Tema 4

Teoría de códigos

4.1. Introducción

Los inicios de la teoría moderna de la comunicación, en la que se incluye la teoría de códigos, se sitúan al final de los años veinte con los trabajos de Ralph Hartley. En 1941, Claude E. Shannon, considerado el padre de la teoría de la información, comienza sus investigaciones en temas de comunicación, sus resultados se publicaron en el trabajo "A Mathematical Theory of Communication" (1948) que es la base de la moderna teoría matemática de la comunicación. Hacia 1950, los trabajos de Richard Hamming y Marcel Golay dieron un mayor impulso a la teoría de códigos. En ellos se construyen, de forma económica y elegante, códigos capaces de corregir un número especificado de errores producidos durante la transmisión. En determinados casos, sus métodos son óptimos, en el sentido de que para transmitir un determinado número de símbolos con una capacidad de corrección marcada, el número de símbolos añadidos es mínimo.

Una comunicación de datos consiste en la transmisión de una secuencia de caracteres de algún alfabeto finito A (normalmente $A = \{0, 1\}$) desde una localización física (fuente) a otra (receptor) a través de un canal de comunicación. En la mayoría de los casos, imperfecciones del canal, denominadas ruido, provocan que algunos caracteres transmitidos sean incorrectamente recibidos por el receptor. Por ello se introducen, de modo sistemático, redundancias en la información, las cuales permiten detectar, e incluso corregir, los errores cuando el mensaje recibido es descodificado.

Ejemplo 4.1.1. Supongamos que un centro de control marítimo debe transmitir a los barcos la ruta que deben seguir para llegar a puerto utilizando un canal binario cuya probabilidad de transmitir incorrectamente un bit es del uno por mil, uno de cada mil bits se recibirá mal. El alfabeto fuente consta

de 4 símbolos {Norte, Sur, Este, Oeste}, cada uno de ellos significa navegar una milla náutica en esa dirección.

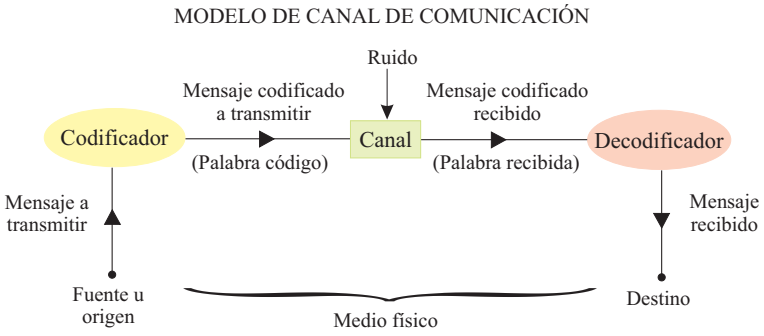
Si usamos como código el conjunto de pares binarios $C = \{00, 01, 10, 11\}$ y se codifica Norte por 00, Oeste por 01, Este por 10 y Sur por 11, la distancia mínima del código es 1 (no detecta errores). Si la ruta que se desea enviar es NNOO quedaría codificada como 00 00 01 01, para facilitar su lectura se insertan espacios entre las palabras. Supongamos que por efecto de las interferencias es recibida como 00 01 01 11, lo que sería interpretado como la ruta NOOS. La probabilidad de que un símbolo fuente sea interpretado correctamente es la misma que la probabilidad de que su palabra correspondiente sea recibida correctamente; para este caso, es del $99,8001$ por ciento; aproximadamente dos de cada mil palabras serán erróneas, lo cual no parece mucho. Sin embargo, si pensamos que el mensaje completo consta de cien símbolos fuente, la probabilidad de que éste sea recibido correctamente en su totalidad es del $81,865$ por ciento.

Si se utiliza el código $C = \{000, 011, 101, 110\}$, codificando Norte por 000, Oeste por 011, Este por 101 y Sur por 110, la distancia mínima es 2 (detecta errores simples), la ruta quedaría codificada como 000 000 011 011. Si se recibe la transmisión 000 010 011 111 se detecta que la segunda y la cuarta palabra son incorrectas, pidiéndose una retransmisión del mensaje. La probabilidad de que una palabra se reciba correctamente es de $99,7003$ por ciento y la de una recepción correcta de un mensaje de cien símbolos fuente desciende hasta un $74,071$ por ciento; pero en un $25,907$ por ciento de los casos se detectará que el mensaje es incorrecto; pidiendo su repetición. En consecuencia, la probabilidad de interpretar mal la recepción de un mensaje baja del $18,135$ por ciento a menos de $0,025$ por ciento.

En el caso de que no fuese posible pedir la retransmisión de la ruta, sería preferible la utilización de un código corrector de errores simples; por ejemplo, $C = \{00000, 01101, 10110, 11011\}$, codificando Norte por 00000, Oeste por 01101, Este por 10110 y Sur por 11011, la ruta quedaría codificada como 00000 00000 01101 01101. Al recibir la transmisión 00000 01000 01101 11101, de nuevo las palabras segunda y cuarta se detectan como incorrectas, pero debido a las capacidades del código son substituidas por 00000 y 01101 respectivamente, obteniendo la ruta correcta. Un símbolo fuente se interpretará correctamente siempre que la palabra correspondiente se reciba correctamente o con un error simple, lo que ocurre en un $99,999$ por ciento de los casos. A costa de incrementar el número de bits transmitidos, la probabilidad de que un mensaje de cien símbolos fuente sea correctamente interpretado aumenta hasta el $99,9$ por ciento.

Un sistema general de comunicación consta de cinco partes. Una fuente

o emisor, un codificador, un canal de comunicación, un descodificador y un destino o receptor.



El canal de comunicación es capaz de admitir en cada instante de tiempo un elemento de un conjunto finito de símbolos $A = \{a_1, a_2, \ldots, a_q\}$, denominado alfabeto del canal o del código; cada uno de los elementos a_i se denomina símbolo del canal o del código. En general, si A es un conjunto finito o alfabeto, una cadena de n símbolos de A se denominan palabra de longitud n; el conjunto de palabras de longitud n formadas por símbolos de A se denota por A^n y el conjunto de palabras de longitud finita formadas por símbolos de A se denota por A^*. Normalmente el alfabeto del canal se representa por un conjunto finito de números naturales, siendo el más utilizado y estudiado el alfabeto binario, $A = \{0, 1\}$. Un canal cuyo alfabeto es el alfabeto binario se denomina canal binario.

El emisor compone los mensajes que se desean enviar a partir de un conjunto finito de símbolos, $S = \{s_1, s_2, \ldots, s_M\}$, denominado alfabeto fuente; cada uno de los elementos s_i se denomina símbolo fuente. De esta forma los mensajes que se desean enviar se consideran palabras de S^*. La tarea del codificador es transformar o *codificar* el mensaje a símbolos del canal. El descodificador realiza la tarea inversa del codificador, transformando los símbolos del canal a símbolos fuente, intentando reconstruir el mensaje original para el receptor.

Definición 4.1.2. Un **código** (**bloque**) C para un alfabeto fuente S y un alfabeto del canal A es una aplicación inyectiva, $C : S \to A^n$. La imagen de la aplicación C se denomina conjunto de palabras código, y sus elementos son las palabras código.

El valor n es la longitud del código, y el número de palabras código, que coinciden con el número de símbolos fuente, es el tamaño del código. Un código de longitud n y tamaño M se denomina un $[n, M]$-código. Un código sobre el alfabeto $A = \{0, 1\}$ se llama código binario, si el alfabeto es

$A = \{0, 1, 2\}$ se llama código ternario. También suele denominarse código al conjunto de palabras código, denotándolo igualmente por C.

Ejemplo 4.1.3.

1. El alfabeto fuente está formado por las cadenas binarias de longitud siete. A cada símbolo fuente le hacemos corresponder una palabra binaria de longitud ocho donde los siete primeros dígitos son los mismos que los del símbolo fuente y el último es un cero o un uno de forma que el número total de dígitos uno en la palabra sea par. Este código se denomina **código control de paridad**.

2. El alfabeto fuente es $\{0, 1\}$ y a cada símbolo fuente le asociamos la terna que consiste en repetir dicho símbolo ($0 \to 000; 1 \to 111$). Este código se denomina **código de triple repetición**.

3. **Código de triple paridad**. Los símbolos fuente son ternas de ceros y unos, abc, y se codifican en cadenas binarias de longitud seis, $abcxyz$, en la forma siguiente: el número de unos en abx es par, el número de unos en acy es par y el número de unos en bcz es par. Por ejemplo, el símbolo fuente 110 se codificaría como 110011. El conjunto de palabras código es $C = \{000000, 100110, 010101, 001011, 110011, 101101, 011110, 111000\}$.

