Homework 4 - Naiara Alonso Montes

Problem 1

Check if a binary linear $[16, 5]$-code with $d_{min} = 8$ satisfies the Griesmer bound.

We are working with a *Reed-Muller* code.

- $n = 16$
- $k = 5$
- $d_{min} = 8$
- $m = 4$
- $r = 1$

$$n = \sum_0^{k-1} \lceil \frac{d_{min}}{2^i} \rceil =$$

$$16 = \lceil \frac{8}{2^0} \rceil + \lceil \frac{8}{2^1} \rceil + \lceil \frac{8}{2^2} \rceil + \lceil \frac{8}{2^3} \rceil + \lceil \frac{8}{2^4} \rceil$$

The provided code satisfies the Griesmer bound.

Construct a generator matrix of the $[16, 5]$-code with $d_{min} = 8$

The generator matrix is constructed by evaluating these polynomials at all points in $\mathbb{F}_4^2$. First we need to get all binary information set of length $= 4$.

In [ ]:
```
import numpy as np

def generate_information_set(length):
    num_combinations = 2**length
    information_set = []
    for i in range(num_combinations):
      binary_string = bin(i)[2:].zfill(length)  # Convert to binary and p
      binary_list = [int(bit) for bit in binary_string]  # Convert to a l
      information_set.append(binary_list)
    return np.array(information_set)

print(generate_information_set(4))
```

```
[[0 0 0 0]
 [0 0 0 1]
 [0 0 1 0]
 [0 0 1 1]
 [0 1 0 0]
 [0 1 0 1]
 [0 1 1 0]
 [0 1 1 1]
 [1 0 0 0]
 [1 0 0 1]
 [1 0 1 0]
 [1 0 1 1]
 [1 1 0 0]
 [1 1 0 1]
 [1 1 1 0]
 [1 1 1 1]]
```

Now, with all information sets, we need k polynomial, with one of them being constant, and the others x_1, x_2, x_3, x_4.

The generator matrix G looks as follow:

$$G = \begin{pmatrix} 1 \\ \text{1 if 1 in position 1 of information set, 0 otherwise} \\ \text{1 if 1 in position 2 of information set, 0 otherwise} \\ \text{1 if 1 in position 3 of information set, 0 otherwise} \\ \text{1 if 1 in position 4 of information set, 0 otherwise} \end{pmatrix} \tag{1}$$

$$G = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix} \tag{2}$$

Let $n = 33$, $k = 19$. Use known non-asymptotic bounds in order to determine the range of possible values of the minimum distance.

Hamming bound

$$d_H = \sum_{i=0}^{\lfloor \frac{d-1}{2} \rfloor} \binom{n}{i} (q-1)^i \leq q^{n-k}$$

- $d = 0; d_H = 0$
- $d = 1; d_H = 1$
- $d = 2; d_H = 1$
- $d = 3; d_H = 34$
- $d = 4; d_H = 34$
- $d = 5; d_H = 562$
- $d = 6; d_H = 562$
- $d = 7; d_H = 6018$
- $d = 8; d_H = 6018$

- $d = 1 > q^{n-k}$

$$d_H = 7;\ d_{min} \leq 7$$

Singleton bound

$$d_S \leq n - k + 1$$

$$d_S \leq 15$$

Gilbert-Varshamov bound

$$d_{GV} = \sum\nolimits_{i=0}^{d-2} \binom{n-1}{i} (q-1)^i \leq q^{n-k}$$

- $d = 0$; $d_{GV} = 0$
- $d = 1$; $d_{GV} = 0$
- $d = 2$; $d_{GV} = 1$
- $d = 3$; $d_{GV} = 33$
- $d = 4$; $d_{GV} = 529$
- $d = 5$; $d_{GV} = 5489$
- $d = 6 > q^{n-k}$

$$d_{GV} = 5;\ d_{min} \leq 5$$

It turns into lower bound

Solution

$$6 \leq d_{min} \leq 7$$

The lower bound is 6 and the upper bound is 7. The results obtained in www.codetables.de/ are the same.

Problem 2

A BCH code of length 31 correcting 2 errors is used for transmitting messages. The primitive polynomial $p(x) = x^5 + x^2 + 1$ was used for constructing the code. At the output of the BSC we observe the sequence $y = 0101000001110101010011001111000$ (the smallest degree is the first). Find the decoded codeword by using the Peterson-Gorenstein-Zierler algorithm.

