Practical Work on Cyclic Codes

ESIEA - 5A

1 Presentation of Cyclic Codes

1.1 Definition of Cyclic Codes

Cyclic codes are a very important subclass of linear codes in which we find the *Reed Solomon* codes (used for NASA spatial communications, the Galileo project, audio CD codes...), Hamming codes (used in the French minitel), BCH codes, parity codes... This practical work aims at studying the class of cyclic codes.

DEFINITION *A code $\mathcal{C}$ is cyclic if the following conditions hold :*

1. *$\mathcal{C}$ is a linear code.*
2. *If $w = (w_1, \ldots, w_n)$ belongs to $\mathcal{C}$, then so does $w' = (w_n, w_1, \ldots, w_{n-1})$.*

..

Without loss of generality, we will work in the binary case (over $\mathbb{F}_2$) and we will identify a vector (i.e. a codeword) $w = (w_1, \ldots, w_n)$ with the polynomial

$$w(x) = \sum_{i=1}^{n-1} w_i.x^{i-1}$$

It means that we will consider the arithmetic in the ring R_n of binary polynomials modulo $x^n + 1$ (or equivalently $x^n + 1$).

The fundamental property is that a shift in a codeword w corresponds to multiplying the corresponding polynomial by the monomial x in R_n.

PROPOSITION *If $w(x)$ is the polynomial representation of a codeword in $\mathcal{C}$, then so is $w(x).f(x)$ for any polynomial f of degree d is such that $d \leq n - 1$.*

1.2 Generator Matrix and Polynomial of a Cyclic Code

Let $g(x)$ be a nonzero polynomial of minimum degree in $\mathcal{C}$. Then $g(x)$ generates the cyclic code $\mathcal{C}$ in the sense that any codeword $w(x) \in \mathcal{C}$ can be written in the form

$$w(x) = f(x).g(x) \quad \text{modulo } x^n + 1$$

for a suitable polynomial f.

The polynomial $g(x)$ is *generator polynomial* of the code $\mathcal{C}$.

Since any cyclic code is also a linear code, it then has a generator matirx as well. The following theorem precises the link between the generator matrix and the generator polynomial.

THEOREM *Let $\mathcal{C}$ a $[n, k, .]$-code. Then there exists a unique polynomial*

$$g(x) = \sum_{i=0}^{n-k} a_i x^i$$

with $a_{n-k} = 1$ such that

- *$g(x)$ is a divisor of $x^n + 1$.*
- *$\mathcal{C}$ is the cyclic code generated by $w = w_0 w_1 \dots w_{n-k} 0 \dots 0$ ($k-1$ zeroes padded at the end).*
- *Codewords $w = w_0 w_1 \dots w_{n-k} 0 \dots 0, \sigma(w) = 0 w_0 w_1 \dots w_{n-k} 0 \dots 0, \dots,$ $\sigma^{k-1}(w) = 0 \dots 0 w_0 w_1 \dots w_{n-k}$ are a basis for $\mathcal{C}$ where σ^i denotes the cyclic shift of order i. The generator matrix is then given by*

$$\begin{pmatrix} w_0 & w_1 & \dots & w_{n-k} & 0 & 0 & \dots & \dots & 0 \\ 0 & w_0 & w_1 & \dots & w_{n-k} & 0 & \dots & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \dots & \vdots & \vdots & \dots \\ 0 & 0 & \dots & \dots & 0 & w_0 & w_1 & \dots & w_{n-k} \end{pmatrix}$$

We have the equivalent Theorem (proof left as an exercise).

THEOREM *Let $\mathcal{C}$ a $[n, k, .]$-code with generator polynomial*

$$g(x) = \sum_{i=0}^{k-1} g_i x^i.$$

with $g_{k-1} = 1$. Then its generator matrix is the $(n-k-1, n)$-matrix given by :

$$\begin{pmatrix} g_0 & g_1 & \dots & w_{k-1} & 0 & 0 & \dots & \dots & 0 \\ 0 & g_0 & g_1 & \dots & g_{k-1} & 0 & \dots & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & \dots & 0 & g_0 & g_1 & \dots & w_{k-1} \end{pmatrix}$$

This polynomial description enables to give an even more compact representation of the code. Indeed, let us compare the memory requirements for different kind of block codes :

— A (n, M, d)-code requires to list (store) the M codewords as well as the coding procedure.

