Gröbner bases for error-correcting codes and their decoding

Mario de Boer and Ruud Pellikaan [1]

Appeared in *Some tapas of computer algebra*
(A.M. Cohen, H. Cuypers and H. Sterk eds.),
Chap. 10, Gröbner bases for codes, pp. 237-259,
Chap. 11, Gröbner bases for decoding, pp. 260-275,
Project 7, The Golay codes, pp. 338-347,
Springer, Berlin 1999,
after the EIDMA/Galois minicourse on Computer Algebra,
September 27-30, 1995, Eindhoven.

[1]Both authors are from the Department of Mathematics and Computing Science, Eindhoven University of Technology, P.O. Box 513, 5600 MB Eindhoven, The Netherlands.

Contents

Chapter 1

Gröbner bases for codes

1.1 Introduction

Coding theory deals with the following topics:

- Cryptography or cryptology. Transmission of secret messages or electronic money, eavesdropping, intruders, authentication and privacy.
- Source coding or data compression. Most data have redundant information, and can be compressed, to save space or to speed up the transmission.
- Error-correcting codes. If the channel is noisy one adds redundant information in a clever way to correct a corrupted message.

In this and the following chapter we are concerned with Gröbner bases and error-correcting codes and their decoding. In Sections 1.2 and 1.3 a kaleidoscopic introduction is given to error-correcting codes centered around the question of finding the minimum distance and the weight enumerator of a code. Section 1.4 uses the theory of Gröbner bases to get all codewords of minimum weight. Section 1.5 gives an elementary introduction to algebraic geometry codes.

All references and suggestions for further reading will be given in the notes of Section 1.6. The beginning of this chapter is elementary and the level is gradually more demanding towards the end.

Notation: The ring of integers is denoted by $\mathbb{Z}$, the positive integers by $\mathbb{N}$ and the non-negative integers by $\mathbb{N}_0$. The ring of integers modulo n is denoted by $\mathbb{Z}_n$. The number of elements of a set S is denoted by $\#S$. A field is denoted by $\mathbb{F}$ and its set of nonzero elements by $\mathbb{F}^*$. The finite field with q elements is denoted by $\mathbb{F}_q$. Vectors are row vectors. The transpose of a matrix M is written as M^T. The inner product of the vectors $\mathbf{x}$ and $\mathbf{y}$ is defined as $\mathbf{x} \cdot \mathbf{y} = \mathbf{x}\mathbf{y}^T = \sum x_i y_i$. The projective space of dimension m over $\mathbb{F}_q$ is denoted by $PG(m, q)$. Variables are denoted in capitials such as $X, Y, Z, X_1, \ldots, X_m$. If I is an ideal and F an element of $\mathbb{F}_q[X_1, \ldots, X_m]$, then $V(I, \mathbb{F})$ denotes the zero set of I in $\mathbb{F}^m$, and

the coset of F modulo I is denoted by f.

1.2 Basic facts from coding theory

Words have a fixed length n, and the letters are from an *alphabet* Q of q elements. Thus words are elements of Q^n. A *code* (dictionary) is a subset of Q^n. The elements of the code are called *codewords*.

1.2.1 Hamming distance

Two distinct words of a code should differ as much as possible. To give this a precise meaning the *Hamming distance* between two words is introduced. If $\mathbf{x}, \mathbf{y} \in Q^n$, then

$$d(\mathbf{x}, \mathbf{y}) = \#\{i \mid x_i \neq y_i\}.$$

Exercise 1.2.1 Show that the Hamming distance is a *metric*. In particular that it satisfies the *triangle inequality*.

$$d(\mathbf{x}, \mathbf{z}) \leq d(\mathbf{x}, \mathbf{y}) + d(\mathbf{y}, \mathbf{z}).$$

The *minimum distance* of a code C is defined as

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

1.2.2 Linear codes

If the alphabet is a finite field, which is the case for instance when $Q = \{0, 1\}$, then Q^n is a vector space. A *linear code* is a linear subspace of $\mathbb{F}_q^n$. If a code is linear of dimension k, then the *encoding*

$$\mathcal{E} : \mathbb{F}_q^k \longrightarrow \mathbb{F}_q^n,$$

from message or source word $\mathbf{x} \in \mathbb{F}_q^k$ to encoded word $\mathbf{c} \in \mathbb{F}_q^n$ can be done efficiently by a matrix multiplication.

$$\mathbf{c} = \mathcal{E}(\mathbf{x}) = \mathbf{x}G,$$

where G is a $k \times n$ matrix with entries in $\mathbb{F}_q$. Such a matrix G is called a *generator matrix* of the code.

For a word $\mathbf{x} \in \mathbb{F}_q^n$ its *support* is defined as the set of non-zero coordinate positions, and its *weight* as the number of elements of its support and denote it by $wt(\mathbf{x})$. The minimum distance of a linear code C is equal to its minimum weight

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

In this chapter a code will always be linear.

The parameters of a code C in $\mathbb{F}_q^n$ of dimension k and minimum distance d will be denoted by $[n,k,d]_q$ or $[n,k,d]$. Then $n-k$ is called the *redundancy*. For an $[n,k,d]$ code C we define the *dual code* $C^\perp$ as

$$C^\perp = \{\mathbf{x} \in \mathbb{F}_q^n \mid \mathbf{c}\cdot\mathbf{x} = 0 \text{ for all } \mathbf{c} \in C\}.$$

Exercise 1.2.2 Let C be a code of length n and dimension k. Show that $C^\perp$ has dimension $n-k$. Let H be a generator matrix for $C^\perp$. Prove that $C = \{\mathbf{c} \in \mathbb{F}_q^n \mid H\mathbf{c}^T = 0\}$. Therefore H is called a *parity check* matrix for C.

Example 1.2.3 The $[7,4,3]$ *Hamming code* has generator matrix G and its dual, the $[7,3,4]$ *Simplex code* has generator matrix H, where

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

Exercise 1.2.4 Let $(I_k|P)$ be a generator matrix of C, where I_k is the $k \times k$ identity matrix. Show that $(-P^T|I_{n-k})$ is a parity check matrix of C.

1.2.3 Weight distribution

Apart from the minimum distance, a code has other important invariants. One of these is the *weight distribution* $\{(i,\alpha_i) \mid i = 0,1,\ldots n\}$, where α_i denotes the number of codewords in C of weight i. The polynomials $W_C(X,Y)$ and $W_C(X)$, defined as

$$W_C(X,Y) = \sum_{i=0}^{n} \alpha_i X^{n-i}Y^i \text{ and } W_C(X) = \sum_{i=0}^{n} \alpha_i X^{n-i}$$

are called the *(homogeneous) weight enumerators* of C. Although there is no apparent relation between the minimum distance of a code and its dual, the weight enumerators satisfy the *MacWilliams identity*.

Theorem 1.2.5 *Let C be an $[n,k]$ code over $\mathbb{F}_q$. Then*

$$W_{C^\perp}(X,Y) = q^{-k}W_C(X+(q-1)Y, X-Y).$$

1.2.4 Automorphisms and isometries of codes

Other important invariants of a code are its group of *automorphisms* and its group of *isometries*.

Let $Perm(n,q)$ be the subgroup of $GL(n,q)$ consisting of permutations of coordinates. Let $Diag(n,q)$ be the subgroup of $GL(n,q)$ consisting of diagonal matirices. Let $Iso(n,q)$ be the subgroup of $GL(n,q)$ which is generated by $Perm(n,q)$ and $Diag(n,q)$.

A code that is the image of C under an element of $Perm(n,q)$ is said to be *equivalent* to C. The subgroup of $Perm(n,q)$ that leaves C invariant is the *automorphism group* of C, $Aut(C)$.

A code that is the image of C under an element of $Iso(n,q)$ is said to be *isometric* to C. The subgroup of $Iso(n,q)$ that leaves C invariant is the *isometry group* of C, $Iso(C)$.

Exercise 1.2.6 Show that $Aut(C) = Aut(C^\perp)$ and similarly for $Iso(C)$.

Exercise 1.2.7 Show that a linear map $\varphi : \mathbb{F}_q^n \to \mathbb{F}_q^n$ is an isometry if and only if φ leaves the Hamming metric invariant, that means that

$$d(\varphi(\mathbf{x}), \varphi(\mathbf{y})) = d(\mathbf{x}, \mathbf{y})$$

for all $\mathbf{x}, \mathbf{y} \in \mathbb{F}_q^n$.

A code of length n is called *cyclic* if the cyclic permutation of coordinates $\sigma(i) = i - 1$ modulo n leaves the code invariant. See Section 1.4.

Exercise 1.2.8 Show that the $[7, 4, 3]$ Hamming code, as defined in Example 1.2.3, is not cyclic, but that it is equivalent to a cyclic code.

1.3 Determining the minimum distance

Given a generator matrix of a code, the problem is to determine the minimum distance of the code. We will give five possible solutions here. All these methods do not have polynomial complexity in n, the lenght of the code. One cannot hope for a polynomial algorithm, since recently it has been proved that this problem is NP complete.

1.3.1 Exhaustive search

This is the first approach that comes to mind. It is the brute force method: generate all codewords and check for their weights.

Since one generates the whole code, other invariants, like the weight distribution, are easy to determine at the same expence. But going through all codewords is the most inefficient way of dealing with the problem.

It is not necessary to consider all scalar multiples $\lambda\mathbf{c}$ of a codeword $\mathbf{c}$ and a nonzero $\lambda \in \mathbb{F}_q$, since they all have the same weight. This improves the complexity by a factor $q - 1$. One can speed up the procedure if one knows more about the code at forehand, for example the automorphism group, in particular for cyclic codes.

By the MacWilliams relations, knowing the weight distribution of a code, one can determine the weight distribution of the dual code by solving linear equations. Therefore it is good to do exhaustive search on whatever code (C or $C^\perp$) has lowest dimension.

Example 1.3.1 The Hamming code. Generating all 16 codewords of the Hamming code yields the following weight distribution of the code:

weight	# codewords
0	1
3	7
4	7
7	1

This could have been achieved by first computing the weight distribution of the dual code (dimension 3) and then applying the MacWilliams transform. Also one can use that the code has an cyclic automorphism group of order 7. Therefore one knows that the number of codewords of weights 3 or 4 are multiples of 7.

Exercise 1.3.2 Does it hold in general that the number of codewords of a given weight in a cyclic code is a multiple of the length? If not, what is the exact relation?

1.3.2 Linear algebra

In a sense the theory of linear codes is "just linear algebra". The determination of the minimum distance can be phrased in these terms as the following exercise shows.

Exercise 1.3.3 The minimum distance is the minimal number of dependent columns in a parity check matrix.

But also for this method one has to look at all possible combinations of columns, and this number grows exponentially.

We give a sketch how the minimum distance of linear codes is determined by the algorithm of Brouwer. Let G be a $k \times n$ generator matrix of G. After a permutation of the columns and rowreductions we may suppose that the first k columns form the $k \times k$ identity matrix. Any linear combination of w rows with non-zero coefficients gives a codeword of weight at least w. In particular, if the code has minimum distance 1, then we will notice this by the fact that one of the rows of G has weight 1. More generaly, we look at all possible linear combinations of w rows for $w = 1, 2, \ldots$ and keep track of the codeword of smallest weight. If we have found a codeword of weight v, then we can restrict the possible number of rows we have to consider to $v - 1$. The lower bound w for the weight of the codewords we generate is raised, and the lowest weight v of a codeword found in the process so far is lowered. Finally v and w meet.

An improvement of this method is obtained if G is of the form $(G_1 \cdots G_l)$ where $G_1, \ldots, G_l$ are matrices such that the first k columns of G_j form the $k \times k$ identity matrix for all $j = 1, \ldots, l$. In this way we know that any linear combination of w rows with non-zero coefficients gives a codeword of weight at least lw.

Exercise 1.3.4 Show that the maximum length of a binary code of dimension 4 and dual distance 3 is 7. What is the maximum length of a q-ary code of dimension k and dual distance 3 ? Hint: use Exercise 1.3.3 and read the next section on finite geometry first.

1.3.3 Finite geometry

It is possible to give the minimum distance of a code a geometric interpretation.

Suppose that the code is *nondegenerate*, this means that there is not a coordinate j such that $c_j = 0$ for all codewords $\mathbf{c}$. For the determination of the minimum distance this is not an important restriction. So every column of the generator matrix G of a $[n, k, d]$ code is not zero and its homogeneous coordinates can be considered as a point in projective space of dimension $k - 1$ over $\mathbb{F}_q$. If two columns are dependent, then they give rise to the same point. In this way we get a set $\mathcal{P}$ of n points (counted with multplicities) in $PG(k - 1, q)$, that are not all contained in a hyperplane. This is called a *projective system*.

A projective system $\mathcal{P}$ of n points $P_1, \ldots, P_n$ in $PG(k - 1, q)$ with $P_j = (g_{1j} : \cdots : g_{kj})$, defines the code C with generator matrix $G = (g_{ij})$. This code depends on the choice of the enumeration of the points of $\mathcal{P}$ and on the choice of the homogeneous coordinates of P_j.

Two projective systems $\mathcal{P}_1$ and $\mathcal{P}_2$ are called equivalent if there exists a projective transformation $\sigma \in PGL(k - 1, q)$ such that $\sigma(\mathcal{P}_1) = \mathcal{P}_2$.

Exercise 1.3.5 Show that we get in this way a one-to-one correspondence between isometry classes of non-degenerate $[n, k, d]$ codes and equivalence classes of projective systems of n points in $PG(k-1, q)$ such that the maximal number of points in a hyperplane (counted with multplicities) is equal to $n - d$.

Example 1.3.6 The 7 columns of the $[7, 3, 4]$ Simplex code, viewed as homogeneous coordinates of points in $PG(2, 2)$, give the seven points of the *Fano plane*. All lines contain three points, so indeed the minimum distance is $7 - 3 = 4$.

Let $F(X, Y, Z) \in \mathbb{F}_q[X, Y, Z]$ be a homogenous polynomial of degree m. Let $\mathcal{P}$ be the set of points $(a : b : c) \in PG(2, q)$ such that $F(a, b, c) = 0$, then we say that $\mathcal{P}$ is a *projective plane curve of degree* m in $PG(2, q)$ and that $F(X, Y, Z) = 0$ is its *defining equation*.

Exercise 1.3.7 What could be said about the minimum distance of the code of a plane curve in $PG(2, q)$ of degree m which has n points ? Notice that the answer depends on whether the defining equation has a linear factor or not. Codes from plane curves are treated more extensively in Section 1.5.

Exercise 1.3.8 The *Klein quartic* is the projective plane curve with defining equation

$$X^3Y + Y^3Z + Z^3X = 0.$$

What are the parameters of the code associated to the Klein quartic over $\mathbb{F}_8$?

A *rational normal curve* in $PG(r,q)$ is the image of the map

$$\varphi : PG(1,q) \longrightarrow PG(r,q)$$

given by $\varphi(x_0 : x_1) = (x_0^r : x_0^{r-1}x_1 : \cdots : x_0x_1^{r-1} : x_1^r)$, or a projective transformation of this image.

Exercise 1.3.9 Show that the $q+1$ points of a rational normal curve in $PG(r,q)$ lie in *general linear position*, that is to say no $r+1$ of these points lie in a hyperplane. What are the parameters of its associated code ?

Exercise 1.3.10 Show that, possibly after a projective change of coordinates, the points of a rational normal curve are zeros of the 2×2 minors of the following matrix

$$\begin{pmatrix} X_0 & X_1 & \dots & X_{r-1} \\ X_1 & X_2 & \dots & X_r \end{pmatrix}.$$

What is the vanishing ideal of a rational normal curve in $PG(r,q)$?

Exercise 1.3.11 The *Hexacode* is the quaternary code with generator matrix

$$G = \begin{pmatrix} 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & \alpha & \alpha^2 \\ 0 & 0 & 1 & 1 & \alpha^2 & \alpha \end{pmatrix},$$

where $\alpha \in \mathbb{F}_4$ is a primitive element satisfying $\alpha^2 + \alpha + 1 = 0$. Show that the last 5 columns of G lie on the conic $X_0^2 + X_1X_2 = 0$ over $\mathbb{F}_4$, which is a rational normal curve. Use Exercise 1.3.9 to show that $d \geq 3$. Show that all 5 lines in $PG(2,4)$ through $(1:0:0)$, corresponding to the first column of G, intersect the remaining 5 points in exactly one point. Conclude that $d \geq 4$. Determine the weight distribution of the code using this geometric setting.

1.3.4 Arrangements of hyperplanes

In this section we consider the dual picture.

Let C be a nondegenerate code. The columns of the generator matrix G can be considered as hyperplanes in $\mathbb{F}_q^k$ or $PG(k-1,q)$. Then column $\mathbf{v}^T$ corresponds to a hyperplane with equation $\sum v_iX_i = 0$. The multiset of hyperplanes will be denoted by $\mathcal{H}$.

Exercise 1.3.12 Show that the weight of a codeword $\mathbf{c} = \mathbf{x}G$ is given by

$$\mathrm{wt}(\mathbf{c}) = n - \text{number of hyperplanes in } \mathcal{H} \text{ through } \mathbf{x},$$

where this number is counted with multiplicities.

Clearly the number of codewords of a certain weight t equals the number of points that are on exactly $n-t$ of the hyperplanes in $\mathcal{H}$. To find a nice expression for this we introduce the following notations. For a subset $J \subseteq \{1, 2, \ldots, n\}$ we define

$$C(J) = \{\mathbf{c} \in C \mid c_j = 0 \text{ for all } j \in J\}$$
$$l(J) = \dim C(J).$$

Under the above correspondence we get the following isomorphism of vector spaces.

$$\bigcap_{j \in J} H_j \cong C(J).$$

Now define

$$\beta_t = \sum_{\#J=t} (q^{l(J)} - 1).$$

Exercise 1.3.13 Let $d^\perp$ denote the minimum distance of the dual code. Then for $t < d^\perp$

$$\beta_t = \binom{n}{t}(q^{k-t} - 1).$$

Exercise 1.3.14 Recall that α_s is the number of codewords of weight s. Prove the following formula

$$\beta_t = \sum_{s=d}^{n-t} \binom{n-s}{t} \alpha_s.$$

by computing the number of elements of the set of pairs

$$\{(J, \mathbf{c}) \mid J \subseteq \{1, 2, \ldots, n\}, \#J = t, \mathbf{c} \in C(J),\ \mathbf{c} \neq 0\}$$

in two different ways.