Some code parameters

- $d = 2t + 1 = 5$
- $b(x) = x^{27} + x^{26} + x^{25} + x^{24} + x^{21} + x^{20} + x^{17} + x^{15} + x^{13} + x^{11} + x^{10} + x^9 +$
- $s_1 = b(\alpha) = \alpha^{27} + \alpha^{26} + \alpha^{25} + \alpha^{24} + \alpha^{24} + \alpha^{21} + \alpha^{20} + \alpha^{17} + \alpha^{15} + \alpha^{13} + \alpha^{11}$

- $s_2 = b(\alpha^2) = s_1^2 = \alpha^{19}$
  $s_1 = b(\alpha^3) = \alpha^{19} + \alpha^{16} + \alpha^{13} + \alpha^{10} + \alpha + \alpha^{29} + \alpha^{20} + \alpha^{14} + \alpha^8 + \alpha^2 + \alpha^{30} +$
- $s_4 = b(\alpha^4) = s_2^2 = \alpha^7$

Algorithm

$$\begin{pmatrix} s_1 & s_2 \\ s_2 & s_3 \end{pmatrix} \begin{pmatrix} \Lambda_2 \\ \Lambda_1 \end{pmatrix} = \begin{pmatrix} -s_3 \\ -s_4 \end{pmatrix} \tag{3}$$

$$\Delta = \alpha^{28} + \alpha^7 = \alpha$$

$$\Delta_2 = \begin{vmatrix} \alpha^{25} & \alpha^3 \\ \alpha^{19} & \alpha^7 \end{vmatrix} = \alpha + \alpha^{22} = \alpha^{26} \tag{4}$$

$$\Delta_1 = \begin{vmatrix} \alpha^3 & \alpha^{19} \\ \alpha^7 & \alpha^3 \end{vmatrix} = \alpha^6 + \alpha^{22} = \alpha^{14} \tag{5}$$

$$\Lambda_2 = \frac{\alpha^{14}}{\alpha} = \alpha^{13}$$

$$\Lambda_1 = \frac{\alpha^{26}}{\alpha} = \alpha^{25}$$

$$\Lambda(x) = 1 + \alpha^{25}x + \alpha^{13}x^2$$

The roots of $\Lambda(x)$ are $x_1 = \alpha^{21}$ and $x_2 = \alpha^{28}$, inverses are α^{10} and α^3.

$$\begin{pmatrix} \alpha^{10} & \alpha^3 \\ \alpha^{20} & \alpha^6 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} \alpha^{25} \\ \alpha^{29} \end{pmatrix} \tag{6}$$

$$y = \alpha^{10} \cdot \alpha^6 + \alpha^{20} \cdot \alpha^3 = \alpha^{16} + \alpha^{23} = \alpha^7$$

$$y_1 = \frac{\begin{vmatrix} \alpha^{10} & \alpha^{25} \\ \alpha^{20} & \alpha^{29} \end{vmatrix}}{\alpha^7} = \frac{\alpha^4}{\alpha^7} = \alpha^{28} \tag{7}$$

$$y_2 = \frac{\begin{vmatrix} \alpha^{25} & \alpha^3 \\ \alpha^{29} & \alpha^6 \end{vmatrix}}{\alpha^7} = \frac{\alpha^{18}}{\alpha^7} = \alpha^{11} \tag{8}$$

Error values: α^{28} and α^{11}

$e(x) = \alpha^{28}x^{21} + \alpha^{11}x^{27}$

$c(x) = b(x) - e(x) = \alpha^7 x^{27} + x^{26} + x^{25} + x^{24} + \alpha^7 x^{21} + x^{20} + x^{17} + x^{15} + x^{13} + x$

Problem 3

A BCH code of length 31 correcting 2 errors is used for transmitting messages. The primitive polynomial $p(x) = x^5 + x^2 + 1$ was used for constructing the code. At the output of the BSC we observe the sequence

$y = 01010000011101010100110011110000$ *(the smallest degree is the first). Find the decoded codeword by using the Berlekamp-Massey algorithm.*

Based on the previous exercise we know:

$$s_1 = b(\alpha) = \alpha^{25};\ s_2 = b(\alpha^2) = s_1^2 = \alpha^{19};\ s_3 = b(\alpha^3) = \alpha^3;\ s_4 = b(\alpha^4) = s_2^2 = \alpha^7$$