— Linear codes $[n, k, d]$ require to store the generator (n, k)-matrix only)

— Cyclic codes are decribed by a single polynomials with $n - k$ coefficients.

Not any polynomial can act as a generator polynomial of some cyclic code. We have the following important proposition.

PROPOSITION *If $g(x)$ is the generator polynomial of the cyclic code $\mathcal{C}$ of length n then $g(x)$ divides $x^n + 1$.* The proof is left as an exercise.

1.3 Encoding and Decoding with Cyclic Codes

To code any word m, we compute the codeword $w = m.G$ where G is the generator matrix. Equivalently, any codeword (with respect to its polynomial representation) is a multiple of the generatpr polynomial g modulo $x^n + 1$ and conversely any multiple of the generator polynomial g modulo $x^n + 1$ is a codeword.

Let us give the check matrix for a cyclic code.

THEOREM *The check matrix for a cyclic code with generator matrix G is*

$$H = \begin{pmatrix} h_k & h_{k-1} & \dots & h_0 & 0 & \dots & 0 \\ 0 & h_k & h_{k-1} & \dots & h_0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \dots & 0 & h_k & h_{k-1} & \dots & h_0 \end{pmatrix}$$

with $h(x) = \sum_{i=0}^{k} h_i x^i$ the polynomial defined by

$$h(x) = \frac{x^n + 1}{g(x)}$$

As a consequence, any polynomial p of degree less than n is a codeword if and only if

$$p(x).h(x) = 0 \quad \text{modulo } x^n + 1$$

If not we have one (or more) error.

The following proposition explains how to compute the error syndrom.

PROPOSITION *Let r a received word with polynomial representation $r(x) = c(x)+e(x)$ where $c(x)$ is the polynomial representation of a codeword and $e(x)$ describes the polynomial representation of the error.*

Then the polynomial representation of the syndrom is given by

$$s(x) = r(x) \quad modulo\ g(x) = e(x) \quad modulo\ g(x)$$

In order to correct the error (Meggit algorithm), if the Hamming weight of the error is less than $\frac{d-1}{2}$ then we correct as $r(x)-(r(x) \quad \text{modulo } g(x))$. In the general case, we use the Berlekamp-Massey algorithm (complexity in $\mathcal{O}(n \log_2(n))$).

2 Practical Work

2.1 Paperwork

Answer the following questions :

— Prove that k-repetition codes ($k = 2p + 1$) are cyclic codes. Give the codeword which generates the full code.

— Prove Proposition 1.1.

— Let $g(X) = 1 + x + x^3$. Give the codeword for the message $m = 1101$. Give the corresponding generator matrix G and verify that the codeword computed from m by $m.G$ is the same as that obtained right before.

— Give the generator polynomial $g(x)$ of the k-repetition code.

— Take $n = 3$. Show that the code $\mathcal{C}$ given by

$$\{(000), (110), (011), (101)\}$$

is cyclic and generated by $1 + x$.

— Let $g(x) = 1 + x + x^3$ of the Hamming code $[7, 4, 3]$. Give the check polynomial and the check matrix. Is the word 1010001 a codeword for this code.

2.2 Computer Work

You have to implement a simple program that will code, check for error and decode a codeword.

For that purpose, the main polynomial operations you need are

— multiplication of polynomials modulo a given polynomial,

— the Euclidean algorithm for binary polynomial which computer the remainder of the polynomial division modulo a given polynomial.

Here is a simple (sample) source code which implements the multiplication of polynomials modulo a given polynomial.

```
/* Multiplication dans GF(2)[X]/p(x) modulo le polynome p(x) dans GF(2)[X] */
/* Exemple ici p(x) = x^8 + x^7 + x^6 + x + 1                              */
#include <stdio.h>
#include <stdlib.h>

int main()
{
unsigned char a, b, p, carry, poly;

a = 0x94;
b = 0x94; /* calcul de a(x).b(x) mod p(x) */
p = 0;
carry = 0;
poly = 0xC3;

while(a && b)
{
if(b & 1) p ^= a;
b >>= 1;
if(a & 0x80) carry = 1;
else carry = 0;
a <<= 1;
if(carry)
a ^= poly;
}
printf("Resulat = %02X\n", p);
}
```

For the second operation, you will use the pseudo-code provided in the paper by written by Sheueling Chang Shantz entitled "From Euclids GCD to Montgomery Multiplication to the Great Divide", Section 2, page 3 (the paper is provided in the moodle repository along with the present document).