1$, and is a *primitive n-th root of unity* if in addition $\xi^s \neq 1$ for $0 < s < n$.

From the above, it follows that:

Proposition 3.2. *A primitive element γ of $\mathbb{F}_q$ is therefore a primitive $q-1$-st root of unity.*

Proof. From Th. 3.2(ii) we have that $\gamma^i = 1$ if and only if $(q-1)|i$, which is certainly true for $i = q-1$ (and implying that $i \geq q-1$), but then we have no such i that is such that $i < q-1$ by the previous argument, which means that $\gamma^{q-1} = 1$, but $\gamma^s \neq 1$ for $0 < s < q-1$, implying that γ is a primitive $q-1$-st root of unity. □

A primitive element γ of $\mathbb{F}_q$ is therefore a primitive $(q-1)$-th root of unity if and only if $n|(q-1)$.

We want to find the irreducible factors of $x^n - 1$ over $\mathbb{F}_q$. There are two possibilities: either $x^n - 1$ has repeated irreducible factors or it does not; the study of cyclic codes focuses primarily on the latter case. From [2003 Huffman & Pless] we know that $x^n - 1$ has no repeated factors if and only if q and n are relatively prime, which is the case we assume in this section.

To help factor $x^n - 1$ over $\mathbb{F}_q$, it is useful to find an extension field $\mathbb{F}_{q^t}$ of $\mathbb{F}_q$ that contains all of its roots. In other words, $\mathbb{F}_{q^t}$ must contain a primitive n-th root of unity, which occurs when $n|(q^t - 1)$ by Th. 3.3. Define the *order* $\mathrm{ord}_n(q)$ of q modulo n to be the smallest positive integer a such that $q^a \equiv 1 \pmod{n}$. $\mathbb{F}_{q^t}$, which contains all the roots of $x^n - 1$, is called a *splitting field of $x^n - 1$ over $\mathbb{F}_q$*.

Let also s be an integer with $0 \leq s < n$. The *q-cyclotomic coset of s modulo n* is the set

$$C_s = \{s, sq, \ldots, sq^{r-1}\} (\mathrm{mod}\ n) \tag{3.3}$$

where r is the smallest positive integer such that $sq^r \equiv s \pmod{n}$. The distinct q-cyclotomic cosets modulo n partition the set of integers $\{0, 1, 2, \ldots, n-1\}$.

We can now introduce the following result on the factorization of $x^n - 1$:

Theorem 3.4. *Let n be a positive integer relatively prime to q. Let $t = \mathrm{ord}_n(q)$. Let α be a primitive n-th root of unity in $\mathbb{F}_{q^t}$.*

i. For each integer s with $0 \leq s < n$, the minimal polynomial of α^s over $\mathbb{F}_q$ is

$$M_{\alpha^s}(x) = \prod_{i \in \mathcal{C}_s} (x - \alpha^i)$$

where C_s is the q-cyclotomic coset of s modulo n.

ii. The conjugates of α^s are the elements α^i with $i \in C_s$ (we recall that two elements of $\mathbb{F}_{q^t}$ which have the same minimal polynomial in $\mathbb{F}_q[x]$ are called conjugate *over $\mathbb{F}_q$).*

iii. Furthermore,

$$x^n - 1 = \prod_s M_{\alpha^s}(x)$$

is the factorization of $x^n - 1$ into irreducible factors over $\mathbb{F}_q$, where s runs through a set of representatives of the q-cyclotomic cosets modulo n.

3.2.2 Basic Theory of Cyclic Codes

We saw above that cyclic codes over $\mathbb{F}_q$ are precisely the ideals of

$$\mathcal{R}_n = \mathbb{F}_q[x]/(x^n - 1)$$

One can show that $\mathbb{F}_q[x]$ is a principal ideal domain, and that the ideals of $\mathcal{R}_n$ are also principal, and hence cyclic codes are principal ideals of $\mathcal{R}_n$. When writing a codeword of a cyclic code as $c(x)$, we technically mean the coset $c(x) + (x^n - 1)$ in $\mathcal{R}_n$, but in order to simplify the notation, we write $c(x)$ even when working in $\mathcal{R}_n$. Thus we think of the elements of $\mathcal{R}_n$ as the polynomials in $\mathbb{F}_q[x]$ of degree less than n with multiplication being carried out modulo $x^n - 1$.

To distinguish the principal ideal $((g(x))$ of $\mathbb{F}_q[x]$ from that ideal in $\mathcal{R}_n$, we use the notation $\langle g(x)\rangle$ for the principal ideal of $\mathcal{R}_n$ generated by $g(x)$. The following theorem states that there is a bijective correspondence between the cyclic codes in $\mathcal{R}_n$ and the monic polynomial divisors of $x^n - 1$.

Theorem 3.5. *Let $\mathcal{C}$ be a nonzero cyclic code in $\mathcal{R}_n$. There exists a polynomial $g(x) \in \mathcal{C}$ with the following properties:*

i. $g(x)$ *is the unique monic polynomial of minimum degree in* $\mathcal{C}$,

ii. $\mathcal{C} = \langle g(x)\rangle$, *and*

iii. $g(x)|(x^n - 1)$.

Let $k = n - \deg(x)$, *and let* $g(x) = \sum_{i=0}^{n-k} g_i x^i$, *where* $g_{n-k} = 1$. *Then:*

iv. the dimension of $\mathcal{C}$ *is* k *and* $\{g(x), xg(x), \ldots, x^{k-1}g(x)\}$ *is a basis for* $\mathcal{C}$,

v. every element of $\mathcal{C}$ *is uniquely expressible as a product* $g(x)f(x)$, *where* $f(x) = 0$ *or* $\deg f(x) < k$,

vi.

$$G = \begin{bmatrix} g_0 & g_1 & g_2 & \cdots & g_{n-k} & & 0 \\ 0 & g_0 & g_1 & \cdots & g_{n-k-1} & g_{n-k} & \\ & \cdots & \cdots & \cdots & \cdots & \cdots & \\ 0 & & g_0 & & \cdots & & g_{n-k} \end{bmatrix}$$

$$\leftrightarrow \begin{bmatrix} g(x) & & & & \\ & xg(x) & & & \\ & & \cdots & & \\ & & & x^{k-1}g(x) & \end{bmatrix} \tag{3.4}$$

is a generator matrix for $\mathcal{C}$, *and*

vii. if α *is a primitive* n*-th root of unity in some extension field of* $\mathbb{F}_q$, *then*

$$g(x) = \prod_s M_{\alpha^s}(x)$$

where the product is over a subset of representatives of the q*-cyclotomic cosets modulo* n.

Let us notice that part (ii) of Th. 3.5 shows that $\mathcal{R}_n$ is a principal ideal ring. Moreover, Th. 3.5 shows that there is a monic polynomial $g(x)$ dividing $x^n - 1$ which generates $\mathcal{C}$. The following corollary states that the monic polynomial dividing $x^n - 1$ which generates $\mathcal{C}$ is unique.

Corollary 3.1. *Let $\mathcal{C}$ be a nonzero cyclic code in $\mathcal{R}_n$. The following are equivalent:*

i. $g(x)$ is the monic polynomial of minimum degree in $\mathcal{C}$.

ii. $\mathcal{C} = \langle g(x) \rangle$, $g(x)$ is monic, and $g(x) | (x^n - 1)$.

The polynomial $g(x)$ of Cor. 3.1 is called *the generator polynomial* of the cyclic code $\mathcal{C}$. By the corollary, this polynomial is both the monic polynomial in $\mathcal{C}$ of minimum degree and the monic polynomial dividing $x^n - 1$ which generates $\mathcal{C}$. So there is a one-to-one correspondence between the nonzero cyclic codes and the divisors of $x^n - 1$, not equal to $x^n - 1$. In order to have a bijective correspondence between all the cyclic codes in $\mathcal{R}_n$ and all the monic divisors of $x^n - 1$, we define the generator polynomial of the zero cyclic code $\{\mathbf{0}\}$ to be $x^n - 1$ (note that $x^n - 1$ equals 0 in $\mathcal{R}_n$).

3.2.3 Zeros of a Cyclic Code and Minimum Distance of Cyclic Codes

Let $t = \mathrm{ord}_n(q)$, and α be a primitive n-th root of unity contained in $\mathbb{F}_{q^t}$. Further, let $\mathcal{C}$ be a cyclic code in $\mathcal{R}_n = \mathbb{F}_q[x]/(x^n - 1)$ with generator polynomial $g(x)$. By Th. 3.4(i) and 3.5(vii), $g(x) = \prod_s M_{\alpha^s}(x) = \prod_s \prod_{i \in C_s} (x - \alpha^i)$, where s runs through some subset of representatives of the q-cyclotomic cosets C_s modulo n. Let $T = \bigcup_s C_s$ be the union of these q-cyclotomic cosets. The roots of unity $\mathcal{Z} = \left\{\alpha^i | i \in T\right\}$ are called the *zeros* of the cyclic code $\mathcal{C}$ and $\left\{\alpha^i | i \notin T\right\}$ are the *nonzeros* of $\mathcal{C}$.

The set T is called the *defining set* of $\mathcal{C}$. Note that if we change the primitive n-th root of unity, we change T; so T is computed with respect to a fixed primitive root.

We have the following relations between T and the generator polynomial $g(x)$:

Corollary 3.2. *Let $t = \mathrm{ord}_n(q)$, and α be a primitive n-th root of unity contained in $\mathbb{F}_{q^t}$. Moreover, let $\mathcal{C}$ be a cyclic code in $\mathcal{R}_n$ with generator polynomial $g(x)$. Also, let T be the defining set computed with respect to α. Then it holds that:*

i. $c(x)$ belongs to $\mathcal{C}$ if and only if $c(\alpha^i) = 0$ for each $i \in T$;

ii. the defining set T, and thus either the set of zeros or the set of nonzeros, completely determines the generator polynomial $g(x)$;

iii. the dimension of $\mathcal{C}$ is $n - |T|$ as $|T|$ is the degree of $g(x)$.

Proof. This is a direct consequence of Th. 3.5 (in particular, see point (vii)). □

With any code, it is important to be able to determine the minimum distance in order to determine its correcting capability; it is then helpful to have bounds on the minimum distance, particularly

lower bounds. Several lower bounds are known for the minimum distance of a cyclic code, and the oldest of these bounds is the Bose–Ray-Chaudhuri–Hocquenghem Bound, usually called BCH bound, fundamental in the definition of BCH codes, which we will discuss in next section. The BCH bound depends on the zeros of the code and especially on the ability to find strings of *consecutive* zeros.

Before proceeding with the BCH bound, we state a lemma that will be used in the proof of the BCH Bound, about the determinant of a matrix, called *Vandermonde matrix*, which we define as:

Definition 3.2. *Let $\alpha_1, \ldots, \alpha_s$ be elements in a field $\mathbb{F}$. The $s \times s$ matrix $V = [v_{i,j}]$, where $v_{i,j} = \alpha_j^{i-1}$ is called a* **Vandermonde matrix**. *The transpose of this matrix is also called a Vandermonde matrix.*

Let us now introduce the afore-mentioned lemma:

Lemma 3.1. *Let* $\det V = \prod_{1 \leq i < j \leq s}(\alpha_j - \alpha_i)$*, then V is nonsingular if the elements $\alpha_1, \ldots, \alpha_s$ are distinct.*

We say that the defining set T for $\mathcal{C}$ contains a set of s *consecutive* elements $\mathcal{S}$ provided there is a set $\{b, b+1, \ldots, b+s-1\}$ of s consecutive integers such that

$$\{b, b+1, \ldots, b+s-1\} \mod n = \mathcal{S} \subseteq T$$

We can now state the following theorem:

Theorem 3.6. *(***BCH Bound***) Let $\mathcal{C}$ be a cyclic code of length n over $\mathbb{F}_q$ with defining set T. Let d denote the minimum weight of $\mathcal{C}$, and assume T contains $\delta - 1$ consecutive elements for some integer δ. Then $d \geq \delta$.*

Proof. By assumptions, the zeros of $\mathcal{C}$ include $\delta - 1$ zeros such that $\alpha^b, \alpha^{b+1}, \ldots, \alpha^{b+\delta-2}$. Let $c(x)$ be a nonzero codeword in $\mathcal{C}$ of weight w, and let

$$c(x) = \sum_{j=1}^{w} c_{i_j} x^{i_j}, \quad \text{where } c_{i_j} \neq 0,\ j = 1, \ldots, w$$

Assume to the contrary of the thesis that $w < \delta$. As $c(\alpha^i) = 0$ for $b \leq i \leq b + \delta - 2$, we can write

$$M\mathbf{u}^T = \mathbf{0}$$

where

$$M = \begin{bmatrix} \alpha^{i_1 b} & \alpha^{i_2 b} & \cdots & \alpha^{i_w b} \\ \alpha^{i_1 (b+1)} & \alpha^{i_2 (b+1)} & \cdots & \alpha^{i_w (b+1)} \\ & & \vdots & \\ \alpha^{i_1 (b+w-1)} & \alpha^{i_2 (b+w-1)} & \cdots & \alpha^{i_w (b+w-1)} \end{bmatrix}$$

and $\mathbf{u} = (c_{i_1}, c_{i_2}, \cdots, c_{i_w})$. Since $\mathbf{u} \neq \mathbf{0}$, M must be a singular matrix (it is indeed equivalent to state that M is nonsingular and that the equation $\mathbf{M}\mathbf{u}^T = \mathbf{0}$ has only the trivial solution $\mathbf{u} = \mathbf{0}$, which is not the case here). Being M singular, we have that $\det M = 0$.

We can write $\det M = \alpha^{(i_1+i_2+\cdots+i_w)b}\det V$, where V is the Vandermonde matrix

$$V = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ \alpha^{i_1} & \alpha^{i_2} & \cdots & \alpha^{i_w} \\ & & \vdots & \\ \alpha^{i_1(w-1)} & \alpha^{i_2(w-1)} & \cdots & \alpha^{i_w(w-1)} \end{bmatrix}$$

Being the α^{i_j} distinct by assumption, by Lemma 3.1 we have that $\det V \neq 0$, from which follows that $\det M \neq 0$, having then a contradiction with the above statement $\det M = 0$. Therefore we conclude that it must be $w \geq \delta$. □

3.2.4 BCH Codes

We will examine here one of the most important families of cyclic codes known as BCH codes, and in next section a subfamily of these codes called Reed-Solomon codes. Because of their burst error-correction capabilities, Reed-Solomon codes are used to improve the reliability of data storage systems.

BCH codes are cyclic codes designed to take advantage of the BCH bound (Th. 3.6). We would like to construct a cyclic code $\mathcal{C}$ of length n over $\mathbb{F}_q$ with simultaneously high minimum weight and high dimension.

The requirement of having high minimum weight, by the BCH bound, can accomplished by choosing a defining set T for $\mathcal{C}$ with a large number of consecutive elements.

Let us first prove the following result:

Proposition 3.3. *The dimension of $\mathcal{C}$ is $n - |T|$.*

Proof. The proof follows from Th. 3.5, as $|T|$ is the degree of $g(x)$), □

Then, as the dimension of $\mathcal{C}$ is $n - |T|$, we would like $|T|$ to be as small as possible. So if we would like $\mathcal{C}$ to have minimum distance at least δ, we can choose a defining set as small as possible that is a union of q-cyclotomic cosets with $\delta - 1$ consecutive elements.

Let δ be an integer with $2 \leq \delta \leq n$. A *BCH code* $\mathcal{C}$ over $\mathbb{F}_q$ of length n and *designed distance* δ is a cyclic code with defining set

$$T = C_b \cup C_{b+1} \cup \cdots \cup C_{b+\delta-2} \tag{3.5}$$

where C_i is the q-cyclotomic coset modulo n containing i.

The following theorem characterizes the minimum distance of a BCH code:

Theorem 3.7. *A BCH code of designed distance δ has minimum weight at least δ.*

Proof. The defining set (3.5) contains $\delta - 1$ consecutive elements. The result follows by the BCH bound. □

By varying the value of b, one produces a variety of codes with possibly different minimum distances and dimensions. When $b = 1$, $\mathcal{C}$ is called a *narrow-sense* BCH code. If $n = q^t - 1$, then $\mathcal{C}$ is called a *primitive* BCH code (as with any cyclic code).