4.1.1. Distancia Hamming

Es de gran utilidad dar una formalización a la idea del número de posiciones en las que difieren dos palabras mediante la noción de distancia entre ellas.

Definición 4.1.4. Sean x e y dos palabras de A^n. La **distancia Hamming** entre x e y, $d(x, y)$, es el número de componentes en las cuales son diferentes. Analíticamente, si $x = (x_1, x_2, \ldots, x_n)$ e $y = (y_1, y_2, \ldots, y_n)$ su distancia es

$$d(x, y) = |\{j \ ; \ 1 \leq j \leq n, \ x_j \neq y_j\}|$$

donde el símbolo $|X|$ denota el cardinal de un conjunto X (número de elementos que contiene).

Si denominamos un "error" simple el cambio de un símbolo por otro distinto en la transmisión de una palabra, la distancia Hamming nos dice el número de cambios necesarios para convertir la palabra código enviada en la palabra efectivamente recibida. Así, sea c una palabra código de longitud n transmitida a través del canal y sea r la palabra recibida; si $d(c, r) = \lambda$ diremos que se ha producido un error de tipo λ o que se han producido λ errores en la transmisión.

Ejemplo 4.1.5. Supongamos que se transmite la palabra $c = 0001$ y se recibe la palabra $r = 0011$, la distancia entre ambas palabras es $d(0001, 0011) = 1$, se ha producido un error de tipo 1 o un error simple. Si se transmite la palabra $c = 100110$ y se recibe la palabra $r = 110100$, se ha producido un error de tipo 2 o un error doble ya que la distancia entre ambas palabras es $d(100110, 110100) = 2$.

El término distancia en el nombre de distancia Hamming es apropiado ya que verifica los axiomas que se piden a una función para ser considerada una distancia o métrica definida en el conjunto A^n. Los axiomas son los siguientes:

Definición 4.1.6. (Axiomas de distancia) Una función $f(x, y)$ definida sobre pares de elementos de un conjunto arbitrario X es una función distancia si verifica las siguientes condiciones:

1. $f(x, y)$ es un número real no negativo.
2. $f(x, y) = 0$ si, y sólo si, $x = y$.
3. Para cualesquiera elementos x e y de X, $f(x, y) = f(y, x)$.
4. Para cualesquiera elementos x, y, z de X, $f(x, z) \leq f(x, y) + f(y, z)$.

La última condición se denomina *desigualdad triangular*. Si pensamos en x, y, z como los vértices de un triángulo, nos indica que la longitud de un lado es siempre menor o igual que la suma de las longitudes de los otros dos.

Proposición 4.1.7. *La distancia Hamming es una función distancia sobre el conjunto A^n siendo A el alfabeto del canal.*

4.1.2. Detección y corrección de errores

El criterio para determinar si una palabra recibida es correcta o no, es simple: si la palabra recibida pertenece al conjunto de palabras código se considera que es la palabra enviada y, en caso contrario, que se ha producido algún error durante la transmisión. Así pues, si el error producido en la transmisión transforma una palabra código en otra palabra código, el descodificador supondrá que la palabra recibida es correcta y el error pasará desapercibido.

Si el objetivo es corregir los posibles errores producidos en la transmisión, se debe establecer un criterio para sustituir las palabras incorrectas recibidas por palabras código, este proceso se denomina **descodificación**. El criterio utilizado en el proceso de descodificación se denomina **descodificación por**

distancia mínima. Consiste en sustituir la palabra recibida r por la palabra código c', siendo ésta la palabra código cuya distancia Hamming a la palabra r sea lo más pequeña posible. Si para alguna palabra r hubiese dos o más palabras código con la misma distancia, se tienen dos posibles alternativas: asignar a la palabra r una de las posibles palabras código de forma arbitraria (**descodificación completa**) o no asignarle ninguna palabra código y notificar que se ha producido un error no corregible (**descodificación incompleta**). La descodificación por distancia mínima hace máxima la probabilidad de una correcta descodificación en la mayoría de modelos de comunicación digital.

Ejemplo 4.1.8. Utilizando el código de triple repetición se recibe la palabra $r = 101$ que no es correcta, las distancias a las dos palabras código posibles son $d(000, 101) = 2$ y $d(111, 101) = 1$; por tanto r se descodifica como la palabra código 111.

Además de la longitud y el tamaño de un código bloque C hay un tercer parámetro de gran importancia, la distancia mínima, que nos permite conocer el número de errores que el código puede detectar o corregir al aplicar los procedimientos descritos anteriormente.

Definición 4.1.9. La **distancia mínima** de un código C, denotada por $d(C)$, es la menor de las distancias entre dos palabras código distintas cualesquiera. Esto es, $d(C) = min\{d(c, c');\ c, c' \in C,\ c \neq c'\}$.

La distancia mínima de un código, con al menos dos palabras código, es siempre un valor mayor o igual que uno, ya que la distancia Hamming entre dos palabras código distintas es siempre mayor o igual que uno.

Ejemplo 4.1.10. El código control de paridad tiene distancia mínima 2. El código de triple repetición tiene distancia mínima 3 al igual que el código de triple paridad.

Definición 4.1.11. Sea λ un número natural. Un código bloque C **detecta λ errores** o es un código **λ-detector** si verifica que para cada palabra código c y cada palabra r obtenida a partir de c cambiando entre 1 y λ símbolos, la palabra r no es una palabra código.

Esta propiedad es importante en los códigos utilizados en aquellas comunicaciones en las que se puede solicitar una retransmisión cuando se detecte error en la transmisión. Ya que, si utilizamos un código detector de λ errores y al transmitir una palabra código se produce un error de tipo λ o menos, tenemos la seguridad de que la palabra recibida no será una palabra código y el descodificador la detectará como errónea.

Proposición 4.1.12. *Un código bloque C detecta λ errores si, y sólo si, su distancia mínima es mayor que λ.*

Demostración. Si la distancia mínima del código es mayor que λ entonces detecta λ errores. En efecto, si se envía una palabra código c y se recibe una palabra r tal que $1 \leq d(c,r) \leq \lambda$, la palabra r no puede ser una palabra código pues de lo contrario la distancia mínima $d(C)$ será menor o igual que $d(c,r) \leq \lambda$.

Recíprocamente, si la distancia mínima del código C es menor o igual que λ entonces el código no detecta λ errores. Sean c, c' dos palabras código cuya distancia Hamming sea la distancia mínima de C. Entonces, $d(c,c') \leq \lambda$ y la palabra código c' se puede obtener modificando a lo sumo λ símbolos de la palabra código c y el código C no detecta λ errores. □

Ejemplo 4.1.13. El código control de paridad detecta errores simples. El código de triple repetición y el código de triple paridad detectan errores dobles.

El concepto similar a la detección de errores para la corrección es algo más sutil.

Definición 4.1.14. Sea λ un número natural. Se dice que un código bloque C **corrige λ errores** o que es **λ-corrector** si verifica que toda palabra recibida con a lo sumo λ errores es descodificada como la palabra código transmitida. Utilizando la descodificación por distancia mínima significa que para cada palabra código c y cada palabra r obtenida modificando a lo sumo λ símbolos de c, la distancia Hamming entre las palabras c y r es menor estrictamente que la distancia entre r y cualquier otra palabra código distinta de la palabra c.

Nota: Siempre se debe tener en cuenta que si se aplica la descodificación por distancia mínima completa, cada vez que la palabra recibida no sea una palabra código se descodificará como una palabra código. Sin embargo, el receptor no tiene la seguridad de si esa palabra código es la realmente transmitida. El descodificador sólo sabe que, utilizando un código corrector de λ errores, si en la transmisión se han producido a lo sumo λ errores la descodificación por distancia mínima proporciona la palabra código enviada.

Proposición 4.1.15. *Un código bloque C corrige λ errores si, y sólo si, su distancia mínima es mayor que 2λ.*

Demostración. Si la distancia mínima de C es mayor que 2λ, sea c una palabra código y r una palabra obtenida modificando a lo sumo λ símbolos

de la palabra c, así $d(r,c) \leq \lambda$. Para toda palabra código c' distinta de c, $d(c',c) \leq d(c',r) + d(r,c)$; despejando y teniendo en cuenta que $d(c',c) \geq d(C) > 2\lambda$, $d(c',r) = d(c',c) - d(r,c) > 2\lambda - \lambda = \lambda$. Por lo tanto, toda palabra código c' distinta de c verifica que $d(r,c') > \lambda$ y la descodificación por distancia mínima asociará correctamente a la palabra r la palabra código c.

