

## SIMULACIÓN DE HAMMING EN UN CANAL SIN RUIDO

El código binario de Hamming de redundancia 3 es un código lineal y, por tanto, destinado a la corrección de los errores o borrados que se puedan producir en la transmisión por un canal con ruido. El objetivo de esta práctica es hacer una simulación de un proceso de transmisión de la información por un canal binario  $\Omega$ , cuya herramienta principal va a ser el código de Hamming indicado. De momento sólo vamos a establecer los procesos de codificación y realizar su decodificación, dejando la descripción del método de corrección para la práctica siguiente. Supongamos que vamos a transmitir un mensaje  $\underline{m}$  escrito en el alfabeto

$$\mathcal{A} = \text{“ABCDEFGHIJKLMNÑOPQRSTUVWXYZ_{\square}abcdefghijklmnopqrstuvwxyz”}$$

por un canal binario. El alfabeto es el dato de partida de la simulación, y puede ser modificado. Para codificar un mensaje  $\underline{m}$ , realizamos los siguientes procesos:

**COD 1.** El primer proceso es una codificación binaria de los símbolos del alfabeto  $\mathcal{A}$ , usando un código en bloque y de longitud la mínima posible (codificación de la fuente). Es decir, como el alfabeto tiene 56 símbolos, dicha longitud mínima es

$$\text{long\_min} = \lceil \log_2(56) \rceil = 6$$

Para codificar cada símbolo del alfabeto  $\mathcal{A}$ , si  $pos$  denota la posición que ocupa dicho símbolo dentro del alfabeto, calculamos la expresión en base dos del número entero  $pos - 1$  y completamos con ceros a la izquierda hasta conseguir longitud 6 (expresión en base 2 de longitud 6 para el entero  $pos - 1$ ). Por ejemplo, si queremos codificar el mensaje “Las”,

símbolo	$pos - 1$	codificación
L	11	[0, 0, 1, 0, 1, 1]
a	28	[0, 1, 1, 1, 0, 0]
s	48	[1, 1, 0, 0, 0, 0]

se obtiene

$$\mathbf{COD\ 1} = [0, 0, 1, 0, 1, 1, 0, 1, 1, 1, 0, 0, 1, 1, 0, 0, 0]$$

**COD 2.** El segundo proceso de codificación tiene como fin la corrección del ruido. Es una codificación que utiliza el código de Hamming binario de redundancia 3. El proceso consiste en multiplicar cada bloque binario de longitud 4 por la matriz generadora

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

(haciendo el producto de matrices en  $\mathbb{Z}_2$ ). En nuestro ejemplo, los bloques que tenemos que codificar son:

$$[0, 0, 1, 0] \quad [1, 1, 0, 1] \quad [1, 1, 0, 0] \quad [1, 1, 0, 0] \quad [0, 0]$$

Para ajustar el último bloque a longitud 4, se rellena con símbolos binarios cualesquiera. Por ejemplo

$$[0, 0, 1, 0] \quad [1, 1, 0, 1] \quad [1, 1, 0, 0] \quad [1, 1, 0, 0] \quad [0, 0, \mathbf{0}, \mathbf{1}]$$

Una vez obtenida la longitud adecuada, multiplicamos cada bloque por la matriz generadora

$$\text{bloque} \cdot \left( \begin{array}{cccc|ccc} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{array} \right) = \text{palabra código}$$

bloque	[0, 0, 1, 0]	[1, 1, 0, 1]	[1, 1, 0, 0]	[1, 1, 0, 0]	[0, 0, 0, 1]
código	[0, 0, 1, 0, 1, 1, 0]	[1, 1, 0, 1, 0, 0, 1]	[1, 1, 0, 0, 1, 1, 0]	[1, 1, 0, 0, 1, 1, 0]	[0, 0, 0, 1, 1, 1, 1]

Por último, el mensaje codificado que se envía por el canal es la concatenación de las palabras obtenidas después de multiplicar por  $G$  (todas son palabras del código de Hamming binario de redundancia 3,  $\mathcal{H}_2(3)$ )

$$[0, 0, 1, 0, 1, 1, 0] \quad [1, 1, 0, 1, 0, 0, 1] \quad [1, 1, 0, 0, 1, 1, 0] \quad [1, 1, 0, 0, 1, 1, 0] \quad [0, 0, 0, 1, 1, 1, 1]$$

es decir, es la secuencia

$$\mathbf{COD\ 2} = 00101101101001110011011001100001111$$

### PROBLEMA 1 (*hay que describir los dos procesos de decodificación*)

Supongamos ahora que recibimos del canal la siguiente secuencia binaria

$$00000001100110010010101001011101001$$

Obtener el mensaje original.

### PROBLEMA 2 (*requiere el uso de una implementación de los procesos de decodificación*)

Supongamos ahora que a un mensaje  $\underline{m}$  escrito en el alfabeto

“ABCDEFGHIJKLMNOPQRSTUVWXYZ<sub>□</sub>,.-;()0123456789abcdeéfgghiíjklmnñoópqrstuvwxyz”

se le aplican los procesos de codificación anteriores y recibimos del canal la secuencia binaria dada en el fichero *datos.txt*. Sabiendo que durante la transmisión por el canal **no se ha producido ruido**, se pide obtener el mensaje original.

Nota: Cuando en el mensaje original aparezcan dos espacios consecutivos lo imprimimos como un cambio de línea.