3.2.5 Reed-Solomon Codes

A *Reed-Solomon code*, abbreviated *RS code*, $\mathcal{C}$ over $\mathbb{F}_q$ is a BCH code of length $n = q - 1$. Then, as $\text{ord}_n(q)$ is the smallest positive integer a such that $q^a \equiv 1 \pmod{n}$, we have that $\text{ord}_n(q) = 1$, being in fact $q^1 \equiv 1 \pmod{n}$. This implies that all irreducible factors of $x^n - 1$ are of degree 1 and all q-cyclotomic cosets modulo n have size 1.

If $\mathcal{C}$ has designed distance δ, the defining set of $\mathcal{C}$ has size $\delta - 1$ and is $T = \{b, b + 1, \cdots, b + \delta - 2\}$ for some integer b.

Let define by $A_q(n, d)$ the maximum number of codewords in a code over $\mathbb{F}_q$ of length n and minimum distance at least d. We have the following result concerning $A_q(n, d)$:

Theorem 3.8. *(***Singleton Bound***) For $d \leq n$,*

$$A_q(n, d) \leq q^{n-d+1} \tag{3.6}$$

Furthermore, if an $[n, k, d]$ linear code over $\mathbb{F}_q$ exists, then $k \leq n - d + 1$.

A code for which equality holds in Eq.(3.6) is called *maximum distance separable*, abbreviated MDS. No code of length n and minimum distance d has more codewords than an MDS code with parameters n and d; equivalently, no code of length n with M codewords has a larger minimum distance than an MDS code with parameters n and M.

By Prop. 3.3, we have that

$$k = n - |T| = n - (\delta - 1) = n - \delta + 1$$

since the defining set of $\mathcal{C}$ has size $\delta - 1$.

By Th. 3.7 we then have that

$$k = n - |T| = n - (\delta - 1) = n - \delta + 1 \geq n - d + 1 \tag{3.7}$$

being $d \geq \delta$.

Finally, by the Singleton bound we have that

$$n - d + 1 \geq k$$

Then Eq.(3.7) becomes

$$k = n - |T| = n - (\delta - 1) = n - \delta + 1 \geq n - d + 1 \geq k \tag{3.8}$$

We see from Eq.(3.8) that both inequalities are equalities implying $d = \delta$ and $k = n - d + 1$ (the latter, being the equality in the Singleton bound verified, means that the code is MDS).

We summarize the above discussion in the following theorem:

Theorem 3.9. *Let $\mathcal{C}$ be an RS code over $\mathbb{F}_q$ of length $n = q - 1$ and designed distance δ. Then:*

i. $\mathcal{C}$ has defining set $T = \{b, b+1, \ldots, b+\delta-2\}$ for some integer b

ii. $\mathcal{C}$ has minimum distance $d = \delta$ and dimension $k = n - d + 1$

iii. $\mathcal{C}$ is MDS.

Chapter 4

Quasicyclic Codes and their Algebraic Structure

This chapter introduces quasicyclic codes and the related algebraic theory. The last part of the chapter focuses on studying quasicyclic codes's algebraic structure by using the tool of Groebner bases of modules; for this chapter, we mainly refer to [LallyPhD].

4.1 Introduction to Quasicyclic Codes

Definition 4.1. *An (n, k) linear block code $\mathcal{C}$ over $\mathbb{F}_q$ is a* **quasicyclic** *code if every cyclic shift of a codeword by r places, for a fixed $1 \leq r < n$, results in another codeword. In other words, for each $c = (c_1, c_2, \ldots, c_n) \in \mathcal{C}$ then $c^{'} = (c_{n-r+1}, \ldots, c_n, c_1, c_2, c_{n-r}) \in \mathcal{C}$.*

Labeling the positions of the n-tuples in $\mathcal{C}$, $1, 2, \ldots, n$, the code is invariant under the permutation π acting on the positions of each codeword:

$$\pi : \ i \to i + r \bmod n, \ i = 1, 2, \ldots, n \tag{4.1}$$

where by n in this notation we practically mean 0 (the standard notation would indeed go from $0, \ldots, n-1$).

It is easy to see that the class of quasicyclic codes is a natural generalization of the class of cyclic codes, the cyclic codes being that subclass where $r = 1$.

Notice that a quasicyclic code $\mathcal{C}$ will also be invariant under cyclic shifts by tr places, for any integer t. We call the smallest number l, such that $\mathcal{C}$ is invariant under l-cyclic shifts the *index* of $\mathcal{C}$. Obviously it is a divisor of the total length of the code, i.e. $n = lm$ for some multiple m.

In general we characterize a quasicyclic code by a generator matrix G, where the rowspace of G is equal to the code. We do not restrict ourselves to the 'strict' definition of generator matrix, which requires the rows of G to be linearly independent, and where the dimension of $\mathcal{C}$ is equal to the number of rows in G. The most intuitive example of a generator matrix for a quasicyclic code over

$\mathbb{F}_q$ of length ml, index l, is of the form

$$G = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1l} & a_{21} & a_{22} & \cdots & a_{2l} & \cdots & a_{m1} & a_{m2} & \cdots & a_{ml} \\ a_{m1} & a_{m2} & \cdots & a_{ml} & a_{11} & a_{12} & \cdots & a_{1l} & \cdots & a_{m-1,1} & a_{m-1,2} & \cdots & a_{m-1,l} \\ \vdots & & & & \ddots & & & & \ddots & \vdots & & & \\ a_{21} & a_{22} & \cdots & a_{2l} & a_{31} & a_{32} & \cdots & a_{3l} & \cdots & a_{11} & a_{12} & \cdots & a_{1l} \end{pmatrix} \tag{4.2}$$

with each $a_{ij} \in \mathbb{F}_q$ and each row being an l-cyclic shift of the previous row. The rowspace of G is invariant under permutation (4.1) and is thus a quasicyclic (ml, k) code where $k \leq m$ the number of rows of G.

Since the code is spanned by all l-cyclic shifts of the top row vector

$$\mathbf{a} = (a_{11}\ a_{12}\ \cdots\ a_{1l}\ a_{21}\ a_{22}\ \cdots\ a_{2l}\ \cdots\ a_{m1}\ a_{m2}\ \cdots\ a_{ml}) \in \mathbb{F}_q^n$$

this is called a *1-generator quasicyclic code*, the top row being that generator. If the rows of G are linearly independent, then we have a (ml, m) code.

4.2 Algebraic Theory of Quasicyclic Codes

Let T be the cyclic shift operator acting on the vectors in $\mathbb{F}_q^n$ so that

$$T\mathbf{v} = T(v_0, v_1, \ldots, v_{n-1}) = (v_{n-1}, v_0, v_1, \ldots, v_{n-2})$$

for $\mathbf{v} \in \mathbb{F}_q^n$. A subset $\mathcal{C}$ of $\mathbb{F}_q^n$ is said to be *invariant* under T if $T\mathcal{C} = \{T\mathbf{v} | \mathbf{v} \in \mathcal{C}\} \subseteq \mathcal{C}$.

Let us define the operator T^i in a recursive way on $\mathbb{F}_q^n$:

$$T^i\mathbf{v} = T(T^{i-1}\mathbf{v}),\ i = 1, 2, \ldots$$

and $T^0 = 1$ is the identity mapping on $\mathbb{F}_q^n$. A quasicyclic (n, k) code $\mathcal{C}$ over $\mathbb{F}_q$ is a linear subspace of $\mathbb{F}_q^n$ which is invariant under T^r for some integer r, $1 \leq r \leq n$. The smallest such integer is called the index of $\mathcal{C}$. We have the following result on the index of a quasicyclic code:

Theorem 4.1. *If a quasicyclic code $\mathcal{C}$ is invariant under T^r, then its index l is a divisor of r, and also a divisor of the total code length n, so that $n = ml$ for some m.*

Notice that if n is prime, the notions of quasicyclic code and of cyclic code of length n coincide.

Let us now look at quasicyclic codes as modules. If $f \in \mathbb{F}_q[x]$ and $n = ml$ then $\mathbb{F}_q^n$ can be viewed as an $\mathbb{F}_q[x]$-module by defining

$$f\mathbf{a} = f(T^l)\mathbf{a}$$

for $\mathbf{a} \in \mathbb{F}_q^n$. With this definition of $\mathbb{F}_q[x]$-multiplication a quasicyclic code of block length $n = ml$ and index l is an $\mathbb{F}_q[x]$-submodule of $\mathbb{F}_q^n$.

Since $x^{n/l} - 1 = x^m - 1$ is the smallest polynomial which annihilates $\mathbb{F}_q^n$, the ideal $I = \langle x^m - 1 \rangle$ is the *annihilator* of $\mathbb{F}_q^n$ denoted $Ann(\mathbb{F}_q^n)$.

Formally, we define the annihilator as follows:

Definition 4.2. *Let R be a ring and M be a module, and choose a subset S of M; the* **annihilator**, *$Ann_R S$, of S is the set of all elements $r \in R$ such that $rs = 0$ for each $s \in S$, is the set of all elements that annihilate S.*

Thus

$$f\mathbf{v} = 0 \text{ for all } f \in I \text{ for all } \mathbf{v} \in \mathbb{F}_q^n$$

We can therefore view $\mathbb{F}_q^n$ as a module over the quotient ring R/I, where $R = \mathbb{F}_q[x]$ and $I = \langle x^m - 1 \rangle$, by defining

$$(f + I)\mathbf{v} = f\mathbf{v}, \ \mathbf{v} \in \mathbb{F}_q^n$$

The quasicyclic codes are just the R/I submodules of $\mathbb{F}_q^n$.

A T^l-subspace $\mathcal{C}$ of $\mathbb{F}_q^n$, i.e. one which is invariant under T^l, is said to have a *single generator* if $\mathcal{C}$ has a basis of the form $\mathbf{c}, T^l\mathbf{c}, T^{2l}\mathbf{c}, \ldots$, for some $\mathbf{c} \in \mathbb{F}_q^n$. Such T^l-subspaces of $\mathbb{F}_q^n$ correspond exactly to those R/I-submodules of $\mathbb{F}_q^n$ which are called 1-generator quasicyclic codes, i.e. where $\mathcal{C} = \langle \mathbf{c} \rangle = \{f\mathbf{c} | f \in R/I\}$ for some $\mathbf{c} \in \mathbb{F}_q^n$ and $\mathcal{C}$ has generator matrix of the form

$$G = \begin{pmatrix} \mathbf{c} \\ T^l\mathbf{c} \\ T^{2l}\mathbf{c} \\ \vdots \\ T^{(m-1)l}\mathbf{c} \end{pmatrix}$$

In the following section we will make use of the theory of Groebner bases to provide a new characterization of quasicyclic codes, exploiting the inherent good properties of these bases.

4.3 Study of Quasicyclic Codes's Algebraic Structure through Groebner Bases of Modules

In [1997 Little, Saints & Heegard], the authors use the theory of Groebner bases of modules to develop a tool for analyzing Hermitian codes. The main idea is to use a **cyclic group** of **automorphisms** of the code to represent it as a module over the polynomial ring $F[x]$.

In [2001 Lally & Fitzpatrick], authors use the same approach as [1997 Little, Saints & Heegard], but to gain insights on the algebraic structure of quasicyclic codes. From now on, unless otherwise specified, by the word "code" we mean "quasicyclic" code.

Let us first explain the philosophy behind [2001 Lally & Fitzpatrick], thereafter we will go into more details of that work. A code $\mathcal{C}$ of index l and length $n = ml$ is an R/I-submodule of the module $(R/I)^l$, where $R = \mathbb{F}_q[x]$, $I = \langle x^m - 1 \rangle \subset R$ and $\mathbb{F}_q$ is a finite field of characteristic p. Each codeword

$$\mathbf{c} = (c_{(0,0)}, \ldots, c_{(0,l-1)}, c_{(1,0)}, \ldots, c_{(1,l-1)}, \ldots, c_{(m-1,0)}, \ldots, c_{(m-1,l-1)}) \in \mathcal{C}$$

is associated with the polynomial vector

$$\mathbf{c} = (c_0 + I, c_1 + I, \ldots, c_{l-1} + I) \in \left(\frac{R}{I}\right)^l$$

where $c_i(x) + I = c_{(0,i)} + c_{(1,i)}x + \cdots + c_{(m-1,i)}x^{m-1} + I \in R/I$. We will usually drop the coset notation, referring to elements $\mathbf{c} + I$ of R/I as just $\mathbf{c}$ and writing

$$\mathbf{c} = (c_0, c_1, \ldots, c_{l-1}) \in \mathcal{C}$$

where $\deg(c_i) < m$ for all $i = 1, 2, \ldots, l-1$ and multiplication is performed mod $x^m - 1$.

Labeling the positions of the n-tuple $\mathbf{c} \in \mathcal{C}$ by $1, 2, \ldots, n$, the code is by definition invariant under the permutations π acting on each codeword's positions

$$\pi : \ i \mapsto i + l \bmod n$$

for $i = 1, \ldots, n$, and hence invariant under the cyclic group of automorphisms

$$\left\{1, \pi, \pi^2, \ldots, \pi^{m-1}\right\}$$

where $x^m = 1$ is the identity permutation. *It is this group of automorphisms which gives rise to the R/I-module structure of quasicyclic codes and allows us to apply the techniques of Groebner bases to the module context.*

There exists a surjective homomorphism ϕ between the polynomial vectors in $\mathbb{F}_q[x]^l = R^l$ and those in $(R/I)^l$ defined by

$$\begin{aligned} \phi : \ R^l &\rightarrow \left(\frac{R}{I}\right)^l \\ \mathbf{f} = (f_1, \ldots, f_l) &\mapsto (f_1 + I, f_2 + I, \ldots, f_l + I) \end{aligned}$$

The kernel of this map is the submodule $\tilde{\mathcal{K}}$ in R^l

$$\tilde{\mathcal{K}} = ker(\phi) = \left\{ \mathbf{f} \in R^l \mid \phi(\mathbf{f}) = \mathbf{0} \in \left(\frac{R}{I}\right)^l \right\}$$

where the zero vector is $\mathbf{0} = (I, I, \ldots, I) \in (R/I)^l$. More specifically,

$$\tilde{\mathcal{K}} = ker(\phi) = \left\{ \mathbf{f} = (f_1, \ldots, f_l) \in R^l \mid f_i = k(x^m - 1) \text{ for some } k \in \mathbb{F}_q[x], \ 1 \leq i \leq l \right\}$$

therefore $\tilde{\mathcal{K}}$ is the submodule generated by the set of elements

$$\begin{aligned} \tilde{\kappa} &= \{(x^m - 1, 0, \ldots, 0), (0, x^m - 1, 0, \ldots, 0), \ldots, (0, \ldots, 0, x^m - 1)\} \\ &= \{\mathbf{k}_i = (x^m - 1)\mathbf{e}_i, \ i = 1, \ldots, l\} \in R^l \end{aligned} \tag{4.3}$$

There exists an isomorphism

$$\frac{R^l}{\tilde{\mathcal{K}}} \cong \left(\frac{R}{I}\right)^l$$

between $(R^l)/\tilde{K}$ and $(R/I)^l$ given by the map

$$\theta : \ (f_1, f_2, \ldots, f_l) + \tilde{K} \mapsto (f_1 + I, f_2 + I, \ldots, f_l + I)$$

and thus a one-to-one correspondence between the submodules $\mathcal{C}$ of $(R/I)^l$ and the preimage submodules $\tilde{\mathcal{C}}$ of R^l which contain $\tilde{\mathcal{K}}$.

Above, one has made use of an idea that is the analogous for submodules of a result we already know for ideals [2006 Cox, Little & O'Shea]:

Proposition 4.1. *Let I be an ideal in $k[x_1, \ldots, x_n]$. The ideals in the quotient ring $k[x_1, \ldots, x_n]/I$ are in one-to-one correspondence with the ideals of $k[x_1, \ldots, x_n]$ containing I (that is, the ideals J satisfying $I \subset J \subset k[x_1, \ldots, x_n]$).*

A k-generator code $\mathcal{C}$ generated by elements $\mathbf{a}_1, \ldots, \mathbf{a}_k \in (R/I)^l$ is the submodule $=\langle \mathbf{a}_1, \ldots, \mathbf{a}_k \rangle \subset (R/I)^l$. Its preimage submodule $\tilde{\mathcal{C}}$ is the R-submodule in R^l generated by $\mathbf{a}_1, \ldots, \mathbf{a}_k \in R^l$ and the elements of $\tilde{\mathcal{K}}$

$$\tilde{\mathcal{C}} = \langle \mathbf{a}_1, \ldots, \mathbf{a}_k, (x^m - 1)\mathbf{e}_1, \ldots, (x^m - 1)\mathbf{e}_l \rangle \subset R^l \tag{4.4}$$

We will now find a Groebner basis of the preimage submodule $\tilde{\mathcal{C}} \subset R^l$. We use the POT monomial order with $\mathbf{e}_1 > \mathbf{e}_2 > \cdots > \mathbf{e}_l$ and the lex order.