Recíprocamente, si la distancia mínima del código es d menor o igual que 2λ, existirán dos palabras código c y c' cuya distancia Hamming es $d(c,c') = d \leq 2\lambda$. Podemos suponer que los d símbolos distintos son los d primeros, así pues $c = (c_1, c_2, \ldots, c_d, c_{d+1}, \ldots, c_n)$ y $c' = (c'_1, c'_2, \ldots, c'_d, c_{d+1}, \ldots, c_n)$ sea r la palabra de longitud n siguiente

$$r = (c'_1, c'_2, \ldots, c'_\lambda, c_{\lambda+1}, c_{\lambda+2}, \ldots, c_d, c_{d+1}, \ldots, c_n)$$

es decir, los primeros λ símbolos iguales a los símbolos de la palabra c' y los restantes como los símbolos de la palabra c, que a partir del $(d+1)$-ésimo símbolo coinciden con los de la palabra c'. De este modo $d(c,r) = \lambda$ y $d(r,c') = d - \lambda \leq \lambda$ (incluso menor estricto). En esta situación, al transmitir la palabra código c se pueden producir λ errores y recibir la palabra r; el proceso de descodificación por distancia mínima no podría (en el mejor de los casos) asignar unívocamente la palabra código c a la palabra recibida r, incluso podría asignarle incorrectamente la palabra código c', lo que contradice la hipótesis sobre la capacidad correctora del código C. □

Ejemplo 4.1.16. El código de triple repetición y código de triple paridad corrigen errores simples. Sea C el código de triple paridad, si se recibe la palabra $r = 111011$, al no ser una palabra código es detectada como errónea. El proceso de descodificación sería el siguiente:
Se calcula la distancia de la palabra r a cada una de las palabras código:

$d(111011, 000000) = 5$, $d(111011, 100110) = 4$,
$d(111011, 010101) = 4$, $d(111011, 001011) = 2$,
$d(111011, 110011) = 1$, $d(111011, 101101) = 3$,
$d(111011, 011110) = 3$, $d(111011, 111000) = 2$.

La palabra código enviada es 001011 y el símbolo fuente es 001.

Por la importancia que tiene el concepto de distancia mínima de un código bloque es usual referirse a un código bloque de longitud n, tamaño M y distancia mínima d como un $[n, M, d]$-código. Los valores longitud n, tamaño M y distancia d se denominan parámetros del código.

4.2. Códigos lineales

Este epígrafe está dedicado al estudio de un tipo de códigos bloque, los códigos lineales. Los códigos lineales son más fáciles de implementar y de analizar que los códigos no lineales debido a la estructura de espacio vectorial que tiene el conjunto de palabras código. Para ello es necesario que el alfabeto del canal sea el conjunto de los enteros módulo un número primo p, $\mathbb{Z}_p = \{0, 1, 2, \ldots, p-1\}$ con las operaciones de suma y producto módulo p. La mayoría de los códigos más interesantes son binarios y ternarios; para ellos se utiliza como alfabeto los conjuntos $\mathbb{Z}_2$ y $\mathbb{Z}_3$, respectivamente.

En los códigos lineales, el conjunto de símbolos fuente también debe tener una estructura de espacio vectorial; para ello, se representa dicho conjunto por el espacio vectorial $(\mathbb{Z}_p)^k$, siendo k un natural lo suficientemente grande para que p^k, el número de vectores de $(\mathbb{Z}_p)^k$, sea mayor o igual que el número de símbolos fuente.

Definición 4.2.1. Un (n, k) **código lineal** sobre un alfabeto del canal $\mathbb{Z}_p$, con $n \geq k$, es una aplicación lineal inyectiva, $C : (\mathbb{Z}_p)^k \to (\mathbb{Z}_p)^n$. La imagen de la aplicación C es un subespacio vectorial de dimensión k de $(\mathbb{Z}_p)^n$, denominado subespacio código. Sus elementos se denominan palabras o vectores código.

Como ocurre en los códigos no lineales, en algunas circunstancias, se denomina también código lineal al subespacio código, denotándolo asimismo por C.

Nótese que el tamaño de un (n, k) código lineal sobre $\mathbb{Z}_p$ es p^k. Por tanto, un (n, k) código lineal sobre $\mathbb{Z}_p$ es un $[n, p^k]$-código bloque. El recíproco no siempre es cierto.

Ejemplo 4.2.2. Los códigos del ejemplo 4.1.3 son códigos lineales binarios. El código control de paridad es $C : (\mathbb{Z}_2)^7 \to (\mathbb{Z}_2)^8$, $C(a_1, a_2, a_3, \ldots, a_7) = (a_1, a_2, a_3, \ldots, a_7, a_8)$, siendo $a_8 = a_1 + a_2 + a_3 + \cdots + a_7$. El código de triple repetición es $C : \mathbb{Z}_2 \to (\mathbb{Z}_2)^3$, $C(a) = (a, a, a)$. El código de triple paridad es $C : (\mathbb{Z}_2)^3 \to (\mathbb{Z}_2)^6$, $C(a, b, c) = (a, b, c, a + b, a + c, b + c)$.

Los parámetros de un código lineal son: la longitud n, la dimensión k y la distancia mínima d. Al hacer referencia a los tres parámetros hablaremos de un (n, k, d) código lineal.

Definición 4.2.3. El **peso** de un vector v de $(\mathbb{Z}_p)^n$, $w(v)$, es el número de entradas no nulas de v, es decir, $w(v) = |\{i;\ v_i \neq 0,\ i = 1, 2, \ldots, n\}|$. El **peso mínimo** de un código lineal C, denotado por $w(C)$, es el menor de los pesos de las palabras código distintas de la palabra cero, es decir, $w(C) = min\{w(v);\ v \in C, v \neq 0\}$.

Una de las propiedades más útiles de los códigos lineales se demuestra en el siguiente resultado.

Teorema 4.2.4. *En un código lineal, la distancia mínima y el peso mínimo coinciden.*

Demostración. Dados dos vectores cualesquiera v, v', el vector $v - v'$ tiene tantas componentes no nulas como componentes diferentes tengan los vectores v y v' entre sí. Por tanto, $d(v, v') = w(v - v')$.

Sean c_1 y c_2 dos vectores código tales que $d(c_1, c_2) = d(C)$. Entonces $d(C) = d(c_1, c_2) = w(c_1 - c_2) \geq w(C)$, pues $c_1 - c_2$ es un vector código no nulo. Recíprocamente, sea c un vector código tal que $w(c) = w(C)$, entonces $w(C) = w(c) = d(c, 0) \geq d(C)$, pues el vector 0 es un vector código distinto del vector c. □

Gracias a este resultado, para calcular la distancia mínima de un código lineal de tamaño M, en lugar de realizar $(M^2 - M)/2$ comparaciones entre palabras código distintas, basta calcular los $M - 1$ pesos de las palabras código no nulas.

4.2.1. Codificación en códigos lineales

Definición 4.2.5. La **matriz generadora** de un (n, k) código lineal C sobre $\mathbb{Z}_p$, G, es la matriz asociada a la aplicación lineal C respecto a las bases canónicas de ambos espacios. Es una matriz de n filas y k columnas de elementos de $\mathbb{Z}_p$, cuyas columnas son las palabras código correspondientes a los k vectores de la base canónica de $(\mathbb{Z}_p)^k$.

$$G^t = \begin{pmatrix} C(1,0,0,\ldots,0,0) \\ C(0,1,0,\ldots,0,0) \\ C(0,0,1,\ldots,0,0) \\ \vdots \\ C(0,0,0,\ldots,1,0) \\ C(0,0,0,\ldots,0,1) \end{pmatrix}$$

Ejemplo 4.2.6. La matriz generadora del código control de paridad es

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

La matriz generadora del código de triple repetición es

$$G = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$$

La matriz generadora del código de triple paridad es

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

Ejemplo 4.2.7. Considera el código lineal binario $C : (\mathbb{Z}_2)^4 \to (\mathbb{Z}_2)^7$ definido por
$C(u_1, u_2, u_3, u_4) = (u_1 + u_3, u_1, u_2, u_2 + u_3, u_2 + u_3 + u_4, u_4, u_1 + u_2 + u_4)$.
Determina los parámetros del código y calcula su matriz generadora.