Exercise 1.3.15 Show that the weight enumerator of C can be expressed in terms of the β_t as follows.

$$W_C(X) = X^n + \sum_{t=0}^{n-d} \beta_t (X-1)^t.$$

Exercise 1.3.16 Prove the following identity either by inverting the formula of Exercise 1.3.14 or by an inclusion/exclusion argument.

$$\alpha_s = \sum_{t=n-s}^{n-d} (-1)^{n+s+t} \binom{t}{n-s} \beta_t.$$

Example 1.3.17 The Hamming code, see Exercise 1.2.3. The seven hyperplanes in $\mathcal{H}$ are given by: $X_1 = 0, X_2 = 0, X_3 = 0, X_4 = 0, X_1 + X_2 + X_3 = 0, X_1 + X_2 + X_4 = 0, X_1 + X_3 + X_4 = 0$. Going through all points $\mathbf{x} \in \mathbb{F}_2^4$ and checking on how many of the hyperplanes in $\mathcal{H}$ they are on, gives after applying Proposition 1.3.15, the weight enumerator of the code.
Computing the $l(J)$ for all J gives the following result:

$\#J$	0	1	2	3	4	4	5	6	7
$l(J)$	4	3	2	1	1 for 7 J	0 for 28 J	0	0	0

Since $\beta_i = 0$ for $i \geq 5$ and there are J of size 4 such that $C(J) \neq \{0\}$ we see that the minimum distance is $d = 3$. To find the weight distribution we compute the β_i.

$$\begin{array}{lcl} \beta_0 & = & 1(2^4-1) = 15 \\ \beta_1 & = & 7(2^3-1) = 49 \\ \beta_2 & = & 21(2^2-1) = 63 \\ \beta_3 & = & 35(2^1-1) = 35 \\ \beta_4 & = & 7(2^1-1) = 7 \\ \beta_5 & = & 0 \\ \beta_6 & = & 0 \\ \beta_7 & = & 0 \end{array} \qquad \begin{array}{lcl} \alpha_0 & = & 1 \\ \alpha_1 & = & 0 \\ \alpha_2 & = & 0 \\ \alpha_3 & = & \beta_4 = 7 \\ \alpha_4 & = & \beta_3 - 4\beta_4 = 7 \\ \alpha_5 & = & \beta_2 - 3\beta_3 + 6\beta_4 = 0 \\ \alpha_6 & = & \beta_1 - 2\beta_2 + 3\beta_3 - 4\beta_4 = 0 \\ \alpha_7 & = & \beta_0 - \beta_1 + \beta_2 - \beta_3 + \beta_4 = 1 \end{array}$$

Exercise 1.3.18 Compute the weight enumerator of the $[7, 3, 4]$ Simplex code and verify MacWilliam's identity.

Exercise 1.3.19 Compute the weight enumerator of the code on the Klein quartic of Exercise 1.3.8.

Exercise 1.3.20 Let C be an $[n, k, n-k+1]$ code. Show that $l(J) = n - \#J$ for all J. Compute the weight enumerator of such a code.

Exercise 1.3.21 Prove that the number $l(J)$ is the same for the codes C and $\mathbb{F}_{q^e}C$ in $\mathbb{F}_{q^e}^n$ for any extension $\mathbb{F}_{q^e}$ of $\mathbb{F}_q$.

Using the Exercises 1.3.14, 1.3.16 and 1.3.21 it is immediate to find the weight distribution of a code over any extension $\mathbb{F}_{q^e}$ if one knows the $l(J)$ over the ground field $\mathbb{F}_q$ for all subsets J of $\{1, \ldots, n\}$. Computing the $C(J)$ and $l(J)$ for a fixed J is just linear algebra. The large complexity for the computation of the weight enumerator and the minimum distance in this way stems from the exponential growth of the number of all possible subsets of $\{1, \ldots, n\}$.

Exercise 1.3.22 Let C be the code over $\mathbb{F}_q$, with q even, with generator matrix H of Exercise 1.2.3. For which q does this code contain a word of weight 7 ?

Exercise 1.3.23 Compare the complexity of the methods "exhaustive search" and "arrangements of hyperplanes" to compute the weight enumerator as a function of q and the parameters $[n, k, d]$ and $d^\perp$.

1.3.5 Algebra

Let the n hyperplanes in $\mathcal{H}$ have equations

$$L_1(X) = L_2(X) = \cdots = L_n(X) = 0,$$

where the L_i are linear forms in the variables $X_1, X_2, \ldots, X_k$ as in the previous section. Then a general codeword is of the form

$$\mathbf{c} = (L_1(\mathbf{x}), L_2(\mathbf{x}), \ldots, L_n(\mathbf{x})),$$

where $\mathbf{x} \in \mathbb{F}_q^k$. Now let I_t be the ideal generated by all products of t distinct $L_i(X)$, so

$$I_t = \left(\prod_{s=1}^{t} L_{i_s}(X) \mid 1 \leq i_1 < i_2 < \cdots < i_t \leq n \right).$$

If Φ_t is the ideal generated by all homogeneous forms of degree t in k variables $X_1, \ldots, X_k$, then clearly $I_t \subseteq \Phi_t$. We have the following.

Exercise 1.3.24 Show that

$$V(I_t, \mathbb{F}_q) = \{\mathbf{x} \in \mathbb{F}_q^k \mid wt(\mathbf{c}) < t, \text{ with } \mathbf{c} = \mathbf{x}G\}.$$

Exercise 1.3.25 Show that

$$d = \min\{t \mid V(I_{t+1}, \mathbb{F}_q) \neq \{0\}\}.$$

Determining whether $V(I_t, \mathbb{F}_q) = \{0\}$ can be done by computing a Gröbner basis for the ideal. Sometimes it is easy to see that $I_t = \Phi_t$, whence one can conclude immediately that $V(I_t, \mathbb{F}_q) = V(\Phi_t, \mathbb{F}_q) = \{0\}$. But in general no polynomial algorithm is known to decide this question.

The ideals I_t are generated by $\binom{n}{t}$ elements and also this makes it infeasible to work with this method when n and t are large.

Contrary to what is done in exhaustive search, here all codewords are considered at once.

Example 1.3.26 The Hamming code, see Exercise 1.2.3. For this code we have

$$\mathbf{c} = (x_1, x_2, x_3, x_4, x_1 + x_2 + x_3, x_1 + x_2 + x_4, x_1 + x_3 + x_4).$$

It is easy to see that $I_1 = (X_1, X_2, X_3, X_4) = \Phi_1$ and hence $d \geq 1$. Also $I_2 = \Phi_2$ and $I_3 = \Phi_3$ is easy to check, so $d \geq 3$. To prove $d = 3$ it is enough to note that I_4 is contained in the ideal (X_1, X_2, X_3), so $(0, 0, 0, 1) \in V(I_4, \mathbb{F}_2)$.

Example 1.3.27 For the Hexacode, see Exercise 1.3.11, it is easy to see that $I_1 = \Phi_1$ and $I_2 = \Phi_2$. We skip the computation of I_3 and compute a Gröbner basis for I_4. The result is:

$$\begin{aligned} I_4 \;=\; & (X_1^4, X_1^3X_2, X_1^3X_3, X_1^2X_2^2, X_1^2X_2X_3, X_1^2X_3^2, X_1X_2^3, X_1X_2^2X_3, \\ & X_1X_2X_3^2, X_1X_3^3, X_2^4, X_2^3X_3, X_2^2X_3^2, X_2X_3^3, X_3^4). \end{aligned}$$

We find that $I_4 = \Phi_4$ and hence $d \geq 4$. Since the rows of G are codewords of weight 4, we can conclude that $d = 4$. For completeness, a Gröbner basis is computed for I_5:

$$\begin{aligned} I_5 \;=\; & (X_1^4X_2 + X_1X_2^4, X_1^4X_3 + X_1X_3^4, X_1^3X_2X_3 + X_1X_2^2X_3^2, \\ & X_1^2X_2^2X_3 + X_1X_2X_3^3, X_1^2X_2X_3^2 + X_1X_2^3X_3, X_2^4X_3 + X_2X_3^4). \end{aligned}$$

Now I_5 is contained in the ideal (X_1, X_2), so $(0, 0, 1) \in V(I_5, \mathbb{F}_4)$ and indeed $d = 4$.

1.4 Cyclic codes

In this section we consider a very important special class of codes: cyclic codes. We will find a nice algebraic description of the codewords of minimal weight in such codes and a way to decode up to half the minimum distance. It is not claimed that this is the most efficient way of treating this problem.

A *cyclic code* is a code C with the following property:

$$\text{if } \mathbf{c} = (c_0, c_1 \ldots, c_{n-1}) \in C, \text{ then } (c_{n-1}, c_0 \ldots, c_{n-2}) \in C.$$

In the context of cyclic codes it is convenient to consider the index i of a word as an element of $\mathbb{Z}_n$, the cyclic group of order n.

Consider the bijection ϕ between $\mathbb{F}_q^n$ and $\mathbb{F}_q[X]/(X^n - 1)$

$$\phi(\mathbf{c}) = c_0 + c_1 X + \cdots + c_{n-1} X^{n-1}.$$

Then ideals in the ring $\mathbb{F}_q[X]/(X^n - 1)$ correspond one-to-one to cyclic codes in $\mathbb{F}_q^n$. In the rest of this chapter we will not distinguish between codewords and the corresponding polynomials under ϕ; we will talk about codewords $c(X)$ when in fact we mean the vector and vice versa.

Since $\mathbb{F}_q[X]/(X^n - 1)$ is a principal ideal ring, every cyclic code C is generated by a unique monic polynomial $g(X)$ of degree at most $n - 1$, the *generator polynomial* $g(X)$:

$$C = \{c(X) \mid c(X) = r(X)g(X) \bmod (X^n - 1),\ r(X) \in \mathbb{F}_q[X]\}.$$

Instead of describing a cyclic code by its generator polynomial $g(X)$, one can describe the code by the set of zeros of $g(X)$ in an extension of $\mathbb{F}_q$.

From now on we assume that n is relatively prime with q. Let α be a primitive n-th root of unity in an extension field $\mathbb{F}_{q^e}$. A subset J of $\mathbb{Z}_n$ is called a *defining set* of a cyclic code C if

$$C = \{c(X) \in \mathbb{F}_q[X]/(X^n - 1) \mid c(\alpha^j) = 0 \text{ for all } j \in J\}.$$

The *complete defining set* $J(C)$ of C is defined as

$$J(C) = \{j \in \mathbb{Z}_n \mid c(\alpha^j) = 0 \text{ for all } c \in C\}.$$

Example 1.4.1 There are exactly two irreducible polynomials of degree 3 in $\mathbb{F}_2[X]$. They are factors of $X^7 + 1$

$$X^7 + 1 = (X + 1)(X^3 + X + 1)(X^3 + X^2 + 1).$$

Let $\alpha \in \mathbb{F}_8$ be a zero of $X^3 + X + 1$. Then α is a primitive element of $\mathbb{F}_8$ and α^2 and α^4 are the remaining zeros of $X^3 + X + 1$. Consider the binary cyclic code C of length 7 with defining set $\{1\}$. Then $J(C) = \{1, 2, 4\}$ and $X^3 + X + 1$ is the generator polynomial of C. The code C is equivalent with the Hamming code.

Exercise 1.4.2 *BCH bound.* Show that a cyclic code has at least minimum distance d if $J(C)$ contains $d-1$ subsequent elements.

Exercise 1.4.3 The *cyclotomic coset* of $j \in \mathbb{Z}_n$ is the set $\{q^i j \mid i \in \mathbb{N}_0\}$. Show that a complete defining set is the union of cyclotomic cosets.

Exercise 1.4.4 Let C be a cyclic code of length 7 over $\mathbb{F}_q$. Show that $\{1, 2, 4\}$ is a complete defining set if q is even.

Exercise 1.4.5 Show that a binary cyclic code of length 11 has minimum distance 1, 2 or 11.

Exercise 1.4.6 Show that the cyclotomic coset of $\{1\}$ in $\mathbb{Z}_{23}$ contains 4 subsequent elements for $q = 2$.

1.4.1 The Mattson-Solomon polynomial

Let $a(X)$ be a word in $\mathbb{F}_q^n$. Let $\alpha \in \mathbb{F}_{q^e}$ be a primitive n-th root of unity. Then the *Mattson-Solomon* (MS) polynomial of $a(X)$ is defined as

$$A(Z) = \sum_{i=1}^{n} A_i Z^{n-i}, \qquad A_i = a(\alpha^i) \in \mathbb{F}_{q^e}.$$

Here too we adopt the convention that the index i is an element of $\mathbb{Z}_n$, so $A_{n+i} = A_i$.

The MS polynomial $A(Z)$ is the *discrete Fourier transform* of the word $a(X)$. Notice that $A_n \in \mathbb{F}_q$.

Proposition 1.4.7
1. *The inverse is given by* $a_j = \frac{1}{n} A(\alpha^j)$.
2. $A(z)$ *is the MS polynomial of a word* $a(X)$ *if and only if* $A_{jq} = A_j^q$ *for all* $j \in \mathbb{Z}_n$.
3. $A(z)$ *is the MS polynomial of a codeword* $a(X)$ *of the cyclic code* C *if and only if* $A_j = 0$ *for all* $j \in J(C)$ *and* $A_{jq} = A_j^q$ *for all* $j = 1, \ldots, n$.

Exercise 1.4.8 Let $\beta \in \mathbb{F}_{q^e}$ be a zero of $X^n - 1$. Show that

$$\sum_{i=1}^{n} \beta^i = \begin{cases} n & \text{if} \quad \beta = 1 \\ 0 & \text{if} \quad \beta \neq 1. \end{cases}$$

Expand $A(\alpha^i)$ using the definitions and use the above fact to prove Proposition 1.4.7(1). Prove the remaining assertions of Proposition 1.4.7.

Let $a(X)$ be a word of weight w. Then the *locators* $x_1, x_2, \ldots, x_w$ of $a(X)$ are defined as

$$\{x_1, x_2, \ldots, x_w\} = \{\alpha^i \mid a_i \neq 0\}.$$

Let $y_j = a_i$ if $x_j = \alpha^i$. Then

$$A_i = a(\alpha^i) = \sum_{j=1}^{w} y_j x_j^i.$$

Consider the product

$$\sigma(Z) = \prod_{j=1}^{w} (1 - x_j Z).$$

Then $\sigma(Z)$ has as zeros the reciprocals of the locators, and is sometimes called the *locator polynomial*. In this chapter and the following on decoding this name is reserved for the polynomial that has the locators as zeros.

Let $\sigma(Z) = \sum_{i=0}^{w} \sigma_i Z^i$. Then σ_i is the *ith elementary symmetric function* in these locators:

$$\sigma_t = (-1)^t \sum_{1 \le j_1 < j_2 < \cdots < j_t \le w} x_{j_1} x_{j_2} \cdots x_{j_t}.$$

The following property of the MS polynomial is called the *generalized Newton identity* and gives the reason for these definitions.

Proposition 1.4.9 *For all i it holds that*

$$A_{i+w} + \sigma_1 A_{i+w-1} + \cdots + \sigma_w A_i = 0.$$

Exercise 1.4.10 Substitute $Z = 1/x_j$ in the equation

$$1 + \sigma_1 Z + \cdots + \sigma_w Z^w = \prod_{j=1}^{w} (1 - x_j Z)$$

and multiply by $y_j x_j^{i+w}$. This gives

$$y_j x_j^{i+w} + \sigma_1 y_j x_j^{i+w-1} + \cdots + \sigma_w y_j x_j^i = 0.$$

Check that summing on $j = 1, \ldots, w$ yields the desired result of Proposition 1.4.9.

Example 1.4.11 Let C be the cyclic code of length 5 over $\mathbb{F}_{16}$ with defining set $\{1, 2\}$. Then this defining set is complete. The polynomial

$$X^4 + X^3 + X^2 + X + 1$$

is irreducible over $\mathbb{F}_2$. Let β be a zero of this polynomial in $\mathbb{F}_{16}$. Then the order of β is 5. The generator polynomial of C is

$$(X + \beta)(X + \beta^2) = X^2 + (\beta + \beta^2)X + \beta^3.$$

So $(\beta^3, \beta + \beta^2, 1, 0, 0) \in C$ and

$$(\beta + \beta^2 + \beta^3, 1 + \beta, 0, 1, 0) = (\beta + \beta^2)(\beta^3, \beta + \beta^2, 1, 0, 0) + (0, \beta^3, \beta + \beta^2, 1, 0)$$

is an element of C. These codewords together with their cyclic shifts and their nonzero scalar multiples give $(5+5)*15 = 150$ words of weight 3.

Using Propositions 1.4.7 and 1.4.9 it will be shown that these are the only codewords of weight 3. Consider the set of equations:

$$\left\{\begin{array}{lllllllll} A_4 & + & \sigma_1 A_3 & + & \sigma_2 A_2 & + & \sigma_3 A_1 & = & 0 \\ A_5 & + & \sigma_1 A_4 & + & \sigma_2 A_3 & + & \sigma_3 A_2 & = & 0 \\ A_1 & + & \sigma_1 A_5 & + & \sigma_2 A_4 & + & \sigma_3 A_3 & = & 0 \\ A_2 & + & \sigma_1 A_1 & + & \sigma_2 A_5 & + & \sigma_3 A_4 & = & 0 \\ A_3 & + & \sigma_1 A_2 & + & \sigma_2 A_1 & + & \sigma_3 A_5 & = & 0 \end{array}\right.$$

If A_1, A_2, A_3, A_4 and A_5 are the coefficients of the MS polynomial of a codeword, then $A_1 = A_2 = 0$. If $A_3 = 0$, then $A_i = 0$ for all i. So we may assume that $A_3 \neq 0$. The above equations imply $A_4 = \sigma_1 A_3$, $A_5 = (\sigma_1^2 + \sigma_2)A_3$ and

$$\left\{\begin{array}{lll} \sigma_1^3 + \sigma_3 & = & 0 \\ \sigma_1^2\sigma_2 + \sigma_2^2 + \sigma_1\sigma_3 & = & 0 \\ \sigma_1^2\sigma_3 + \sigma_2\sigma_3 + 1 & = & 0. \end{array}\right.$$

Subsitition of $\sigma_3 = \sigma_1^3$ in the remaining equations yields

$$\left\{\begin{array}{lll} \sigma_1^4 + \sigma_1^2\sigma_2 + \sigma_2^2 & = & 0 \\ \sigma_1^5 + \sigma_1^3\sigma_2 + 1 & = & 0. \end{array}\right.$$

Multiplying the first equation with σ_1 and adding to the second one gives

$$1 + \sigma_1\sigma_2^2 = 0.$$

Thus $\sigma_1 = \sigma_2^{-2}$ and

$$\sigma_2^{10} + \sigma_2^5 + 1 = 0.$$

This last equation has 10 solutions in $\mathbb{F}_{16}$, and we are free to choose A_3 from $\mathbb{F}_{16}^*$. This gives in total 150 solutions.

