TGR 2 - Codificación de canal

Software de comunicaciones

Iñaki Amatria Barral
i.amatria@udc.es

Mayo 2020 – Universidade da Coruña

1. Calcula la distancia mínima del siguiente código con $k = 2$ y $n = 5$:

mensaje	código
00	00000
01	10101
10	01010
11	11011

(a) ¿Cuántos errores puede detectar este código? ¿Cuántos puede corregir?

Solución Para saber cuantos errores puede detectar y corregir el código, tenemos que hallar la distancia mínima entre dos palabras código cualesquiera. Como existen pocas combinaciones, en concreto $C_4^2 = 6$, vamos a listarlas todas y calcular la distancia mínima

$d(c_0, c_2) = d(c_2, c_3) = 2$
$d(c_0, c_1) = d(c_1, c_3) = 3$
$d(c_0, c_3) = 4$
$d(c_1, c_2) = 5$

Ahora que conocemos la distancia mínima del código —$d_{min} = 2$— podemos usar las siguientes fórmulas para calcular los parámetros que pide el ejercicio

$$e_d = d_{min} - 1 \qquad e_c = \left\lfloor \frac{d_{min} - 1}{2} \right\rfloor$$

Así, el código puede detectar 1 error y no corrige ninguno.

(b) ¿Puedes encontrar una matriz generadora para el ćodigo?

Solución No, no se puede encontrar una matriz generadora para el código. Rápidamente podemos deducir que no existe tal matriz, ya que la suma de c_1 y c_2 no es una palabra código. Es decir, las palabras código no forman un subespacio vectorial, por tanto no podemos generar una matriz de cambio de base.

2. Considera el código lineal $(9,5)$ definido por

$$[d_1, d_2, d_3, d_4, d_5, d_1+d_2+d_4+d_5, d_1+d_3+d_4+d_5, d_1+d_2+d_3+d_5, d_1+d_2+d_3+d_4]$$

donde d_1, d_2, d_3, d_4 y d_5 son los bits correspondientes al mensaje.

(a) Determina la matriz generadora del código.

Solución De forma natural obtenemos

$$G_{5\times 9} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \end{bmatrix}$$

(b) Calcula la palabra código correspondiente al mensaje 01010.

Solución Para calcular la palabra código c correspondiente a la palabra fuente $s = [01010]$, solo tenemos que calcular

$$c_{1\times 9} = s_{1\times 5} \times G_{5\times 9}$$

De esta forma, la palabra código c correspondiente al mensaje s es 010100110.

(c) Halla la distancia mínima.

Solución Para hallar la distancia mínima, tenemos que calcular el menor peso de todas las palabras código distintas de cero. Es decir, tenemos que hallar

$$min\{w(c); c \in C, c \neq 0\}$$

Para calcular el mínimo peso de todas las posibles palabras código distintas de cero, tenemos que calcular el peso de todas las combinaciones —$C_5^1 + C_5^2 + C_5^3 + C_5^4 + C_5^5 = 31$— lineales posibles para una matriz con 5 filas.

$$\begin{aligned}
&w(f_1+f_2) = w(f_1+f_3) = w(f_1+f_4) = w(f_1+f_5) = 3 \\
&w(f_2) = w(f_3) = w(f_4) = w(f_5) = w(f_2+f_3) = w(f_2+f_4) = \ldots = 4 \\
&w(f_1) = w(f_1+f_2+f_3) = w(f_1+f_2+f_4) = w(f_1+f_2+f_5) = \ldots = 5 \\
&w(f_1+f_2+f_3+f_4) = w(f_1+f_2+f_3+f_5) = w(f_1+f_2+f_4+f_5) = \ldots = 7 \\
&w(f_1+f_2+f_3+f_4+f_5) = 8
\end{aligned}$$

Podemos ver que el menor peso es 3, por lo que podemos concluir que la distancia mínima del código es 3.

(d) ¿Puedes diseñar un código $(8,5)$ con la misma distancia mínima?

Solución No, con un bit menos de paridad no podemos diseñar un código con las mismas prestaciones. Esto se debe a que con solo 3 bits de redundancia, existirá, como mínimo, una fila en la matriz P que es combinación lineal de otras dos filas de esa matriz. Por tanto, existirá una palabra con peso 2 y el código tendrá distancia mínima 2.

(e) ¿Puedes diseñar un código $(10,5)$ con mayor distancia mínima?

Solución Sí, con un bit más de redundancia podemos diseñar el código con matriz generadora

$$G_{5\times 10} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 \end{bmatrix}$$

con distancia mínima 4.

3. Un código bloque lineal sistemático $(8,4)$ tiene la siguiente matriz de paridad:

$$P = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix}$$

(a) Halla la matriz generadora y la matriz control de paridad del código.

Solución Como estamos ante un código bloque lineal sistemático, podemos calcular las matrices generadoras y de paridad como

$$G_{k\times n} = \begin{bmatrix} I_{k\times k} & | & P_{k\times l} \end{bmatrix} \qquad H_{l\times n} = \begin{bmatrix} P^T_{l\times k} & | & I_{l\times l} \end{bmatrix}$$

entonces, en nuestro caso, G y H quedan definidas como

$$G_{4\times 8} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{bmatrix} \qquad H_{4\times 8} = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Además, podemos comprobar rápidamente que las dos matrices son correctas, ya que

$$G_{4\times 8} \times H^T_{8\times 4} = 0_{4\times 4}$$

(b) Descodifica la palabra $r = [11111000]$ utilizando la teoría de síndromes.