Let $\tilde{\mathcal{C}}$ be a submodule in R^l which contains $\tilde{\mathcal{K}} = \langle \{(x^m - 1)\mathbf{e}_i \mid i = 1, \ldots, l\} \rangle$. Let $\tilde{\mathcal{G}}$ be minimal Groebner basis of $\tilde{\mathcal{C}}$ with respect to the formerly defined fixed monomial order, such that $\tilde{\mathcal{C}} = \langle \tilde{\mathcal{G}} \rangle$. Since $\tilde{\mathcal{K}} \subset \tilde{\mathcal{C}}$ contains a vector $(x^m - 1)\mathbf{e}_i$ for each $i = 1, \ldots, l$ with $\mathrm{LM}\left((x^m - 1)\mathbf{e}_i\right)$ in i-th position, there exists $\mathbf{g} \in \tilde{\mathcal{G}}$ such that $\mathrm{LM}(\mathbf{g})$ divides $\mathrm{LM}\left((x^m - 1)\mathbf{e}_i\right)$ and hence the leading monomial of $\mathbf{g}$ is in i-th position; thus, for each $i = 1, \ldots, l$ there exists $\mathbf{g} \in \tilde{\mathcal{G}}$ such that $\mathrm{LM}(\mathbf{g}) = X\mathbf{e}_i$ for some power $X = x^\beta \in R$ (*remark*: we dot use for these objects the name *monomial*, as we already use this in the context of modules; further, we talk about powers and not power products, since we are dealing with the one-variable case, therefore we are dealing with trivial products, in just one variable).

Let $\mathbf{g}_1$ and $\mathbf{g}_2$ be two elements in $\tilde{\mathcal{G}}$ as in the above paragraph, with leading monomials in the same position, say $\mathrm{LM}(\mathbf{g}_1) = X\mathbf{e}_i$ and $\mathrm{LM}(\mathbf{g}_2) = Y\mathbf{e}_i$ for some $i,\ 1 \leq i \leq l$. Since one of $X\mathbf{e}_i$ or $Y\mathbf{e}_i$ can be reduced by the other in R^l, it follows that if $\tilde{\mathcal{G}}$ is a minimal Groebner basis, then it contains exactly l elements, each with leading monomial in a different position, and we write $\tilde{\mathcal{G}} = \{\mathbf{g}_1, \ldots, \mathbf{g}_l\}$, and by reordering we may suppose $\mathrm{LM}(\mathbf{g}_i) = X_i\mathbf{e}_i,\ i = 1, \ldots, l$ for some powers $X_i \in R$.

Since in our case we have that $\mathrm{LM}(\mathbf{g}_i) = X_i\mathbf{e}_i,\ i = 1, \ldots, l$, by the properties of POT monomial order $\mathbf{e}_1 > \mathbf{e}_2 > \cdots > \mathbf{e}_l$, there are non nonzero monomials occurring in the j-th position for all $j,\ 1 \leq j \leq i - 1$ and a nonzero leading monomial must occur in the i-th position.

Therefore we have that $\mathbf{g}_i$ has the form $\mathbf{g}_i = (0, \ldots, 0, g_{ii}, g_{i,i+1}, \ldots, g_{il}) \in R^l$ and $g_{ii} \neq 0$, i.e. the elements of $\tilde{\mathcal{G}}$ have the form:

$$\begin{aligned}
\mathbf{g}_1 &= (g_{11}, g_{12}, g_{13}, \ldots, g_{1l}) && (4.5)\\
\mathbf{g}_2 &= (0, g_{22}, g_{23}, \ldots, g_{2l}) && (4.6)\\
\mathbf{g}_3 &= (0, 0, g_{33}, \ldots, g_{3l}) && (4.7)\\
&\vdots \\
\mathbf{g}_3 &= (0, 0, \ldots, 0, g_{ll}) && (4.8)
\end{aligned}$$

where the monic polynomials $0 \neq g_{ii} \in R,\ i = 1, \ldots, l$ are called the *diagonal elements* of $\tilde{\mathcal{G}}$.

We now investigate the form of a Groebner Basis (GB) $\tilde{\mathcal{G}} = \{\mathbf{g}_1, \mathbf{g}_2, \ldots, \mathbf{g}_l\}$ of the submodule $\tilde{\mathcal{C}} \subset R^l$, and some properties which result from such form.

Let us first notice that the diagonal elements of a GB $\tilde{\mathcal{G}}$ of $\tilde{\mathcal{C}}$ can be seen as a generalization of the generating polynomial of a cyclic code (see Eq.(3.4)).

The diagonal elements fulfil the following property:

Theorem 4.2. *For each $i,\ 1 \leq i \leq l$, the diagonal element g_{ii} divides $x^m - 1$.*

Proof. We saw before that, since $\tilde{\mathcal{K}} \subset \tilde{\mathcal{C}}$ contains a vector $(x^m - 1)\mathbf{e}_i$ for each $i = 1, \ldots, l$ with $\mathrm{LM}\left((x^m - 1)\mathbf{e}_i\right)$ in i-th position, there exists $\mathbf{g} \in \tilde{\mathcal{G}}$ such that $\mathrm{LM}(\mathbf{g})$ divides $\mathrm{LM}\left((x^m - 1)\mathbf{e}_i\right)$ and hence the leading monomial of $\mathbf{g}$ is in i-th position, that is the diagonal element g_{ii} divides $x^m - 1$. □

Each nonzero element of $\tilde{\mathcal{C}}$ may be expressed in the form $(0, \ldots, 0, c_r, \ldots, c_l)$, where $r \geq 1$ and $c_r \neq 0$. We now prove that c_r is divisible by the corresponding diagonal element g_{rr}:

Theorem 4.3. *If an element $\mathbf{f} \in \tilde{\mathcal{C}}$ has leading monomial in the r-th position, then g_{rr} divides the polynomial entry in this r-th position.*

Proof. By writing $(0, \ldots, 0, c_r, \ldots, c_l)$ as an $\mathbb{F}_q[x]$-linear combination $\sum_{i=1}^{l} a_i \mathbf{g}_i$, from Eqs.(4.5)÷(4.8) we can say that

$$\begin{aligned} c_r &= \sum_{i=1}^{l} a_{ir} g_{ir} \\ &= \sum_{i=1}^{r} a_{ir} g_{ir} \end{aligned}$$

where the second equality comes again from the Eqs.(4.5)÷(4.8), i.e. from the fact that $g_{ir} = 0,\ i = r+1, \ldots, l$.

But the $g_{ir} = 0,\ i = 1, \ldots, r-1$ do not contribute to c_r as well, being $\mathbf{c} = (0, \ldots, 0, c_r, \ldots, c_l)$, and therefore $\mathbf{g}_1, \ldots, \mathbf{g}_{r-1}$ does not play a role in generating $\mathbf{c}$ since, if they would, we would have nonzero elements in the first $r-1$ components of $\mathbf{c}$. □

Before proceeding, let us introduce a definition which we will use in a while:

Definition 4.3. *Let us write the polynomial $f \in \mathbb{F}_q[x],\ f \neq 0$, as a sum of terms in decreasing order $f = a_1 x^{\alpha_1} + a_2 x^{\alpha_2} + \cdots + a_s x^{\alpha_s}$, where $0 \neq a_i \in \mathbb{F}_q$, and $x^{\alpha_1} > x^{\alpha_2} > \cdots > x^{\alpha_s}$. We define the* **leading power** *of f, written $\mathrm{LP}(f)$, to be $\mathrm{LP}(f) = x^{\alpha_1}$.*

We will now prove two more results telling more about $\tilde{\mathcal{G}} = \{\mathbf{g}_1, \mathbf{g}_2, \ldots, \mathbf{g}_l\}$.

Theorem 4.4. *Let $\tilde{\mathcal{G}}$ be a Reduced Groebner Basis (RGB) of $\mathcal{C}$. Then for each $i,\ 1 \leq i \leq l$, the diagonal element g_{ii} is a monic polynomial and the nonzero off-diagonal polynomials g_{ij} of the element $\mathbf{g}_i$ are restricted in degree such that*

$$\deg(g_{ij}) < \deg(g_{jj}) \leq m$$

for all $j,\ 1 \leq i < j \leq l$.

Proof. Since $\tilde{\mathcal{G}}$ is an RGB, $\mathrm{LC}(\mathbf{g}_i) = 1$ for each $i,\ 1 \leq i \leq l$. Now $\mathrm{LM}(\mathbf{g}_i) = \mathrm{LP}(g_{ii}\mathbf{e}_i)$ and so $\mathrm{LC}(\mathbf{g}_i) = \mathrm{LC}(g_{ii}) = 1$. Thus g_{ii} is monic for each $i,\ 1 \leq i \leq l$.

Furthermore, being $\tilde{\mathcal{G}}$ an RGB, the are no nonzero polynomials in $\mathbf{g}_i$ that are divisible by $\mathrm{LM}(\mathbf{g}_j)$ for any $j \neq i,\ 1 \leq j \leq l$. Thus for each $i,\ 1 \leq i \leq l,\ \mathbf{g}_i = (0, \ldots, 0, g_{ii}, g_{i,i+1}, \ldots, g_{il}) \in \tilde{\mathcal{G}}$, $\mathrm{LM}(\mathbf{g}_j)$ does not divide any monomials in $g_{ij}\mathbf{e}_i$, for each $j,\ i < j \leq l$, and hence $\mathrm{LP}(g_{jj})$ does not divide g_{ij} for each $j,\ 1 \leq i < j \leq l$ as terms in R.

The latter is true in $R = \mathbb{F}_q[x]$ if and only if $\deg(g_{ij}) < \deg(g_{jj})$ for each $i, j,\ 1 \leq i < j \leq l$. Finally, as g_{jj} divides $x^m - 1$ for each $j,\ 1 \leq j \leq l$ then $\deg(g_{jj}) \leq m$. □

Theorem 4.5. *Let $\tilde{\mathcal{G}}$ be an RGB of $\tilde{\mathcal{C}}$. The element $\mathbf{g}_i$ is contained in $\tilde{\mathcal{K}}$ if and only if $\mathbf{g}_i = (0, \ldots, 0, g_{ii}, g_{i,i+1}, \ldots, g_{il}) = (x^m - 1)\mathbf{e}_i$.*

Proof. The element $\mathbf{g}_i \in \tilde{\mathcal{G}}$ has leading monomial in the i-th position and $g_{ii} | x^m - 1$. Assume $\mathbf{g}_i \in \tilde{\mathcal{K}}$, where $\tilde{\mathcal{K}} = \langle \{(x^m - 1)\mathbf{e}_i \mid i = 1, \ldots, l\} \rangle$, then the polynomial g_{ii} must be a multiple of $x^m - 1$, and so we have $g_{ii} = x^m - 1$.

Therefore, $\mathbf{g}_i = (0, \ldots, 0, x^m - 1, g_{i,i+1}, \ldots, g_{il})$ and since $(x^m - 1)\mathbf{e}_i \in \tilde{\mathcal{C}}$, then $\mathbf{f} = \mathbf{g}_i - (x^m - 1)\mathbf{e}_i = (0, \ldots, 0, 0, g_{i,i+1}, \ldots, g_{il}) \in \tilde{\mathcal{C}}$.

Since $\tilde{\mathcal{G}}$ is a GB for $\tilde{\mathcal{C}}$, the element $\mathbf{f} \in \tilde{\mathcal{C}}$ can be reduced to $\mathbf{0}$ modulo $\tilde{\mathcal{G}}$. Now, since no $\mathrm{LM}(\mathbf{g}_j)$ will divide $\mathbf{f}$ for $j = 1, \ldots, i$ (see Eqs.(4.5)÷(4.8)) we must have that the element $\mathbf{f}$ can actually be reduced to $\mathbf{0}$ modulo $\mathbf{g}_{i+1}, \mathbf{g}_{i+2}, \ldots, \mathbf{g}_l$.

Since $\deg(g_{ij}) < \deg(g_{jj})$ for each $j > i$ it follows that $\mathbf{g}_{i,i+1} = \cdots = \mathbf{g}_{il} = 0$, and then

$$\begin{aligned} \mathbf{g}_i &= (0, \ldots, 0, x^m - 1, g_{i,i+1}, \ldots, g_{il}) \\ &= (0, \ldots, 0, x^m - 1, 0, \ldots, 0) = (x^m - 1)\mathbf{e}_i \end{aligned}$$

The converse is simple to prove. If $\mathbf{g}_i = (x^m - 1)\mathbf{e}_i$, being $\tilde{\mathcal{K}} = \langle \{(x^m - 1)\mathbf{e}_i \mid i = 1, \ldots, l\} \rangle$, then $\mathbf{g}_i \in \tilde{\mathcal{K}}$ by definition. □

Any triangular set $\tilde{\mathcal{G}}$ is a Groebner basis of the submodule of $\mathbb{F}_q[x]^l$ that it generates. The condition that the submodule should contain $\tilde{\mathcal{K}}$ is equivalent to the existence of a matrix $\tilde{A} \in \mathrm{Mat}_l(\mathbb{F}[x])$ (with $\mathrm{Mat}_l(\mathbb{F}[x])$ indicating the square matrices of size $l \times l$ defined over $\mathbb{F}[x]$) such that

$$\tilde{A}\tilde{G} = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1l} \\ a_{21} & a_{22} & \cdots & a_{2l} \\ \vdots & \vdots & \ddots & \vdots \\ a_{l1} & a_{l2} & \cdots & a_{ll} \end{pmatrix} \begin{pmatrix} g_{11} & g_{12} & \cdots & g_{1l} \\ 0 & g_{22} & \cdots & g_{2l} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & g_{ll} \end{pmatrix} = (x^m - 1)I \tag{4.9}$$

where I is the identity matrix. It is immediate to prove that

Proposition 4.2. *The matrix $\tilde{A}$ is upper triangular.*

Proof. From $\tilde{A}\tilde{G} = (x^m - 1)I$ we have that $\tilde{A} = (x^m - 1)I\tilde{G}^{-1} = (x^m - 1)\tilde{G}^{-1}$, but being $\tilde{G}$ upper triangular, also $\tilde{G}^{-1}$ is upper triangular, and so is $\tilde{A}$ too. □

We will now prove that the entries of $\tilde{A}$ can be computed recursively from those of $\tilde{G}$:

Theorem 4.6. *The set $\tilde{\mathcal{G}}$ is a Groebner basis of a submodule $\mathbb{F}_q[x]^l$ containing $\tilde{\mathcal{K}}$ if and only if there exist a_{ij} for $1 \leq i, j \leq l$ satisfying*

$$a_{ij} = \begin{cases} 0 & \text{if } j < i \\ \frac{x^m - 1}{g_{ii}} & \text{if } j = i \\ \frac{-1}{g_{jj}} \left(\sum_{k=i}^{j-1} a_{ik} g_{kj} \right) & \text{if } j > i \end{cases} \tag{4.10}$$

Moreover the corresponding equations with the roles of g_{ij}, a_{ij} interchanged also hold, and $m - \deg(g_{ii}) = \deg(a_{ii})$ for all i. The Groebner basis is reduced if and only if $\deg(g_{ii}) > \deg(g_{ji})$ for all $j < i$, if and only if $\deg(a_{ii}) > \deg(a_{ij})$ for all $j > i$.