Exercise 1.4.12 Let C be the code of the previous example. Compute the number of codewords of weight 3 with the help of Exercise 1.3.20.

Exercise 1.4.13 Let C be a cyclic code of length 7 over $\mathbb{F}_q$ with defining set $\{1, 2, 4\}$. Show that $d(C) > 3$ if q is odd.

1.4.2 Codewords of minimal weight

The following way to get all minimal codewords of cyclic codes uses the theory of Gröbner bases.

Let C be a cyclic code of length n over $\mathbb{F}_q$ with defining set $J(C)$. Let $\mathbb{F}_{q^e}$ be an extension of $\mathbb{F}_q$ that contains an n-th root of unity. Let $\mathcal{S}_C(w)$ be the following

system of equations:

$$\begin{cases} A_{w+1} + \sigma_1 A_w + \cdots + \sigma_w A_1 = 0 \\ A_{w+2} + \sigma_1 A_{w+1} + \cdots + \sigma_w A_2 = 0 \\ \vdots \quad\quad\quad\quad\quad\quad \vdots \quad\quad \vdots \\ A_{w+n} + \sigma_1 A_{w+n-1} + \cdots + \sigma_w A_n = 0 \\ \text{for all } j \in J(C) \quad A_j = 0 \\ \text{for all } j \in \mathbb{Z}_n \quad A_{qj} = A_j^q. \end{cases}$$

In this system both the A_i and the σ_i are indeterminates.

From the properties of the MS polynomial stated in Propositions 1.4.7 and 1.4.9 we see that codewords of weight at most w give solutions of the system $\mathcal{S}_C(w)$, and that conversely any solution to the system comes from a codeword of weight at most w. The exact relation is as follows.

Theorem 1.4.14 *The solutions $(A_0, A_1, \ldots, A_{n-1})$ to $\mathcal{S}_C(w)$ over $\mathbb{F}_{q^e}$ are the coefficients of the MS polynomials of codewords of weight at most w.*

Corollary 1.4.15 *The minimum distance d is equal to the smallest value of w such that $\mathcal{S}_C(w)$ has a non-zero solution over $\mathbb{F}_{q^e}$. Each solution $(A_0, A_1, \ldots, A_{n-1})$ to $\mathcal{S}_C(d)$ over $\mathbb{F}_{q^e}$ corresponds one-to-one to a codeword of minimal weight.*

We conclude that the codewords of minimal weight in a cyclic code can be determined by solving a system of equations in the polynomial ring $\mathbb{F}_q[A_0, A_1, \ldots, A_{n-1}, \sigma_0, \sigma_1, \ldots, \sigma_d]$. Solving the system can be done by computing a Gröbner basis for the ideal defined by $\mathcal{S}_C(d)$. This method will be applied to the ternary Golay code in a project in this book.

Exercise 1.4.16 Let C be a cyclic code of length 7 over $\mathbb{F}_q$, q even, with defining set $\{1, 2, 4\}$. Show that the number of codewords of weight 3 is equal to $7(q-1)$.

1.5 Codes from varieties

Consider a geometric object $\mathcal{X}$ with a subset $\mathcal{P}$ consisting of n distinct points which are listed by $P_1, \ldots, P_n$. Suppose that we have a vector space L over $\mathbb{F}_q$ of functions on $\mathcal{X}$ with values in $\mathbb{F}_q$. Thus $f(P_i) \in \mathbb{F}_q$ for all i and $f \in L$. In this way one has an evaluation map

$$ev_{\mathcal{P}} : L \longrightarrow \mathbb{F}_q^n$$

which is defined by $ev_{\mathcal{P}}(f) = (f(P_1), \ldots, f(P_n))$. If this evaluation map is linear, then its image is a linear code.

In the following, $\mathcal{X}$ is a subset of an *affine variety*, that is the common set of zeros in affine space of some given set of polynomials. The points $P_1, \ldots, P_n$ are called *rational* when they have coordinates in $\mathbb{F}_q$. The functions will be polynomial functions.

Extending a reduction order on the set of monomials to a function on all polynomials gives an example of an order function. A special kind of order function is a weight function. The theory of Gröbner bases is used to show the existence of certain weight functions.

These order functions will be used to define codes and to derive a bound for the minimum distance for these codes which is similar to the BCH bound for cyclic codes.

1.5.1 Order and weight functions

Let $\mathbb{F}$ be a field. In this chapter an $\mathbb{F}$-algebra is a commutative ring with a unit that contains $\mathbb{F}$ as a unitary subring. Let R be an $\mathbb{F}$-algebra. An *order function* on R is a map

$$\rho : R \longrightarrow \mathbb{N}_0 \cup \{-\infty\},$$

that satisfies the following conditions:

- $(O.0)$ $\rho(f) = -\infty$ if and only if $f = 0$
- $(O.1)$ $\rho(\lambda f) = \rho(f)$ for all nonzero $\lambda \in \mathbb{F}$
- $(O.2)$ $\rho(f+g) \leq \max\{\rho(f), \rho(g)\}$ and equality holds when $\rho(f) < \rho(g)$
- $(O.3)$ If $\rho(f) < \rho(g)$ and $h \neq 0$, then $\rho(fh) < \rho(gh)$
- $(O.4)$ If $\rho(f) = \rho(g)$, then there exists a nonzero $\lambda \in \mathbb{F}$ such that $\rho(f - \lambda g) < \rho(g)$.

for all $f, g, h \in R$. Here $-\infty < n$ for all $n \in \mathbb{N}_0$.

Example 1.5.1 Let $R = \mathbb{F}[X_1, \ldots, X_m]$. Let $\prec$ be a reduction order on the monomials in $X_1, \ldots, X_m$ which is isomorphic to the ordinary order on $\mathbb{N}$. The lexicographical total degree order is isomorphic with $(\mathbb{N}, <)$, but the lexicographical order is not if $m > 1$. Let the sequence $(F_i | i \in \mathbb{N})$ be an enumeration of the monomials in increasing order, so $F_i \prec F_{i+1}$ for all i. They form a basis of R over $\mathbb{F}$. So every nonzero polynomial F has a unique representation

$$F = \sum_{i \leq j} \lambda_i F_i,$$

where $\lambda_i \in \mathbb{F}$ and $\lambda_j \neq 0$. Define $\rho(F) = j - 1$. Then ρ is an order function on R.

Exercise 1.5.2 Let R be an $\mathbb{F}$-algebra. Show that there exists a sequence $(f_i | i \in \mathbb{N})$ which is a basis of R over $\mathbb{F}$ such that $\rho(f_i) < \rho(f_{i+1})$.

Let $L(l)$ be the vector space with basis $f_1, \ldots, f_l$. Let $l(i,j)$ be the smallest l such that $f_i f_j \in L(l)$. Prove that $l(i,j)$ is strictly increasing in both arguments. Such a sequence is called *well-behaving*.

Let R be an $\mathbb{F}$-algebra. A *weight function* on R is an order function on R that satisfies furthermore

$$(O.5) \quad \rho(fg) = \rho(f) + \rho(g)$$

for all $f, g \in R$. Here $-\infty + n = -\infty$ for all $n \in \mathbb{N}_0$.

If ρ is a weight function and $\rho(f)$ is divisible by an integer $d > 1$ for all $f \in R$, then $\rho(f)/d$ is again a weight function. Thus we may assume that the greatest common divisor of the integers $\rho(f)$ with $0 \neq f \in R$ is 1.

A *degree function* on R is a map that satisfies conditions $(O.0)$, $(O.1)$, $(O.2)$ and $(O.5)$. It is clear that condition $(O.3)$ is a consequence of $(O.5)$.

Example 1.5.3 The standard example of an $\mathbb{F}$-algebra R with a degree function ρ is obtained by taking $R = \mathbb{F}[X_1, \ldots, X_m]$ and $\rho(F) = \deg(F)$, the degree of $F \in R$. It is a weight function if and only if $m = 1$.

Let $\mathbf{w} = (w_1, \ldots, w_m)$ be an m-tuple of positive integers called *weights*. The *weighted degree* of $\alpha \in \mathbb{N}_0^m$ and the corresponding monomial X^α is defined as

$$\mathrm{wdeg}(X^\alpha) = \mathrm{wdeg}(\alpha) = \sum \alpha_l w_l,$$

and of a nonzero polynomial $F = \sum \lambda_\alpha X^\alpha$ as

$$\mathrm{wdeg}(F) = \max\{\ \mathrm{wdeg}(X^\alpha) \mid \lambda_\alpha \neq 0\ \}.$$

The *lexicographical total weighted degree order* $\prec_{\mathbf{w}}$ on $\mathbb{N}_0^m$ is defined as $\alpha \prec_{\mathbf{w}} \beta$ if and only if either $\mathrm{wdeg}(\alpha) < \mathrm{wdeg}(\beta)$ or $\mathrm{wdeg}(\alpha) = \mathrm{wdeg}(\beta)$ and $\alpha \prec_L \beta$, and similarly for the monomials.

Exercise 1.5.4 Show that wdeg is a degree function on $\mathbb{F}[X_1, \ldots, X_m]$ and that $\prec_{\mathbf{w}}$ is a reduction order that is isomorphic with $(\mathbb{N}, <)$.

Exercise 1.5.5 Let R be an $\mathbb{F}$-algebra with a weight function. Show that the set of elements of weight zero is equal to $\mathbb{F}^*$.

Example 1.5.6 Consider the $\mathbb{F}$-algebra

$$R = \mathbb{F}[X, Y]/(X^5 - Y^4 - Y).$$

Assume that R has a weight function ρ. Let x and y be the cosets in R of X and Y, respectively. Then $x^5 = y^4 + y$. Now $y \notin \mathbb{F}$, so $\rho(y) > 0$ by Exercise 1.5.5, and $\rho(y^4) = 4\rho(y) > \rho(y)$ by $(O.5)$. Thus $\rho(y^4 + y) = \rho(y^4)$ by $(O.2)$. Therefore

$$5\rho(x) = \rho(x^5) = \rho(y^4 + y) = 4\rho(y)$$

Thus the only possible solution is $\rho(x) = 4$ and $\rho(y) = 5$.

Exercise 1.5.7 Let $R = \mathbb{F}[X, Y]/(X^3Y + Y^3 + X)$. Show by the same reasoning as in the example above that $\rho(x) = 2$ and $\rho(y) = 3$ if there exists a weight function ρ on R. Prove that there exists no weight function on R.

Let $\mathcal{M}$ be the set of monomials in $X_1, \ldots, X_m$. The *footprint* or Δ-*set* of a finite set $\mathcal{B}$ of polynomials is defined by

$$\Delta(\mathcal{B}) = \mathcal{M} \setminus \{lm(BM) | B \in \mathcal{B}, B \neq 0, M \in \mathcal{M}\}.$$

If $\mathcal{B}$ is a Gröbner basis for the ideal I in R, then the cosets modulo I of the elements of the footprint $\Delta(\mathcal{B})$ form a basis of R/I.

Exercise 1.5.8 Let $\prec_{\mathbf{w}}$ be the lexicographical total weighted degree order on the monomials in X and Y with weights 4 and 5 for X and Y, respectively. Show that

$$\{X^iY^j | i,j \in \mathbb{N}_0,\ j < 4\}$$

is the footprint of $X^5 + Y^4 + Y$ with respect to the reduction order $\prec_{\mathbf{w}}$.

Prove that the degree function wdeg is injective on this footprint. Let $(F_l | l \in \mathbb{N})$ be an enumeration of this footprint such that $\mathrm{wdeg}(F_l) < \mathrm{wdeg}(F_{l+1})$ for all l. Let $R = \mathbb{F}[X,Y]/(X^5 + Y^4 + Y)$. Let f_i be the coset of F_i in R. Thus $(f_l | l \in \mathbb{N})$ is a basis of R over $\mathbb{F}$. Define $\rho_l = 4i + 5j$ if $f_l = x^i y^j$.

Let $L(l)$ be the vector space with $f_1, \ldots, f_l$ as basis. Let $l(i,j)$ be the smallest l such that $f_i f_j \in L(l)$. Prove that $\rho_l = \rho_i + \rho_j$ if $l = l(i,j)$.

Show that there exists a weight function on R as a conclusion of the above results or as a special case of the following.

Theorem 1.5.9 *Let I be an ideal in $\mathbb{F}[X_1, \ldots, X_m]$ with Gröbner basis $\mathcal{B}$ with respect to $\prec_{\mathbf{w}}$. Suppose that the elements of the footprint of I have mutually distinct weighted degrees and that every element of $\mathcal{B}$ has two monomials of highest weighted degree in its support. Then there exists a weight function ρ on $R = \mathbb{F}[X_1, \ldots, X_m]/I$ with the property that $\rho(f) = wdeg(F)$, where f is the coset of F modulo I, for all polynomials F.*

Exercise 1.5.10 Let $R = \mathbb{F}[X,Y]/(X^a + Y^b + G(X,Y))$, where $gcd(a,b) = 1$ and $\deg(G) < b < a$. Show that R has a weight function ρ such that $\rho(x) = b$ and $\rho(y) = a$.

Exercise 1.5.11 Let ρ be a weight function. Let $\Gamma = \{\rho(f) | f \in R,\ f \neq 0\}$. We may assume that the greatest common divisor of Γ is 1. Then Γ is called the set of *non-gaps*, and the complement of Γ in $\mathbb{N}_0$ is the set of *gaps*. Show that the number of gaps of the weight function of Example 1.5.10 is equal to $(a-1)(b-1)/2$.

1.5.2 A bound on the minimum distance

We denote the coordinatewise multiplication on $\mathbb{F}_q^n$ by $*$. Thus $\mathbf{a}*\mathbf{b} = (a_1b_1, \ldots, a_nb_n)$ for $\mathbf{a} = (a_1, \ldots, a_n)$ and $\mathbf{b} = (b_1, \ldots, b_n)$. The vector space $\mathbb{F}_q^n$ becomes an $\mathbb{F}_q$-algebra with the multiplication $*$.

Let R be an *affine* $\mathbb{F}_q$-algebra. That is to say $R = \mathbb{F}_q[X_1, \ldots, X_m]/I$, where I is an ideal of $\mathbb{F}_q[X_1, \ldots, X_m]$. Let $\mathcal{P} = \{P_1, \ldots, P_n\}$ consist of n distinct points of the zero set of I in $\mathbb{F}_q^m$. Consider the evaluation map

$$ev_{\mathcal{P}} : R \longrightarrow \mathbb{F}_q^n,$$

defined as $ev_{\mathcal{P}}(f) = (f(P_1), \ldots, f(P_n))$.

Exercise 1.5.12 Show that $ev_{\mathcal{P}}$ is well defined and a morphism of $\mathbb{F}_q$-algebras, that means that this map is $\mathbb{F}_q$-linear and $ev_{\mathcal{P}}(fg) = ev_{\mathcal{P}}(f) * ev_{\mathcal{P}}(g)$ for all $f, g \in R$. Prove that the evaluation map is surjective.

Assume that R has an order function ρ. Then there exists a well-behaving sequence $(f_i \mid i \in \mathbb{N})$ of R over $\mathbb{F}_q$ by Exercise 1.5.2. So $\rho(f_i) < \rho(f_{i+1})$ for all i. Let $\mathbf{h}_i = ev_{\mathcal{P}}(f_i)$. Define

$$C(l) = \{\mathbf{c} \in \mathbb{F}_q^n | \mathbf{c} \cdot \mathbf{h}_j = 0 \text{ for all } j \leq l\}.$$

The map $ev_{\mathcal{P}}$ is surjective, so there exists an N such that $C(l) = 0$ for all $l \geq N$.

Let $\mathbf{y} \in \mathbb{F}_q^n$. Consider

$$s_{ij}(\mathbf{y}) = \mathbf{y} \cdot (\mathbf{h}_i * \mathbf{h}_j).$$

Then $S(\mathbf{y}) = (s_{ij}(\mathbf{y}) | 1 \leq i, j \leq N)$ is the *matrix of syndromes* of $\mathbf{y}$.

Exercise 1.5.13 Prove that

$$S(\mathbf{y}) = HDH^T,$$

where D is the $n \times n$ diagonal matrix with $\mathbf{y}$ on the diagonal and H is the $N \times n$ matrix with rows $\mathbf{h}_1, \ldots, \mathbf{h}_N$. Use this fact to show that

$$\text{rank } S(\mathbf{y}) = wt(\mathbf{y}).$$

Let $L(l)$ be the vector space with basis $f_1, \ldots, f_l$. Let $l(i,j)$ be the smallest l such that $f_i f_j \in L(l)$. Define

$$N(l) = \{(i,j) \mid l(i,j) = l + 1\}.$$

Let $\nu(l)$ be the number of elements of $N(l)$.

Exercise 1.5.14 Show that $i_1 < \cdots < i_t \leq r$ and $j_t < \cdots < j_1 \leq r$, if $(i_1, j_1), \ldots, (i_t, j_t)$ is an enumeration of the elements of $N(l)$ in increasing order with respect to the lexicographical order.

Exercise 1.5.15 Suppose that $\mathbf{y} \in C(l) \setminus C(l+1)$. Prove that

$$s_{i_u j_v}(\mathbf{y}) = \begin{cases} 0 & \text{if } u + v \leq t \\ \text{not zero} & \text{if } u + v = t + 1. \end{cases}$$

Use this fact together with Exercises 1.5.13 and 1.5.14 to prove that

$$wt(\mathbf{y}) \geq \nu(l).$$

Define

$$d_{ORD}(l) = \min\{\nu(l') | l' \leq l\}$$

$$d_{ORD,\mathcal{P}}(l) = \min\{\nu(l') | l' \geq l,\ C(l') \neq C(l'+1)\}.$$

As a consequence of the definitions and Exercise 1.5.15 we get the following theorem.