Solución Antes de descodificar r, tenemos que comprobar si es una palabra código. Para ello, tenemos que calcular

$$r_{1\times 8} \times H^T_{8\times 4}$$

Si el resultado es $0_{1\times 4}$, podemos concluir que la palabra recibida es una palabra código y proceder con la descodificación. En otro caso, antes de descodificar, tenemos que usar el resultado de la multiplicación para calcular que máscara de error produce ese resultado y sumárselo a la palabra recibida para corregirla.

En nuestro caso, el resultado de la multiplicación es 0111. Es decir, la palabra recibida no es una palabra código. Entonces, tenemos que encontrar una máscara de error que al multiplicar por H^T nos de 0111 para poder corregir el error que se ha producido.

Rápidamente podemos ver que la máscara 00010000 tiene como resultado 0111. Por tanto, podemos corregir el error cambiando el cuarto bit de la palabra recibida. De esta forma, la palabra corregida es 11101000, que corresponde con la palabra fuente 1110.

Es importante recordar que no tenemos certeza de que la palabra corregida corresponda con la palabra enviada en primer lugar. Este código tiene distancia mínima 4, es decir, puede corregir 1 error. Si se producen errores como 01001100 o 00101001, que también tienen síndrome 0111, se van a corregir, de nuevo, cambiando el cuarto bit de la palabra recibida, pero la palabra descodificada no será, en ningún caso, la palabra que pretendía transmitir el emisor.

4. Diseña un código bloque lineal con tasa $k/n > 1/3$ y con el tamaño de palabra código más pequeño posible que sea capaz de corregir 1 error o detectar 3 errores.

   Solución Para diseñar un código bloque lineal que sea capaz de corregir 1 error o detectar 3 errores, tenemos que ser capaces de encontrar un código con distancia mínima

$$e_d = d_{min} - 1 \implies d_{min} = e_d + 1 = 4$$

   Ahora, vamos a listar todos los posibles valores para k y n que satisfacen $k/n > 1/3$ y nos quedaremos con la primera combinación que cumpla que $d_{min} = 4$

k	n	k/n	d_{min}
2	3	2/3	1
2	4	1/2	2
2	5	2/5	3
3	4	3/4	1
3	5	3/5	2
3	6	1/2	3
3	7	3/7	4

   Para $k = 3$ y $n = 7$, el código bloque lineal queda definido por la matriz generadora

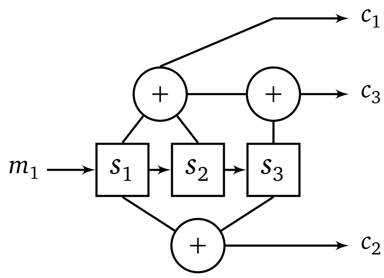
$$G_{3\times 6} = \begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 & 1 \end{bmatrix}$$

5. Un código convolucional $k = 1$ tiene la siguiente matriz generadora:

$$G = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}$$

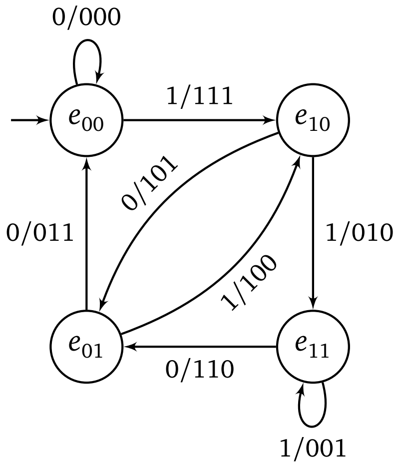
   (a) Dibuja el codificador.

   Solución



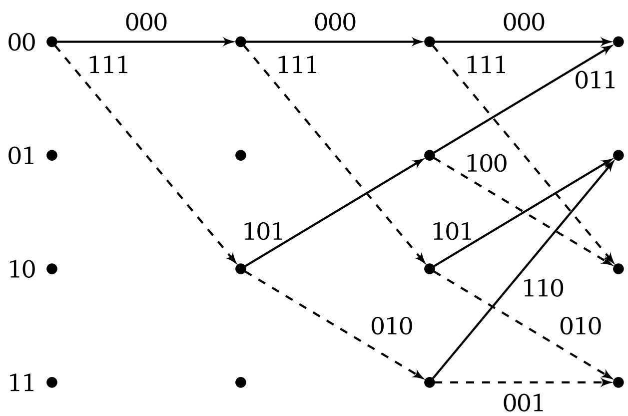
(b) Dibuja el diagrama de estados.

Solución



(c) Dibuja el diagrama trellis.

Solución



(d) Halla la distancia libre.

Solución Para hallar la distancia libre, tenemos que calcular el menor peso de las palabras código distintas de cero que se forman sin repetir ningún estado. Usando el diagrama de estados del apartado b, las dos secuencias de estados que cumplen esta condición son: (1) $\{e_{00}, e_{10}, e_{01}, e_{00}\}$ y (2) $\{e_{00}, e_{10}, e_{11}, e_{01}, e_{00}\}$.

Respectivamente, estas secuencias forman las palabras 111 101 011 y 111 010 110 011 que tienen peso 7 y 8. Por tanto, la distancia libre del código es 7, lo que significa que el ćodigo puede corregir $e_c = \lfloor (7-1)/2 \rfloor = 3$ errores cada 7 bits.

(e) Descodifica la secuencia recibida 010 111 110 010 011. ¿En qué posiciones hubo errores durante la transmisión?

Solución De acuerdo con el diagrama trellis, y siguiendo el camino

$$00 \to 00 \to 10 \to 11 \to 01 \to 00$$

que tiene distancia $1+0+1+1+0=3$, descodificamos la palabra 01100. Además, podemos comprobar que la secuencia recibida contenía 3 errores en las posiciones 2, 7 y 10. Ya que la secuencia código corregida es 000 111 010 110 011.