Solución. La longitud del código es 7 y la dimensión es 4. Calculamos las imágenes de los vectores de la base,

$C(1, 0, 0, 0) = (1, 1, 0, 0, 0, 0, 1)$, $C(0, 1, 0, 0) = (0, 0, 1, 1, 1, 0, 1)$,
$C(0, 0, 1, 0) = (1, 0, 0, 1, 1, 0, 0)$, $C(0, 0, 0, 1) = (0, 0, 0, 0, 1, 1, 1)$,

por tanto su matriz generadora es

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

Una segunda ventaja de los códigos lineales es la utilización de la matriz generadora para la codificación. Ahora no es necesario mantener en memoria todas las correspondencias entre símbolos fuente y palabras código, es suficiente tener la matriz generadora del código. Si $G = (g_{ij})$ es dicha matriz y $u = (u_1, u_2, u_3, \ldots, u_k)$ es un símbolo fuente, para calcular la palabra código correspondiente basta calcular Gu^t.

$$(C(u))^t = Gu^t = \begin{pmatrix} g_{11} & g_{12} & \cdots & g_{1k} \\ g_{21} & g_{22} & \cdots & g_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ g_{n1} & g_{n2} & \cdots & g_{nk} \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_k \end{pmatrix}$$

Ejemplo 4.2.8. Consideremos el $(7, 4)$ código lineal binario C con matriz generadora G:

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

la palabra código correspondiente al símbolo fuente $u = 1010$ es $C(u) = (Gu^t)^t = 0011101$.

Definición 4.2.9. Un (n, k) código lineal C es un **código sistemático** si las k primeras filas de su matriz generadora G forman la matriz identidad de orden k, I_k. Es decir, $G = \begin{pmatrix} I_k \\ Q \end{pmatrix}$ siendo Q una matriz de $n - k$ filas y k columnas. En ese caso, se dice que la matriz generadora del código está en forma estándar o que es una matriz generadora estándar. Un (n, k) código lineal C es un **código separable** si entre las filas de su matriz generadora están presentes las filas de I_k, la matriz identidad de orden k.

Si C es un código sistemático, al codificar un vector $u = (u_1, u_2, u_3, \ldots, u_k)$ de $(\mathbb{Z}_p)^k$, el vector código $C(u) = (u_1, u_2, u_3, \ldots, u_k, v_{k+1}, \ldots, v_n)$ tiene dos partes claramente diferenciadas. Las primeras k entradas forman el elemento u de $(\mathbb{Z}_p)^k$, la información que se desea enviar, son los **dígitos de información**. Las $n - k$ entradas restantes forman la redundancia añadida por el código, se denominan **dígitos de control**. Si C es un código separable, los k dígitos que forman el vector fuente aparecen en el vector código correspondiente, no necesariamente ordenados ni en las k primeras componentes. Se

puede hablar también de dígitos de información y dígitos de control. Todo código sistemático es un código separable.

Ejemplo 4.2.10. Los códigos lineales control de paridad, triple repetición y triple paridad son códigos sistemáticos.

El $(7, 4)$ código lineal binario C con matriz generadora G, es un código separable, pero no es sistemático.

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

Los cuatro dígitos que forman el símbolo fuente aparecen respectivamente en las posiciones tercera, quinta, sexta y segunda, pues $C(u_1, u_2, u_3, u_4) = (u_1 + u_2 + u_3, u_4, u_1, u_1 + u_4, u_2, u_3, u_1 + u_2 + u_4)$

Para un (n, k) código lineal C, siempre es posible hallar k entradas de los vectores código que permiten hallar el vector fuente correspondiente; dichas posiciones, que no son únicas, se denominan **posiciones de información** y corresponden a k filas de la matriz generadora del código que sean linealmente independientes; las $(n - k)$ entradas restantes se denominan **posiciones de control**.

Ejemplo 4.2.11. El código del ejemplo 4.2.8 no es un código separable. Se pueden tomar como posiciones de información las cuatro primeras entradas de las palabras código, las cuatro primeras filas de la matriz generadora son linealmente independientes, las tres últimas serán posiciones de control. Por igual motivo, se pueden considerar las cuatro últimas entradas como posiciones de información y las tres primeras como posiciones de control.

4.2.2. Descodificación en códigos lineales

Estudiamos a continuación cómo aprovechar la estructura lineal de los códigos para corregir los errores producidos en la transmisión. Para ello introducimos un nueva matriz asociada a un código lineal, la matriz control de paridad.

Definición 4.2.12. Sea c un vector código recibido como el vector r, se define el **vector error** producido en la transmisión de c como el vector $e = r - c$; equivalentemente, $r = c + e$.

Definición 4.2.13. Sea C un (n,k) código lineal sobre $\mathbb{Z}_p$. Una matriz $H \in \mathcal{M}_{(n-k)\times n}(\mathbb{Z}_p)$ y rango $(n-k)$ es una **matriz control de paridad** para el código C si $Hv^t = (0)$, para todo vector código v de C.

Una matriz control de paridad H para un código C es la matriz de coeficientes de un sistema de $(n-k)$ ecuaciones lineales con n incógnitas, homogéneas e independientes, que caracterizan el subespacio vectorial de las palabras código. Así pues, un mismo código lineal puede tener distintas matrices control de paridad; basta cambiar el sistema de ecuaciones que lo determine por uno equivalente. Recíprocamente una misma matriz H puede ser matriz control de paridad de varios códigos lineales; si bien, todos ellos tendrán el mismo subespacio código.

El siguiente resultado nos permite caracterizar una matriz control de paridad de un código lineal mediante su matriz generadora.

Teorema 4.2.14. *Sea C un (n,k) código lineal sobre $\mathbb{Z}_p$ con matriz generadora G. Una matriz H de $\mathcal{M}_{(n-k)\times n}(\mathbb{Z}_p)$ y rango $(n-k)$ es una matriz control de paridad para el código C si, y sólo si, el producto de la matriz H por la matriz de G es la matriz cero, es decir, $HG = (0)$.*

Demostración. Denotemos por $g_1, g_2, \ldots, g_k$ los k vectores código que forman las columnas de la matriz G. Se tiene que $\{g_1, g_2, \ldots, g_k\}$ es una base del subespacio código de C y la igualdad $HG = (0)$ es equivalente a que $H(g_i)^t = (0)$ para todo $i = 1, 2, \ldots, k$.

Si H es una matriz control de paridad para el código C entonces $Hc^t = (0)$ para todo vector código c de C, en particular para los vectores código $g_1, g_2, \ldots, g_k$. Por tanto, $H(g_i)^t = (0)$ para todo $i = 1, 2, \ldots, k$.

Recíprocamente, supongamos que $H(g_i)^t = (0)$ para todo $i = 1, 2, \ldots, k$. Puesto que todo vector código c es combinación lineal de los vectores g_i, existen $\alpha_1, \alpha_2, \ldots, \alpha_k \in \mathbb{Z}_p$ tales que $c = \alpha_1 g_1 + \alpha_2 g_2 + \cdots + \alpha_k g_k$. Por lo tanto, $Hc^t = H(\alpha_1 g_1 + \alpha_2 g_2 + \cdots + \alpha_k g_k)^t = \alpha_1(H(g_1)^t) + \alpha_2(H(g_2)^t) + \cdots + \alpha_k(H(g_k)^t) = \alpha_1(0) + \alpha_2(0) + \cdots + \alpha_k(0) = (0)$. □

Para calcular una matriz control de paridad de un (n,k) código lineal C con matriz generadora G podemos seguir cualquiera de los dos métodos siguienes:

1. calcular un sistema de ecuaciones lineales homogéneas e independientes que caractericen el subespacio código de C, basándonos en que los vectores columna de la matriz G forman una base de dicho subespacio,

2. utilizar la caracterización que proporciona el teorema 4.2.14 y calcular una matriz H de $\mathcal{M}_{(n-k)\times n}(\mathbb{Z}_p)$ y rango $(n-k)$ tal que $HG = (0)$.

Lo que supone resolver un sistema de k ecuaciones lineales homogéneas con n incógnitas y tomar $(n-k)$ soluciones linealmente independientes.