Theorem 1.5.16 *The numbers $d_{ORD,\mathcal{P}}(l)$ and $d_{ORD}(l)$ are lower bounds for the minimum distance of $C(l)$:*

$$d(C(l)) \geq d_{ORD,\mathcal{P}}(l) \geq d_{ORD}(l).$$

Exercise 1.5.17 *Reed-Solomon codes.* Let $R = \mathbb{F}_q[X]$. Let ρ be the order function defined as $\rho(f) = \deg(f)$. Let α be a primitive element of $\mathbb{F}_q$. Let $n = q - 1$ and $\mathcal{P} = \{\alpha^0, \ldots, \alpha^{n-1}\}$.

Prove that $(X^{i-1}|i \in \mathbb{N})$ is a well-behaving sequence and $l(i,j) = i + j - 1$.

Show that $C(l)$ is a cyclic code with defining set $\{0, 1, \ldots, l-1\}$ and $d_{ORD}(l) = l + 1$. Thus the BCH bound is obtained.

Exercise 1.5.18 Let ρ be a weight function and $(f_i|i \in \mathbb{N})$ a well-behaving sequence. Let $\rho_i = \rho(f_i)$. Show that $N(l) = \{(i,j)|\rho_i + \rho_j = \rho_{l+1}\}$.

Exercise 1.5.19 This is a continuation of Exercise 1.5.8 with $\mathbb{F} = \mathbb{F}_{16}$. Prove that $d_{ORD}(l) = \nu(l) = l - 5$ for all $l \geq 17$ and verify the numbers in the following table.

l	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
ρ_l	0	4	5	8	9	10	12	13	14	15	16	17	18	19	20	21
$\nu(l)$	2	2	3	4	3	4	6	6	4	5	8	9	8	9	10	12
$d_{ORD}(l)$	2	2	3	3	3	4	4	4	4	5	8	8	8	9	10	12

Show that there are exactly 64 zeros of the ideal $X^5 + Y^4 + Y$ with coordinates in $\mathbb{F}_{16}$. Denote this zero set by $\mathcal{P}$. Determine $d_{ORD,\mathcal{P}}(l)$ for all l.

Exercise 1.5.20 Suppose that ρ is a weight function. Let γ be the number of gaps. Show that $d_{ORD}(l) \geq l + 1 - \gamma$.

Exercise 1.5.21 *Reed-Muller codes.* Let $R = \mathbb{F}_q[X_1, \ldots, X_m]$ and let ρ be the order function associated to the lexicographical total degree order on the monomials of R. Let $n = q^m$. Let $\mathcal{P} = \{P_1, \ldots, P_n\}$ be an enumeration of the q^m points of $\mathbb{F}_q^m$.

Show that $\nu(l) = \prod(\epsilon_i + 1)$ and $d_{ORD}(l) = (\sum \epsilon_i) + 1$ when $f_{l+1} = \prod X_i^{\epsilon_i}$.

Now suppose that $f_{l+1} = X_m^{r+1}$. Then $\{f_i|i \leq l\}$ is the set of monomials of degree at most r. The corresponding words $\{\mathbf{h}_i|i \leq l\}$ generate $RM_q(r,m)$, the Reed-Muller code over $\mathbb{F}_q$ of order r in m variables. So $C(l)$ is the dual of $RM_q(r,m)$which is in fact equal to $RM_q((q-1)m - r - 1, r)$.

Write $r + 1 = \rho(q-1) + \mu$ with $\rho, \mu \in \mathbb{N}_0$ such that $\mu < q - 1$. Prove that $d(C(l)) = d_{ORD,\mathcal{P}}(l) = (\mu + 1)q^\rho$.

1.6 Notes

We use [9, 14, 15] as a reference for the theory of Gröbner bases, and [33, 34] for the theory of error-correcting codes. The computer algebra packages Axiom [29], GAP [20] and Macaulay [38] are used for the computations.

The weightenumerator and MacWilliams identity is treated in [33, 34].

See [16] and the project on Golay codes in this book for more about automorphism groups of codes and its connection with designs.

For an algorithm to compute the automorphism group of a code we refer to [32].

For questions concerning complexity issues in coding theory we refer to [7]. The recent proof of the NP completeness of finding the minimum distance of a linear code is in [41]. This answers a problem posed in [11]. For cyclic codes there is an algorithm [8] to compute the weight enumerator that is much faster than the methods presented here.

See [13] for the tables of optimal q-ary codes for $q = 2$, 3 and 4. There is an online connection to the latest state of the table [12] which can also be used to propose a new worldrecord. The algorithm of Brouwer is incorporated in the coding theory package GUAVA [6, 37].

For finite geometry and projective system we refer to [27, 40].

The treatment of the weight enumerator in Section 1.3.4 is from [30, 40] and this way of computing the weight distribution has been implemented by [10].

The treatment of the Mattson-Solomon polynomial can be found in [33, 34]. The proof of Proposition 1.4.7 is from [33, Chap 6] or [34, Chap 8 §6]. The proof of Proposition 1.4.9 is from [34, Chap 8 §6 Theorem 24]. The relation with the ordinary Newton identies is explained in [34, Chap 8 §6 (52)].

The method in Section 1.4.2 to get the minimal codewords of cyclic codes is from [1, 2, 3, 4, 5]. This can be generalized to all linear codes as will be explained in the next chapter.

Goppa [21, 22, 23, 24, 25] used algebraic curves to construct codes. These codes are called nowadays geometric Goppa codes or algebraic geometry codes, and give asymptotically good codes, even better than the Gilbert-Varshamov bound [40]. The mathematics is quite deep and abstract. For the construction and the parameters of these codes one needs the theory of algebraic curves or algebraic function fields of one variable [39], in particular the Theorem of *Riemann-Roch*. The asymptotically good codes require the knowledge of *modular curves*. Several authors [17, 18, 19, 28, 31] have proposed a more elementary approach to algebraic geometry codes and this new method has much to do with Gröbner bases [36].

The notion of order order and weight functions and its relation with coding theory is developped in [26, 36].

Section 1.5 is from [28, 31, 36]. Theorem 1.5.9 is from [36]. The values of a order function form a semigroup in case of a weight function. The order bound is called the Feng-Rao bound and is computed in terms of the properties of the semigroup [31]. The way Reed-Muller codes are treated in Exercise 1.5.21 is from [26, 35].

A classical treatment of algebraic geometry codes is given in [39, 40].

Bibliography

[1] D. Augot, "Description of minimum weight codewords of cyclic codes by algebraic systems," Un. de Sherbrooke, preprint Aug. 1994.

[2] D. Augot, "Algebraic characterization of minimum codewords of cyclic codes," *Proc. IEEE ISIT'94,* Trondheim, Norway, June 1994.

[3] D. Augot, "Newton's identities for minimum codewords of a family of alternant codes," preprint 1995.

[4] D. Augot, P. Charpin and N. Sendrier, "Weights of some binary cyclic codewords throughout Newton's identities," *Eurocode '90*, Springer-Verlag, Berlin 1990.

[5] D. Augot, P. Charpin and N. Sendrier, "Studying the locator polynomial of minimum weight codewords of BCH codes," *IEEE Trans. Inform. Theory*, vol. -38, pp. 960-973, 1992.

[6] R. Baart, J. Cramwinckel and E. Roijackers, "GUAVA, a coding theory package," Delft Un. Techn., June 1994.

[7] A. Barg, "Complexity issues in coding theory," to appear in *Handbook of Coding Theory,* (V.S. Pless, W.C. Huffman and R.A. Brualdi eds.), Elsevier.

[8] A. Barg and I. Dumer, "on computing the weight spectrum of cyclic codes," *IEEE Trans. Inform. Theory*, vol. 38, pp. 1382-1386, 1992.

[9] T. Becker and V. Weispfenning, *Gröbner basis; a computational approach to commutative algebra,* Springer, Berlin 1993.

[10] M. Beckker and J. Cramwinckel, "Implementation of an algorithm for the weight distribution of block codes," Modelleringcolloquium, Eindhoven Un. Techn., Sept. 1995.

[11] E.R. Berlekamp, R.J. McEliece and H.C.A. van Tilborg, "On the inherent intractibility of certain coding problems," *IEEE Trans. Inform. Theory,* vol. -24, pp. 384-386, 1978.

[12] A.E. Brouwer, http://www.win.tue.nl/win/math.dw.voorlincod.html

[13] A.E. Brouwer and T. Verhoeff, "An updated table of minimum-distance bounds for binary linear codes," *IEEE Trans. Inform. Theory,* vol. -39, pp. 662-677, Mar. 1993.

[14] A.M. Cohen, "Gröbner bases, an introduction," this volume.

[15] D. Cox, J. Little and D. O'Shea, *Ideals, varieties and algorithms; an introduction to computational algebraic geometry and commutative algebra,* Springer, Berlin 1992.

[16] H. Cuypers and H. Sterk, "Working with permutation groups," this volume.

[17] G.-L. Feng and T.R.N. Rao, "A simple approach for construction of algebraic-geometric codes from affine plane curves," *IEEE Trans. Inform. Theory*, vol. -40, pp. 1003-1012, July 1994.

[18] G.-L. Feng and T.R.N. Rao, "Improved geometric Goppa codes Part I, Basic Theory," *IEEE Trans. Inform. Theory*, vol. -41, pp. 1678-1693, Nov. 1995.

[19] G.-L. Feng, V. Wei, T.R.N. Rao and K.K. Tzeng, "Simplified understanding and efficient decoding of a class of algebraic-geometric codes," *IEEE Trans. Inform. Theory*, vol. -40, pp. 981-1002, July 1994.

[20] GAP: Group algorithms and programming, Math. Dept. RWTH Aachen, gap-forummath.rwth.aachen

[21] V.D. Goppa, "Codes associated with divisors," *Probl. Peredachi Inform.* vol. 13 (1), pp. 33-39, 1977. Translation: *Probl. Inform. Transmission*, vol. 13, pp. 22-26, 1977.

[22] V.D. Goppa, "Codes on algebraic curves," *Dokl. Akad. Nauk SSSR,* vol. 259, pp. 1289-1290, 1981. Translation: *Soviet Math. Dokl.*, vol. 24, pp. 170-172, 1981.

[23] V.D. Goppa, "Algebraico-geometric codes," *Izv. Akad. Nauk SSSR*, vol. 46, 1982. Translation: *Math. USSR Izvestija*, vol. 21, pp. 75-91, 1983.

[24] V.D. Goppa, "Codes and information," *Russian Math. Surveys*, vol. 39, pp. 87-141, 1984.

[25] V.D. Goppa, *Geometry and codes*, Mathematics and its Applications, vol. 24, Kluwer Acad. Publ., Dordrecht 1991.

[26] P. Heijnen and R. Pellikaan, "Geneneralized Hamming weights of q-ary Reed-Muller codes," to appear in *IEEE Trans. Inform. Theory.*

[27] J.W.P. Hirschfeld and J.A. Thas, *General Galois geometries,* Oxford University Press, Oxford 1991.

[28] T. Høholdt, J.H. van Lint and R. Pellikaan, "Algebraic geometry codes," to appear in *Handbook of Coding Theory,* (V.S. Pless, W.C. Huffman and R.A. Brualdi eds.), Elsevier.

[29] R.D. Jenks and R.S. Sutor, *Axiom. The scientific computation system,* Springer-Verlag, New York 1992.

[30] G.L. Katsman and M.A. Tsfasman, "Spectra of algebraic-geometric codes," *Probl. Peredachi Inform*, vol. 23 (4), pp. 19-34, 1987. Translation: Probl. Inform. Transmission, vol. 23, pp. 262-275, 1987.

[31] C. Kirfel and R. Pellikaan, "The minimum distance of codes in an array coming from telescopic semigroups," *IEEE Trans. Inform. Theory,* vol. 41, pp. 1720-1732, Nov. 1995.

[32] J. Leon, "Computing the automorphism groups of error-correcting codes," *IEEE Trans. Inform. Theory,* vol. 28, pp. 496-511, 1982.

[33] J.H. van Lint, *Introduction to coding theory,* Graduate Texts in Math. **86**, Springer-Verlag, Berlin 1982.

[34] F.J. MacWilliams and N.J.A. Sloane, *The theory of error-correcting codes,* North-Holland Math. Library vol. 16, North-Holland, Amsterdam 1977.

[35] R. Pellikaan, "The shift bound for cyclic, Reed-Muller and geometric Goppa codes," *Proceedings AGCT-4, Luminy 1993*, Walter de Gruyter, Berlin 1996, pp. 155-175.

[36] R. Pellikaan, "On the existence of order functions," submitted to the proceedings of the Second Shanghai Conference on Designs, Codes and Finte Geometry, 1996.

[37] J. Simonis, "GUAVA: A computer algebra package for coding theory," *Proc Fourth Int. Workshop Algebraic Combinatorial Coding Theory,* Novgorod, Russia, Sept. 11-17, 1994, pp. 165-166.

[38] Ma. Stillman, Mi. Stillman and D. Bayer, Macaulay user manual.

[39] H. Stichtenoth, *Algebraic function fields and codes*, Universitext, Springer-Verlag, Berlin 1993.

[40] M.A. Tsfasman and S.G. Vlăduţ, *Algebraic-geometric codes,* Mathematics and its Application vol. 58, Kluwer Acad. Publ., Dordrecht 1991.

[41] A. Vardy, "The intractibility of computing the minimum distance of a code," to appear in *IEEE Trans. Inform. Theory.*

Chapter 2

Gröbner bases for decoding

2.1 Introduction

From the previous chapter one might get the impression that the theory of error-correcting codes is equivalent to the theory of finite geometry or arrangements over finite fields. This is not true from a practical point of view. A code is useless without a decoding algorithm. For engineers the total performance of the encoding and decoding scheme is important.

An introduction to the decoding problem is given in Section 2.2. In Section 2.3 we first restrict ourselves to cyclic codes where the system of syndrome equations can be explicitely solved using Gröbner basis techniques and later, in Section 2.5, to arbitrary linear codes. Although this method decodes up to half the true minimum distance, the complexity is not polynomial, because there is no polynomial algorithm known to compute Gröbner bases. The algorithms of Euclid, Sugiyama and Berlekamp-Massey give an efficient way to decode cyclic codes is by solving the key equation.

All references and suggestions for further reading will again be given in the notes of Section 2.6.

2.2 Decoding

Let C be a linear code. Decoding is the inverse operation of encoding. A *decoder* is a map

$$\mathcal{D} : \mathbb{F}_q^n \longrightarrow C \cup \{?\},$$

such that $\mathcal{D}(\mathbf{c}) = \mathbf{c}$ for all $\mathbf{c} \in C$. Let $\mathbf{y}$ be a *received word*. Then $\mathcal{D}(\mathbf{y})$ is a codeword or equal to ?, in case of a *decoding failure*

Decoding by *error detection* does the following. Let H be a parity check matrix of C. The output of the decoder is $\mathbf{y}$ if $\mathbf{y}H^T = 0$, and ? otherwise.

If the received word $\mathbf{y}$ is again a codeword, but not equal to the one sent, then the decoder gives $\mathbf{y}$ as output and we have a *miscorrection* also called a *decoding error*.
Let $C \subseteq \mathbb{F}_q^n$ be the code with minimum distance d that is used to transmit information over a noisy channel. If the codeword $\mathbf{c}$ is transmitted at one side of the channel and $\mathbf{y}$ is received at the other end, then we say that the *error* $\mathbf{e} = \mathbf{y} - \mathbf{c}$ has occurred:

$$\mathbf{y} = \mathbf{c} + \mathbf{e}.$$

A decoder $\mathcal{D}$ is called a *minimum distance decoder* if $\mathcal{D}(\mathbf{y})$ is a codeword that is nearest to $\mathbf{y}$ with respect to the Hamming metric for all $\mathbf{y}$.

Minimum distance decoding is similar to finding a codeword of minimal weight. If $\mathbf{y}$ is a received word, then one has to find a word in the coset $\mathbf{y} + C$ of minimal weight. Such a word is called a *coset leader*. Having a list of all coset leaders requires a memory of q^{n-k} of such elements and is only efficient for codes of small redundancy.

If the Hamming weight of the error-vector is at most $\lfloor (d-1)/2 \rfloor$, then $\mathbf{c}$ is the unique codeword which has the smallest distance to $\mathbf{y}$, so the error can be corrected. The value $t = \lfloor (d-1)/2 \rfloor$ is called the *error-correcting capability* or *capacity* of the code.

Let H be a parity check matrix for C, so $\mathbf{c}H^T = 0$ for all $\mathbf{c} \in C$. After receiving $\mathbf{y}$ one computes the vector of *syndromes*

$$\mathbf{s} = \mathbf{y}H^T.$$

Since $\mathbf{y} = \mathbf{c} + \mathbf{e}$ we have that $\mathbf{s} = \mathbf{y}H^T = \mathbf{c}H^T + \mathbf{e}H^T = \mathbf{e}H^T$ and the problem becomes: given $\mathbf{s}$, find a vector $\mathbf{e}$ of lowest Hamming weight such that $\mathbf{e}H^T = \mathbf{s}$.

A decoder $\mathcal{D}$ is called a *bounded distance decoder* which *corrects t errors* if $\mathcal{D}(\mathbf{y})$ is a codeword that is nearest to $\mathbf{y}$ for all $\mathbf{y}$ such that $d(\mathbf{y}, C) \leq t$. We say that $\mathcal{D}$ *decodes up to half the minimum distance* if it corrects $\lfloor (d-1)/2 \rfloor$ errors.

Proposition 2.2.1 *Let C be a linear code in $\mathbb{F}_q^n$ with parity check matrix H. Suppose we have a received word $\mathbf{y}$ with error vector $\mathbf{e}$ and we know a set J with at most $d(C) - 1$ elements and that contains the set of error positions. Then the error-vector $\mathbf{e}$ is the unique solution for $\mathbf{x}$ of the following linear equations:*

$$\mathbf{x}H^T = \mathbf{y}H^T \quad \textit{and} \quad x_j = 0 \ \textit{ for } \ j \notin J.$$

Exercise 2.2.2 Prove Proposition 2.2.1 and deduce that the syndrome of a received word with at most $\lfloor (d-1)/2 \rfloor$ errors is unique.

Proposition 2.2.1 shows that error decoding can be reduced to the problem of finding the error positions. If we want to decode all received words with t errors, then there are $\binom{n}{t}$ possible t-sets of error positions one has to consider. This number grows exponentially with n if t/n tends to a non-zero real number. The

decoding problem is hard. Only for special families of codes this problem has an efficient solution with practical applications. We will consider only bounded distance decoders.

