

Ejercicio 1. (20%) Considera el código lineal binario C dado por la matriz generadora

$$G = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix} \quad C: K \times n$$

- (a) Escribe todas las palabras del código C.
- (b) ¿Cuáles son los parámetros de C?
- (c) ¿Cuántos errores puede detectar C?, ¿Cuántos borrones puede corregir C?, ¿Cuántos errores puede corregir C?
- (d) Calcula el polinomio de pesos de C.
- (e) Se tiene un canal donde la probabilidad de error de cada bit es 1/11, y se usa el código C para codificar la información. ¿Cuál es la probabilidad de que se reciba una palabra código diferente a la palabra código enviada? (un error no detectable)
- (f) ¿Es C un código auto-ortogonal?

a) $q^k = \text{palabras del código}$

$$2^3 = 8 \text{ palabras}$$

$$\mathbb{F}_2^3 \longrightarrow \mathbb{F}_2^6$$

$$\vec{m} \xrightarrow{\quad} \vec{m} \cdot G$$

$$(m_0, m_1, m_2)$$

$$(0, 0, 0) \longrightarrow (0, 0, 0, 0, 0, 0)$$

$$(0, 0, 1) \longrightarrow (1, 1, 1, 1, 1, 1)$$

$$(0, 1, 0) \longrightarrow (0, 1, 1, 0, 0, 0)$$

$$(0, 1, 1) \longrightarrow (1, 0, 0, 1, 1, 1)$$

$$(1,0,0) \longrightarrow (1,1,0,0,0,0) 2 -$$

$$(1,0,1) \longrightarrow (0,0,1,1,1,1) 4 -$$

$$(1,1,0) \longrightarrow (1,0,1,0,0,0) 2 -$$

$$(1,1,1) \longrightarrow (0,1,0,1,1,1) 4 -$$

b) $[n, k, d]_2$ $\begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$

$[6, 3, 2]_2$ - $d=2$ debido a que
es el menor peso de las
palabras del código.

c)

- Podemos detectar: t errores, $t < d$

Como $d=2$, podemos detectar $t=1$
errores.

- Podemos corregir $\lceil \frac{d}{2} \rceil$ errores, $\lceil \frac{d}{2} \rceil < d$

Dado que $d=2$, $\lceil \frac{d}{2} \rceil = 1$ error.

Podemos corregir 1

- Podemos corregir + errores, $2+cd$

Como $d=2$, no podemos corregir ningún error.

d) Polinomio de pesos

$$W(x) = \sum_{i=0}^n a_i x^i = \sum_{i=0}^6 a_i x^i =$$
$$= [1 + 3x^2 + 3x^4 + x^6]$$

.....

- (e) Se tiene un canal donde la probabilidad de error de cada bit es $1/11$, y se usa el código C para codificar la información. ¿Cuál es la probabilidad de que se reciba una palabra código diferente a la palabra código enviada? (un error no detectable)
- (f) ¿Es C un código auto-ortogonal?

c) $p=1/11$ $\left(1-p\right)^n \left(\text{wt}\left(\frac{p}{1-p}\right)-1\right)$

$$\left(1 - \frac{1}{11}\right)^6 \left(\text{wt}\left(\frac{1111}{1100}\right) - 1\right) =$$

$$\begin{aligned}
 &= \left(\frac{10}{11} \right)^6 \left(\text{wI} \left(\frac{-1}{10} \right) - 1 \right) = \\
 &= \left(\frac{10}{11} \right)^6 \left(1 + 3 \cdot 10^{-2} + 3 \cdot 10^{-4} + 10^{-6} - 1 \right) = \\
 &= 0'5645 \cdot 0'030301 = \boxed{0'017}
 \end{aligned}$$

}) Aquel que esté contenido en su dual ($C^{\perp\perp}$)
 C no es un código auto-ortogonal

Ejercicio 2. (15%) Sea C el código Reed-Solomon sobre \mathbb{F}_5 con longitud 3, dimensión 1 y cuyos puntos de evaluación son $x_1 = 0, x_2 = 1, x_3 = 2$.

- (a) ¿Cuál es la capacidad correctora de C ?
- (b) Sea $r = (0, 0, 1)$ una palabra recibida. Decodifica r usando el algoritmo de decodificación para códigos Reed-Solomon. Es decir, con la notación vista en clase y en el libro, calcula los polinomios Q_0 y Q_1 que dan el polinomio evaluador g .
- (c) Justifica que no se puede usar para C el algoritmo de decodificación en lista para códigos Reed-Solomon de forma que la capacidad correctora aumente.