Iter 1

- $\Delta = \Lambda_0 \cdot s_1 = 1 \cdot \alpha^{25} = \alpha^{25}$
- $B(x) = x \cdot B(x) = x \cdot 1 = x$
- $T(x) = \Lambda(x) + \Delta B(x) = 1 + \alpha^{25}x$
- $B(x) = \Delta^{-1}\Lambda(x) = \alpha^6 \cdot 1 = \alpha^6$
- $L = r - L = 1 - 0 = 1$
- $\Lambda(x) = T(x) = 1 + \alpha^{25}x$

Iter 2

- $\Delta = \Lambda_0 \cdot s_2 + \Lambda_1 \cdot s_1 = 1 \cdot \alpha^{19} + \alpha^{25} \cdot \alpha^{25} = \alpha^{19} + \alpha^{19} = 0$
- $B(x) = x \cdot B(x) = x \cdot \alpha^6 = \alpha^6 x$

Iter 3

- $\Delta = \Lambda_0 \cdot s_3 + \Lambda_1 \cdot s_2 = 1 \cdot \alpha^3 + \alpha^{25} \cdot \alpha^{19} = \alpha^3 + \alpha^{13} = \alpha^7$
- $B(x) = x \cdot B(x) = x \cdot \alpha^6 x = \alpha^6 x^2$
- $T(x) = \Lambda(x) + \Delta B(x) = (1 + \alpha^{25}x) + \alpha^7(\alpha^6 x^2) = 1 + \alpha 25x + \alpha^{13}x^2$
- $B(x) = \Delta^{-1}\Lambda(x) = \alpha^{24} \cdot (1 + \alpha^{25}x) = (\alpha^{24} + \alpha^{18}x)$
- $L = r - L = 3 - 1 = 2$
- $\Lambda(x) = T(x) = 1 + \alpha 25x + \alpha^{13}x^2$

Iter 2

- $\Delta = \Lambda_0 \cdot s_4 + \Lambda_1 \cdot s_3 + \Lambda_2 \cdot s_2 = 1 \cdot \alpha^7 + \alpha^{25} \cdot \alpha^3 + \alpha^{13} \cdot \alpha^{19} = \alpha^7 + \alpha^{28} + \alpha =$
- $B(x) = x \cdot B(x) = x \cdot \alpha^6 = \alpha^{24}x + \alpha^{18}x^2$

r	Δ	$B(x)$	$T(x)$	$\Lambda(x)$	L
0	0	1		1	0
1	α^{25}	α^6	$1+\alpha^{25}x$	$1+\alpha^{25}x$	1
2	0	$\alpha^6 x$	$1+\alpha^{25}x$	$1+\alpha^{25}x$	1
3	α^7	$\alpha^{24}+\alpha^{18}x$	$1+\alpha^{25}x+\alpha^{13}x^2$	$1+\alpha^{25}x+\alpha^{13}x^2$	2
4	0	$\alpha^{24}x+\alpha^{18}x^2$	$1+\alpha^{25}x+\alpha^{13}x^2$	$1+\alpha^{25}x+\alpha^{13}x^2$	2

$\Lambda(x) = 1 + \alpha^{25}x + \alpha^{13}x^2$

The obtained $\Lambda(x)$ is the same, so the result is also the same, error locators are inverses of the roots. Error locators: α^{21}, α^{27}.

Problem 4

Find the GCD D of two integers $a = 265$ and $b = 95$ by using the Euclidean algorithm.

Find the representation of the found GCD $D = la + jb$, where l and j are integers.