Exercise 2.2.3 Assume that the channel is a *q-ary symmetric channel*. This means that the probability that the symbol $x \in \mathbb{F}_q$ is changed in the symbol $y \in \mathbb{F}_q$ is the same for all $x, y \in \mathbb{F}_q$ and $x \neq y$, and does not depend on the position. The probability that a fixed symbol is changed in another symbol, distinct from the original one, is called the *crossover* probability and is denoted by P. Prove that the probability that an error vector $\mathbf{e}$ is equal to the word $\mathbf{c}$ of weight t is given by

$$\text{Prob}\{\mathbf{e} = \mathbf{c}\} = \left(\frac{P}{q-1}\right)^t (1-P)^{n-t}.$$

Show that the *undetected error probability* is given by

$$W_C\left(1-P, \frac{P}{q-1}\right) - (1-P)^n,$$

where $W_C(X, Y)$ is the homogeneous weightenumerator of C.

2.3 Decoding cyclic codes with Gröbner bases

Let C be an $[n, k, d]$ cyclic code with generator polynomial $g(X)$ and defining set $J = \{j_1, \ldots, j_r\}$. Let $\mathbb{F}_{q^e}$ an extension of $\mathbb{F}_q$ that contains all the zeros of $g(X)$. Let $\alpha \in \mathbb{F}_{q^e}$ be a primitive n-th root of unity. Then a parity check matrix of C is

$$H = \begin{pmatrix} 1 & \alpha^{j_1} & \alpha^{2j_1} & \cdots & \alpha^{(n-1)j_1} \\ 1 & \alpha^{j_2} & \alpha^{2j_2} & \cdots & \alpha^{(n-1)j_2} \\ 1 & \alpha^{j_3} & \alpha^{2j_3} & \cdots & \alpha^{(n-1)j_3} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & \alpha^{j_r} & \alpha^{2j_r} & \cdots & \alpha^{(n-1)j_r} \end{pmatrix}.$$

Now let $\mathbf{e} = e(X)$ be an error vector of a received word $\mathbf{y} = y(X)$. Then $\mathbf{s} = \mathbf{y}H^T = \mathbf{e}H^T$ and

$$s_i = y(\alpha^{j_i}) = e(\alpha^{j_i})$$

is the ith component of $\mathbf{s}$ for $i = 1, \ldots, r$. It is more convenient to consider the extension $\hat{H}$ of the matrix H, where $\hat{H}$ is the $n \times n$ matrix with ith row

$$(1 \ \alpha^i \ \alpha^{2i} \ \cdots \ \alpha^{(n-1)i})$$

for $i = 1, \ldots, n$. Define $\hat{\mathbf{s}} = \mathbf{e}\hat{H}^T$. The jth component of $\hat{\mathbf{s}}$ is

$$\hat{s}_j = e(\alpha^j) = \sum_{i=0}^{n-1} \alpha^{ij}$$

for $j = 1, \ldots, n$. If $j \in J(C)$, then $\hat{s}_j = e(\alpha^j) = y(\alpha^j)$, so these syndromes are *known*.

From now on $\hat{s}_j$ will be denoted by s_j. Notice that the old s_i is now denoted by s_{j_i}.

Let $\mathbf{e} = e(X)$ be an error vector with error positions $i_1, i_2, \ldots, i_t$ and error values $e_{i_1}, e_{i_2}, \ldots, e_{i_t}$. Then the known syndromes will be

$$s_j = \sum_{m=1}^{t} e_{i_m} (\alpha^{i_m})^j, \qquad j \in J(C).$$

Consider the following system of equations over $\mathbb{F}_{q^e}[X_1, \ldots, X_v, Y_1, \ldots, Y_v]$:

$$\mathcal{S}(\mathbf{s}, v) = \begin{cases} \sum_{m=1}^{v} Y_m X_m^j &= s_j & \text{for } j \in J \\ Y_m^q &= Y_m & \text{for } m = 1, \ldots, v \\ X_m^n &= 1 & \text{for } m = 1, \ldots, v. \end{cases}$$

Conclude that $X_m = \alpha^{i_m}$ and $Y_m = e_{i_m}$ for $m = 1, \ldots, t$ is a solution of $\mathcal{S}(\mathbf{s}, t)$.

Exercise 2.3.1 Show that the equation $\sum_{m=1}^{v} Y_m X_m^{jq} = s_{jq}$ is a consequence of $\mathcal{S}(\mathbf{s}, v)$ for all $j \in J$.

Example 2.3.2 Let $J = \{1, 2\}$. If C is a cyclic code with defining set J, then its minimum distance is at least 3 by the BCH bound. So one can correct at least 1 error. The equations

$$\begin{cases} Y_1 X_1 &= s_1 \\ Y_1 X_1^2 &= s_2 \end{cases}$$

imply that the error position is $x_1 = s_2/s_1$ if there is exactly one error. If moreover $q = 2$, then $s_2 = s_1^2$, so $x_1 = s_1$.

We have the following.

Proposition 2.3.3 *Suppose that t errors occurred and $t \leq (d-1)/2$. Then the system $\mathcal{S}(\mathbf{s}, v)$ over $\mathbb{F}_{q^e}$ has no solution when $v < t$, and a unique solution up to permutations, corresponding to the error vector of lowest weight that satisfies the syndrome equations when $v = t$. The X_i of the solution are the error-locators and the Y_i the corresponding error values. If $v > t$, then for every j the system has a solution with $X_1 = \alpha^j$.*

Exercise 2.3.4 Prove Proposition 2.3.3 using Proposition 2.2.1.

The system $\mathcal{S}(\mathbf{s}, v)$ defines an ideal in the ring $\mathbb{F}_{q^e}[X_1, \ldots, X_v, Y_1, \ldots, Y_v]$. By abuse of notation we denote this ideal also by $\mathcal{S}(\mathbf{s}, v)$. The zero set of this ideal gives the error vector that occurred during the transmission. Gröbner basis techniques can be used to find the solutions of the equations.

Let $\prec_L$ be the lexicographic order with $Z_1 \prec_L Z_2 \prec_L \cdots \prec_L Z_w$. Then $\prec_L$ is an *elimination order*, that is to say it satisfies the following property.

Proposition 2.3.5 *Let I be an ideal in $\mathbb{F}[Z_1, Z_2, \ldots, Z_w]$. Let $\mathcal{G}$ be a Gröbner basis of I with respect $\prec_L$. Then $\mathcal{G} \cap \mathbb{F}[Z_1, Z_2, \ldots, Z_i]$ is a Gröbner basis of $I \cap \mathbb{F}[Z_1, Z_2, \ldots, Z_i]$.*

Let I be an ideal in $\mathbb{F}[Z_1, Z_2, \ldots, Z_w]$ with finitely many zeros over $\bar{\mathbb{F}}$ which are all defined over $\mathbb{F}$. Let V be the zero set in $\mathbb{F}^w$ of the ideal I. Then the zero set of $I \cap \mathbb{F}[Z_1, Z_2, \ldots, Z_i]$ is equal to the projection of V on the first i coordinates. This fact and Proposition 2.3.5 have a direct application to our problem of finding the solutions to system $\mathcal{S}(\mathbf{s}, v)$. Indeed, if $(x_1, \ldots, x_v)$ is the X-part of a solution to $\mathcal{S}(\mathbf{s}, v)$, then also any permutation of the x_i will be a solution (apply the same permutation to the Y-part of the solution). Hence every error-locator will appear as the first coordinate of a solution to $\mathcal{S}(\mathbf{s}, v)$. Thus we have sketched the proof of the following.

Proposition 2.3.6 *Suppose that t errors occurred and $t \leq (d-1)/2$. Let $g(X_1)$ be the monic generator of the ideal $\mathcal{S}(\mathbf{s}, t) \cap \mathbb{F}_{q^e}[X_1]$. Then the zeros of g are the error-locators.*

Before giving the final algorithm for the decoding, we must worry about one more thing: we assumed we knew how many errors occurred (the v occurring in system $\mathcal{S}(\mathbf{s}, v)$). Now note that the work required to solve the system $\mathcal{S}(\mathbf{s}, v)$ for large v is much more than for small v, and remark that in general words with many errors occur less often than words with few or no errors. The following theorem leads the way to an algorithm that implements this idea.

Theorem 2.3.7 *Suppose t errors occurred and $t \leq (d-1)/2$. Denote the monic error-locator polynomial by $l(X_1)$, that is to say $l(x) = 0$ if and only if x is an error-locator. Let $g(X_1)$ be the monic generator of the ideal $\mathcal{S}(\mathbf{s}, v) \cap \mathbb{F}_{q^e}[X_1]$, with $\mathcal{S}(\mathbf{s}, v)$ the ideal in $\mathbb{F}_{q^e}[X_1, \ldots, X_v, Y_1, \ldots, Y_v]$. Then*

$$g(X_1) = \begin{cases} 1 & \text{if } v < t \\ l(X_1) & \text{if } v = t \\ X_1^n - 1 & \text{if } v > t \end{cases}$$

Exercise 2.3.8 Show that in Proposition 2.3.3 and Theorem 2.3.7 it is allowed to replace the assumption "$t \leq (d-1)/2$" by the weaker statement "the received word has a unique closest codeword".

Exercise 2.3.9 Let $\mathcal{S}'(\mathbf{s}, v)$ be the system of equations which is obtained by replacing the equation $Y_m^q = Y_m$ in $\mathcal{S}(\mathbf{s}, v)$ by $Y_m^{q-1} = 1$ for all $m = 1, \ldots, v$. So the variables Y_m disappear if $q = 2$. How should Proposition 2.3.3 and Theorem 2.3.7 be restated for $\mathcal{S}'(\mathbf{s}, v)$?

We are now ready to state the algorithm to decode cyclic codes.

Algorithm 2.3.10
input($\mathbf{y}$);
$\mathbf{s} := \mathbf{y}H^T$;

if $s_j = 0$ for all $j \in J$
then output($\mathbf{y}$); stop; {no errors occurred}
else $v := 1$;
  $\mathcal{G} := \{1\}$;
  while $1 \in \mathcal{G}$ do
    $\mathcal{S} := \{\sum_{m=1}^{v} Y_m X_m^j - s_j, j \in J\} \cup \{Y_m^q - Y_m, X_m^n - 1,\ m = 1, \ldots, v\}$;
    $\mathcal{G} :=$ Gröbner($\mathcal{S}$);
    $v := v + 1$;
  od;
  {$1 \notin \mathcal{G}$ so there are solutions}
  $g(X_1) :=$ the unique element of $\mathcal{G} \cap \mathbb{F}_{q^e}[X_1]$};
  if $\deg(g(X_1)) > v$
  then output(?); stop { too many errors }
  else error-locators := {zeros of $g(Z_1)$}
    find error vector $\mathbf{e}$ by solving the linear equations
    as in Proposition 2.2.1
    output($\mathbf{y} - \mathbf{e}$)

We will treat an example in the project on the Golay codes.

2.3.1 One-step decoding of cyclic codes

In the system of equations $\mathcal{S}(\mathbf{s}, v)$ the syndromes s_j are considered to be known constants. In this section we treat the syndromes as variables and consider the corresponding system of equations

$$\mathcal{S}(v) = \begin{cases} \sum_{m=1}^{v} Y_m X_m^j &= S_j & \text{for } j \in J \\ Y_m^q &= Y_m & \text{for } m = 1, \ldots, v \\ X_m^n &= 1 & \text{for } m = 1, \ldots, v. \end{cases}$$

to define an ideal in the ring

$$\mathbb{F}_{q^e}[X_1, \ldots, X_v, Y_1, \ldots, Y_v, S_j, j \in J].$$

This of course has the advantage that we have to solve these equations only once, and that this can be done before we start to use the code. This is called the *preprocessing* of the decoding algorithm. In the actual running of the algorithm the values of the syndromes s_j of a received word are substituted in the variables S_j for $j \in J$.

Exercise 2.3.11 Let $\prec$ be a reduction order on the monomials $X_1, \ldots, X_v$, $Y_1, \ldots, Y_v$ and $S_j, j \in J$ such that the variables $S_j, j \in J$ are larger than $X_1, \ldots, X_v$ and $Y_1, \ldots, Y_v$. Show that $\mathcal{S}(v)$ is a Gröbner basis with respect to $\prec$.

The exercise gives the impression that we are done. But we have to elimination the variables $X_2, \ldots, X_v$ and $Y_1, \ldots, Y_v$. Therefore the variables X_1, $S_j, j \in J$

need to be smaller than $X_2, \ldots, X_v$, $Y_1, \ldots, Y_v$.

As an example, we have applied one-step decoding to binary cyclic codes with defining sets $\{1,3\}$, $\{1,3,5\}$ and $\{1,3,5,7\}$, respectively. Remark that the complete defining sets contain $\{1,2,3,4\}$, $\{1,2,3,4,5,6\}$ and $\{1,\ldots,8\}$, respectively. From the BCH-bound we know that these codes can correct $2,3$ and 4 errors, respectively. The Gröbner basis is computed with a lexicographic order in a way such that the basis contains a polynomial in X_1 and the syndrome-variables S_j. We consider binary codes. Thus the error values are always 1. Therefore we delete the variables Y_i in the equations. The equations of the form $X_m^n = 1$ are also left out. So the number of solutions is not finite anymore. The results are as follows.

Example 2.3.12 $q = 2$, $\{1,3\} \subseteq J(C)$.

$$\mathcal{S} = \left\{ \begin{array}{ccccccc} X_1 & + & X_2 & - & S_1 & = & 0 \\ X_1^3 & + & X_2^3 & - & S_3 & = & 0 \end{array} \right.$$

Order: $X_2 > X_1 > S_3 > S_1$
Error-locator polynomial with $X = X_1$:

$$S_1X^2 + S_1^2X + (S_1^3 + S_3).$$

Example 2.3.13 $q = 2$, $\{1,3,5\} \subseteq J(C)$.

$$\mathcal{S} = \left\{ \begin{array}{ccccccccc} X_1 & + & X_2 & + & X_3 & - & S_1 & = & 0 \\ X_1^3 & + & X_2^3 & + & X_3^3 & - & S_3 & = & 0 \\ X_1^5 & + & X_2^5 & + & X_3^5 & - & S_5 & = & 0 \end{array} \right.$$

Order: $X_3 > X_2 > X_1 > S_5 > S_3 > S_1$
Error-locator polynomial:

$$(S_3 + S_1^3)X^3 + (S_3S_1 + S_1^4)X^2 + (S_5 + S_3S_1^2)X + (S_5S_1 + S_3^2 + S_3S_1^3 + S_1^6).$$

Example 2.3.14 $q = 2$, $\{1,3,5,7\} \subseteq J(C)$.

$$\mathcal{S} = \left\{ \begin{array}{ccccccccccc} X_1 & + & X_2 & + & X_3 & + & X_4 & - & S_1 & = & 0 \\ X_1^3 & + & X_2^3 & + & X_3^3 & + & X_4^3 & - & S_3 & = & 0 \\ X_1^5 & + & X_2^5 & + & X_3^5 & + & X_4^5 & - & S_5 & = & 0 \\ X_1^7 & + & X_2^7 & + & X_3^7 & + & X_4^7 & - & S_7 & = & 0 \end{array} \right.$$

Order: $X_4 > X_3 > X_2 > X_1 > S_7 > S_5 > S_3 > S_1$
Error-locator polynomial:

$$\begin{array}{c} (S_1^6 + S_3^2 + S_5S_1 + S_3S_1^3)X^4 + (S_5S_1^2 + S_3^2S_1 + S_3S_1^4 + S_1^7)X^3 + \\ (S_7S_1 + S_5S_3 + S_3S_1^5 + S_1^8)X^2 + (S_7S_1^2 + S_5S_1^4 + S_3^3 + S_3S_1^6)X + \\ (S_7S_3 + S_7S_1^3 + S_5^2 + S_5S_3S_1^2 + S_5S_1^5 + S_3^3S_1 + S_3S_1^7 + S_1^{10}). \end{array}$$

Example 2.3.15 The error-locator polynomial for the 6-error correcting binary BCH code took four hours using Axiom. The coefficient of X^i has 20, 20, 22, 22, 20, 24 and 46 terms for $i = 6, 5, \ldots, 1$ and 0, respectively.

Exercise 2.3.16 Give S_i weighted degree i and let $\mathrm{wdeg}(X) = 1$. Notice that in the above examples the error-locator polynomial is homogeneous of total weighted degree $\binom{t+1}{2}$ if the BCH bound is $2t+1$. Show that this is always the case.

Looking at the formulas for the $2, 3$ and 4 error-correcting BCH codes one gets the impression that the number of terms grows exponentially (we do not know whether this is a fact). Thus specializing the values for the syndromes still would not give a decoding algorithm of polynomial complexity.

It is a priori not clear that substituting values for the syndromes in the variables after elimination gives the same answer as the original method with the syndromes as constants.

To make this point clear we introduce some notation. Let $\mathcal{G}$ be a subset of the polynomial ring in the variables $S_j, j \in J$, $X_1, \ldots, X_v$ and more. Then $\mathcal{G}_1$ is the subset of $\mathcal{G}$ of polynomials in the variables $S_j, j \in J$ and X_1 only. Let $\mathbf{s} = (s_j, j \in J)$ be a vector with coordinates in $\mathbb{F}_q$. Then $\mathcal{G}_1(\mathbf{s})$ is the set obtained from $\mathcal{G}_1$ by substituting the value s_j in S_j for all elements of $\mathcal{G}_1$ and $j \in J$.

Let $\prec_E$ be an elimination order on the monomials $X_1, \ldots, X_v, Y_1, \ldots, Y_v$ and $S_j, j \in J$ with the variables $X_1, \ldots, X_v$ and $Y_1, \ldots, Y_v$ larger than $S_j, j \in J$. That the one-step method works is stated as a fact in the following

Theorem 2.3.17 *Let $\mathcal{G}$ be a Gröbner basis of $\mathcal{S}(t)$ with respect to $\prec_E$. Let $\mathbf{y}$ be a received word such that t errors occurred. Let $\mathbf{s}$ be its syndrome. Assume that the closest codeword to $\mathbf{y}$ is unique. Then $\mathcal{G}_1$ is the Gröbner basis of $\mathcal{S}(t) \cap \mathbb{F}_q[X_1, S_j, j \in J]$ and $\mathcal{G}_1(\mathbf{s})$ generates the ideal of the error-locator polynomial.*

The proof relies on the fact that $\mathcal{S}(t)$ has a finite number of solutions.