Proof. We already know from Prop. 4.2 that $\tilde{A}$ is upper triangular, so we have that $a_{ij} = 0$ if $j < i$. Further, by writing the equations associated to the diagonal elements of $(x^m - 1)I$ in Eq. (4.9), we have:

$$\begin{aligned} a_{11}g_{11} &= x^m - 1 && (4.11) \\ a_{21}g_{12} + a_{22}g_{22} &= x^m - 1 && (4.12) \\ &\vdots && (4.13) \\ \sum_{i=1}^{l} a_{li}g_{il} &= x^m - 1 && (4.14) \end{aligned}$$

and knowing that $a_{ij} = 0$ if $j < i$, we get that

$$a_{ij} = \frac{x^m - 1}{g_{ii}} \text{ if } j = i \tag{4.15}$$

At last, writing the equations corresponding the upper triangular part of $(x^m - 1)I$ in Eq. (4.9), we have:

$$\begin{aligned} a_{11}g_{12} + a_{12}g_{22} &= 0 && (4.16) \\ a_{11}g_{13} + a_{12}g_{23} + a_{13}g_{33} &= 0 && (4.17) \\ &\vdots && (4.18) \end{aligned}$$

from which we get

$$a_{12} = \frac{-1}{g_{22}}(a_{11}g_{12}) \tag{4.19}$$

$$a_{13} = \frac{-1}{g_{33}}(\sum_{k=1}^{2} a_{1k}g_{k3}) \tag{4.20}$$

$$\vdots \tag{4.21}$$

from which we can generalize to

$$a_{ij} = \frac{-1}{g_{jj}}(\sum_{k=i}^{j-1} a_{ik}g_{kj}) \text{ if } j > i \tag{4.22}$$

Eq. (4.10) holds also when interchanging the roles of g_{ij} and a_{ij}, if we assume to work over the field of fractions of $\mathbb{F}_q[x]$, indeed in this case the factors are invertible.

Suppose that the Groebner basis is reduced so that $\deg(g_{ii}) > \deg(g_{ji})$ for all $j < i$ (see Th. 4.4). The equation

$$a_{ii}g_{i,i+1} + a_{i,i+1}g_{i+1,i+1} = 0$$

e.g. $a_{11}g_{12} + a_{12}g_{22} = 0$ in Eq. (4.9), implies either that

$$g_{i,i+1} = a_{i,i+1} = 0$$

as we know from Eq. (4.10) that $a_{ii} \neq 0$ and from [2001 Lally & Fitzpatrick] (pg. 159) we know that $g_{i+1,i+1} \neq 0$, which follows indirectly from the fact that the diagonal components of $\tilde{G}$, g_{ii}, divide $x^m - 1$ for all i), or that

$$\deg(a_{ii}) + \deg(g_{i,i+1}) = \deg(a_{i,i+1}) + \deg(g_{i+1,i+1})$$

from which it follows that

$$\deg(a_{ii}) - \deg(a_{i,i+1}) = \deg(g_{i+1,i+1}) - \deg(g_{i,i+1}) > 0$$

where the inequality sign follows from Th. 4.4.

Using an induction argument, if $\deg(a_{ii}) > \deg(a_{ij})$ for $j = i+1, \ldots, k-1$ and $\deg(a_{ik}) \geq \deg(a_{ii})$ then the last summand on the left-hand side of the equation

$$\deg(a_{ii})\deg(g_{ik}) + \deg(a_{i,i+1})\deg(g_{i+1,k}) + \cdots + \deg(a_{i,k-1})\deg(g_{k-1,k}) + \deg(a_{ik})\deg(g_{kk}) = 0$$

has degree strictly greater than the degrees of the others (see Th. 4.4) and, as just mentioned above, $\deg(a_{ik}) \geq \deg(a_{ii}) > \deg(a_{ij})$, $j = i+1, \ldots, k-1$), which is a contradiction; indeed in that case, the equation above would then simplify to $a_{ik}g_{kk} = 0$, and since we know that $g_{kk} \neq 0$, which implies that $a_{ik} = 0$, with $k > i$, but this contradicts Eq. (4.10). Hence, $\deg(a_{ik}) < \deg(a_{ii})$ for all $j > i$ by induction, and the proof in one direction is complete. The converse is true by a symmetrical argument. □

A generating set for the code $\mathcal{C} \subseteq (R/I)^l$ can be found by mapping each element $\mathbf{g}_i \in \tilde{\mathcal{G}}$, $1 \leq i \leq l$, of the GB for $\tilde{\mathcal{C}} \subseteq R^l$ into $(R/I)^l$

$$\begin{aligned} \phi \quad &: \quad \mathbf{g}_i = (0, \ldots, 0, g_{ii}, g_{i,i+1}, \ldots, g_{il}) \\ &\mapsto \quad (0+I, \ldots, 0+I, g_{ii}+I, g_{i,i+1}+I, \ldots, g_{il}+I) \in \left(\frac{R}{I}\right)^l \end{aligned} \tag{4.23}$$

When $\tilde{\mathcal{G}}$ is an RGB, considering the elements $\mathbf{g}_i \in \tilde{\mathcal{G}}$ such that $\mathbf{g}_i \notin \tilde{\mathcal{K}},\ 1 \leq i \leq l$, these elements have polynomial entries of degree less than m, and we can therefore drop the coset notation and write the corresponding i-the generator of the code $\mathcal{C}$ as

$$\phi(\mathbf{g}_i) = (0, \ldots, 0, g_{ii}, g_{i,i+1}, \ldots, g_{il}) \in \mathcal{G}$$

where the entries of $\phi(\mathbf{g}_i)$ are identical to those in $\mathbf{g}_i = (0, \ldots, 0, g_{ii}, g_{i,i+1}, \ldots, g_{il}) \in \tilde{\mathcal{G}}$.

Moreover, an element $\mathbf{g}_i \in \tilde{\mathcal{G}}$ such that $\mathbf{g}_i \in \tilde{\mathcal{K}}$, then satisfies

$$\phi(\mathbf{g}_i) = (0, \ldots, 0) \in \left(\frac{R}{I}\right)^l$$

i.e. the zero element in $(R/I)^l$ and is thus omitted from the generating set for $\mathcal{C}$. We call this generating set $\mathcal{G}$ a *GB generating set* for the code $\mathcal{C}$, written $\mathcal{C} = \langle \mathcal{G} \rangle$; dropping the coset notation, we can write

$$\mathcal{G} = \left\{ \phi(\mathbf{g}_i) \mid \mathbf{g}_i \in \tilde{\mathcal{G}}, \mathbf{g}_i \notin \tilde{\mathcal{K}},\ 1 \leq i \leq l \right\}$$

Example 4.1

Let us consider a code of index $l = 3$ and length $n = ml = 21$, thus $m = 7$, over $\mathbb{F}_2$ generated by elements

$$\begin{aligned} \mathbf{a}_1 &= (x^5 + x^4 + 1, x^4 + x^3 + x + 1, x^4 + x^3 + x^2) \\ \mathbf{a}_2 &= (x^4 + x^3 + x^2 + 1, x, x^4 + x^3 + x + 1) \end{aligned}$$

in $(R/I)^l = (\mathbb{F}_2[x]/\langle x^7 - 1 \rangle)^3$. The GB for $\tilde{\mathcal{C}} = \langle \mathbf{a}_1, \mathbf{a}_2, (x^7 - 1)\mathbf{e}_1, (x^7 - 1)\mathbf{e}_2, (x^7 - 1)\mathbf{e}_3 \rangle \in R^l$ (see Eq. (4.4)) comprises the rows of

$$\begin{pmatrix} f_2 & f_1^2 & x^2 \\ 0 & f_3 & f_1 f_3 \\ 0 & 0 & x^7 - 1 \end{pmatrix} \tag{4.24}$$

with $f_1 = x + 1,\ f_2 = x^3 + x + 1,\ f_3 = x^3 + x^2 + 1$ so that $x^7 - 1 = f_1 f_2 f_3$, so the diagonal components of (4.24) are divisors of $x^7 - 1$. The corresponding RGB generating set of $\mathcal{C}_1$ consists of the rows of

$$\begin{pmatrix} f_2 & f_1^2 & x^2 \\ 0 & f_3 & f_1 f_3 \end{pmatrix}$$

since the third row of (4.24) is mapped by ϕ to the zero element of $(R/I)^l$, i.e. $(0, 0, 0)$.

Let us now see how this can be implemented through Singular:

```
ring R = 2,x,(c,lp);
% 2 indicates the binary field; (c,lp) indicates the POT extension of lex order
% c indicates downward ordering on the standard basis elements
> vector a1=[x5+x4+1,x4+x3+x+1,x4+x3+x2];
> vector a2=[x4+x3+x2+1,x,x4+x3+x+1];
> vector a3=[x7-1,0,0];
> vector a4=[0,x7-1,0];
> vector a5=[0,0,x7-1];
```

```
> module M=a1,a2,a3,a4,a5;
> print(M);
x5+x4+1,  x4+x3+x2+1,x7+1,0,   0,
x4+x3+x+1,x,          0,   x7+1,0,
x4+x3+x2, x4+x3+x+1, 0,   0,   x7+1
> option(redSB);
> module N=std(M);
> print(N);
0,   0,        x3+x+1,
0,   x3+x2+1,  x2+1
x7+1,x4+x2+x+1,x2
> N;
N[1]=[0,0,x7+1];
N[2]=[0,x3+x2+1,x4+x2+x+1]
N[3]=[x3+x+1,x2+1,x2]
> poly f1=x+1;
> poly f2=x3+x+1;
> poly f3=x3+x2+1;
> f2*gen(1)+f1^2*gen(2)+x2*gen(3);
[x3+x+1,x2+1,x2]
> f3*gen(2)+f1*f3*gen(3);
[0,x3+x2+1,x4+x2+x+1]
% the above verifies that we get the first two rows of (4.24) correctly
```

We will prove in a while a results which will tell us how, from the diagonal elements $g_{ii},\ 1 \leq i \leq l$ of a GB, we can determine the dimension of the code. Let us first introduce two results, which we will then use to prove the afore-mentioned result on the code's dimension:

Definition 4.4. *Given $f \in \mathbb{F}_q[x]$, we call the unique remainder $r \in \mathbb{F}_q[x]$, reduced with respect to $\mathcal{G}$, with $\mathcal{G}$ Groebner basis, the* **normal form** *of f with respect to $\mathcal{G}$, denoted $N_{\mathcal{G}}(f)$.*

Theorem 4.7. *Let us consider the quotient module R^l/M, for M a submodule of R^l and $\mathcal{G} = \{\mathbf{g}_1, \ldots, \mathbf{g}_t\}$ a Groebner basis for M. The set of all cosets containing monomials $\mathbf{M}$ in R^l such that $\mathrm{LM}(\mathbf{g}_i)$ does not divide $\mathbf{X}$ for all $i = 1, \ldots, t$, is a basis for the $\mathbb{F}_q$ vector space R^l/M.*

Let us now see how we can determine the code's dimension from the diagonal elements $g_{ii},\ 1 \leq i \leq l$:

Theorem 4.8. *The dimension k of the code $\mathcal{C} \subseteq (R/I)^l$ is the number of monomials $X\mathbf{e}_i = x^s\mathbf{e}_i$ not in normal form in $\tilde{\mathcal{C}}$, whose exponents satisfy $s \leq m-1$. Thus the dimension of $\mathcal{C}$ is*

$$k = ml - \sum_{i=1}^{l} \deg(g_{ii})$$

where $\tilde{\mathcal{G}} = \{\mathbf{g}_1, \ldots, \mathbf{g}_l\}$ is a GB for $\tilde{\mathcal{C}}$.

Proof. From Th. 4.7 a basis for $R^l/\tilde{\mathcal{C}}$ as a vector space over $\mathbb{F}_q$ is the set of all cosets of monomials $\mathbf{X}$ in R^l such that $\mathrm{LM}(\mathbf{g}_i)$ does not divide $\mathbf{X}$, i.e., the cosets of all monomials in

$$\{\mathbf{X} = x^r\mathbf{e}_i \mid 0 \leq r \leq \deg(g_{ii}) - 1\}$$

Similarly the set of all cosets of the monomials in $\{\mathbf{X} = x^r\mathbf{e}_i \mid 0 \leq r \leq m-1\}$ is a basis of the $\mathbb{F}_q$-vector space $R^l/\tilde{\mathcal{K}}$ since $\{(x^m-1)\mathbf{e}_i \mid i = 1, \ldots, l\}$ is a GB of $\tilde{\mathcal{K}}$.

Since

$$\begin{aligned}\dim(\tilde{\mathcal{C}}/\tilde{\mathcal{K}}) &= \dim(\tilde{\mathcal{C}}) - \dim(\tilde{\mathcal{K}}) \\ &= \dim(R^l) - \dim(\tilde{\mathcal{K}}) - \Big[\dim(R^l) - \dim(\tilde{\mathcal{C}})\Big] \\ &= \dim(R^l/\tilde{\mathcal{K}}) - \dim(R^l/\tilde{\mathcal{C}})\end{aligned}$$

we have that the dimension of $\tilde{\mathcal{C}}/\tilde{\mathcal{K}}$ as an $\mathbb{F}_q$-vector space is the number of monomials in $\{\mathbf{X} = x^r\mathbf{e}_i \mid 0 \leq r \leq m-1\}$, subtracted those in $\{\mathbf{X} = x^r\mathbf{e}_i \mid 0 \leq r \leq \deg(g_{ii}) - 1\}$, i.e. the dimension of $\tilde{\mathcal{C}}/\tilde{\mathcal{K}}$ is the number of monomials $x^s\mathbf{e}_i$ such that $\deg(g_{ii}) \leq s \leq m-1,\ i = 1, \ldots, l$, that is the set of monomials $x^s\mathbf{e}_i$ not in normal form in $\tilde{\mathcal{C}}$ (being $s \geq \deg(g_{ii})$ with $\tilde{\mathcal{G}}$ GB of $\tilde{\mathcal{C}}$), where $s \leq m-1$.

The cosets of $\tilde{\mathcal{C}}/\tilde{\mathcal{K}}$ are in one-to-one correspondence with the elements of $\mathcal{C} \subseteq (R/I)^l$ due to isomorphism

$$\theta : \ (g_1, \ldots, g_l) + \tilde{\mathcal{K}} \mapsto (g_1 + I, \ldots, g_l + I)$$

and therefore the dimension k of the code $\mathcal{C}$ is the number of elements in the set

$$\{\mathbf{X} = x^s\mathbf{e}_i \mid \deg(g_{ii}) \leq s \leq m-1\}$$

and therefore, subtracting for each i, from the m elements such that $x^s \mid 0 \leq m-1$ the $\deg(g_{ii})$ elements such that $x^s \mid 0 \leq \deg(g_{ii}) - 1$, we have that the dimension k is given by

$$\begin{aligned}k &= \sum_{i=1}^{l}(m - \deg(g_{ii})) \\ &= ml - \sum_{i=1}^{l}(\deg(g_{ii}))\end{aligned}$$

□

Example 4.2

Continuing from Example 4.1, we have that, being $l = 3,\ m = 7$ and the diagonal elements $g_{11} = x^3 + x + 1,\ g_{22} = x^3 + x^2 + 1,\ g_{33} = x^7 - 1$, the dimension k of the code $\mathcal{C}$ is

$$k = ml - \sum_{i=1}^{l}(\deg(g_{ii})) = 7 \cdot 3 - (3 + 3 + 7) = 8$$

Chapter 5

Decoding Issues related to Quasicyclic Codes

This chapter, after a short summary on quasicyclic codes' algebraic structure, deals with the decoding of quasicyclic codes in Groebner basis form, and then with the decoding of restriction-1 1-generator quasicyclic codes. Later on, some decoding algorithms for Reed-Solomon codes are discussed, and the last section deals with the decoding of quasicyclic codes formed by blocks, constituted in turn by Reed-Solomon codes.

5.1 Summary on Quasicyclic Codes' Algebraic Structure

Let $\mathcal{C}$ be a quasicyclic code of length lm and index l over $\mathbb{F}_q$. In the former Chapter, we have discussed a unique reduced triangular generating set of a code $\mathcal{C}$, regarded as a R-submodule of R^l, where $R = \mathbb{F}_q[x]/\langle x^m - 1\rangle$.

The code is the image of an $\mathbb{F}_q[x]$-submodule $\tilde{\mathcal{C}}$ of $\mathbb{F}_q[x]^l$ containing $\tilde{\mathcal{K}} = \langle (x^m - 1)\mathbf{e}_i,\ i = 1, \ldots, l\rangle$, under the natural homomorphism $\varphi :\ \mathbb{F}_q[x]^l \to R^l,\ (c_1, \ldots, c_l) \mapsto (c_1 + \langle x^m - 1\rangle, \ldots, c_l + \langle x^m - 1\rangle)$. The Reduced Triangular Basis (RTB) of $\tilde{\mathcal{C}}$ is a unique set of vectors $\tilde{\mathcal{G}} = \{\mathbf{g}_i = (0, \ldots, g_{ii}, \ldots, g_{il}),\ i = 1, \ldots, l\}$ where:

i. the diagonal element g_{ii} is monic and nonzero;

ii. for the off-diagonal elements it is $\deg(g_{ki}) < \deg(g_{ii})$ for $k < i$.