Ejemplo 4.2.15. Calcula una matriz control de paridad para el $(7,4)$ código lineal binario C cuya matriz generadora es G,

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

Solución. Método 1: como la matriz del sistema de ecuaciones lineal homogéneo que deben verificar el conjunto de palabras código. Un vector $x = (x_1, x_2, x_3, \ldots, x_7) \in (\mathbb{Z}_2)^7$ pertenecen al conjunto de palabras código si es combinación lineal de los elementos de la base y eso ocurre cuando la matriz

$$(Gx^t) = \begin{pmatrix} 1 & 0 & 1 & 0 & x_1 \\ 1 & 0 & 0 & 0 & x_2 \\ 0 & 1 & 0 & 0 & x_3 \\ 0 & 1 & 1 & 0 & x_4 \\ 0 & 1 & 1 & 1 & x_5 \\ 0 & 0 & 0 & 1 & x_6 \\ 1 & 1 & 0 & 1 & x_7 \end{pmatrix}$$

tiene el mismo rango que la matriz G; que, en este caso, es cuatro. Para ello, deben ser nulos todos los menores de orden cinco. Utilizando la técnica de orlado de menores, es suficiente que sean nulos aquellos menores que se obtengan completando un menor de orden cuatro de la matriz G no nulo; por ejemplo, el formado por la segunda, tercera, cuarta y quinta fila (posiciones de información),

$$det \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \end{pmatrix} = 1$$

los posibles menores de orden cinco orlados del anterior son

$$det\begin{pmatrix}1&0&1&0&x_1\\1&0&0&0&x_2\\0&1&0&0&x_3\\0&1&1&0&x_4\\0&1&1&1&x_5\end{pmatrix},\quad det\begin{pmatrix}1&0&0&0&x_2\\0&1&0&0&x_3\\0&1&1&0&x_4\\0&1&1&1&x_5\\0&0&0&1&x_6\end{pmatrix},\quad det\begin{pmatrix}1&0&0&0&x_2\\0&1&0&0&x_3\\0&1&1&0&x_4\\0&1&1&1&x_5\\1&1&0&1&x_7\end{pmatrix}$$

igualándolos a cero nos dan respectivamente

$$x_1+x_2+x_3+x_4=0,\quad x_4+x_5+x_6=0,\quad x_2+x_3+x_4+x_5+x_7=0$$

la matriz de coeficientes del sistema es la matriz H

$$H=\begin{pmatrix}1&1&1&1&0&0&0\\0&0&0&1&1&1&0\\0&1&1&1&1&0&1\end{pmatrix}$$

Método 2: como una matriz $H\in\mathcal{M}_{3\times7}(\mathbb{Z}_2)$ de rango 3 tal que $HG=(0)$. Denotemos por $x=(x_1,x_2,x_3,\ldots,x_7)\in(\mathbb{Z}_2)^7$ un vector fila arbitrario de la matriz H buscada, x debe verificar que $xG=(0)$, equivalentemente $G^tx^t=(0)$. Se obtiene el siguiente sistema de cuatro ecuaciones lineales y homogéneas con siete incógnitas

$$\begin{pmatrix}1&1&0&0&0&0&1\\0&0&1&1&1&0&1\\1&0&0&1&1&0&0\\0&0&0&0&1&1&1\end{pmatrix}\begin{pmatrix}x_1\\x_2\\x_3\\x_4\\x_5\\x_6\\x_7\end{pmatrix}=\begin{pmatrix}0\\0\\0\\0\end{pmatrix}$$

La matriz de coeficientes tiene rango cuatro; por tanto, para resolver el sistema, se deben despejar cuatro incógnitas en función de otras tres (denominados parámetros), con el fin de conseguir un sistema de cuatro ecuaciones con cuatro incógnitas que tenga rango cuatro. Consideramos x_1, x_6 y x_7, como parámetros; obteniendo el sistema

$$\begin{pmatrix}1&0&0&0\\0&1&1&1\\0&0&1&1\\0&0&0&1\end{pmatrix}\begin{pmatrix}x_2\\x_3\\x_4\\x_5\end{pmatrix}=\begin{pmatrix}x_1+x_7\\x_7\\x_1\\x_6+x_7\end{pmatrix}$$

Despejando, se obtiene que

$$\left.\begin{array}{rcl}x_2&=&x_1+x_7\\x_3&=&x_1+x_7\\x_4&=&x_1+x_6+x_7\\x_5&=&x_6+x_7\end{array}\right\}$$

para conseguir tres soluciones linealmente independientes, damos a los parámetros los siguientes valores:

- $x_1 = 1, x_6 = 0, x_7 = 0$, con lo cual $x_2 = 1, x_3 = 1, x_4 = 1, x_5 = 0$.
- $x_1 = 0, x_6 = 1, x_7 = 0$, con lo cual $x_2 = 0, x_3 = 0, x_4 = 1, x_5 = 1$.
- $x_1 = 0, x_6 = 0, x_7 = 1$, con lo cual $x_2 = 1, x_3 = 1, x_4 = 1, x_5 = 1$.

Colocando por filas estas tres soluciones se obtiene la matriz H,

$$H = \begin{pmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 & 1 \end{pmatrix}$$

Definición 4.2.16. Se define el **síndrome** de un vector v de $(\mathbb{Z}_p)^n$ como el vector $sin(v) = vH^t \in (\mathbb{Z}_p)^{n-k}$.

Nota: Por la propia definición de matriz control de paridad, un vector v de $(\mathbb{Z}_p)^n$ es un vector código si, y sólo si, su síndrome es el vector cero, $sin(v) = 0$. En consecuencia, el proceso de detección de errores en los códigos lineales se simplifica: recibido un vector r se calcula su síndrome, $sin(r)$; si es nulo, la transmisión se considera correcta; en caso contrario, se han producido errores en la transmisión. Se elimina de esta forma la necesidad comparar la palabra recibida con cada una de las palabras código.

Tabla de síndromes. Como la matriz control de paridad de un (n, k) código lineal tiene rango $n - k$, el número de posibles síndromes es p^{n-k} pues todo vector de $(\mathbb{Z}_p)^{n-k}$ es síndrome de algún vector de $(\mathbb{Z}_p)^n$. A cada vector s_i de $(\mathbb{Z}_p)^{n-k}$ se le asocia el vector v_i de $(\mathbb{Z}_p)^n$ del menor peso posible cuyo síndrome sea s_i, si hay más de una posible elección se elige uno de ellos arbitráreamente. El vector v_i se denomina representante del síndrome s_i.

La tabla de síndromes de un código C se construye disponiendo en p^{n-k} filas y 2 columnas los vectores v_i y sus correspondientes síndromes.

Representante	*Síndrome*
$v_1 = 0$	$s_1 = 0$
v_2	s_2
v_3	s_3
$\cdots$	$\cdots$
v_i	s_i
$\cdots$	$\cdots$
$v_{p^{n-k}}$	$s_{p^{n-k}}$

La construcción de la tabla de síndromes se realiza de forma iterativa:

- Paso 1. Se consideran los vectores $v_1 = 0$ de $(\mathbb{Z}_p)^n$ y $s_1 = 0$ de $(\mathbb{Z}_p)^{n-k}$.
- Paso 2. Para $i = 2, 3, \ldots, p^{n-k}$; se busca un vector v_i de $(\mathbb{Z}_p)^n$ del menor peso posible cuyo síndrome sea distinto de los síndromes anteriores, $s_1, s_2, s_3, \ldots, s_{i-1}$. Para ello se comienza con los vectores de peso 1, luego los de peso 2 y así sucesivamente.

El **proceso de descodificación** mediante la tabla de síndromes es el siguiente: recibido un vector r, se calcula su síndrome, $sin(r) = rH^t$. Si no es nulo, se localiza dicho vector en la segunda columna de la tabla. Se considera que v_i, el representante correspondiente a dicho síndrome, es el error cometido. Se descodifica el vector r como el vector código $c = r - v_i$.

Ejemplo 4.2.17. Sea C el código lineal binario de triple paridad. Una matriz control de paridad H para C es

$$H = \begin{pmatrix} 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 \end{pmatrix}$$

Realizando el proceso descrito para calcular los representantes, obtenemos la tabla de síndromes,

Representante	*Síndrome*
000000	000
100000	110
010000	101
001000	011
000100	100
000010	010
000001	001
100001	111

Si recibimos, por ejemplo, el vector $r = 101000$ lo descodificaremos como sigue:

- Calculamos el síndrome de r,

$$sin(r) = rH^t = \begin{pmatrix} 1 & 0 & 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 1 \end{pmatrix}$$

- El representante con síndrome 101 es $v_i = 010000$.

- El vector código transmitido es $c = r - v_i = 101000 - 010000 = 111000$. Al ser C un código sistemático, el símbolo fuente es 111.

Al realizar la descodificación utilizando la tabla de síndromes, los errores que se corrigen son exactamente los representantes de los síndromes, aunque no sean los errores producidos.

Por ejemplo, continuando con el código binario de triple paridad, supongamos que se transmite la palabra código $c = 101101$. Si la palabra recibida fuese $r = 011101$ entonces $sin(r) = 011$, siendo su representante la palabra 001000; así pues se interpreta el error producido, 110000, como el error simple 001000.

Nótese que el síndrome 111 tiene más de un posible representante pues $sin(100001) = sin(010010) = sin(001100) = 111$, en la tabla se ha elegido al primer vector como representante de dicho síndrome. Así pues, si al transmitir la palabra código $c = 101101$ se recibe la palabra $r = 001100$, $sin(r) = 111$ y r se descodifica correctamente como la palabra c; sin embargo, si se recibe la palabra $r' = 111111$, $sin(r') = 111$ y se descodifica incorrectamente como $c' = 011110$.