Nota: Se recalca que la pregunta (b) debe resolverse usando el algoritmo específico para códigos Reed-Solomon. Se considera este sencillo código y esta palabra recibida para que resolver el sistema lineal no tenga dificultad.

a) $[3, 1, 3]_2 \quad d = n - k + 1 = 3$

$$t = \left[\frac{d-1}{2} \right] = \left[\frac{3-1}{2} \right] = \boxed{1}$$

$$b) r = (0, 0, 1); x_1=0, x_2=1, x_3=2$$

$$l_0 = n - 1 - t = 3 - 1 - 1 = 1$$

$$l_n = n - 1 - t - (k - 1) = 3 - 1 - 1 - (1 - 1) = 1$$

$$M = \begin{pmatrix} 1 & x_1 & x_1^2 & \dots & x_1^{l_0} & \cdots & x_1 & x_1x_2 & \dots & x_1x_n \\ 0 & x_2 & x_2^2 & \dots & x_2^{l_0} & \cdots & x_2 & x_2x_3 & \dots & x_2x_n \\ \vdots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & x_n & x_n^2 & \dots & x_n^{l_0} & \cdots & x_n & x_nx_{n-1} & \dots & x_nx_n \end{pmatrix}$$

$$M = \left(\begin{array}{cc|cc} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 2 & 1 & 2 \end{array} \right)$$

- 3 ecuaciones y 4 incógnitas
 → Sistema indeterminado
 $Q_{0,0}, Q_{0,1}, Q_{1,0}, Q_{1,1}$

$$\begin{array}{c} F_1 \\ F_2 \\ F_3 \end{array} \left[\begin{array}{cccc|c} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 2 & 1 & 2 & 0 \end{array} \right] \quad \begin{array}{c} Q_{0,0}=0 \\ Q_{0,1}=0 \end{array}$$

$$\left[\begin{array}{cccc|c} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ -1 & 0 & 1 & 2 & 0 \end{array} \right] \quad F_3 = F_3 - 2F_2$$

$$\left[\begin{array}{cccc|c} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 2 & 0 \end{array} \right] \quad F_3 = F_3 + F_1$$

$$Q_{1,0} + 2Q_{1,1} = 0 \rightarrow Q_{1,0} = -2Q_{1,1}$$

$$\left\{ \begin{array}{l} Q_{0,0} = 0 \\ Q_{0,1} = 0 \\ Q_{1,0} = -2Q_{1,1} \\ Q_{1,1} = \alpha \end{array} \right. \xrightarrow{\alpha=1} \left\{ \begin{array}{l} Q_{0,0} = 0 \\ Q_{0,1} = 0 \\ Q_{1,0} = -2 \\ Q_{1,1} = 1 \end{array} \right.$$

Solución: $(0, 0, -2, 1)$

$$\Rightarrow Q_0(x) = \sum_{j=0}^{l_0} Q_{0,j} x^j = \sum_{j=0}^1 Q_{0,j} x^j = 0 + 0 = 0$$

$$\Rightarrow Q_1(x) = \sum_{j=1}^{l_1} Q_{1,j} x^j = \sum_{j=1}^1 Q_{1,j} x^j = Q_{1,1} x^1 = -2 + x$$

$$g(x) = - \frac{Q_0(x)}{Q_1(x)} = - \frac{0}{-2+x} = \boxed{0}$$

Ejercicio 2. (15%) Sea C el código Reed-Solomon sobre \mathbb{F}_5 con longitud 3, dimensión 1 y cuyos puntos de evaluación son $x_1 = 0, x_2 = 1, x_3 = 2$.

(a) ¿Cuál es la capacidad correctora de C ?

(b) Sea $\mathbf{r} = (0, 0, 1)$ una palabra recibida. Decodifica \mathbf{r} usando el algoritmo de decodificación para códigos Reed-Solomon. Es decir, con la notación vista en clase y en el libro, calcula los polinomios Q_0 y Q_1 que dan el polinomio evaluador g .

(c) Justifica que se puede usar para C el algoritmo de decodificación en lista para códigos Reed-Solomon de forma que la capacidad correctora aumente. ¿Para qué?

códigos Reed-Solomon de forma que la capacidad correctora aumente. ¿Para qué valor mínimo de ℓ (el tamaño máximo de la lista output) la capacidad correctora aumenta?

Nota: Se recalca que la pregunta (b) debe resolverse usando el algoritmo específico para códigos Reed-Solomon. Se considera este sencillo código y esta palabra recibida para que resolver el sistema lineal no tenga dificultad.

$$(*) \quad \frac{k}{n} < \frac{1}{\ell+1} + \frac{1}{n}$$

$$\checkmark \quad \frac{1}{3} < \frac{1}{\ell+1} + \frac{1}{3}$$

$$0 < \frac{1}{\ell+1} \Rightarrow 0 < 1$$

Se verifica siempre,
por tanto siempre es cierto.

(Rte)

$$v, l=3 \\ \tau < n \cdot \frac{\ell}{\ell+1} - \frac{\ell}{2} (k-1)$$

- Para $\ell=2$

$$\tau < 3 \cdot \frac{2}{3} - \frac{2}{2} \cancel{\int^0_{\ell-1}}$$

$$\tau < 2; \text{ No mejora}$$

fz ℓ

- Para $\ell=3$

$$3 \cancel{\int^0_{\ell-1}}$$

$$\tau < 3 \cdot \frac{9}{4} = \frac{27}{4}$$

$\tau < \frac{9}{4} = 2.25$; Por tanto $\tau = 2$ y
la capacidad correctora
aumenta.

- - - - - - - - -

Por ejemplo, si se manda $(0,0,0)$, hay 2
errores y recibes $(1,2,0)$.

La lista output va a ser $(1,1,1)(2,2,2)(0,0,0)$
que son las 3 palabras de C que están a
distancia 2 de la recibida.

3 palabras = valor de $\ell = 3$

Por tanto saber el tamaño de f a 3 no
fendría sentido.