The set $\mathcal{G}_1(\mathbf{s})$ consist of polynomials in the variable X_1. In examples the greatets common divisor of $\mathcal{G}_1(\mathbf{s})$ is among its elements, that is to say that $\mathcal{G}_1(\mathbf{s})$ is a (nonreduced) Gröbner bases. It is conjectured that this is always the case.

2.4 The key equation

Let C be a cyclic code of length n such that $\{1, 2, \ldots, \delta - 1\} \subset J(C)$. From the BCH bound we see that the minimum distance of C is at least δ. In this section we will give a decoding algorithm for such a code which has an efficient implementation and is used in practice. A drawback of the algorithm is that it only corrects errors of weight at most $(\delta - 1)/2$, whereas the true minimum distance can be larger than δ. An example of this phenomenon will be treated

in the project on the Golay codes.

The algorithms in this section work for cyclic codes that have any $\delta-1$ consecutive elements in their complete defining set. We leave it to the reader to make the necessary adjustments in the case where these elements are not $\{1, 2, \ldots, \delta-1\}$.

Let α be a primitive n-th root of unity. Let $\mathbf{c} = c(X) \in C$ be the transmitted codeword that is received as $\mathbf{y} = y(X) = c(X) + e(X)$, with $w = wt(\mathbf{e}) \leq (\delta-1)/2$. The support of $\mathbf{e}$ will be denoted by I. We then can compute the syndromes

$$s_i = A_i = e(\alpha^i) = y(\alpha^i) \qquad \text{for } i \in J(C),$$

where the A_i are the coefficients of the MS polynomial of $e(X)$, see Section 1.4.1. Since $\{1, 2, \ldots, \delta-1\} \subseteq J(C)$ and $2w \leq \delta-1$ we know all $A_1, A_2, \ldots, A_{2w}$. Write σ_i for the i-th symmetric function of the error positions and form the following set of generalized Newton identities, see Proposition 1.4.9

$$\begin{cases} A_{v+1} + \sigma_1 A_v + \cdots + \sigma_v A_1 = 0 \\ A_{v+2} + \sigma_1 A_{v+1} + \cdots + \sigma_v A_2 = 0 \\ \qquad\qquad\qquad \vdots \qquad\qquad \vdots \\ A_{2v} + \sigma_1 A_{2v-1} + \cdots + \sigma_v A_v = 0. \end{cases} \tag{2.1}$$

From the system with $v = w$ we have to find the σ_i. After we have done this, we can find the polynomial

$$\sigma(Z) = 1 + \sigma_1 Z + \sigma_2 Z^2 + \cdots + \sigma_w Z^w$$

which has as its zeros the reciprocals of the error locations. Finding the zeros of this polynomial is an easy task. We return to the problem of finding the coefficients σ_i.

Exercise 2.4.1 Consider the system of equations (2.1) as linear in the unknown $\sigma_1, \ldots, \sigma_w$ with coefficients in $\mathbb{F}_q(A_1, \ldots, A_{2w})$ the field of rational functions in $A_1, \ldots, A_{2w}$, which are treated now as variables. Then

$$\sigma_i = \frac{\Delta_i}{\Delta_0}$$

where Δ_i is the determinant of a certain $w \times w$ matrix according to Cramers rule. Then the Δ_i are polynomials in the A_i. Conclude that

$$\Delta_0 X^w + \Delta_1 X^{w-1} + \cdots + \Delta_w$$

is a closed form of the *generic* error-locator polynomial.

Substitute $A_{2i+1} = S_{2i+1}$ and $A_{2i} = S_i^2$ and compare the result with Examples 2.3.12, 2.3.13 and 2.3.14.

Exercise 2.4.2 Show that the matrix $(A_{i+j-1} | 1 \leq i, j \leq v)$ is nonsingular if and only if $v = w$, the number of errors. Hint: try to write the matrix as a triple product of matrices of known rank as done in Exercise 1.5.13.

The algorithm of *Arimoto-Peterson-Gorenstein-Zierler* (APGZ) solves the systems of linear equations (2.1) for $v = 1, \ldots, w$ by Gaussian elimination.

Exercise 2.4.3 What is the complexity of the algorithm of APGZ ?

Write

$$S(Z) = \sum_{i=1}^{\delta-1} A_i Z^{i-1},$$

then an alternative way of formulating (2.1) is that there exist polynomials $q(Z)$ and $r(Z)$ such that

$$\sigma(Z)S(Z) = q(Z)Z^{\delta-1} + r(Z), \quad \deg(r(Z)) \leq w-1,$$

or that there exists a polynomial $\omega(Z)$ of degree at most $w-1$ such that

$$\omega(Z) \equiv \sigma(Z)S(Z) \bmod Z^{\delta-1}. \tag{2.2}$$

This is called the *key equation*.

Exercise 2.4.4 Check that

$$\omega(Z) = \sum_{i \in I} e_i \alpha^i \prod_{j \in I \setminus \{i\}} (1 - \alpha^j Z),$$

by rewriting $\omega(Z)/\sigma(Z) \bmod Z^{\delta-1}$.

Exercise 2.4.5 Let $\sigma'(Z)$ be the formal derivative of $\sigma(Z)$. Show *Forney's formula* for the error values:

$$e_i = -\frac{\omega(\alpha^{-i})}{\sigma'(\alpha^{-i})}$$

for all error positions i. The polynomial $\omega(Z)$ is called the *error evaluator polynomial*.

We will discuss two algorithms that are faster than the one proposed in Exercise 2.4.3.

2.4.1 The algorithms of Euclid and Sugiyama

The *Euclidean algorithm* is a well known algorithm that can be used to compute the *greatest common divisor* of two univariate polynomials. We assume that the reader is familiar with this algorithm. In order to fix a notation, suppose we want to compute $\gcd(r_{-1}(Z), r_0(Z))$. Then the Euclidean algorithm proceeds as follows.

$$\begin{array}{rcllcll}
r_{-1}(Z) & = & q_1(Z)r_0(Z) & + & r_1(Z), & \deg(r_1) & < & \deg(r_0) \\
r_0(Z) & = & q_2(Z)r_1(Z) & + & r_2(Z), & \deg(r_2) & < & \deg(r_1) \\
 & \vdots & & \vdots & & & \vdots & \\
r_{j-2}(Z) & = & q_j(Z)r_{j-1}(Z) & + & r_j(Z), & \deg(r_j) & < & \deg(r_{j-1}) \\
r_{j-1}(Z) & = & q_{j+1}(Z)r_j(Z). & & & & &
\end{array}$$

From this we can conclude that $\gcd(r_{-1}(Z), r_0(Z)) = r_j(Z)$. The key equation can be solved with the algorithm of *Sugiyama* in the following way.

Algorithm 2.4.6 Set

$$r_{-1}(Z) = Z^{\delta-1}, \quad r_0(Z) = S(Z), \quad U_{-1}(Z) = 0, \quad U_0(Z) = 1,$$

and proceed with the algorithm of Sugiyama until an $r_k(Z)$ is reached such that

$$\deg(r_{k-1}(Z)) \geq \frac{1}{2}(\delta - 1) \qquad \text{and} \qquad \deg(r_k(Z)) \leq \frac{1}{2}(\delta - 3),$$

also updating

$$U_i(Z) = q_i(Z)U_{i-1}(Z) + U_{i-2}(Z).$$

Then the error-locator and evaluator polynomial are

$$\begin{aligned} \sigma(Z) &= \epsilon U_k(Z) \\ \omega(Z) &= (-1)^k \epsilon r_k(Z) \end{aligned}$$

where ϵ is chosen such that $\sigma_0 = \sigma(0) = 1$.

Exercise 2.4.7 Show that the $\sigma(Z)$ and $\omega(Z)$ resulting from the algorithm satisfy
1. $\omega(Z) = \sigma(Z)S(Z) \bmod Z^{\delta-1}$
2. $\deg(\sigma(Z)) \leq \frac{1}{2}(\delta - 1)$
3. $\deg(\omega(Z)) \leq \frac{1}{2}(\delta - 3)$.

We will not prove the correctness of the algorithm. The algorithm of Sugiyama is used to decode in the project on Golay codes.

2.4.2 The algorithm of Berlekamp-Massey

The algorithm of *Berlekamp-Massey* is an example of *dynamic programming* The algorithm is iterative, and in the j-th iteration the following problem is solved: find the pair $(\sigma_j(Z), \omega_j(Z))$ such that

1. $\sigma_j(0) = 1$
2. $\sigma_j(Z)S(Z) = \omega_j(Z)(\bmod Z^j)$
3. $d_j = \max\{\deg(\sigma_j), \deg(\omega_j) + 1\}$ is minimal.

It is rather technical to work out what has to be updated when proceeding to the next iteration. After the algorithm we will give a few remarks on the variables that are used.

Algorithm 2.4.8

1. $j = 0; \quad \sigma_0 = -\omega_0' = 1; \quad \sigma_0' = \omega_0 = 0; \quad d_0 = 0; \quad \Delta = 1.$

2. Δ_j = coefficient of Z^j in $\sigma_j(Z)S(Z) - \omega_j(Z)$.

3. If $\Delta_j = 0$ then
   $d_{j+1} := d_j$; $\sigma_{j+1} := \sigma_j$; $\omega_{j+1} := \omega_j$;
   $\sigma'_{j+1} := Z\sigma'_j$; $\omega'_{j+1} := Z\omega'_j$

4. If $\Delta_j \neq 0$ and $2d_j > j$ then
   $d_{j+1} := d_j$; $\sigma_{j+1} := \sigma_j - \Delta_j\Delta^{-1}\sigma'_j$; $\omega_{j+1} := \omega_j - \Delta_j\Delta^{-1}\omega'_j$;
   $\sigma'_{j+1} := Z\sigma'_j$; $\omega'_{j+1} := Z\omega'_j$

5. If $\Delta_j \neq 0$ and $2d_j \leq j$ then
   $d_{j+1} := j + 1 - d_j$; $\sigma_{j+1} := \sigma_j - \Delta_j\Delta^{-1}\sigma'_j$; $\omega_{j+1} := \omega_j - \Delta_j\Delta^{-1}\omega'_j$;
   $\Delta := \Delta_j$; $\sigma'_{j+1} := Z\sigma_j$; $\omega'_{j+1} := Z\omega_j$

6. If S_{j+1} is known then $j := j + 1$ and go to step 2; otherwise stop.

In the algorithm, the variables σ'_j and ω'_j are auxiliary. The Δ_j measures how far a solution to the j-th iteration is from being a solution to the $(j+1)$-th iteration. If $\Delta_j = 0$, the solution passes to the next iteration. If $\Delta_j \neq 0$, then the solution must be adjusted in such a way that the resulting $d_{j+1} = \max\{\deg(\sigma_{j+1}), \deg(\omega_{j+1}) + 1\}$ is minimal. In order to minimize this degree, the two cases 4 and 5 have to be distinguished.

Notice that in the algorithm of Sugiyama the degree of the polynomial decreases during the algorithm, whereas in the Berlekamp-Massey algorithm the degree of the polynomial increases. This is an advantage, since error vectors of small weight are more likely to occur than those of high weight.

2.5 Gröbner bases and arbitrary linear codes

We will start by a general construction of a code, and later show that in fact this gives all linear codes.

Let $\mathcal{P} = \{P_1, P_2, \ldots, P_n\} \subseteq \mathbb{F}_q^m$ be the set of zeros of a set of polynomials $\mathcal{G} = \{G_1, \ldots, G_u\}$ in $\mathbb{F}_q[X_1, X_2, \ldots, X_m]$. Let I be the ideal generated by $\mathcal{G}$. Define the ring R as

$$R = \mathbb{F}_q[X_1, \ldots, X_m]/I,$$

Let $F_1, F_2, \ldots, F_r$ be a basis of the $\mathbb{F}_q$-vector subspace L of R. Consider the evaluation map

$$ev_{\mathcal{P}} : L \longrightarrow \mathbb{F}_q^n.$$

The codes we consider here are

$$C = Im(ev_{\mathcal{P}})^{\perp}.$$

Thus $H = (F_i(P_j))$ is a parity check matrix of C. After introducing this algebraic setting, it is clear how Gröbner bases can be used for the decoding

problem. Let d be the minimum distance of C. Suppose we receive a vector $\mathbf{y}$ and we want to decode t errors, with $t \leq \lfloor (d-1)/2 \rfloor$. Then, after computing the syndromes

$$s_i = \sum_{j=1}^{n} y_j F_i(P_j)$$

we can form the following system of equations $\mathcal{S}(\mathbf{s}, v)$

$$\begin{cases} \sum_{j=1}^{v} Y_j F_i(X_{1j}, \ldots, X_{mj}) &= s_i & \text{for } i = 1, \ldots, r \\ G_i(X_{1j}, \ldots, X_{mj}) &= 0 & \text{for } j = 1, \ldots, v \text{ and } i = 1, \ldots, u \\ Y_j^q &= Y_j & \text{for } j = 1, \ldots, t, \end{cases}$$

with variables $X_{1j}, \ldots, X_{mj}$ for the coordinates of a copy of $\mathbb{F}_q^m$ for all $j = 1, \ldots, v$, and the variables $Y_1, \ldots, Y_v$ for the error values in $\mathbb{F}_q$. As in the case with cyclic codes, we see that if $(\mathbf{x}_1, \ldots, \mathbf{x}_v, \mathbf{y}_1, \ldots, \mathbf{y}_v)$, with $\mathbf{x}_j = (x_{1j}, \ldots, x_{mj})$, is a solution to $\mathcal{S}(\mathbf{s}, v)$, then so is

$$(\mathbf{x}_{\pi(1)}, \ldots, \mathbf{x}_{\pi(v)}, \mathbf{y}_{\pi(1)}, \ldots, \mathbf{y}_{\pi(v)}),$$

for any permutation π of $\{1, \ldots, v\}$. Hence a Gröbner basis $\mathcal{G}$ for the ideal $\mathcal{S}(\mathbf{s}, t)$ with respect to the lexicographic order with

$$Y_m > \cdots > Y_1 > X_{mv} > \cdots > X_{1v} > \cdots > X_{m1} > \cdots > X_{11}$$

will have elements that are polynomials in $X_{m1}, \ldots, X_{11}$ only. These elements generate the ideal $\mathcal{S}(\mathbf{s}, v) \cap \mathbb{F}_q[X_{11}, \ldots, X_{m1}]$. This intersection has no solution when $v < t$. If $v = t$, then the intersection is the *error-locator ideal*, that means that it has the set of error postions as zero set in $\mathbb{F}_q^m$. The error values can be found as before for cyclic codes with Proposition 2.2.1.

Example 2.5.1 Let C be an $[n, k, d]$ linear code with $r \times n$ parity check matrix H, where $r = n - k$. Consider the n columns of H as points $P_1, \ldots, P_n \in \mathbb{F}_q^r$ and set $\mathcal{P} = \{P_1, \ldots, P_n\}$. Then $\mathcal{P}$ is finite, so it is an algebraic set:

$$\mathcal{P} = V(I, \mathbb{F}_q), \quad I = \{G \in \mathbb{F}_q[X_1, \ldots, X_r] \mid G(P_1) = \cdots = G(P_n) = 0\}.$$

If we take as an r-dimensional vector space L the coordinate functions

$$L = \langle X_1, \ldots, X_r \rangle,$$

then it is clear that $C = Im(ev_{\mathcal{P}})^{\perp}$.

Exercise 2.5.2 Describe the Hamming code by the above method. What is the vanishing ideal in $\mathbb{F}_2[X_1, X_2, X_3]$ if one applies the above procedure to the Hamming code ?

Although in principle every linear code could be described and decoded in this way, the large number of variables will make it very impractical. The following exercise relaxes the number of variables a bit.

Exercise 2.5.3 Let C be an q-ary $[n, k, d]$ code. Let $r = n-k$. Let $H = (h_{ij})$ be a parity check matrix of C. Let m be a positive integer such that $q^m \geq n$. Show that there exist n distinct points $P_1, \ldots, P_n$ in $\mathbb{F}_q^m$ and polynomials $F_1, \ldots, F_r$ in $\mathbb{F}_q[X_1, \ldots, X_m]$ such that $F_i(P_j) = h_{ij}$.

Example 2.5.4 Let C be a cyclic code with defining set J. Instead of treating this as an arbitrary linear code as in the previous example, it is better to use the structure of the parity check matrix, as follows. Take $\mathcal{P} = \{1, \alpha, \ldots, \alpha^{n-1}\} \subseteq \mathbb{F}_{q^e}$, the set of n-th roots of unity. Hence

$$I = (X^n - 1)\mathbb{F}_{q^e}[X].$$

If we take for L the vector space

$$L = \langle X^j \mid j \in J \rangle$$

over $\mathbb{F}_{q^e}$, it is clear that C is a code as described above, and that the system $\mathcal{S}(\mathbf{s}, t)$ we have to solve, equals the one we already met in Section 2.3.

One-step decoding is done in the same way as for cyclic codes by treating the s_j as variables and the corresponding Theorem 2.3.17 holds.

The same methods applies for getting the minimal weight codewords of a linear code.

2.6 Notes

That the general decoding problem is hard can be made precise in terms of complexity theory. See [3, 5].

Formulas for the probability of a decoding error or failure for several decoders and the relation with the weightenumerator is given in [6, 24]. Some history of the origins of decoding algorithms can be found in [2].

The original idea of one-step decoding is from [9, 10] and [30]. See also [38].

The method to decode cyclic codes up to half the actual minimum distance using Gröbner basis is from [11, 12, 13]. The extension to arbitrary linear codes is from [17, 18]. Theorem 2.3.17 is from [17, 18, 25]. The conjecture concerning $\mathcal{G}_1(\mathbf{s})$ is from [25]. The remark in Exercise 2.3.11 is from [25]. In this paper the work of [15] is used to transform a Gröbner basis of a zero dimensional ideal with respect to one reduction order into a Gröbner basis with respect to another one. The decoding is considerably faster by this method as is seen in the Project on the Golay code. Decoding constacyclic codes in Lee metric by the use of Gröbner bases is explained in [28]

A more efficient way to decode cyclic codes is by solving the key equation [1, 4, 20, 27, 31, 37]. The formula for the error values is from [19].

The material of Section 2.4 is from [6, 7, 26, 32]. This formulation of the Berlekamp-Massey algorithm is from [14].

For Reed-Solomon codes a hybrid of the algorithm of Berlekamp-Massey and Gröbner bases techniques is given in [39, 40, 41] to get all closest codewords of a received word.