The set $\mathcal{G}$ in R^l consisting of the elements of $\tilde{\mathcal{G}}$ not mapped to zero under φ forms an R-generating set for the code $\mathcal{C}$. The dimension of code $\mathcal{C}$ can be obtained directly from the diagonal elements as $lm - \sum_{i=1}^{l} \deg(g_{ii}) = \sum_{i=1}^{l} (m - \deg(g_{ii}))$.

5.2 Decoding of Quasicyclic Codes in Groebner Basis Form

For any given information vector

$$\mathbf{v} = (v_1, \ldots, v_l) \in (R/\langle x^m - 1\rangle)^l$$

where $\deg(v_i) < m - \deg(g_i^i)$, the codeword $\mathbf{c} = (c_1, \ldots, c_l) \in \mathcal{C}$ is

$$\mathbf{c} = v_1\mathbf{g}_1 + v_2\mathbf{g}_2 + \cdots + v_l\mathbf{g}_l \tag{5.1}$$

where $\mathbf{g}_i$ is the i-th row of the matrix $\tilde{G}$ representing the triangular set of rows $\tilde{\mathcal{G}} = \{\mathbf{g}_1, \ldots, \mathbf{g}_l\}$

$$\tilde{G} = \begin{pmatrix} g_{11} & g_{12} & \cdots & g_{1l} \\ 0 & g_{22} & \cdots & g_{2l} \\ & \cdots & \cdots & \\ 0 & 0 & \cdots & g_{ll} \end{pmatrix} \tag{5.2}$$

Then we can rewrite Eq. (5.1) as

$$\mathbf{c} = (v_1 g_{11}, v_1 g_{12} + v_2 g_{22}, \ldots, v_1 g_{1l} + v_2 g_{2l} + \cdots + v_l g_{ll}) \bmod x^m - 1 \tag{5.3}$$

Suppose that $\mathbf{c} = (c_1, c_2, \ldots, c_l) \in \mathcal{C}$ is transmitted over a channel and an erroneous vector $\mathbf{r} = (r_1, r_2, \ldots, r_l) \in (R/\langle x^m - 1\rangle)^l$ is received. We denote by $\mathbf{e}$ the error vector

$$\mathbf{e} = (e_1, e_2, \ldots, e_l) \in (R/\langle x^m - 1\rangle)^l$$

where

$$\mathbf{r} = \mathbf{c} + \mathbf{e}$$

and the component polynomials are

$$r_i = c_i + e_i \text{ for all } i,\ 1 \le i \le l$$

We denote by $\mathrm{wt}(a)$ the number of nonzero coefficients in a. The weight of the vector $\mathbf{v} = (v_1, \ldots, v_l) \in (R/\langle x^m - 1\rangle)^l$, written $\mathrm{wt}(\mathbf{v})$, is equal to

$$\mathrm{wt}(\mathbf{v}) = \sum_{j=1}^{l} wt(v_j)$$

that is, the sum of the partial weights in $\mathbf{v}$ where $v_j,\ 1 \le j \le l$, is a vector component. We denote by $d_i^*,\ 1 \le i \le l$, the BCH distance of the cyclic code generated by $g_{ii} \ne x^m - 1,\ 1 \le i \le l$, and

$$d_i^* = \#ConsecutiveRoots(g_{ii}) + 1$$

in the splitting field $\mathbb{F}_{q^t}$ of $x^m - 1$ over $\mathbb{F}_q$.

From Eq. (5.3) we have that the first partial codeword

$$c_1 = v_1 g_{11} \bmod x^m - 1$$

is a codeword in the cyclic code generated by g_{11}, and so the polynomial

$$r_1 = c_1 + e_1 = v_1 g_{11} \bmod x^m - 1$$

can be decoded correctly to c_1 if the error polynomial e_1 is of weight at most

$$t_1 = \left\lfloor \frac{d_1^* - 1}{2} \right\rfloor$$

The information polynomial v_1 can be retrieved by direct division, being

$$\deg(v_1) < m - \deg(g_{11})$$

and therefore

$$\begin{aligned} c_1 &= v_1 g_{11} \bmod x^m - 1 \\ &= v_1 g_{11} \end{aligned}$$

The polynomial

$$\begin{aligned} r_2 &= c_2 + e_2 \\ &= v_1 g_{12} + v_2 g_{22} + e_2 \bmod x^m - 1 \end{aligned}$$

thus

$$\begin{aligned} r_2^{'} &= r_2 - v_1 g_{12} \\ &= v_2 g_{22} + e_2 \bmod x^m - 1 \end{aligned}$$

can be decoded correctly to $c_2^{'} = v_2 g_{22}$, which is a codeword in the cyclic code generated by g_{22}, if the error polynomial has weight

$$\mathrm{wt}(e_2) \leq t_2 = \left\lfloor \frac{d_2^* - 1}{2} \right\rfloor$$

We can compute directly the partial codeword

$$c_2 = c_2^{'} + v_1 g_{12} \bmod x^m - 1$$

since we have that

$$\deg(v_2) < m - \deg(g_{22})$$

In a similar way, for each $i = 2, \ldots, l$

$$\begin{aligned} r_i &= c_i + e_i \\ &= \sum_{j=1}^{i} v_j g_{ji} + e_i \bmod x^m - 1 \end{aligned}$$

where

$$\begin{aligned} r_i^{'} &= r_i - \sum j = 1^{i-1} v_j g_{ji} \\ &= v_i g_{ii} + e_i \bmod x^m - 1 \end{aligned}$$

can be decoded correctly to $c_i^{'} = v_i g_{ii} \in \langle g_{ii} \rangle \subseteq R/\langle x^m - 1 \rangle$, if the weight of the error polynomial is such that

$$\mathrm{wt}(e_i) \leq t_i = \left\lfloor \frac{d_i^* - 1}{2} \right\rfloor$$

The partial codeword

$$c_i = c_i^{'} + \sum_{j=1}^{i-1} v_j g j i \bmod x^m - 1$$

and the information polynomial

$$v_i = \frac{c_i^{'}}{g_{ii}}$$

can be found directly if

$$\deg(v_i) < m - \deg(g_{ii})$$

If for some $i,\ 1 \leq i \leq l$, there exists $\mathbf{g}_i \in \tilde{\mathcal{G}}$ for which $\phi(\mathbf{g}_i) \notin \mathcal{G}$ then $\mathbf{g}_i \in \tilde{\mathcal{K}}$ with $g_{ii} = x^m - 1$ which is mapped to the zero vector in $(R/\langle x^m - 1\rangle)^l$. In this case we can omit the information component v_i and the generator $\mathbf{g}_i$ from the encoding procedure in Eq. (5.3), and we get

$$\begin{aligned} r_i &= c_i + e_i \\ &= \sum_{j=1}^{i-1} v_j g_{ji} + e_i \bmod x^m - 1 \end{aligned} \tag{5.4}$$

where

$$r_i - \sum_{j=1}^{i-1} v_j g_{ji} = e_i \bmod x^m - 1 \tag{5.5}$$

and

$$c_i = r_i - e_i \bmod x^m - 1$$

can be computed from the previous steps, Eq. (5.4) and Eq. (5.5), without any need for cyclic decoding at this stage. The polynomial r_i can then contain an arbitrary number of errors and we then write

$$t_i = m \quad \text{when} \quad g_{ii} = x^m - 1$$

Summarizing, we can then decode each polynomial r_i to c_i and thus retrieve the original codeword

$$\mathbf{c} = (c_1, \ldots, c_l) \in \mathcal{C}$$

subject to the condition

$$\mathrm{wt}(e_i) \leq t_i \quad \text{for each } i,\ 1 \leq i \leq l$$

where the last equation imposes an upper bound on the number of errors which can be successfully decoded.

Let us introduce the following definition [LallyPhD]:

Definition 5.1. *Let the set $\tilde{\mathcal{G}} = \{\mathbf{g}_1, \mathbf{g}_2, \ldots, \mathbf{g}_l\} \subseteq R^l$ be a GB for a submodule $\tilde{\mathcal{C}} \subseteq R^l$. We say that $\tilde{\mathcal{G}}$ is an* **r-level** *GB for $\tilde{\mathcal{C}}$ if there exists $\mathbf{g}_r \in \tilde{\mathcal{G}},\ 1 \leq r \leq l$, such that $\mathbf{g}_r \notin \tilde{\mathcal{K}}$ and $\mathbf{g}_j \in \tilde{\mathcal{K}}$ for all $j,\ r < j \leq l$. The corresponding GB generating set $\mathcal{G}$ for the code $\mathcal{C}$ we also term an* **r-level** *GB generating set as it contains at most r generators*

$$\mathcal{G} = \left\{ \phi(\mathbf{g}_i) | g_i \in \tilde{\mathcal{G}},\ \mathbf{g}_i \notin \tilde{\mathcal{K}},\ 1 \leq i \leq r \right\}$$

A particular case of the algorithm described above, is that of a 1-level GB code. A code $\mathcal{C} \subseteq (R/\langle x^m - 1\rangle)^l$ of index l and length ml has a 1-level GB if it is generated by a single generator $\mathbf{g}_1 \in (R/\langle x^m - 1\rangle)^l$ of the form

$$\mathbf{g}_1 = (g_{11}, f_1 g_{11}, \ldots, f_{l-1} g_{11}) \in (R/\langle x^m - 1\rangle)^l$$

where g_{11} divides $x^m - 1$ and $\deg(f_i) < m - \deg(g_{11})$, $1 \leq i \leq l-1$. Each codeword $\mathbf{c} = (c_1, c_2, \ldots, c_l) \in \mathcal{C}$ has the form

$$\begin{aligned} \mathbf{c} &= a\mathbf{g}_1 \\ &= (a_1 g_{11}, a f_1 g_{11}, \ldots, a f_{l-1} g_{11}) \bmod x^m - 1 \end{aligned} \tag{5.6}$$

where a is an information polynomial with $\deg(a) < m - \deg(g_{11})$. If the transmitted codeword is erroneously received as

$$\mathbf{r} = \mathbf{c} + \mathbf{e}$$

where $\mathbf{e}$ is the error vector

$$\mathbf{e} = (e_1, e_2, \ldots, e_l) \in \left(\frac{R}{I}\right)^l$$

and component polynomials are

$$r_i = c_i + e_i \text{ for all } i,\ 1 \leq i \leq l$$

The first received polynomial

$$r_1 = c_1 + e_1 = a g_{11} + e_1 \bmod x^m - 1$$

can be decoded correctly to c_1 by cyclic decoding if

$$\mathrm{wt}(e_1) \leq t_1 = \left\lfloor \frac{d_1^* - 1}{2} \right\rfloor$$

and the remaining partial codewords are given by (see Eq. (5.6))

$$c_i = a f_{i-1} g_{11} = f_{i-1} c_1 \bmod x^m - 1 \quad 2 \leq i \leq l$$

and $t_i = m,\ i = 2, \ldots, l$. We then see that a code in 1-level GB form can be decoded correctly if at most t_1 errors occur in the first m-tuple of received digits and any number of errors can occur in the remaining $(l-1)m$ digits.

Notice that since $f_i,\ 1 \leq i \leq l-1$ are not necessarily relatively prime to $x^m - 1$, the partial codewords do not all lie in the same cyclic code. Thus the information vector a may not be recoverable from any r_i other than r_1 (if f_i, not relatively prime with respect to $x^m - 1$, it is mapped to zero by ϕ). In the next section, we will see how to impose restrictions such that this will be possible [LallyPhD].

5.3 Decoding of Restriction-1 1-generator Quasicyclic Codes

Given the 1-generator code with generator $\mathbf{g} = (g_1, g_2, \ldots, g_l)$ we can permute the "partial generators" of this vector to obtain a generator $\sigma\mathbf{g} = (g_{\sigma(1)}, g_{\sigma(2)}, \ldots, g_{\sigma(l)})$ of an equivalent code with the same weight structure.

Let us introduce the following theorem:

Theorem 5.1. *A 1-generator code generator code generated by* $\mathbf{g} = (g_1, g_2, \ldots, g_l)$*, of dimension* $k = m - \deg(g)$*, where* $g = \gcd(g_1, g_2, \ldots, g_l, x^m - 1)$*, has a 1-level GB and all codes generated by permutations of its partial generators have 1-level GBs if and only if* $\mathbf{g}$ *is of the form*

$$\mathbf{g} = (f_1 g, f_2 g, \ldots, f_l g) \in (R/\langle x^m - 1\rangle)^l, \ f_i \in R \tag{5.7}$$

where $g | x^m - 1$*,* $\gcd(f_i, (x^m - 1)/g) = 1$ *and* $\deg(f_i) < m - \deg(g), \ 1 < i \leq l$*.*

We refer to 1-generator codes with generator $\mathbf{g}$ of the form as in Eq. (5.7) as *restriction-1 codes*.

We now present a decoding algorithm for a subclass of restriction-1 1-generator codes, which can correct a greater range of error patterns than the 1-level decoding algorithm discussed in Section 5.2.

Let m be relatively prime to the characteristic of the field $\mathbb{F} = \mathbb{F}_q$. Let $\mathcal{C} \subset (R/\langle x^m - 1\rangle)^l$ be a 1-generator code of index l and length ml, generated by the single generator of the form

$$\mathbf{g} = (f_1 g, f_2 g, \ldots, f_l g) \in (R/\langle x^m - 1\rangle)^l, \ f_i \in R \tag{5.8}$$

where $g | x^m - 1$, $\gcd(f_i, x^m - 1) = 1$ and $\deg(f_i) < m - \deg(g), \ 1 \leq i \leq l$. From Th. 5.1 we see that the code $\mathcal{C}$ is a restriction-1 1-generator code, of dimension $k = m - \deg(g)$ and minimum distance

$$d_{min} \geq ld^* = l(\#ConsecutiveRoots(g) + 1) \tag{5.9}$$

Each codeword $\mathbf{c} = (c_1, c_2, \ldots, c_l)$ has the form

$$\begin{aligned} \mathbf{c} &= a\mathbf{g} \\ &= (af_1 g, af_2 g, \ldots, af_l g) \bmod x^m - 1 \end{aligned} \tag{5.10}$$

where $a \in R$ is an information polynomial of $\deg(a) < m$, and each partial codeword $af_i g, \ 1 \leq i \leq l$, is a partial codeword in the cyclic code of length m generated by $\gcd(f_i g, x^m - 1) = g$ for all $i = 1, \ldots, l$. If any partial codeword $c_i, \ 1 \leq i \leq l$, of the codeword $\mathbf{c} \in \mathcal{C}$ is known then all other partial codewords can be found from the equations (see Eq. (5.10)):

$$c_j = c_i f_i^{-1} f_j \bmod x^m - 1 \tag{5.11}$$

for each $j \neq i, \ 1 \leq j \leq l$, since $\gcd(f_i, x^m - 1) = 1$ and f_i^{-1} exists modulo $x^m - 1$.