$$r_0 = a;\ r_1 = b$$

$$x_0 = 1;\ x_1 = 0$$

$$y_0 = 0;\ y_1 = 1$$

$$r_2 = r_0 + q_1 \cdot r_1 = a - q_1 \cdot b = 265 - 2 \cdot 95 = 75$$

$$x_2 = x_0 + q_1 \cdot x_1 = 1 - 2 \cdot 0 = 1$$

$$y_2 = y_0 + q_1 \cdot y_1 = 0 - 2 \cdot 1 = -2$$

$$r_3 = r_1 + q_2 \cdot r_2 = 95 - 1 \cdot 75 = 20$$

$$x_3 = x_1 + q_2 \cdot x_2 = 0 - 1 \cdot 1 = -1$$

$$y_3 = y_1 + q_2 \cdot y_2 = 1 - 1 \cdot (-2) = 3$$

$$r_4 = r_2 + q_3 \cdot r_3 = 75 - 3 \cdot 20 = 15$$

$$x_4 = x_2 + q_3 \cdot x_3 = 1 - 3 \cdot (-1) = 4$$

$$y_4 = y_2 + q_3 \cdot y_3 = -2 - 3 \cdot 3 = -11$$

$$r_5 = r_3 + q_4 \cdot r_4 = 20 - 1 \cdot 15 = 5$$

$$x_5 = x_3 + q_4 \cdot x_4 = -1 - 1 \cdot 4 = -5$$

$$y_5 = y_3 + q_4 \cdot y_4 = 3 - 1 \cdot (-11) = 14$$

$$r_6 = r_4 + q_5 \cdot r_5 = 15 - 3 \cdot 5 = 0$$

Solution

$$5 = (-5) \cdot 256 + 14 \cdot 95$$

Find the GCD of two polynomials with coefficients in $GF(5)$, $a(x) = x^3 + x^2 + x + 1$ and $b(x) = x^2 + x + 3$

Find the representation of the GCD in the form $l(x)a(x) + j(x)b(x)$ where $l(x)$ and $j(x)$ are polynomial with coefficients in $GF(5)$.

- $r_0(x) = a(x);\ r_1(x) = b(x)$

- $x_0(x) = 1;\ x_1(x) = 0$
- $y_0(x) = 0;\ y_1(x) = 1$
- $r_2(x) = r_0(x) + q_1(x) \cdot r_1(x) = a(x) - q_1(x) \cdot b(x) = (x^3 + x^2 + x + 1) - (x)(x$
- $x_2(x) = x_0(x) + q_1(x) \cdot x_1(x) = 1 - (x) \cdot 0 = 1$
- $y_2(x) = y_0(x) + q_1(x) \cdot y_1(x) = 0 - (x) \cdot 1 = 4x$
- $r_3(x) = r_1(x) + q_2(x) \cdot r_2(x) = (x^2 + x + 3) - (2x)(3x + 1) = 4x + 3$
- $x_3(x) = x_1(x) + q_2(x) \cdot x_2(x) = 0 - (2x)(1) = 3x$
- $y_3(x) = y_1(x) + q_2(x) \cdot y_2(x) = 1 - (2x)(4x) = 1 - 8x^2 \equiv 1 - 3x^2 \equiv 2x^2 + 1$
- $r_4(x) = r_2(x) + q_3(x) \cdot r_3(x) = (3x + 1) - 2(4x + 3) = 0$

Solution

$$4x + 3 = 3x(x^3 + x^2 + x + 1) + (2x^2 + 1)(x^2 + x + 3)$$

A BCH code of length 32 correcting 1 errors is used for transmitting messages. The primitive polynomial $p(x) = x^5 + x^2 + 1$ was used for constructing the code. At the output of the BSC we observe the sequence $y = 01010000011101010100110011111000$. Find the decoded codeword by using the Euclidean algorithm.

- $a(x) = x^{2t} \equiv x^4$
- $b(x) = \alpha^7 x^3 + \alpha^3 x^2 + \alpha^{19} x + \alpha^{25}$

Euclidean algorithm, stop condition, degree of $r_n < t$

- $r_2 = x^4 - (\alpha^7 x^3 + \alpha^3 x^2 + \alpha^{19} x + \alpha^{25})(\alpha^{24} x + \alpha^{20}) = x^2 + \alpha^{12} x + \alpha^{14}$
- $y_2 = 0 - 1 \cdot (\alpha^{24} x + \alpha^{20})$
- $r_3 = (\alpha^7 x^3 + \alpha^3 x^2 + \alpha^{19} x + \alpha^{25}) - (x^2 + \alpha^{12} x + \alpha 14)(\alpha^7 x + \alpha^{12}) = \alpha^{12}$
- $y_3 = 1 + (\alpha^{24} x + \alpha^{20})(\alpha^7 + \alpha^{12}) = \alpha^{18} + \alpha^{12} x + x^2 = 1 + \alpha^{25} x + \alpha^{13} x^2$

$\Lambda(x) = 1 + \alpha^{25} x + \alpha^{13} x^2$

$\Omega(x) = \alpha^{12}$

```
In [9]: !jupyter nbconvert --to html HW4.ipynb
```