Decoding arbitrary linear codes with Gröbner bases is from [17, 18]. This method can also be applied to get all minimal weight codewords as explained for cyclic codes in the previous chapter.

There are many papers on decoding algebraic geometry codes and we refer to the literature [8, 16, 21, 22, 23, 29].

The Berlekamp-Massey algorithm is generalized to polynomials in several variables by [34, 35, 36]. This theory has very much to do with the theory of Gröbner bases, but it solves another problem than Buchbergers algorithm. The algorithm is implemented in the decoding of algebraic geometry codes. See the literature cited above and [33]. The name *footprint* for the Δ-set is from [8].

Bibliography

[1] S. Arimoto, "Encoding and decoding of p-ary group codes and the correction system," (in Japanese) *Inform. Processing in Japan,* vol. 2, pp. 320-325, Nov. 1961.

[2] A. Barg, "At the dawn of the theory of codes," *Math. Intelligencer,* vol. 15, No. 1, pp. 20-27, 1993.

[3] A. Barg, "Complexity issues in coding theory," to appear in *Handbook of Coding Theory,* (V.S. Pless, W.C. Huffman and R.A. Brualdi eds.), Elsevier.

[4] E.R. Berlekamp, *Algebraic coding theory,* Aegon Park Press, Laguna Hills CA, 1984.

[5] E.R. Berlekamp, R.J. McEliece and H.C.A. van Tilborg, "On the inherent intractibility of certain coding problems," *IEEE Trans. Inform. Theory,* vol. IT-24, pp. 384-386, 1978.

[6] R.E. Blahut, *Theory and practice of error control codes*, Addison-Wesley, Reading 1983.

[7] R.E. Blahut, *Fast algorithms for digital signal processing*, Addison-Wesley, Reading 1985.

[8] R.E. Blahut, *Introduction to algebraic coding,* book in prepartation.

[9] A. Brinton Cooper III, "Direct solution of BCH decoding equations," *Communication, Control and Signal Processing,* Elsevier Sc. Publ., pp. 281-286, 1990.

[10] A. Brinton Cooper III, "Finding BCH error locator polynomials in one step," *Electronic Letters*, vol. 27 ,pp. 2090-2091, 1991.

[11] X. Chen, I.S. Reed, T. Helleseth and T.K. Truong, "Algebraic decoding of cyclic codes: a polynomial point of view," *Contemporary Math.* vol. 168, pp. 15-22, 1994.

[12] X. Chen, I.S. Reed, T. Helleseth and T.K. Truong, "Use of Gröbner bases to decode binary cyclic codes up to the true minimum distance," *IEEE Trans. Inform. Theory,* vol. IT-40, pp. 1654-1661, Sept. 1994.

[13] X. Chen, I.S. Reed, T. Helleseth and T.K. Truong, "General principles for the algebraic decoding of cyclic codes," *IEEE Trans. Inform. Theory,* vol. IT-40, pp. 1661-1663, Sept. 1994.

[14] J.L. Dornstetter, "On the equivalence of Berlekamp's and Euclid's algorithm," *IEEE Trans. Inform. Theory,* vol. IT-33, pp. 428-431, May 1987.

[15] J.C. Faugère, P. Gianni, D. Lazard and T. Mora, "Efficient computation of zero-dimensional Gröbner bases by a change of ordering," *Journ. Symb. Comp.,* vol. 16, pp. 329-344, 1993.

[16] G.-L. Feng and T.R.N. Rao, "Decoding of algebraic geometric codes up to the designed minimum distance," *IEEE Trans. Inform. Theory*, vol. IT-39, pp. 37-45, Jan. 1993.

[17] J. Fitzgerald, *Applications of Gröbner bases to linear codes,* Ph.D. Thesis, Louisiana State Un., Aug. 1996.

[18] J. Fitzgerald and R.F. Lax, "Decoding affine variety codes using Gröbner bases," to appear in *Designs, Codes and Cryptography.*

[19] G.D. Forney Jr., "On decoding BCH codes," *IEEE Trans. Inform. Theory,* vol. 11, pp. 549-557, 1965.

[20] D.C. Gorenstein and N. Zierler, "A class of error-correcting codes in p^m symbols," *Journ. SIAM,* vol. 9, pp. 207-214, 1961.

[21] T. Høholdt, J.H. van Lint and R. Pellikaan, "Algebraic geometry codes," to appear in *Handbook of Coding Theory,* (V.S. Pless, W.C. Huffman and R.A. Brualdi eds.), Elsevier.

[22] T. Høholdt and R. Pellikaan, "On decoding algebraic-geometric codes," *IEEE Trans. Inform. Theory,* vol. 41, pp. 1589-1614, Nov. 1995.

[23] J. Justesen, K.J. Larsen, H. Elbrønd Jensen, A. Havemose and T. Høholdt, "Construction and decoding of a class of algebraic geometric codes," *IEEE Trans. Inform. Theory*, vol. IT-35, pp. 811-821, July 1989.

[24] T. Kløve and V.I. Korzhik, *Error detecting codes,* Kluwer Acad. Publ. , Dordrecht 1995.

[25] P. Loustaunau and E.V. York, "On the decoding of cyclic codes using Gröbner bases," preprint # 249, Dept. Math., Univ. of Notre Dame, Sept. 1996.

[26] F.J. MacWilliams and N.J.A. Sloane, *The theory of error-correcting codes,* North-Holland Math. Library vol. 16, North-Holland, Amsterdam 1977.

[27] J.L. Massey, "Shift-register synthesis and BCH decoding," *IEEE Trans. Inform. Theory* vol. IT-15, pp.122-127, January 1969.

[28] J. Maucher and R. Kötter, "Decoding constacyclic codes in Lee- and Mannheim metric by the use of Gröbner bases," preprint, August 1996.

[29] R. Pellikaan, "On the efficient decoding of algebraic-geometric codes," *Proceedings of Eurocode 92, CISM Courses and Lectures*, vol. 339, pp. 231-253, Springer-Verlag, Wien-New York, 1993.

[30] W.T. Penzhorn, "On the fast decoding of binary BCH codes," *Proc. IEEE Int. Symp. Inform. Theory,* San Antonio, pp. 103, Jan. 1993.

[31] W.W. Peterson, "Encoding and error-correction procedures for the Bose-Chauduri codes," *IRE Trans. Inform. Theory,* vol. IT-6, pp.459-470, 1960.

[32] W.W. Peterson and E.J. Weldon, *Error-correcting codes,* MIT Pres, Cambridge 1977.

[33] K. Saints and C. Heegard, "Algebraic-geometric codes and multidimensional cyclic codes: A unified theory and algorithms for decoding using Gröbner bases," *IEEE Trans. Inform. Theory,* vol. 41, pp. 1733-1751, Nov. 1995.

[34] S. Sakata, "On determining the independent point set for doubly periodic arrays and encoding two-dimensional cyclic codes and their duals," IEEE Trans. Inform. Theory, vol. IT-27, pp. 556-565, 1981.

[35] S. Sakata, "Finding a minimal set of linear recurring relations capable of generating a given finite two-dimensional array," *Journal of Symbolic Computation,* vol. 5, pp. 321-337, 1988.

[36] S. Sakata, "Extension of the Berlekamp-Massey algorithm to N dimensions," *Information and Computation,* vol. 84, pp. 207-239, 1990.

[37] Y. Sugiyama, M. Kasahara, S. Hirasawa and T. Namekawa, "A method for solving the key equation for decoding Goppa codes," *Information and Control,* vol. 27, pp. 87-99, 1975.

[38] H.-J. Weber, *Algebraische Algorithmen zur Decodierung zyklischer Codes,* Master's Thesis, Univ. Dortmund, March 1994.

[39] D-J. Xin, "New approach to decoding Reed-Solomon codes based on generalized rational interpolation," in *Proc. Sixth Swedish-Russian International Workshop Inform. Trans.* August 1993, pp. 219-223.

[40] D-J. Xin, "Homogeneous interpolation problem and key equation for decoding Reed-Solomon codes," *Science in China* (Series A), vol. 37 No. 11, Nov. 1994.

[41] D-J. Xin, " Extension of the Welch-Berlekamp theorem and universal stragegy of decoding algorithm beyond BCH bound," *Science in China* (Series A), vol. 38 No. 11, Nov. 1995.

Chapter 3

Project: The Golay codes

3.1 Introduction

In this project we will give examples of methods described in the previous chapters on finding the minimum weight codewords, the decoding of cyclic codes and working with the Mathieu groups. The codes that we use here are the well known *Golay codes*. These codes are among the most beautiful objects in coding theory, and we would like to give some reasons why.

There are two Golay codes: the ternary cyclic code $\mathcal{G}_{11}$ and the binary cyclic code $\mathcal{G}_{23}$.

The ternary Golay code $\mathcal{G}_{11}$ has parameters $[11, 6, 5]$, and it is the unique code with these parameters. The automorphism group $Aut(\mathcal{G}_{11})$ is the Mathieu group M_{11}. The group M_{11} is simple, 4-fold transitive and has size $11 \cdot 10 \cdot 9 \cdot 8$. The supports of the codewords of weight 5 form the blocks of a 4-design, the unique Steiner system $S(4, 5, 11)$. The ternary Golay code is a perfect code, this means that the Hamming spheres of radius $(d-1)/2 = 2$ centered at the codewords of $\mathcal{G}_{11}$ exactly cover the whole space $\mathbb{F}_3^{11}$. The code $\mathcal{G}_{11}$ can be uniquely extended to a $[12, 6, 6]$ code, which we will denote by $\mathcal{G}_{12}$. The code $\mathcal{G}_{12}$ is self-dual and $Aut(\mathcal{G}_{12}) = M_{12}$: the simple, 5-fold transitive Mathieu group of size $12{\cdot}11{\cdot}10{\cdot}9{\cdot}8$. The supports of the codewords of weight 6 in $\mathcal{G}_{12}$ form a 5-design, the unique $S(5, 6, 12)$.

The binary Golay code $\mathcal{G}_{23}$ has similar properties. Its parameters are $[23, 12, 7]$, and it is the unique code with these parameters. The automorphism group $Aut(\mathcal{G}_{23})$ is the Mathieu group M_{23}. The group M_{23} is simple, 4-fold transitive and has size $23 \cdot 22 \cdot 21 \cdot 20 \cdot 48$. The supports of the codewords of weight 7 form the blocks of a 4-design, the unique Steiner system $S(4, 7, 23)$. The binary Golay code is a perfect code, so the Hamming spheres of radius 3 centered at the codewords of $\mathcal{G}_{11}$ exactly cover the whole space $\mathbb{F}_2^{23}$. The code $\mathcal{G}_{23}$ can

be uniquely extended to a $[24, 12, 8]$ code, which we will denote by $\mathcal{G}_{24}$. The code $\mathcal{G}_{24}$ is self-dual and $Aut(\mathcal{G}_{24}) = M_{24}$: the simple, 5-fold transitive Mathieu group of size $24 \cdot 23 \cdot 22 \cdot 21 \cdot 20 \cdot 48$. The supports of the codewords of weight 8 in $\mathcal{G}_{24}$ form a 5-design, the unique $S(5, 8, 24)$.

3.2 Minimal weight codewords of $\mathcal{G}_{11}$

$\mathcal{G}_{11}$ is the ternary cyclic code of length 11 with defining set $J = \{1\}$. It is a $[11, 6, d]$ code with complete defining set $J(\mathcal{G}_{11}) = \{1, 3, 4, 5, 9\}$. The generator polynomial is

$$g(X) = \prod_{j \in J(\mathcal{G}_{11})} (X - \alpha^j) = 2 + X^2 + 2X^3 + X^4 + X^5.$$

From the BCH bound we see that $d \geq 4$, and by computing Gröbner bases we will show that in fact $d = 5$. Moreover we will determine all codewords of minimal weight.

First we consider the system $\mathcal{S}_{\mathcal{G}_{11}}(4)$:

$$\mathcal{S}_{\mathcal{G}_{11}}(4) = \begin{cases} A_5 + \sigma_1 A_4 + \sigma_2 A_3 + \sigma_3 A_2 + \sigma_4 A_1 &= 0 \\ A_6 + \sigma_1 A_5 + \sigma_2 A_4 + \sigma_3 A_3 + \sigma_4 A_2 &= 0 \\ \vdots & \vdots \ \ \vdots \\ A_4 + \sigma_1 A_3 + \sigma_2 A_2 + \sigma_3 A_1 + \sigma_4 A_0 &= 0 \\ A_j = 0 \ \text{ for } \ j \in J(\mathcal{G}_{11}) & \\ A_{3j} = A_j^3 \ \text{ for } \ j = 1, \ldots, 11. & \end{cases}$$

Using $A_{3i} = A_i^3$ we can express every A_i with $i \in \{1, 2, \ldots, 10\} \setminus J(\mathcal{G}_{11})$ as a power of A_2 (this can be done since all of these i form a single cyclotomic coset). Setting $A_i = 0$ for $i \in J(\mathcal{G}_{11})$ and writing $A_2 = a$ and $A_0 = b$ this reduces $\mathcal{S}_{\mathcal{G}_{11}}(4)$ to

$$\mathcal{S}_{\mathcal{G}_{11}}(4) = \begin{cases} \sigma_3 a &= 0 \\ a^3 + \sigma_4 a &= 0 \\ a^9 + \sigma_1 a^3 &= 0 \\ a^{81} + \sigma_1 a^9 + \sigma_2 a^3 &= 0 \\ \sigma_1 a^{81} + \sigma_2 a^9 + \sigma_3 a^3 &= 0 \\ a^{27} + \sigma_2 a^{81} + \sigma_3 a^9 + \sigma_4 a^3 &= 0 \\ b + \sigma_1 a^{27} + \sigma_3 a^{81} + \sigma_4 a^9 &= 0 \\ \sigma_1 b + \sigma_2 a^{27} + \sigma_4 a^{81} &= 0 \\ a + \sigma_2 b + \sigma_3 a^{27} &= 0 \\ \sigma_1 a + \sigma_3 b + \sigma_4 a^{27} &= 0 \\ \sigma_2 a + \sigma_4 b &= 0 \\ b^3 - b &= 0. \end{cases}$$

Computing a Gröbner basis $\mathcal{G}$ with respect to the lexicographic order with

$$\sigma_4 > \sigma_3 > \sigma_2 > \sigma_1 > b > a$$

gives $\mathcal{G} = \{b, a\}$ and hence there are no nonzero codewords of weight at most 4. We conclude $d \geq 5$, and even $d = 5$, since the weight of the generator polynomial is $wt(g(X)) = 5$. To determine the minimum weight codewords we consider the system $\mathcal{S}_{\mathcal{G}_{11}}(5)$:

$$\mathcal{S}_{\mathcal{G}_{11}}(5) = \begin{cases} A_6 + \sigma_1 A_5 + \sigma_2 A_4 + \sigma_3 A_3 + \sigma_4 A_2 + \sigma_5 A_1 &= 0 \\ A_7 + \sigma_1 A_6 + \sigma_2 A_5 + \sigma_3 A_4 + \sigma_4 A_3 + \sigma_5 A_2 &= 0 \\ \vdots & \vdots \ \vdots \\ A_5 + \sigma_1 A_4 + \sigma_2 A_3 + \sigma_3 A_2 + \sigma_4 A_1 + \sigma_5 A_0 &= 0 \\ A_i = 0 \ \text{ for } \ i \in J(\mathcal{G}_{11}) \\ A_{3i} = A_i^3 \ \text{ for } \ i = 0, \ldots, 10 \end{cases}$$

Again we can reduce the system as we did in the system $\mathcal{S}_{\mathcal{G}_{11}}(4)$ and compute its Gröbner basis with respect to the lexicographic order with

$$\sigma_5 > \sigma_4 > \sigma_3 > \sigma_2 > \sigma_1 > b > a.$$

After 2 minutes using Axiom or 10 minutes using Macaulay, the resulting basis $\mathcal{G}$ is

$$\mathcal{G} = \begin{cases} \sigma_5 a + 2a^{31} + 2a^9, \sigma_4 a + a^3, \sigma_3 a + 2a^{107} + a^{41} + 2a^{19}, \\ \sigma_2 a + a^{79} + 2a^{35} + a^{13}, \sigma_1 a + a^{29} + 2a^7, \\ b + a^{77} + 2a^{55} + a^{33} + a^{11}, a^{133} + 2a^{111} + 2a^{89} + 2a^{67} + a^{45} + a, \end{cases}$$

where $a = A_2$ and $b = A_0$. From the triangular form of the basis $\mathcal{G}$, it is easy to see that the number of codewords of weight 5 in $\mathcal{G}_{11}$ equals the number of nonzero solutions to

$$f(X) = X^{133} + 2X^{111} + 2X^{89} + 2X^{67} + X^{45} + X = 0$$

in $\mathbb{F}_{3^5}$. We determine these solutions in the following exercise.

Exercise 3.2.1 Let $\alpha \in \mathbb{F}_{3^5}$ be a primitive element. Now show
1. $f(1) = 0$;
2. $f(\alpha^2) = 0$ (you can use a computer algebra package for this);
3. $f(\alpha^{11} X) = \alpha^{11} f(X)$.

Conclude from this that the complete set of zeros of $f(X)$ in $\mathbb{F}_{3^5} \setminus \{0\}$ is

$$M = \{\alpha^{i+11j} \mid i \in \{0, 1, \ldots, 10\} \setminus J(\mathcal{G}_{11}),\ j \in \{0, 1, \ldots, 21\}\}.$$

So the number of codewords of weight 5 is $\#M = 132$ and the locators of these words (i.e. the polynomials having as zeros the reciprocals of positions where the codewords have a nonzero value) are given by

$$\sigma(X, a) = \begin{cases} (a^{30} + a^8)X^5 + 2a^2 X^4 + \\ (a^{106} + 2a^{40} + a^{18})X^3 + \\ (2a^{78} + a^{34} + 2a^{12})X^2 + \\ (2a^{28} + a^6)X + 1, \end{cases}$$

with $a \in M$.
Since the code is cyclic, any shift of a codeword of weight 5 is again a codeword of weight 5. We can recognize this fact from M in the following way.

Exercise 3.2.2 Show that there exists a primitive 11-th root of unity β such that $\sigma(X, \alpha^{11}a) = \sigma(\beta X, a)$ for all $a \in M$.

Now we can conclude that the codewords of weight 5 consist of the 6 codewords with locator polynomials $\sigma(X, a)$, $a \in \{1, \alpha^2, \alpha^6, \alpha^7, \alpha^8, \alpha^{10}\}$, their cyclic shifts, and their non-zero multiples in $\mathbb{F}_3^{11}$.