We denote by D the minimum distance of the code $\mathcal{C}$; according to [LallyPhD], we can choose the multipliers $f_1, f_2, \ldots, f_l$ in the generator $\mathbf{g}$ to find a code of the same length, index and dimension for which D is as large as possible and often very much larger than the lower bound ld^*,

$$D = d_{min} \geq ld^*$$

Suppose that the codeword $\mathbf{c} = (c_1, c_2, \ldots, c_l) \in \mathcal{C}$ is transmitted over a channel and the vector $\mathbf{r} = (r_1, r_2, \ldots, r_l) \in (R/\langle x^m - 1\rangle)^l$ is erroneously received. We denote by $\mathbf{e}$ the error vector

$$\mathbf{e} = (e_1, e_2, \ldots, e_l) \in (R/\langle x^m - 1\rangle)^l$$

where

$$\mathbf{r} = \mathbf{c} + \mathbf{e} = (c_1 + e_1, c_2 + e_2, \ldots, c_l + e_l)$$

and the component polynomials

$$r_i = c_i + e_i \text{ for all } i, \ 1 \leq i \leq l$$

We now describe a bounded distance decoding algorithm for this code which can correct any pattern of at most t errors where

$$t = \min \begin{cases} l \left\lfloor \frac{d^*-1}{2} \right\rfloor + l - 1 \\ \lfloor \frac{D-1}{2} \rfloor \end{cases}$$

and if $t < \lfloor (D-1)/2 \rfloor$ can also correct certain patterns of up to $\lfloor (D-1)/2 \rfloor$ errors. The algorithm is based on successive repetitions of the same cyclic decoding procedure:

Input: $\mathbf{r} = (r_1, r_2, \ldots, r_l),\ \mathbf{g} = (f_1 g, f_2 g, \ldots, f_l g)$
Initialization: set $\mathbf{c} = (0, 0, \ldots, 0),\ i = 0$
while $i < l$ **do**
  i=i+1;
  Decode r_i as a cyclic codeword in $\langle g \rangle \subseteq R/\langle x^m - 1 \rangle$
  if r_i decodes to a cyclic codeword $c_i^{'}$ **then**
    Compute $c_j^{'} = c_i^{'} f_i^{-1} f_j \bmod x^m - 1$ for all $j \neq i,\ 1 \leq j \leq l$
    Construct the codeword $\mathbf{c}^{'} = (c_1^{'}, c_2^{'}, \ldots, c_l^{'}) \in \mathcal{C} \subseteq (R/\langle x^m - 1 \rangle)$
    if $wt(|\mathbf{c}^{'} - \mathbf{r}|) \leq \lfloor (D-1)/2 \rfloor$ **then**
      set $\mathbf{c} = \mathbf{c}^{'}$
      break;
    else
      next;
    end if
  else
    if r_i not decoded **then**
      next;
    end if
  end if
end while
Output: $\mathbf{c} = (c_1, \ldots, c_l)$

Example 5.1

Let us make a couple of examples, with $d^* = \#ConsecutiveRoots(g) + 1$ in turn assumed to be even and odd.

Consider the binary code $\mathcal{C}(28, 3, 16)$ of index $l = 4$ and length $ml = 7l = 28$ generated by

$$\mathbf{g} = (1 + x + x^2 + x^4, 1 + x + x^2 + x^4, 1 + x + x^2 + x^4, 1 + x + x^2 + x^4) \in (R/\langle x^7 - 1 \rangle)^4$$

The designed minimum distance of the cyclic code $\langle 1 + x + x^2 + x^4 \rangle \subseteq R/\langle x^7 - 1 \rangle$ is $d^* = 4$. The minimum distance of the code $\mathcal{C} \subseteq (R/\langle x^7 - 1 \rangle)^4$ is $D = 16 = ld^*$ and so the decoding algorithm introduced above can correct all patterns with the following number of errors

$$t = \min \begin{cases} l \left\lfloor \frac{d^*-1}{2} \right\rfloor + l - 1 = 4 \left\lfloor \frac{4-1}{2} \right\rfloor + 4 - 1 = 7 \\ \lfloor \frac{D-1}{2} \rfloor = \lfloor \frac{16-1}{2} \rfloor = 7 \end{cases}$$

i.e. $t = 7$ errors.

Consider the binary code $\mathcal{C}(28, 4, 12)$ of index $l = 4$ and length $ml = 7l = 28$ generated by

$$\mathbf{g} = (1 + x + x^3, 1 + x + x^3, 1 + x + x^3, 1 + x + x^3) \in (R/\langle x^7 - 1\rangle)^4$$

The designed minimum distance of the cyclic code $\langle 1 + x + x^3\rangle \subseteq R/\langle x^7 - 1\rangle$ is $d^* = 3$. The minimum distance of the code $\mathcal{C} \subseteq (R/\langle x^7 - 1\rangle)^4$ is $D = 12 = ld^*$ and so the decoding algorithm introduced above can correct all patterns with the following number of errors

$$t = \min \begin{cases} l \left\lfloor \frac{d^*-1}{2} \right\rfloor + l - 1 = 4 \left\lfloor \frac{3-1}{2} \right\rfloor + 4 - 1 = 7 \\ \left\lfloor \frac{D-1}{2} \right\rfloor = \left\lfloor \frac{12-1}{2} \right\rfloor = 5 \end{cases}$$

i.e. $t = 5$ errors.

5.4 RS Decoding Algorithms

Let us fix a field $\mathbb{F}_q$ and a primitive element α, and consider the Reed-Solomon code $\mathcal{C} \subset \mathbb{F}_q/\langle x^{q-1} - 1\rangle$ given by a generator polynomial

$$g = (x - \alpha) \cdots (x - \alpha^{d-1}) \tag{5.12}$$

of degree $d - 1$. We assume that $d = 2t + 1$ for some $t \geq 1$, then by the following result

Proposition 5.1. *Let $\mathcal{C}$ be a code with minimum distance d. All error vectors of weight $\leq d - 1$ can be detected. Moreover, if $d \geq 2t + 1$, then all error vectors of weight $\leq t$ can be corrected by nearest neighbor decoding.*

any error vector of weight t or less should be correctable.

Let $c = \sum_{j=0}^{q-2} c_j x^j$ be a codeword of $\mathcal{C}$. Since $\mathcal{C}$ has generator polynomial $g(x)$, then in $\mathbb{F}_q[x]$, c is divisible by g. Suppose that c is transmitted, but there are some errors, so the received word is $y = c + e$ for some $e = \sum_{i \in I} e_i x^i$, where I is called the set of *error locations* and the coefficients e_i are known as the *error values*. To decode we must thus solve the following problem:

Problem 5.1

Given a received codeword y, determine the set of error locations I and the error values e_i.

We can determine whether errors have occurred by computing the values $E_j = y(\alpha^j)$, $j = 1, \ldots, d-1$. If $E_j = y(\alpha^j) = 0$ for all $j = 1, \ldots, d - 1$, then y is divisible by g (see Eq.(5.12)). Assuming the error vector has a weight at most t, y must be the codeword we intended to send.

If some $E_j \neq 0$, it means there are errors; we can try to use the information included in the E_j to solve Problem 5.4. Note that the E_j are the values of the error polynomial for $j = 1, \ldots, d - 1$:

$$E_j = y(\alpha^j) = c(\alpha^j) + e(\alpha^j) = e(\alpha^j) \tag{5.13}$$

being c a multiple of g (see Eq.(5.12)). The polynomial

$$S(x) = \sum_{j=1}^{d-1} E_j x^{j-1} \tag{5.14}$$

is called the *syndrome polynomial* for y. We can see from Eq.(5.14) that this polynomial has degree lower than or equal to $d-2$.

By extending the definition of $E_j = e(\alpha^j)$ to all exponents j we can consider the formal power series

$$E(x) = \sum_{j=1}^{\infty} E_j x^{j-1} \tag{5.15}$$

Suppose we knew the error polynomial e. Then, from Eq.(5.13) we have that

$$E_j = \sum_{i \in I} e_i (\alpha^j)^i = \sum_{i \in I} e_i (\alpha^i)^j \tag{5.16}$$

Let us make some manipulations with Eq.(5.15). First, let us plug Eq.(5.16) into Eq.(5.15), obtaining:

$$E(x) = \sum_{j=1}^{\infty} E_j x^{j-1} = \sum_{j=1}^{\infty} \sum_{i \in I} e_i (\alpha^i)^j x^{j-1}$$

We can also write that

$$\begin{aligned} E(x) &= \sum_{j=1}^{\infty} \sum_{i \in I} e_i (\alpha^i)^j x^{j-1} \\ &= \sum_{i \in I} \sum_{j=1}^{\infty} e_i \alpha^i (\alpha^i x)^{j-1} \\ &= \sum_{i \in I} e_i \alpha^i \sum_{j=1}^{\infty} (\alpha^i x)^{j-1} \\ &= \sum_{i \in I} e_i \alpha^i \sum_{k=0}^{\infty} (\alpha^i x)^k \end{aligned}$$

where the last equality follows from the index change $k = j - 1$. We can further write:

$$\begin{aligned} E(x) &= \sum_{i \in I} e_i \alpha^i \sum_{k=0}^{\infty} (\alpha^i x)^k \\ &= \sum_{i \in I} \frac{e_i \alpha^i}{1 - \alpha^i x} \end{aligned}$$

which we can rewrite as

$$\begin{aligned} E(x) &= \sum_{i \in I} \frac{e_i \alpha^i}{1 - \alpha^i x} \\ &= \sum_{i \in I} e_i \alpha^i \frac{\prod_{j \neq i,\ j \in I} (1 - \alpha^j x)}{\prod_{i \in I} (1 - \alpha^i x)} \\ &= \frac{\Omega(x)}{\Lambda(x)} \end{aligned} \tag{5.17}$$

where

$$\Omega(x) = \sum_{i\in I} e_i\alpha^i \prod_{j\neq i,\ j\in I}(1-\alpha^j x) \tag{5.18}$$

$$\Lambda(x) = \prod_{i\in I}(1-\alpha^i x) \tag{5.19}$$

The roots of Λ are precisely the α^{-i} for $i \in I$. Since the error locations can be determined from these roots, we call Λ the *error locator polynomial*. From Eqs.(5.18)-(5.19) we see that

$$\deg(\Omega) \leq \deg(\Lambda) - 1$$

Moreover

$$\Omega(\alpha^{-i}) = e_i\alpha^i \prod_{j\neq i,\ j\in I}(1-\alpha^j\alpha^{-i}) \neq 0 \tag{5.20}$$

hence Ω has no roots in common with Λ (being $j \neq i$ in Eq.(5.20), see also Eq.(5.19)). From this, and the fact that Ω and Λ are products of linear terms (see (5.18)-(5.19)), we deduce that the polynomials Ω and Λ must be relatively prime.

Let us consider the "tail" of the series E and do some manipulations of it

$$\begin{aligned}
E(x) - S(x) &= \sum_{j=d}^{\infty}(\sum_{i\in I} e_i(\alpha^i)^j)x^{j-1} \\
&= \sum_{i\in I}\sum_{j=d}^{\infty} e_i\alpha^i(\alpha^i x)^{j-1} \\
&= \sum_{i\in I} e_i\alpha^i \sum_{j=d}^{\infty}(\alpha^i x)^{j-1} \\
&= \sum_{i\in I} e_i\alpha^i(\alpha^i x)^{d-1}\sum_{j=1}^{\infty}(\alpha^i x)^{j-1} \\
&= \sum_{i\in I} e_i\alpha^i(\alpha^i x)^{d-1}\sum_{k=0}^{\infty}(\alpha^i x)^{k} \\
&= \sum_{i\in I} e_i\alpha^i(\alpha^i x)^{d-1}\frac{1}{1-\alpha^i x} \\
&= x^{d-1}\sum_{i\in I} e_i\alpha^{id}\frac{1}{1-\alpha^i x} \\
&= x^{d-1}\frac{\sum_{i\in I} e_i\alpha^{id}\prod_{j\neq i,\ j\in I}(1-\alpha^j x)}{\prod_{i\in I}(1-\alpha^i x)} \\
&= x^{d-1}\frac{\Gamma(x)}{\Lambda(x)}
\end{aligned} \tag{5.21}$$

where $\Gamma(x) = \sum_{i\in I} e_i\alpha^{id}\prod_{j\neq i,\ j\in I}(1-\alpha^j x)$ and $\Lambda(x)$ as already defined in Eq.(5.19), with $\deg(\Gamma(x)) \leq \deg(\Lambda(x)) - 1$.

From Eq.(5.17) we have that

$$\Omega = \Lambda E \tag{5.22}$$

and from Eq.(5.21) we have that

$$E = S + x^{d-1}\frac{\Gamma}{\Lambda} \tag{5.23}$$

and therefore, plugging (5.23) into (5.22), we obtain

$$\begin{aligned} \Omega &= \Lambda E \\ &= \Lambda S + x^{d-1}\Gamma \\ &= \Lambda S + x^{2t}\Gamma \end{aligned} \tag{5.24}$$

where the last equality comes from the previous assumption $d = 2t + 1$, and thus $d - 1 = 2t$.

It might be in some cases more convenient to look at (5.24) as a congruence, i.e.

$$\Omega \equiv \Lambda S \bmod x^{2t} \tag{5.25}$$

Conversely, if (5.25) holds, there is some polynomial Γ such that (5.24) holds. The congruence (5.25) is called the *key equation* for decoding.

The derivation of the key equation (5.25) assumed that e was known; let us now consider an actual decoding problem, and assume an error vector of weigth at most t.

Given the received word y, S is computed:

$$S(x) = \sum_{j=1}^{d-1} E_j x^{j-1} = \sum_{j=1}^{d-1} y(\alpha^j) x^{j-1}$$

The key equation (5.25) can then be viewed as a relation between the known polynomials $S,\ x^{2t}$, and the unknowns $\Omega,\ \Lambda$.

Suppose that a solution $(\overline{\Omega}, \overline{\Lambda})$ of the key equation is found, which satisfies the following *degree conditions*

$$\begin{cases} \deg(\overline{\Lambda}) \leq t \\ \deg(\overline{\Omega}) < \deg(\overline{\Lambda}) \end{cases} \tag{5.26}$$

where the first condition comes from the fact that we assumed the error vector of weight at most t and the second is a direct consequence of Eqs.(5.18)-(5.19), and in which $\overline{\Omega},\ \overline{\Lambda}$ are relatively prime.

The following result tells us that a solution of the key equation (5.25) satisfying the degree conditions (5.26) is unique, up to a constant multiple:

Theorem 5.2. *Let S be the syndrome polynomial corresponding to a received word y with an error of weight at most t. Up to a constant multiple, there exists a unique solution (Ω, Λ) of (5.25) that satisfies the degree conditions (5.26), and in which Ω and Λ are relatively prime.*

Proof. We know that the error locator Λ and the corresponding Ω give one such solution; let $(\overline{\Omega}, \overline{\Lambda})$ be any other. Let us write the congruences

$$\overline{\Omega} \equiv \overline{\Lambda} S \bmod x^{2t} \tag{5.27}$$

$$\Omega \equiv \Lambda S \bmod x^{2t} \tag{5.28}$$

and multiplying (5.27) by Λ and (5.28) by $\overline{\Lambda}$ we get

$$\overline{\Omega}\Lambda \equiv \overline{\Lambda}S\Lambda \bmod x^{2t} \tag{5.29}$$
$$\Omega\overline{\Lambda} \equiv \Lambda S\overline{\Lambda} \bmod x^{2t} \tag{5.30}$$

and subtracting (5.29) and (5.30) we obtain

$$\overline{\Omega}\Lambda \equiv \Omega\overline{\Lambda} \bmod x^{2t}$$

Since the degree conditions (5.26) are satisfied for both solutions, both sides of this congruence are actually polynomials of degree at most $2t-1$, so it is

$$\overline{\Omega}\Lambda = \Omega\overline{\Lambda} \tag{5.31}$$

Now, from (5.31) we can write

$$\Lambda = \Omega\overline{\Lambda}/\overline{\Omega} \tag{5.32}$$

and since $\overline{\Lambda},\ \overline{\Omega}$ are relatively prime, we have that no factor of $\overline{\Lambda}$ is canceled by any factor of $\overline{\Omega}$, and thus from (5.32) we see that $\overline{\Lambda}|\Lambda$. Similarly we can show that $\Lambda|\overline{\Lambda}$, therefore $\Lambda,\ \overline{\Lambda}$ differ at most by a constant multiple. Similarly for $\Omega,\ \overline{\Omega}$ and since from (5.31) we can write

$$\overline{\Omega}/\Omega = \overline{\Lambda}/\Lambda$$

the constants must agree. □

Given a solution of (5.25) for which the conditions of Th. 5.2 are satisfied, we can then determine the solutions of $\overline{\Lambda} = 0$ in $\mathbb{F}^*$, and hence the error locations; if α^{-i} appears as a root, then $i \in I$ is an error location.

Th. 5.2 and the previous discussion show that solving the decoding problem 5.1 can be accomplished by solving the key equation (5.25). We will now see how the theory of module Groebner bases proves to be useful here.