Cuestiones:

- ¿El vector $r - v_i$ es un vector código? Sí, pues el síndrome de $r - v_i$ es $sin(r) - sin(v_i) = 0$.

- ¿Esta regla de descodificación coincide con la descodificación por distancia mínima? Sí, $d(c, r) = w(v_i)$. Sea c' un vector código distinto del vector c; dado que $sin(r - c') = sin(r) = sin(v_i)$, por la elección del representante del síndrome se verifica que $w(v_i) \leq w(r - c')$, equivalentemente $d(c, r) \leq d(c', r)$.

- ¿Qué podemos asegurar sobre los representantes de los síndromes si la distancia mínima del código es $2\lambda + 1$? En este caso el código es corrector de λ errores, por lo tanto entre los representantes estarán al menos todos los vectores de peso a lo sumo λ.

- ¿Cómo se distingue entre una descodificación completa e incompleta? Si nuestro objetivo es únicamente corregir los errores de peso menor o igual que λ, podemos considerar una tabla de descodificación incompleta, en la cual sólo están las clases cuyos representantes son todos los vectores de peso menor o igual que λ. Así, la elección de los representantes es inmediata y la tabla es de menor tamaño. El único cambio

en el algoritmo de descodificación es: si el síndrome la palabra recibida r, $sin(r)$, no aparece en la tabla de síndromes incompleta, debemos comunicar que se han producido más de λ errores y no es posible la descodificación.

Ejemplo 4.2.18. La tabla incompleta de síndromes del código binario de triple paridad es:

Representante	*Síndrome*
000000	000
100000	110
010000	101
001000	011
000100	100
000010	010
000001	001

4.3. Códigos Hamming

Los códigos de Hamming, definidos independientemente por Marcel Golay (1949) y Richard Hamming (1950), son una familia de códigos lineales correctores de errores simples y con un sencillo algoritmo de descodificación, especialmente en el caso binario. Si bien es posible construir códigos de Hamming sobre alfabetos más generales, sólo los construiremos para el caso binario.

Definición 4.3.1. Sea m un número natural y H la matriz de $\mathcal{M}_{m\times(2^m-1)}(\mathbb{Z}_2)$ cuyas columnas son todos los vectores no nulos de $(\mathbb{Z}_2)^m$ colocados en orden lexicográfico; de esta manera, la j-ésima columna de H es la representación binaria del número natural j, $1 \leq j \leq 2^m - 1$. Un código lineal binario que tenga a H como matriz control de paridad se denomina **código de Hamming binario**, denotado por $H(m, 2)$.

Al introducir un código lineal vía una matriz control de paridad, estamos determinando sólo el subespacio código y no el código en sí mismo. De todas formas, todos los códigos considerados poseen esencialmente las mismas propiedades, pues el único cambio es la correspondencia entre vectores fuente y vectores código.

Observaciones:

1. Tiene sentido considerar la matriz H como matriz control de paridad de un código lineal porque al tener todos los vectores no nulos de $(\mathbb{Z}_2)^m$, entre sus columnas están las correspondientes a la base canónica del espacio y su rango será m.

2. Del orden de la matriz H se obtiene que la longitud de un código de Hamming binario $H(m,2)$ es $n = 2^m - 1$, su dimensión es $k = 2^m - m - 1$.

3. La distancia mínima de un código de Hamming $H(m,2)$ es 3. Todo vector de peso 1 tiene por síndrome la traspuesta de una columna de la matriz H, al ser todas no nulas no puede tener síndrome nulo y por lo tanto no es un vector código. Todo vector de peso 2 tiene por síndrome la traspuesta de la suma de dos columnas de H, como son todas distintas entre sí, no puede tener síndrome nulo y por lo tanto no es un vector código. Pero sí existen vectores código de peso 3, por ejemplo el vector con todas las entradas nulas excepto las tres primeras, $1110\ldots0$.

4. Los códigos de Hamming binarios corrigen exactamente los errores de peso 1. Por los parámetros de $H(m,2)$, el número de filas en la tabla de síndromes es $2^{n-k} = 2^m$. Por la distancia mínima, es un código corrector de errores simples; la tabla tiene, al menos, 2^m filas, una fila para los vectores código y $n = 2^m - 1$ filas para los n errores de peso 1. En otras palabras, los representantes de la tabla de síndromes son exactamente los $n + 1$ vectores de peso ≤ 1.

5. En algunos textos, se propone una definición más amplia de código de Hamming binario. Se toma como matriz H cualquier matriz de $\mathcal{M}_{m\times(2^m-1)}(\mathbb{Z}_2)$ cuyas columnas son todos los vectores no nulos de $(\mathbb{Z}_2)^m$, sin importar el orden. Los códigos obtenidos no tienen el mismo subespacio código, pero siguen teniendo las mismas propiedades esenciales ya que únicamente se han permutado las posiciones de las componentes de los vectores código.

Ejemplo 4.3.2. El código de Hamming $H(3,2)$ es un $(7,4)$ código lineal binario que tiene por matriz control de paridad la matriz

$$H = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix}$$

El código de Hamming $H(4,2)$ es un $(15,11)$ código lineal binario que tiene por matriz control de paridad la matriz

$$H = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix}$$

La elección de la ordenación de las columnas de la matriz H no es arbitraria, está motivada para definir un algoritmo de descodificación que evita la utilización de tablas. El algoritmo debe centrarse en la corrección de los errores simples, el tipo de error que un código de Hamming puede corregir. Sea $e_j = 0 \dots 010 \dots 0 \in (\mathbb{Z}_2)^n$ con una única entrada no nula en la posición j, su síndrome es $sin(e_j) = e_j H^t = (h_j)^t$, la j-ésima columna de H traspuesta que, por la ordenación establecida, es la representación binaria del número natural j. Así pues, todo vector recibido con un error simple en la j-ésima posición tiene por síndrome el valor j escrito en binario. Este hecho permite definir el siguiente algoritmo de descodificación:

Recibido un vector $r \in (\mathbb{Z}_2)^m$, se calcula su síndrome, $sin(r) = rH^t$. Si $sin(r) = 0$, el vector r se considera el vector código transmitido. Si $sin(r) \neq 0$, se supone que se ha producido un error simple; $sin(r)$ indica, en binario, la posición errónea. Se corrige el error sin más que modificar el valor del dígito en dicha posición.

Ejemplo 4.3.3. Utilizando el código de Hamming $H(3, 2)$, supongamos que el vector código enviado $c = 0110011$, ha sido recibido como $r = 0110001$. El proceso de descodificación sería: Se calcula el síndrome de r

$$sin(r) = rH^t = \begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 0 \end{pmatrix}$$

que es la representación binaria del 6. Por tanto, interpretamos que se ha producido un error simple en la sexta posición, se cambia el 0 de la sexta posición por un 1 y obtenemos el vector código $c = 0110011$.

La obtención del vector fuente a partir de un vector código, una vez corregido el error si lo hubiera, depende del código $H(m, 2)$ elegido entre todos los códigos que tienen a la matriz H como matriz control de paridad. En la práctica, la elección está determinada. Se considera el código lineal separable que asocia a un vector fuente $u = (u_1, u_2, \dots, u_k)$, $k = 2^m - m - 1$, el vector código cuyas entradas en las $2^m - m - 1$ posiciones no potencia de 2 son las entradas del vector u colocadas por orden (dígitos de información), y en las m entradas restantes los valores correspondientes para ser un vector código de Hamming (dígitos de control), $c = (x_1, x_2, u_1, x_3, u_2, u_3, \dots, u_{k-2}, u_{k-1}, u_k)$.

En términos de la matriz generadora del código significa que, suprimiendo las filas que están en los lugares potencia de 2, se obtiene I_k, la matriz identidad de orden k.

De esta forma, el vector fuente correspondiente a un vector código $c = (c_1, c_2, c_3, c_4, c_5, c_6, \ldots, c_{n-2}, c_{n-1}, c_n)$ es el vector de $2^m - m - 1$ componentes $u = (c_3, c_5, c_6, \ldots, c_{n-2}, c_{n-1}, c_n)$, obtenido de c suprimiendo las m entradas correspondientes a las posiciones de control, es decir, las posiciones potencia de 2.