Exercise 3.2.3 Let α again be a primitive element in $\mathbb{F}_{3^5}$, then $\beta = \alpha^{22}$ is a fixed 11-th root of unity. Check that the zeros of the 6 polynomials $\sigma(X, a)$ are:

polynomial	$\{i \mid \beta^{-i}$ is a zero$\}$
$\sigma(X, 1)$	2, 6, 7, 8, 10
$\sigma(X, \alpha^2)$	3, 4, 9, 10, 11
$\sigma(X, \alpha^6)$	1, 5, 8, 9, 11
$\sigma(X, \alpha^7)$	1, 2, 8, 10, 11
$\sigma(X, \alpha^8)$	2, 3, 5, 7, 9
$\sigma(X, \alpha^{10})$	3, 5, 8, 10, 11

Let $\mathcal{B}$ consists of the 6 subsets in $\{1, \ldots, 11\}$ of size 5 in the table and their cyclic shifts modulo 11. Then $|\mathcal{B}| = 66$. Show that $\mathcal{B}$ is the set of blocks of a 4-design, the Steiner system $S(4, 5, 11)$.

3.3 Decoding of $\mathcal{G}_{23}$ with Gröbner bases

Let $\mathcal{G}_{23}$ be the binary cyclic code of length 23 with defining set $J = \{1\}$. Then the complete defining set is $J(\mathcal{G}_{23}) = \{1, 2, 3, 4, 6, 8, 9, 12, 13, 16, 18\}$ and the code has parameters $[23, 12, d]$. The BCH bound states that $d \geq 5$ but in fact $d = 7$. This can be checked in the same way as we did in the previous section for the ternary Golay code. The computer algebra packages we tried, did not perform very well on the systems $\mathcal{S}_{\mathcal{G}_{23}}(w)$. Since the minimum distance is 7, $\mathcal{G}_{23}$ should be able to correct errors of weight at most 3. In this example we will decode a word with three errors.

Take

$$\mathbb{F}_{2^{11}} = \mathbb{F}_2[\beta]/(\beta^{11} + \beta^2 + 1)$$

and set $\alpha = \beta^{89}$. Then β is a primitive element of $\mathbb{F}_{2^{11}}$ and α has order 23.

The generator polynomial of the code is

$$g(X) = \prod_{j \in J(\mathcal{G}_{23})} (X - \alpha^j) = 1 + X + X^5 + X^6 + X^7 + X^9 + X^{11}.$$

Suppose we send the codeword $g(X)$, which corresponds to the binary vector

$$\mathbf{c} = (1, 1, 0, 0, 0, 1, 1, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)$$

over a noisy channel, and the following error occurs during transmission:

$$\mathbf{e} = (1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0).$$

As a result at the other end of the channel the following vector will be received:

$$\mathbf{y} = (0,1,0,1,0,1,1,1,0,1,0,1,0,0,0,0,0,1,0,0,0,0,0),$$

corresponding to the polynomial

$$r(X) = X + X^3 + X^5 + X^6 + X^7 + X^9 + X^{11} + X^{17}.$$

We will now decode the recieved word by applying the decoding algorithm. First we compute the syndrome:

$$s_1 = H\mathbf{y} = r(\alpha) = \alpha+\alpha^3+\alpha^5+\alpha^6+\alpha^7+\alpha^9+\alpha^{11}+\alpha^{17} = \beta^9+\beta^6+\beta^3+\beta^2+1.$$

Since $s_1 \neq 0$ we see that errors have occured during transmission.

We already remarked that the Y_i variables can be disposed of by setting them equal to 1, since 1 is the only error value that can occur.

Following the algorithm of Section 2.3 we set

$$\mathcal{S} = \{X_1 + \beta^9 + \beta^6 + \beta^3 + \beta^2 + 1, X_1^{23} + 1\}$$

and can conclude that there are no solutions since s_1 is not a 23-rd root of unity.

In the next step we set

$$\mathcal{S} = \{X_2 + X_1 + \beta^9 + \beta^6 + \beta^3 + \beta^2 + 1, X_2^{23} + 1, X_1^{23} + 1\}$$

and compute its Gröbner basis with respect to the lexicographic order with $X_2 > X_1$:

$$\mathcal{G} = \{1\}.$$

Since $1 \in \mathcal{G}$ there is no solution to these syndrome equations and we proceed with the loop of the algorithm. We set

$$\mathcal{S} = \{X_3 + X_2 + X_1 + \beta^9 + \beta^6 + \beta^3 + \beta^2 + 1, X_3^{23} + 1, X_2^{23} + 1, X_1^{23} + 1\},$$

and a Gröbner basis with respect to the lexicographic order with $X_3 > X_2 > X_1$ is computed:

$$\begin{cases} X_3 + X_2 + X_1 + \beta^9 + \beta^6 + \beta^3 + \beta^2 + 1, \\ X_2^2 + X_2X_1 + (\beta^9 + \beta^6 + \beta^3 + \beta^2 + 1)X_2 + X_1^2 + \\ \qquad +(\beta^9 + \beta^6 + \beta^3 + \beta^2 + 1)X_1 + \beta^6 + \beta^5 + \beta^2, \\ X_1^3 + (\beta^9 + \beta^6 + \beta^3 + \beta^2 + 1)X_1^2 + (\beta^6 + \beta^5 + \beta^2)X_1 + \beta^9 + \beta^5 + \beta^3. \end{cases}$$

This took 8 minutes using Axiom. We did the same computation with $X_j^{24} + X_j$ instead of $X_j^{23} + 1$ for $j = 1, 2, 3$ and it took only 90 seconds.

Now $1 \notin \mathcal{G}$ and there are solutions to the syndrome equations. The error locator polynomial is

$$g(X_1) = X_1^3 + (\beta^9 + \beta^6 + \beta^3 + \beta^2 + 1)X_1^2 + (\beta^6 + \beta^5 + \beta^2)X_1 + \beta^9 + \beta^5 + \beta^3$$

and its zeros are the error locators $\{\alpha^0, \alpha^3, \alpha^{17}\}$. Hence the errors occurred at positions 0, 3 and 17 and the word that was sent is

$$\mathbf{y} - (1,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0) =$$

$$(1,1,0,0,0,1,1,1,0,1,0,1,0,0,0,0,0,0,0,0,0,0).$$

We have recovered the transmitted codeword $\mathbf{c}$.

3.4 One-step decoding of $\mathcal{G}_{23}$

In this paragraph we will decode all error patterns of weight 3 that can occur in a codeword of the code $\mathcal{G}_{23}$ at once by computing the Gröbner basis for variable syndromes S. Apart from the advantage that all syndromes are treated at once, it also has the advantage that the computations take place over the field $\mathbb{F}_2$ instead of the large field $\mathbb{F}_{2^{11}}$. The system of equations is:

$$\mathcal{S} = \begin{cases} X_3 + X_2 + X_1 + S &= 0 \\ X_3^{23} + 1 &= 0 \\ X_2^{23} + 1 &= 0 \\ X_1^{23} + 1 &= 0. \end{cases}$$

The outcome of this set of equations is quite complicated. The result is much simpler if we consider the following set of equations.

$$\mathcal{S}' = \begin{cases} X_3 + X_2 + S + X_1 &= 0 \\ X_3^{24} + X_3 &= 0 \\ X_2^{24} + X_2 &= 0 \\ X_1^{24} + X_1 &= 0. \end{cases}$$

With the lexicographic order with $X_3 > X_2 > X_1 > S$ the computer was still not finished with its computations after 24 hours. Loustaunau and York did this example where they started with the above system, which is a Gröbner bases with respect to lexicographic order with $S > X_3 > X_2 > X_1$, and transformed it into a Gröbner bases with respect to lexicographic order with $X_3 > X_2 > X_1 > S$ as explained in the Notes of Chapter 2.2. Using the lexicographic order with $X_3 > X_2 > S > X_1$ we obtain the Gröbner basis:

$$\mathcal{G} = \begin{cases} X_3 + X_2 + S + X_1, \\ X_2^{24} + X_2, \\ X_2^2 S + X_2^2 X_1 + X_2 S^2 + X_2 X_1^2 + S^{256} + S^3 + S^2 X_1 + S X_1^2, \\ g(X_1), \\ X_1^{24} + X_1, \end{cases}$$

with

$$g(X_1) = \begin{cases} (S^{256} + S^3)X_1^{21} &+ (S^{257} + S^4)X_1^{20} + \\ (S^{260} + S^7)X_1^{17} &+ (S^{261} + S^8)X_1^{16} + \\ (S^{32} + S^9)X_1^{15} &+ (S^{33} + S^{10})X_1^{14} + \\ (S^{34} + S^{11})X_1^{13} &+ (S^{35} + S^{12})X_1^{12} + \\ (S^{36} + S^{13})X_1^{11} &+ (S^{37} + S^{14})X_1^{10} + \\ (S^{38} + S^{15})X_1^{9} &+ (S^{39} + S^{16})X_1^{8} + \\ (S^{40} + S^{17})X_1^{7} &+ (S^{64} + S^{41})X_1^{6} + \\ (S^{272} + S^{42})X_1^{5} &+ (S^{273} + S^{66} + S^{43} + S^{20})X_1^{4} + \\ (S^{44} + S^{21})X_1^{3} &+ (S^{68} + S^{45})X_1^{2} + \\ (S^{276} + S^{46})X_1 &+ (S^{277} + S^{70} + S^{47} + S). \end{cases}$$

We conclude that for a general syndrome S we find the error-locator polynomial

$$\gcd(g(X_1), X_1^{23} + 1).$$

These computations took 120 seconds using Axiom. The original set of equations $\mathcal{S}$ took 150 seconds. Macaulay did both these computations on the same computer in 3 seconds.

Exercise 3.4.1 Notice that the coefficient of X^i is divisible by $S^{23} + 1$ for all i. Denote $g(X_1)/(S^{23} + 1)$ by $h(X_1)$.

Exercise 3.4.2 Suppose $s = x_1 + x_2 + x_3$ with $x_j \in \mathbb{F}_{2^{11}}$ and $x_j^{23} = 1$ for all j. Show that $s^{23} = 1$ if and only if $x_i = x_j$ for some i, j with $1 \leq i < j \leq 3$.

Exercise 3.4.3 Compute $gcd(h(X_1), X_1^{23} + 1)$ with Euclid's algorithm in the ring $\mathbb{F}_q(S)[X_1]$ and show that it is a polynomial of degree 3 in X_1 and rational functions in S as coefficients.

3.5 The key equation for $\mathcal{G}_{23}$

In this section we will use the Euclidean algorithm to decode an error that occured during the transmission of a codeword of the binary Golay code $\mathcal{G}_{23}$. As we mentioned in Section 2.4, decoding a cyclic code C by solving the key equation only works for errors of weight at most $(\delta-1)/2$, where δ is is maximal such that $\{1, 2, \ldots, \delta - 1\} \subset J(C)$. In the case of the binary Golay code, this means we can only expect to decode errors of weight at most 2 in this way.

As in the previous section, we assume that the transmitted codeword was $g(X)$. Suppose the following error occurs:

$$\mathbf{e} = (1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0).$$

Then the received word is

$$\mathbf{y} = (0,1,0,0,0,1,1,1,0,1,0,1,0,0,0,0,0,1,0,0,0,0,0),$$

corresponding to the polynomial

$$r(X) = X + X^5 + X^6 + X^7 + X^9 + X^{11} + X^{17}.$$

After we receive this word, we can compute the following syndromes:

$$\begin{array}{lllllll} s_1 & = & r(\alpha) & & & = & \beta^{10} + \beta^9 + \beta^7 + \beta^6 + 1 \\ s_2 & = & r(\alpha^2) & = & s_1^2 & = & \beta^7 + \beta^5 + \beta^2 + \beta \\ s_3 & = & r(\alpha^3) & = & s_1^{256} & = & \beta^8 + \beta^7 + \beta^6 + \beta^5 \\ s_4 & = & r(\alpha^4) & = & s_1^4 & = & \beta^{10} + \beta^5 + \beta^4 + \beta^3 + \beta^2. \end{array}$$

Following Section 2.4 we define

$$S(Z) = s_1 + s_2 Z + s_3 Z^2 + s_4 Z^3$$

and we start the Euclidean algorithm on $S(Z)$ and Z^4. We find

$$Z^4 = S(Z)q_1(Z) + r_1(Z),$$

with

$$q_1(Z) = (\beta^9 + \beta^3 + \beta^2 + 1)Z + \beta^{10} + \beta^9 + \beta^5 + \beta$$

and

$$\begin{aligned} r_1(Z) &= (\beta^{10} + \beta^9 + \beta^7 + \beta^6 + \beta^5 + \beta^4)Z^2 + \\ &\quad (\beta^{10} + \beta^9 + \beta^7 + \beta^5 + \beta^4 + \beta^3)Z + \\ &\quad (\beta^9 + \beta^6 + \beta^2 + 1). \end{aligned}$$

In the following step we get

$$S(Z) = r_1(Z)q_2(Z) + r_2(Z),$$

with

$$q_2(Z) = (\beta^{10} + \beta^3 + \beta^2 + 1)Z + (\beta^{10} + \beta^7 + \beta^6 + \beta)$$

and

$$r_2(Z) = (\beta^7 + \beta^6 + \beta^3 + \beta^2 + \beta + 1).$$

Since $\deg(r_1(Z)) \geq 2$ and $\deg(r_2(Z)) \leq 1$ we can stop the algorithm and compute

$$\begin{aligned} U_2(Z) &= q_2(Z)U_1(Z) + U_0(Z) \\ &= q_2(Z)q_1(Z) + 1 \\ &= (\beta^9 + \beta^8 + \beta^6)Z^2 + \\ &\quad (\beta^7 + \beta^6 + \beta^3 + \beta^2 + \beta + 1)Z + \\ &\quad \beta^9 + \beta^8 + \beta^7 + \beta^3 + \beta^2 + \beta + 1. \end{aligned}$$

From this we find

$$\sigma(Z) = U_2(Z)/(\beta^9 + \beta^8 + \beta^7 + \beta^3 + \beta^2 + \beta + 1) =$$

$$(\beta^{10} + \beta^9 + \beta^7 + \beta^6)Z^2 + (\beta^{10} + \beta^9 + \beta^7 + \beta^6 + 1)Z + 1.$$

Since the zeros of $\sigma(Z)$ are $Z = 1$ and $Z = \alpha^6$, we conclude that the error locators are 1 and α^{17} and thus that the error vector is

$$\mathbf{e} = (1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0).$$

We retrieve the transmitted codeword by computing $\mathbf{c} = \mathbf{y} - \mathbf{e}$.

Exercise 3.5.1 Do the same example with the algorithm of Berlekamp-Massey instead of Euclid's algorithm.

3.6 Exercises

Let C be the binary cyclic code C of length 15 with defining set $J = \{1, 3, 5\}$. In the following, $\alpha \in \mathbb{F}_{16}$ will denote a primitive element satisfying

$$\alpha^4 + \alpha + 1 = 0.$$

Exercise 3.6.1 Show that the complete defining set is given by

$$J(C) = \{1, 2, 3, 4, 5, 6, 8, 9, 10, 12\},$$

and that C has generator polynomial

$$g(X) = 1 + X + X^2 + X^4 + X^5 + X^8 + X^{10}.$$

Determine the dimension of the code and apply the BCH bound on the minimum distance.

In order to find the true minimum distance of C, we will determine all codewords of weight 7.

Exercise 3.6.2 Write down the equations of the system $\mathcal{S}_C(7)$ and reduce the system by setting $A_0 = b$ and $A_7 = a$ and expressing everything in a, b and $\sigma_1, \sigma_2, \ldots, \sigma_7$. Compute a Gröbner basis for the ideal defined by $\mathcal{S}_C(7)$ and answer the following questions:

1. How many codewords of weight 7 does C have?
2. Determine a set M and polynomials $\sigma(X, a)$ such that $\sigma(X, a)$ has as zeros the locators of a codeword of weight 7 is and only if $a \in M$.
3. Prove that $\sigma(X, \alpha^i) = \sigma(\alpha^{13i}X, 1)$. What does this show?

We will now use code C to decode a word that is a transmitted codeword in which errors have occured. First we choose a codeword in C.

Exercise 3.6.3 Pick your favorite polynomial $m(X) \in \mathbb{F}_2[X]$ of degree at most 4 and encode it by computing

$$c(X) = m(X)g(X) \bmod (X^{15} + 1).$$

Now choose a random binary error vector $\mathbf{e}$ of weight at most 3 and compute the word $\mathbf{r}$ that is received at the other end of the channel:

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

We will decode the received codeword using all the algorithms we have discussed. If you want you can exchange the word $\mathbf{r}$ you ave chosen with someone else and try to decode the word "he/she sent you".

Exercise 3.6.4 Compute the syndromes $s_1 = r(\alpha)$, $s_3 = r(\alpha^3)$ and $s_5 = r(\alpha^5)$ and proceed with Algorithm 2.3.10. You have to use a computer algebra package that can compute Gröbner bases over $\mathbb{F}_{16}$. Compare your result with the codeword that was sent.

Now compute all syndromes $s_1, s_2, \ldots, s_6$ and define the syndrome polynomial

$$S(Z) = s_1 + s_2 Z + s_3 Z^2 + s_4 Z^3 + s_5 Z^4 + s_6 Z^5.$$

Set

$$\sigma(Z) = 1 + \sigma_1 Z + \sigma_2 Z^2 + \sigma_3 Z^3.$$

We want to determine the σ_i such that $\sigma(Z)$ has as its zeros the reciprocals of the error positions of $\mathbf{e}$. We have seen two algorithms for this.

Exercise 3.6.5 Apply Sugiyama's algorithm to the situation here: compute the greatest common divisor of Z^6 and $S(Z)$ until the stop criterion of the algorithm is reached. Determine $\sigma(Z)$ from this and determine its zeros and thus the error positions. Compare your result with the codeword that was sent.

Exercise 3.6.6 Determine $\sigma(Z)$ by applying the Berlekamp-Massey algorithm. Again find the error locators and compare this with your result from the previous exercise.

If the number of errors that were made during transmission is equal to 3, we can use the formulas we found by one-step decoding.

Exercise 3.6.7 Lookup in Example 2.3.13 the formula corresponding to a 3-error correcting binary BCH code, substitute the syndromes you have computed, and determine the zeros and hence the error positions of the equation.