Given the integer t and $S \in \mathbb{F}_q[x]$, consider the set of all pairs $(\Omega,\ \Lambda) \in \mathbb{F}_q[x]^2$ satisfying (5.25):

$$K = \left\{(\Omega, \Lambda) :\ \Omega \equiv \Lambda S \bmod x^{2t}\right\} \tag{5.33}$$

K is a $\mathbb{F}_q[x]$-submodule of $\mathbb{F}_q[x]^2$, and every element of K can be written as a combination, with polynomial coefficients, of the two generators

$$\mathbf{g}_1 = (x^{2t}, 0) \tag{5.34}$$
$$\mathbf{g}_2 = (S, 1) \tag{5.35}$$

We will now show that (5.34)-(5.35) is a Groebner basis for K with respect to one monomial order on $\mathbb{F}_q[x]^2$. Moreover, one of the special solutions $(\Lambda, \Omega) \in K$ given by Th. 5.2 is guaranteed to occur in a Groebner basis for K with respect to a second monomial order on $\mathbb{F}_q[x]^2$. These results will form the ground for *two different decoding methods*, which we will discuss later on.

Let us first investigate some facts about submodules of $\mathbb{F}_q[x]^2$ and monomial orders. We here restrict our attention to submodules $M \subset \mathbb{F}_q[x]^2$. The following result tells us that a k-vector space $k[x]^2/M$ is finite-dimensional if and only if M has generators of a certain form (we will see later on that the generators of (5.33), i.e. (5.34)-(5.35) are actually of this kind).

Proposition 5.2. *Let k be any field, and M be a submodule of $k[x]^2$. Let $>$ be any monomial order on $k[x]^2$. Then the following conditions are equivalent:*

i. The k-vector space $k[x]^2/M$ is finite-dimensional.

ii. $\langle \mathrm{LT}_>(M)\rangle$ contains elements of the form $x^u\mathbf{e}_1 = (x^u, 0)$ and $x^v\mathbf{e}_2 = (0, x^v)$ for some $u, v \geq 0$.

Proof. Let $\mathcal{G}$ be a Groebner basis for M with respect to the monomial order $>$. As we know from ideals theory, the elements of $k[x]^2/M$ are linear combinations of monomials in the complement of $\langle \mathrm{LT}_>(M)\rangle$. There is a finite number of such monomials if and only if $\langle \mathrm{LT}_>(M)\rangle$ contains multiples of both $\mathbf{e}_1$ and $\mathbf{e}_2$ (in other words, we do not end up with a complement of $\langle \mathrm{LT}_>(M)\rangle$ that correspond to an infinite region in the plane identified by $\mathbf{e}_1$ and $\mathbf{e}_2$). □

Every submodule we consider from now on in the present section, will satisfy the equivalent conditions in Prop. 5.2.

The monomial orders that we consider in decoding can be described as follows:

Definition 5.2. *Let $r \in \mathbb{Z}$, we define an order $>_r$ by the following rules:*

i. $x^m\mathbf{e}_i >_r x^n\mathbf{e}_i$ if $m > n$ and $i = 1, 2$;

ii. $x^m\mathbf{e}_2 >_r x^n\mathbf{e}_1$ if and only if $m + r \geq n$.

For example, with $r = 2$, the monomials in $k[x]^2$ are ordered by $>_2$ as follows: $\mathbf{e}_1 <_2 x\mathbf{e}_1 <_2 x^2\mathbf{e}_1 <_2 \mathbf{e}_2 <_2 x^3\mathbf{e}_1 <_2 \cdots$ where e.g. $x^2\mathbf{e}_1 <_2 \mathbf{e}_2$, as $m + r = 0 + 2 \geq n = 2$.

Let us now see how Groebner bases for submodules with respect to $>_r$ orders look like:

Proposition 5.3. *Let M be a submodule of $k[x]^2$, and fix $r \in \mathbb{Z}$. Assume $\langle \mathrm{LT}_>(M)\rangle$ is generated by $x^u\mathbf{e}_1 = (x^u, 0)$ and $x^v\mathbf{e}_2 = (0, x^v)$ for some $u, v \geq 0$. Then a subset $\mathcal{G} \subset M$ is a reduced Groebner basis of M with respect to $>_r$ if and only if $\mathcal{G} = \{\mathbf{g}_1 = (g_{11}, g_{12}),\ \mathbf{g}_2 = (g_{21}, g_{22})\}$, where the g_i satisfy the following two properties:*

i. $\mathrm{LT}(\mathbf{g}_1) = x^u\mathbf{e}_1$ and $\mathrm{LT}(\mathbf{g}_2) = x^v\mathbf{e}_2$ for u, v as above;

ii. $\deg(g_{21}) < u$ and $\deg(g_{12}) < v$.

Proof. Suppose $\mathcal{G}$ is a subset of M satisfying conditions (i),(ii). By (i), the leading terms of the elements of $\mathcal{G}$ generate $\langle \mathrm{LT}(M)\rangle$, so by definition $\mathcal{G}$ is a Groebner basis for M.

Condition (ii) implies that no terms in $\mathbf{g}_1$ can be removed from $\mathcal{G}$ by division with respect to $\mathbf{g}_2$, and similarly for the terms in $\mathbf{g}_2$, so $\mathcal{G}$ is reduced.

Conversely, if $\mathcal{G}$ is a reduced Groebner basis for M with respect to $>_r$ it must contain exactly two elements, say $\mathbf{g}_1,\ \mathbf{g}_2$. Because of the assumption that $\langle \mathrm{LT}_>(M)\rangle$ is generated by $x^u\mathbf{e}_1 = (x^u, 0)$ and $x^v\mathbf{e}_2 = (0, x^v)$ for some $u, v \geq 0$, and since we assume here that $\mathcal{G}$ is a reduced Groebner basis, condition (i) must hold.

Finally, again because we assume $\mathcal{G}$ is a reduced Groebner basis, condition (ii) must hold. □

An important consequence of Prop. 5.3 is the following observation:

Corollary 5.1. *Let $\mathcal{G} = \{(S,1),(x^{2t},0)\}$ be the generators for the module K of solutions of the key equation* (5.25) *in the decoding problem with syndrome S. Then $\mathcal{G}$ is a Groebner basis for K with respect to the order $>_{\deg(S)}$.*

Note that $\mathrm{LT}_{>_{\deg(S)}}((S,1)) = (0,1) = \mathbf{e}_2$, indeed:

$$\begin{aligned} \mathrm{LT}_{>_{\deg(S)}}((S,1)) &= \mathrm{LT}_{>_{\deg(S)}}((\mathrm{LP}(S),1)) \\ &= \mathrm{LT}_{>_{\deg(S)}}((x^{\deg(S)},1)) \\ &= \mathrm{LT}_{>_{\deg(S)}}(x^{\deg(S)}\mathbf{e}_1 + \mathbf{e}_2) = \mathbf{e}_2 \end{aligned} \tag{5.36}$$

where the first equality uses Def. 4.3, and the last one comes from applying Def. 5.2. Equation (5.36) means that the module of solutions of the key equation always satisfies the finiteness condition from Prop. 5.2.

Let us finally introduce a definition and a general result, before going to the main point of this section.

Definition 5.3. *Let M be a nonzero submodule of $k[x]^2$. A* **minimal element** *of M with respect to a monomial order $>$ is a $\mathbf{g} \in M \setminus \{0\}$ such that $\mathrm{LT}(\mathbf{g})$ is minimal with respect to $>$.*

For example, $(S,1)$ is minimal with respect to the order $>_{\deg(S)}$ in $\langle (S,1),(x^{2t},0)\rangle$, indeed

$$\mathbf{e}_2 = \mathrm{LT}\left((S,1)\right) <_{\deg(S)} \mathrm{LT}\left((x^{2t},0)\right) = x^{2t}\mathbf{e}_1$$

being $m + r = 0 + \deg(S) < n = 2t$, since $\deg(S) \leq d - 2$ (see Eq.(5.14)), and we have assumed $d = 2t + 1$ at the beginning of the present section, therefore $\deg(S) \leq 2t - 1$, which leads to $m + r = 0 + \deg(S) \leq 2t - 1 < n = 2t$.

Notice that the leading terms $\mathrm{LT}\left((x^{2t},0)\right) = x^{2t}\mathbf{e}_1$ and $\mathrm{LT}\left((S,1)\right) = \mathbf{e}_2$ generate $\langle \mathrm{LT}(K)\rangle$ with respect to the $>_{\deg(S)}$ order.

The following result tells us that once we fix an order $>_r$, a minimal element for M with respect to that order is guaranteed to appear in a Groebner basis for M with respect to $>_r$.

Proposition 5.4. *Fix any $>_r$ order on $k[x]^2$, and let M be a submodule. Every Groebner basis for M with respect to $>_r$ contains a minimal element of M with respect to $>_r$.*

We will now discuss the main point of this section. Let us first prove a result that states that the special solution of the key equation (5.25) guaranteed by Th. 5.2 can be characterized as the minimal element of the module K with respect to a suitable order:

Proposition 5.5. *Let $\mathbf{g} = (\overline{\Omega}, \overline{\Lambda})$ be a solution of the key equation (5.25) satisfying the degree conditions (5.26) and with components relatively prime (which is unique up to constant multiple by* Th. 5.2*). Then $\mathbf{g}$ is a minimal element of K under the $>_{-1}$ order.*

Proof. An element $\overline{\mathbf{g}} = (\overline{\Omega}, \overline{\Lambda}) \in K$ satisfies $\deg(\overline{\Lambda}) > \deg(\overline{\Omega})$ if and only if its leading term with respect to $>_{-1}$ is a multiple of $\mathbf{e}_2$. The elements of K given by Th. 5.2 have this property and have minimal possible $\deg(\Lambda)$ (by uniqueness statement of Th. 5.2), so their leading term is minimal among leading terms which are multiples of $\mathbf{e}_2$.

Aiming for a contradiction, suppose that $\overline{\mathbf{g}}$ is not minimal, or equivalently that there is some nonzero $\mathbf{h} = (A, B)$ in K such that $\mathrm{LT}(\mathbf{h}) <_{-1} \mathrm{LT}(\overline{\mathbf{g}})$. Then by what said above, $\mathrm{LT}(\mathbf{h})$ must be a multiple of $\mathbf{e}_1$, i.e. it $\mathrm{LT}(\mathbf{h})$ must appear in A, so

$$\deg(\overline{\Lambda}) > \deg(A) \geq \deg(B) \tag{5.37}$$

where the first inequality in (5.37) comes from the assumption that $\mathbf{h}$ is minimal (and in particular, with respect to $\overline{\mathbf{g}}$).

Being both $\mathbf{h}$ and $\overline{\mathbf{g}}$ solutions of the key equation we have that

$$A \equiv SB \bmod x^{2t} \tag{5.38}$$
$$\overline{\Omega} \equiv S\overline{\Lambda} \bmod x^{2t} \tag{5.39}$$

Multiplying (5.38) by $\overline{\Lambda}$ and (5.39) by B:

$$\overline{\Lambda}A \equiv \overline{\Lambda}SB \bmod x^{2t} \tag{5.40}$$
$$B\overline{\Omega} \equiv BS\overline{\Lambda} \bmod x^{2t} \tag{5.41}$$

and subtracting (5.40) and (5.41), we have

$$\overline{\Lambda}A \equiv B\overline{\Omega} \bmod x^{2t} \tag{5.42}$$

Now, by the degree conditions (5.26) we know that $\deg(\overline{\Lambda}) \leq t$ and $\deg(\overline{\Omega}) < \deg(\overline{\Lambda})$, therefore $\deg(\overline{\Omega}) \leq t - 1$.

Further, from (5.37) it follows that $\deg(A) \leq t - 1$ (being $\deg(A) < \deg(\overline{\Lambda})$ and $\deg(\overline{\Lambda}) \leq t$).

Therefore, while the product on the left side of (5.42) has degree at most $2t - 1$, the product on the right side has degree at most $2t - 2$, which is an absurd. □

Combining Prop. 5.5 and Prop. 5.4, we see that the special solution of the key equation that we seek can be found in a Groebner basis for K with respect to the $>_{-1}$ order. This leads to at least *two possible ways to decode*:

i. We can use the generating set
$$\big\{(S, 1), (x^{2t}, 0)\big\}$$
for K, apply Buchberger's algorithm, and compute a Groebner basis for K with respect to $>_{-1}$ *directly*. Then the minimal element $\overline{g}$ will appear in the Groebner basis (see Prop. 5.4).

ii. We can alternatively make use of Corollary 5.1); since $\mathcal{G} = \big\{(S, 1), (x^{2t}, 0)\big\}$ is already a Groebner basis for K with respect to the order $>_{deg(S)}$, we can *convert* $\big\{(S, 1), (x^{2t}, 0)\big\}$ into a Grorbner basis $\mathcal{G}'$ for the same module, but with respect to the $>_{-1}$ order (using an appropriate algorithm, i.e. the FGLM algorithm). Once we get the basis with respect to the $>_{-1}$, we proceed as in approach (i), since we know by Prop. 5.4 that the minimal element in K will be an element of $\mathcal{G}'$.

Another possibility would be to build up the desired solution of the key equation inductively, solving the congruences

$$\Omega \equiv \Lambda S \bmod x^l$$

for $l = 1, \ldots, 2t$ in turn. This approach gives one way to understand the operations from the well-known Reed-Solomon decoding algorithm due to Berlekamp and Massey.

Example 5.2

We will now see a practical example of application the decoding method (i) above. We will accompany the calculations with the corresponding implementation in Singular.

Let us consider the Reed-Solomon code over $\mathbb{F}_9$ with $k = 3$ (we then also know that $n = q - 1 = 9 - 1 = 8,\ d = q - k = n - k + 1 = 8 - 3 + 1 = 6,\ t = \lfloor \frac{d-1}{2} \rfloor = \lfloor \frac{6-1}{2} \rfloor = 2$) and generator matrix

$$G = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & \alpha & \alpha^2 & \alpha^3 & \alpha^4 & \alpha^5 & \alpha^6 & \alpha^7 \\ 1 & \alpha^2 & \alpha^4 & \alpha^6 & 1 & \alpha^2 & \alpha^4 & \alpha^6 \end{pmatrix} \tag{5.43}$$

Let us now recall the following result:

Proposition 5.6. *Let $\mathcal{C}$ be the Reed-Solomon code of dimension k and minimum distance $d = q-k$ over $R = \mathbb{F}_q$. Then the generator polynomial of $\mathcal{C}$ has the form*

$$g = (x - \alpha) \cdots (x - \alpha^{q-k-1}) = (x - \alpha) \cdots (x - \alpha^{d-1})$$

where α is a root of the minimal polynomial associated to $\mathbb{F}_q$.

For example, the generator polynomial for the code we consider in this example is

$$g = (x - \alpha)(x - \alpha^2)(x - \alpha^3)(x - \alpha^4)(x - \alpha^5) \tag{5.44}$$

where α is a root of the minimal polynomial associated to $\mathbb{F}_9$:

```
> ring R=(3^2,alpha),x,(lp,c);
> minpoly;
1*alpha^2+2*alpha^1+2*alpha^0
> poly g=(x-alpha)*(x-alpha2)*(x-alpha3)*(x-alpha4)*(x-alpha5);
> g;
x5-x4+alpha*x3+x2+alpha*x+alpha3
```

We use the TOP ordering (command *(lp,c)*), since the order $>_{-1}$ corresponds to it.