Para calcular la matriz generadora del código de Hamming $H(m, 2)$ se resuelve el sistema de m ecuaciones lineales con $2^m - 1$ incógnitas $Hx^t = (0)$, despejando las incógnitas en las posiciones potencia de 2 (los dígitos de control) en función de las incógnitas en las posiciones no potencia de dos (los dígitos de información). Se calcula una base del subespacio código y colocando los vectores en el orden adecuado se obtiene la matriz generadora.

Ejemplo 4.3.4. Calcula la matriz generadora del código de Hamming $H(3, 2)$.
Para $m = 3$, la matriz control de paridad H es

$$H = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix}$$

se resuelve el sistema

$$Hx^t = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

Despejando los dígitos de control x_1, x_2, x_4 en función de los de información se obtiene

$$\left.\begin{array}{rcl} x_1 & = & x_3 + x_5 + x_7 \\ x_2 & = & x_3 + x_6 + x_7 \\ x_4 & = & x_5 + x_6 + x_7 \end{array}\right\}$$

Para conseguir los vectores de la base se van dando a los parámetros los siguiente valores,

- $x_3 = 1, x_5 = 0, x_6 = 0$ y $x_7 = 0$, con lo cual $x_1 = 1, x_2 = 1, x_4 = 0$.

- $x_3 = 0, x_5 = 1, x_6 = 0$ y $x_7 = 0$, con lo cual $x_1 = 1, x_2 = 0, x_4 = 1$.
- $x_3 = 0, x_5 = 0, x_6 = 1$ y $x_7 = 0$, con lo cual $x_1 = 0, x_2 = 1, x_4 = 1$.
- $x_3 = 0, x_5 = 0, x_6 = 0$ y $x_7 = 1$, con lo cual $x_1 = 1, x_2 = 1, x_4 = 1$.

Colocando los vectores en columnas y en el orden adecuado se obtiene la matriz G,

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

Ejemplo 4.3.5. Calcula la matriz generadora del código de Hamming $H(4, 2)$. Para $m = 4$, la matriz control de paridad H es

$$H = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix}$$

se resuelve el sistema $Hx^t = (0)$, despejando los dígitos de control x_1, x_2, x_4, x_8 en función de los de información, se obtiene

$$\left.\begin{array}{rcl} x_1 & = & x_3 + x_5 + x_7 + x_9 + x_{11} + x_{13} + x_{15} \\ x_2 & = & x_3 + x_6 + x_7 + x_{10} + x_{11} + x_{14} + x_{15} \\ x_4 & = & x_5 + x_6 + x_7 + x_{12} + x_{13} + x_{14} + x_{15} \\ x_8 & = & x_9 + x_{10} + x_{11} + x_{12} + x_{13} + x_{14} + x_{15} \end{array}\right\}$$

Para conseguir los vectores de la base se van dando a los parámetros los siguiente valores,

- $x_3 = 1, x_5 = 0, x_6 = 0, x_7 = 0, x_9 = 0, x_{10} = 0, x_{11} = 0, x_{12} = 0,$ $x_{13} = 0, x_{14} = 0, x_{15} = 0$, con lo cual $x_1 = 1, x_2 = 1, x_4 = 0, x_8 = 0$.
- $x_3 = 0, x_5 = 1, x_6 = 0, x_7 = 0, x_9 = 0, x_{10} = 0, x_{11} = 0, x_{12} = 0,$ $x_{13} = 0, x_{14} = 0, x_{15} = 0$, con lo cual $x_1 = 1, x_2 = 0, x_4 = 1, x_8 = 0$.
- $x_3 = 0, x_5 = 0, x_6 = 1, x_7 = 0, x_9 = 0, x_{10} = 0, x_{11} = 0, x_{12} = 0,$ $x_{13} = 0, x_{14} = 0, x_{15} = 0$, con lo cual $x_1 = 0, x_2 = 1, x_4 = 1, x_8 = 0$.

- $x_3 = 0, x_5 = 0, x_6 = 0, x_7 = 1, x_9 = 0, x_{10} = 0, x_{11} = 0, x_{12} = 0,$
  $x_{13} = 0, x_{14} = 0, x_{15} = 0$, con lo cual $x_1 = 1, x_2 = 1, x_4 = 1, x_8 = 0$.
- $x_3 = 0, x_5 = 0, x_6 = 0, x_7 = 0, x_9 = 1, x_{10} = 0, x_{11} = 0, x_{12} = 0,$
  $x_{13} = 0, x_{14} = 0, x_{15} = 0$, con lo cual $x_1 = 1, x_2 = 0, x_4 = 0, x_8 = 1$.
- $x_3 = 0, x_5 = 0, x_6 = 0, x_7 = 0, x_9 = 0, x_{10} = 1, x_{11} = 0, x_{12} = 0,$
  $x_{13} = 0, x_{14} = 0, x_{15} = 0$, con lo cual $x_1 = 0, x_2 = 1, x_4 = 0, x_8 = 1$.
- $x_3 = 0, x_5 = 0, x_6 = 0, x_7 = 0, x_9 = 0, x_{10} = 0, x_{11} = 1, x_{12} = 0,$
  $x_{13} = 0, x_{14} = 0, x_{15} = 0$, con lo cual $x_1 = 1, x_2 = 1, x_4 = 0, x_8 = 1$.
- $x_3 = 0, x_5 = 0, x_6 = 0, x_7 = 0, x_9 = 0, x_{10} = 0, x_{11} = 0, x_{12} = 1,$
  $x_{13} = 0, x_{14} = 0, x_{15} = 0$, con lo cual $x_1 = 0, x_2 = 0, x_4 = 1, x_8 = 1$.
- $x_3 = 0, x_5 = 0, x_6 = 0, x_7 = 0, x_9 = 0, x_{10} = 0, x_{11} = 0, x_{12} = 0,$
  $x_{13} = 1, x_{14} = 0, x_{15} = 0$, con lo cual $x_1 = 1, x_2 = 0, x_4 = 1, x_8 = 1$.
- $x_3 = 0, x_5 = 0, x_6 = 0, x_7 = 0, x_9 = 0, x_{10} = 0, x_{11} = 0, x_{12} = 0,$
  $x_{13} = 0, x_{14} = 1, x_{15} = 0$, con lo cual $x_1 = 0, x_2 = 1, x_4 = 1, x_8 = 1$.
- $x_3 = 0, x_5 = 0, x_6 = 0, x_7 = 0, x_9 = 0, x_{10} = 0, x_{11} = 0, x_{12} = 0,$
  $x_{13} = 0, x_{14} = 0, x_{15} = 1$, con lo cual $x_1 = 1, x_2 = 1, x_4 = 1, x_8 = 1$.

Colocando los vectores en columnas y en el orden adecuado se obtiene la matriz G,

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

Ejemplo 4.3.6. Consideremos de nuevo el código de Hamming $H(3,2)$. El vector fuente $u = 1010$ se codifica como el vector código $c = 1011010$. Al transmitirlo se produce un error doble y recibimos el vector $r = 1010110$. Como el síndrome de r es $sin(r) = 001$, el proceso de descodificación aplicado al vector r interpreta que se ha producido un error simple en el primer dígito, descodificando r como el vector código $c' = 0010110$ y como vector fuente $u' = 1110$.

Como pone de manifiesto el ejemplo anterior, los códigos de Hamming interpretan los errores dobles (en general, cualquier error detectado) como un error simple y los corrigen como tal, aumentando el número de posiciones erróneas. Para evitar esta situación, de cada código de Hamming binario, $H(m,2)$, se construye un nuevo código, denominado código de **Hamming ampliado**, denotado por $HA(m,2)$, añadiendo a cada vector código de $H(m,2)$ un dígito de control de paridad al final; es decir, un dígito 0 ó 1 para que el peso de cada vector en el nuevo código sea par. Este proceso de ampliación se puede aplicar a cualquier código lineal binario.

La longitud del código $HA(m,2)$ es $n = 2^m$, una unidad más que la longitud de $H(m,2)$; su dimensión $k = 2^m - m - 1$, la misma que $H(m,2)$ pues no se ha modificado el número de vectores código; en consecuencia, hay un dígito de control más, el último, que es la suma de todos los dígitos anteriores. Por la construcción, todo vector código de $H(m,2)$ de peso 3, el peso mínimo de dicho código, pasa a tener peso 4 en el código $HA(m,2)$, pues el nuevo dígito de control será un uno. El peso mínimo de un código de Hamming ampliado, $HA(m,2)$, es 4. En resumen, el código $HA(m,2)$ es un $(2^m, 2^m - m - 1, 4)$ código lineal binario.