We assume to transmit the codeword $c = 1 + x + x^2 + x^3 + x^4 + x^5 + x^6 + x^7 \sim 11111111$, corresponding to the first row of matrix G in (5.43):

```
> poly c=1+x+x2+x3+x4+x5+x6+x7;
```

We can verify that c is actually a codeword, by verifying that the remainder of the division by the generator polynomial g is zero:

```
> division(c,g);
[1]:
   _[1,1]=x2-x+alpha5
[2]:
   _[1]=0
```

Let us assume that during the transmission of c, an error $e = 2x + 2 \sim 22000000$, such that the received message is

$$y = c + e = 00111111 \sim x^2 + x^3 + x^4 + x^5 + x^6 + x^7$$

```
> poly e=2x+2;
> poly y=c+e;
y;
x7+x6+x5+x4+x3+x2
```

The syndrome of the received codeword is

$$s(x) = \sum_{j=1}^{d-1} y(\alpha^j)x^{j-1} = \sum_{j=1}^{5} y(\alpha^j)x^{j-1} = y(\alpha) + y(\alpha^2)x + y(\alpha^3)x^2 + y(\alpha^4)x^3 + y(\alpha^5)x^4$$

```
> poly s1=subst(y,x,alpha);
> poly s2=subst(y,x,alpha2);
> poly s3=subst(y,x,alpha3);
> poly s4=subst(y,x,alpha4);
> poly s5=subst(y,x,alpha5);
> poly s=s1+s2*x+s3*x2+s4*x3+s5*x4;
> s;
alpha7*x4+alpha2*x2+alpha3*x+alpha6
```

At this point, we define the module M from the generators $\mathbf{g}_1 = (x^{2t}, 0),\ \mathbf{g}_2 = (s(x), 1) \in \mathbb{F}_9[x]^2$:

```
> vector g1=[x4,0];
> vector g2=[s,1];
> module M=g1,g2;
> M;
M[1]=x4*gen(1);
M[2]=alpha7*x4*gen(1)+alpha2*x2*gen(1)+alpha3*x*gen(1)+alpha6*gen(1)+gen(2)
> print(M);
x4,alpha7*x4+alpha2*x2+alpha3*x+alpha6,
0, 1
```

We now calculate the RGB of M with respect to $>_{-1}$:

```
> option(redSB);
> module G=std(M);
```

```
> G;
G[1]=x2*gen(2)-x*gen(1)+alpha5*x*gen(2)+alpha5*gen(1)+alpha7*gen(2)
G[2]=x2*gen(1)+alpha*x*gen(1)-gen(1)+alpha6*gen(2)
```

Since the first generator of the GB is smaller, with respect to $>_{-1}$, than the second one, then $\mathbf{g}_1 = (\alpha^5 - x, \alpha^7 + \alpha^5 x + x^2)$, this is the minimal element $\overline{\mathbf{g}}$ we look for:

```
> G[1]<G[2];
1   % the condition at the line above is verified
> print(G);
-x+alpha5,          x2+alpha*x-1,
x2+alpha5*x+alpha7,alpha6
```

and therefore $(\overline{\Omega}, \overline{\Lambda}) = (\alpha^5 - x, \alpha^7 + \alpha^5 x + x^2)$ is the solution of the key equation which verifies the conditions of Prop. 5.5:

```
> poly E=G[1][1];
> E;
-x+alpha5
> poly L=G[1][2];
> L;
x2+alpha5*x+alpha7
```

Before going ahead, let us state the following proposition [2007 Martinez Moro, Munuera Gomez & Ruano], which will be useful in next steps:

Proposition 5.7. *Let us consider the error locator polynomial* $\Lambda(x) = \prod_{i \in I}(1 - \alpha^i x)$ *and the error evaluator polynomial* $\Omega(x) = \sum_{i \in I} e_i \alpha^i \prod_{j \neq i,\ j \in I}(1 - \alpha^j x)$. *Then:*

i. if $\eta_1, \ldots, \eta_r$ *are the roots of* $\Lambda(x)$, *then its inverses* $\eta_1^{-1}, \ldots, \eta_r^{-1}$ *are the locations of the errors;*

ii. given the values of $\alpha_1, \ldots, \alpha_r$ *corresponding to the error locations as per point (i), the error values are*

$$e_j = \frac{-\Omega(\alpha_j^{-1})}{\Lambda'(\alpha_j^{-1})}, \quad j = 1, \ldots, r \tag{5.45}$$

where $\Lambda'(x)$ *indicates the first derivative of* $\Lambda(x)$. *Equation (5.45) is called the* Forney formula *for the error value.*

The errors positions are then determined from the roots of $\Lambda(x) = \alpha^7 + \alpha^5 x + x^2$, and to calculate them we will make an exhaustive search, considering as values for the substitution α^i with the exponent i going from 1 to the code length $n = 8$:

```
> subst(L,x,alpha);
alpha7
```

```
> subst(L,x,alpha2);
alpha5
> subst(L,x,alpha3);
-1
> subst(L,x,alpha4);
alpha5
> subst(L,x,alpha5);
1
> subst(L,x,alpha6);
-1
> subst(L,x,alpha7);
0
> 1/alpha7;
alpha
> subst(L,x,alpha8);
0
> 1/alpha8;
1
```

We obtained that $\alpha^7,\ \alpha^8$ are the roots of $\overline{\Lambda}$. Being $\eta_1 = \alpha^0 = (\alpha^8)^{-1},\ \eta_2 = \alpha^1 = (\alpha^7)^{-1}$, we have that the errors have occurred in the positions 1 and 2 (or equivalently, in the independent term and in the one in x).

Now that we have determined the errors positions, we can obtain the errors values too, by using the evaluator polynomial $\overline{\Omega}(x) = \alpha^5 - x$, and the first derivative of the locator polynomial, i.e. $\overline{\Lambda}^{'} = \frac{d\overline{\Lambda}}{dx} = \alpha^5 + 2x$ (see Prop. 5.7 (ii)):

$$\begin{aligned} e_1 &= \frac{-\Omega(\alpha^8)}{\Lambda'(\alpha^8)} = \frac{\alpha^2}{\alpha^6} = -1 \\ e_2 &= \frac{-\Omega(\alpha^7)}{\Lambda'(\alpha^7)} = \frac{\alpha^6}{\alpha^2} = -1 \end{aligned}$$

```
> L;
x2+alpha5*x+alpha7
> poly LP=2x+alpha5;
> LP;
-x+alpha5
> subst(-E,x,alpha8);
alpha2
> subst(LP,x,alpha8);
alpha6
> poly e1=alpha2/alpha6;
> e1;
-1
> subst(-E,x,alpha7);
alpha6
> subst(LP,x,alpha7);
alpha2
> poly e2=alpha6/alpha2;
```

```
> e2;
-1
> poly e_rx=e1+e2*x;
> e_rx;
-x-1
```

Therefore the error is $e(x) = -1 - x \sim e = 22000000$ and the decoded codeword is

$$c_{dec}(x) = y(x) - e(x) = 1 + x + x^2 + x^3 + x^4 + x^5 + x^6 + x^7 \sim 11111111$$

which is correct as we expect, since two errors occurred and our error correction capability is up to $t = 2$ errors.

```
> poly c_dec=y-e_rx;
> c_dec;
x7+x6+x5+x4+x3+x2+x+1
```

Let us assume now that we have $t = 3$ errors.

```
> poly c=1+x+x2+x3+x4+x5+x6+x7;
> poly e=2+2x+2x2;
> poly y=c+e;
> y;
x7+x6+x5+x4+x3
... % we calculate s and M as above
> option(redSB);
> module G=std(M);
> G;
G[1]=x2*gen(2)+alpha5*x*gen(1)+alpha2*x*gen(2)+alpha7*gen(1)+alpha5*gen(2)
G[2]=x2*gen(1)+alpha6*x*gen(1)+alpha5*x*gen(2)+alpha5*gen(1)+alpha3*gen(2)
> print(G);
alpha5*x+alpha7,   x2+alpha6*x+alpha5,
x2+alpha2*x+alpha5,alpha5*x+alpha3
> G[1]<G[2];
1
> poly E=G[1][1];
> poly L=G[1][2];
> subst(L,x,alpha);
alpha6
> subst(L,x,alpha2);
alpha3
> subst(L,x,alpha3);
-1
> subst(L,x,alpha4);
alpha
> subst(L,x,alpha5);
alpha
> subst(L,x,alpha6);
```

```
alpha5
> subst(L,x,alpha7);
alpha6
> subst(L,x,alpha8);
-1
```

and we see from the fact that we cannot find a root of $\Lambda(x)$, that we are not capable to correct the 3 errors that have occurred. Now, in this case we could not correct the errors, but the fact to have more than 2 errors (our RS code's correcting capability) could alternatively have lead to the case that we were detecting another codeword, still part of the codebook, but different than the transmitted one.

5.5 Decoding of QC Codes Formed by RS Codes

Example 5.3

Let us see an example of QC code over $\mathbb{F}_q = \mathbb{F}_9$, with index $l = 2$, and length $n = m \cdot l = 8 \cdot 2 = 16$ (m is what we have called n in Example 5.2), generated by

$$\mathbf{g} = (f_1 g, f_2 g) \in \left(\frac{R}{I}\right)^2 \tag{5.46}$$

where g as in (5.44), with $g | x^m - 1$, $\gcd(f_i, x^m - 1) = 1$ and $\deg(f_i) < m - \deg(g)$, $1 \leq i \leq l$; further, we have that in our case $R/I = \mathbb{F}_9/\langle x^8 - 1 \rangle$. The minimum distance is $D = l \cdot d = 2 \cdot 6 = 12$ (where d is the distance of the RS code considered also in Example 5.2), and the decoding algorithm we have introduced in Section 5.3, can correct at most t errors, where $t = \left\lfloor \frac{D-1}{2} \right\rfloor = \left\lfloor \frac{12-1}{2} \right\rfloor = \left\lfloor \frac{11}{2} \right\rfloor = 5$.

Let us assume we want to transmit the information polynomial a (which in our example is $a = x^2 - x + \alpha^5$), with $\deg(a) < m - \deg(g)$. As we know, each codeword has the form

$$\mathbf{c} = a\mathbf{g} = (af_1 g, af_2 g) \bmod x^m - 1 \tag{5.47}$$

We report below the Singular code:

```
> ring R=(3^2,alpha),x,(lp,c);
> poly g=(x-alpha)*(x-alpha2)*(x-alpha3)*(x-alpha4)*(x-alpha5);   % generator polynomial
> division(x8-1,g);
[1]:
   _[1,1]=x3+x2+alpha3*x+alpha   % quotient
[2]:
   _[1]=0   % remainder
% from the above we see that g|x^8-1
> poly a=x2-x+alpha5;   % information polynomial
```

We now consider $f_1,\ f_2$ which appear in (5.47), e.g. $f_1 = x^2,\ f_2 = 1$, and calculate their inverses. We want to find f_1^{-1} such that $f_1 f_1^{-1} = 1$, i.e. $x^2 f_1^{-1} = 1$, and this is $f_1^{-1} = x^6$, since $x^2 x^6 = x^8 \equiv 1 \bmod x^8 - 1$. Trivially, $f_2^{-1} = 1$.

We report below the QC encoding procedure:

```
> division(a*f1*g,x8-1);
[1]:
   _[1,1]=x+1
[2]:
   _[1]=x7+x6+x5+x4+x3+x2+x+1   % a*f1*g mod x^8-1
> poly c1=x7+x6+x5+x4+x3+x2+x+1;
> division(a*f2*g,x8-1);
[1]:
   _[1,1]=0
[2]:
   _[1]=x7+x6+x5+x4+x3+x2+x+1   % a*f2*g mod x^8-1
> poly c2=x7+x6+x5+x4+x3+x2+x+1;
> vector c=[c1,c2];   % transmitted codeword of the QC code
> print(c);
[x7+x6+x5+x4+x3+x2+x+1,x7+x6+x5+x4+x3+x2+x+1]
```

Some errors happen during the transmission and affect our QC codeword $\mathbf{c} = (c_1, c_2)$:

```
> poly e1=2x+2;
> poly e2=2x7+2;
> poly r1=c1+e1;
> poly r2=c2+e2;
> vector r=[r1,r2];
> print(r);
[x7+x6+x5+x4+x3+x2,x6+x5+x4+x3+x2+x]
```

We then decode r_1 as we did for the RS code in section 5.4 (RS code which constitutes the $l = 2$ blocks of the QC code considered in this example), as we considered the same codeword, i.e. $c_1 = 1 + x + x^2 + x^3 + x^4 + x^5 + x^6 + x^7$ and the same error, i.e. $e = 2 + 2x$, obtaining $c_{dec} = c_1$. At this point we apply the algorithm introduced in Section 5.3 for decoding of restriction-1 1-generator QC codes:

```
> poly c1_prime=c1;
> c1_prime;
x7+x6+x5+x4+x3+x2+x+1
> poly c2_prime=c1_prime*inv_f1*f2;
> division(c2_prime,x8-1);
[1]:
   _[1,1]=x5+x4+x3+x2+x+1
[2]:
   _[1]=x7+x6+x4+x3+x2+x+1   % c2_prime mod x^8-1
> c2_prime=x7+x6+x5+x4+x3+x2+x+1;
> vector c_prime=[c1_prime,c2_prime];
> c_prime-r;
x7*gen(2)+x*gen(1)+gen(1)+gen(2)
> print(c_prime-r);
[x+1,x7+1]
```

and since $wt(|\mathbf{c}^{'} - \mathbf{r}|) = 4 \leq \left\lfloor \frac{D-1}{2} \right\rfloor = 5$, we then exit the WHILE cycle when executing the decoding

algorithm of Section 5.3, since our QC code can correct up to $\min\left(l\left\lfloor\frac{d-1}{2}+l-1\right\rfloor,\left\lfloor\frac{D-1}{2}\right\rfloor\right) = \min(5,5) = 5$:

Let us now see what happens when we have 3 errors in the block corresponding to $l = 1$ and 2 errors in the block corresponding to $l = 2$; this is still within the error correcting capabilities of our QC code, i.e. 5.

```
> poly e1=2+2x+2x2;
> poly e2=2+2x;
> poly r1=c1+e1;
> r1;
x7+x6+x5+x4+x3
> poly r2=c2+e2;
x7+x6+x5+x4+x3+x2
> vector r=[r1,r2];
```

We cannot decode r_1 (see last part of Example 5.2). We therefore move to the second WHILE cycle, i.e. $i + 1 = 2$, where we can decode the RS codeword c_2

```
> poly c2_prime=1+x+x2+x3+x4+x5+x6+x7;
> poly c1_prime=c2_prime*inv_f2*f1;
> division(c1_prime,x8-1);
[1]:
   _[1,1]=x+1
[2]:
   _[1]=x7+x6+x4+x3+x2+x+1   % c1_prime mod x^8-1
> c1_prime=x7+x6+x4+x3+x2+x+1;
> vector c_prime=[c1_prime,c2_prime];
> c_prime-r;
x2*gen(1)+x*gen(1)+x*gen(2)+gen(1)+gen(2)
> print(c_prime-r);
[x2+x+1,x+1]
```

and since $\mathrm{wt}(\mathbf{c}^{'}-\mathbf{r}) = 5 \leq \left\lfloor\frac{D-1}{2}\right\rfloor = 5$, our QC code can handle the fact that the first block cannot be decoded, recovering the whole information from the second block.

Bibliography

[1970 Hartley] *Rings, Modules and Linear Algebra*, B. Hartley & T.O. Hawkes, Chapman & Hall Mathematics Series, Cambridge University Press, 1970.

[1997 Little, Saints & Heegard] *On the structure of Hermitian codes*, J. Little, K. Saints & C. Heegard, J. Pure Appl. Algebra, no. 121, pp. 293-314, 1997.

[LallyPhD] *PhD dissertation*, K. Lally.

[2001 Lally & Fitzpatrick] *Algebraic structure of quasicyclic codes*, K. Lally & P. Fitzpatrick, Discrete Applied Mathematics, no. 111, pp. 157-175, 2001.

[2002 Lally] *Quasicyclic codes - some practical issues*, K. Lally, International Symposium on Information Theory (ISIT), June-July 2002.

[lecture cyclic] http://www.math.msu.edu/∼jhall/classes/codenotes/Cyclic.pdf

[2003 Huffman & Pless] *Fundamentals of Error-Correcting Codes*, W.C. Huffman & V. Pless, Cambridge University Press, 2003, ISBN 0-521-78280-5.

[2004 Cox, Little & O'Shea] *Using Algebraic Geometry*, D. Cox, J.Little & D. O'Shea, Springer, 2004, ISBN 0-387-20706-6.

[2005 Greuel, Pfister & Schoenemann] *SINULAR 3.0, A Computer Algebra System for Polynomial Computations*, G.-M. Greuel, G.Pfister & H. Schoenemann, Centre for Computer Algebra, University of Kaiserslautern, http://www.singular.uni-kl.de, 2005.

[2006 Cox, Little & O'Shea] *Ideals, Varieties and Algorithms*, D. Cox, J.Little & D. O'Shea, Springer, 2006, ISBN 978-0-387-35650-1.

[2007 Martinez Moro, Munuera Gomez & Ruano] *Bases de Groebner: aplicaciones a la codificacion algebraica*, E. Martinez Moro, C. Munuera Gomez & D. Ruano Benito, Escuela Venezolana de Matematicas, 2007, ISBN 978-980-261-087-7.