El sistema de ecuaciones lineales del código ampliado tiene ahora una incógnita más y una ecuación más, $c_1 + c_2 + \cdots + c_{n-1} + c_n = 0$. Es decir, un vector binario c^* de longitud 2^m pertenece al código $HA(m,2)$ si, y sólo si, tiene peso par y el vector c, formado por las $2^m - 1$ primeras componentes de c^*, pertenece al código $H(m,2)$. Por lo tanto, la matriz control de paridad de un código $HA(m,2)$, denotada por H^*, se obtiene a partir de H, la matriz control de paridad del código $H(m,2)$, añadiendo a la derecha una columna de ceros y, en la parte inferior, una fila de unos.

$$H^* = \begin{pmatrix} & & & & & 0 \\ & & & & & 0 \\ & & H & & & \vdots \\ & & & & & 0 \\ 1 & 1 & 1 & \cdots & 1 & 1 \end{pmatrix}$$

A su vez, si el código $H(m,2)$ tiene a la matriz G como matriz generadora,

entonces el código $HA(m,2)$ tiene por matriz generadora $G^* = \begin{pmatrix} G \\ B \end{pmatrix}$, siendo B una matriz fila añadida a la matriz G de tal forma que todas las columnas de la matriz G^* tengan peso par; es decir, basta ampliar los vectores de la base del código $H(m,2)$, para obtener una base de $HA(m,2)$. Obsérvese que las posiciones potencia de dos siguen siendo las posiciones de control y las posiciones de información no han variado.

Ejemplo 4.3.7. El código de Hamming ampliado $HA(3,2)$, al igual que el código $H(3,2)$, tiene dimensión cuatro (16 vectores código), pero su longitud es ahora $2^m = 2^3 = 8$. Su matriz control de paridad es H^*,

$$H^* = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

La matriz generadora, G^*, para este código se obtiene a partir de G, la matriz generadora del código $H(3,2)$, añadiendo una fila cuyos valores sean la suma de los dígitos de cada columna de la matriz G,

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

Como se ha comentado, el peso mínimo de un código de Hamming ampliado, $HA(m,2)$, es 4; lo que nos va a permitir distinguir los errores dobles de los errores simples.

Sea $e_j = 0 \dots 010 \dots 0$ el vector de $(\mathbb{Z}_2)^n$ con una única entrada no nula en la posición $j, 1 \leq j \leq n = 2^m$. Su síndrome, $sin(e_j)$, es la traspuesta de la columna j-ésima de la matriz H^*. Por la construcción de H^*, para los valores de j menores estrictamente que 2^m, las m primeras componentes de $sin(e_j)$, son la representación binaria del natural j y la última componente es 1; para $j = 2^m$, $sin(e_{2^m}) = 00 \dots 001$. Lo mismo sucede para todo vector recibido con un error simple. Por otra parte, sea e un vector de $(\mathbb{Z}_2)^m$ con dos únicas entradas no nulas, $e = 0 \dots 010 \dots 010 \dots 0$, su síndrome, $sin(e)$, es la traspuesta de la suma de las dos columnas de H^* correspondientes a las entradas no nulas de e, siendo su última componente igual a cero. Por

lo tanto, la última componente de $sin(e)$ será igual a cero. Así pues, un vector recibido con un error doble no tiene por síndrome la traspuesta de una columna de dicha matriz, lo que es fácilmente observable pues su último dígito es cero. Estos hechos nos permiten definir el siguiente algoritmo de descodificación:

Recibido un vector r, se calcula su síndrome,

$$sin(r) = r(H^*)^t = s_1 s_2 s_3 \dots s_{m-1} s_m s_{m+1} = s|s_{m+1}$$

siendo $s = s_1 s_2 s_3 \dots s_{m-1} s_m$ la palabra binaria formada por las m primeras componentes de $sin(r)$.

1. Si $s_{m+1} = 0$ y
   - *a)* $s = 0$, es decir $sin(r) = 0$, se considera que r es el vector código enviado.
   - *b)* $s \neq 0$, se considera que se han producido al menos dos errores en la transmisión y no es posible la descodificación.
2. Si $s_{m+1} = 1$ y
   - *a)* $s = 0$, se considera que se ha producido un error simple en la n-ésima posición.
   - *b)* $s \neq 0$, se considera que se ha producido un error simple en la j-ésima posición ($1 \leq j < n = 2^m$), donde j es el número natural cuya representación binaria es s.

Ejemplo 4.3.8. Consideremos el código de Hamming ampliado $HA(3, 2)$, con matriz generadora G^*,

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

El vector fuente $u = 1011$, que en el código $H(3, 2)$ se codifica como 0110011, es codificado ahora como $c = 01100110$. Si c es recibido como $r = 01100010$, su síndrome es $r(H^*)^t = 110|1$, esto nos indica que se ha producido un error simple en la sexta posición, correspondiente al valor binario 110. Si c

es recibido como $r = 01100111$, su síndrome es $r(H^*)^t = 000|1$, esto nos indica que se ha producido un error simple, pero esta vez en la última posición. Por último, si c es recibido como $r = 00100010$, al calcular su síndrome obtenemos $r(H^*)^t = 100|0$, esto nos indica que se ha producido un error al menos doble que no es posible corregir.

4.4. Códigos Reed-Muller y códigos Golay

Otras familias de códigos lineales son los códigos de Reed-Muller y los códigos de Golay. Los **códigos de Reed-Muller**, introducidos en 1954, deben su nombre a sus descubridores, David E. Muller e Irving S. Reed. Es una de las colecciones de códigos binarios más antiguas y han sido ampliamente utilizadas; por ejemplo, en la transmisión de fotografías del planeta Marte.

Para cada número natural $m \geq 1$ el código de Reed-Muller $R(m)$ es un código lineal binario de parámetros $(2^{m+1}, m+1, 2^{m-1})$ cuya matriz generadora, denotada R_m, tiene por filas todos los vectores binarios de longitud $m+1$ cuyo último dígito es 1, colocados en orden lexicográfico.

Las matrices generadoras para los códigos Reed-Muller $R(1)$, $R(2)$, $R(3)$ y $R(4)$ son respectivamente R_1, R_2, R_3 y R_4;

$$(R_1)^t = \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}, \quad (R_2)^t = \begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix}, \quad (R_3)^t = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

$$R_1 = \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix} \quad R_2 = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{pmatrix} \quad R_3 = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix}$$

$$(R_4)^t = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

Es posible dar una definición recursiva de la matriz generadora del código Reed-Muller.

- La matriz generadora para $R(1)$ es $R_1 = \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}$
- Si R_m es la matriz generadora del código $R(m)$, entonces la matriz generadora de $R(m+1)$ es R_{m+1}

$$R_{m+1} = \left(\begin{array}{cc} 0 & \\ 0 & \\ \cdots & R_m \\ 0 & \\ 0 & \\ \hline 1 & \\ 1 & \\ \cdots & R_m \\ 1 & \\ 1 & \end{array}\right)$$

En 1949, Marcel Golay, en un trabajo de una página de extensión, dio la definición de una familia de cuatro códigos lineales sistemáticos, dos binarios y dos ternarios, dando las matrices generadoras de dichos códigos, sin ninguna indicación de cómo las obtuvo.

El **código de Golay** binario G_{24} es un código de parámetros $(24, 12, 8)$ cuya matriz generadora es $G = \begin{pmatrix} I_{12} \\ A \end{pmatrix}$, siendo I_{12} la matriz identidad de orden 12 y A la matriz cuadrada de orden 12 y simétrica,

$$A = \begin{pmatrix} 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \end{pmatrix}$$

La matriz $H = (A\ I_{12})$ es una matriz control de paridad para el código G_{24}, también lo es la traspuesta de la matriz generadora, $(I_{12}\ A)$.

El código de Golay binario G_{23} es un código de parámetros $(23, 12, 7)$ cuya matriz generadora se obtiene de la matriz G suprimiendo la última fila de la matriz A. Este código tiene la particularidad de que corrige exactamente todos los errores de hasta peso tres. Si se amplía el código G_{23} como se hizo con los códigos de Hamming se obtiene el código de Golay G_{24}.

El código de Golay ternario G_{12} es un código de parámetros $(12, 6, 6)$ cuya matriz generadora es $G = \begin{pmatrix} I_6 \\ A \end{pmatrix}$ siendo I_6 la matriz identidad de orden 6 y A la matriz cuadrada de orden 6 y simétrica,

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

Al igual que en el caso binario las matrices $H = (-A\ I_6)$ y $G^t = (I_6\ A)$ son matrices control de paridad del código G_{12}.

El código de Golay ternario G_{11} es un código de parámetros $(11, 6, 5)$ cuya matriz generadora se obtiene de la matriz G suprimiendo la última fila de la matriz A. Este código tiene la particularidad de que corrige exactamente todos los errores de hasta peso dos.