UNIVERSIDAD CENTRAL DE VENEZUELA
FACULTAD DE CIENCIAS
ESCUELA DE MATEMÁTICA

Estudio de la relación existente entre los Códigos Binarios Perfectos y los Sistemas de Steiner

Trabajo Especial de Grado presentado ante la ilustre Universidad Central de Venezuela por el **Br. Alex Padrón** para optar al título de Licenciado en Matemática.

Tutor: Irene Santos.

Caracas, Venezuela
Octubre 2014

Nosotros, los abajo firmantes, designados por la Universidad Central de Venezuela como integrantes del Jurado Examinador del Trabajo Especial de Grado titulado "**Estudio de la relación existente entre los Códigos Binarios Perfectosy los Sistemas de Steiner**", presentado por el **Br. Alex Padrón**, titular de la Cédula de Identidad **17.427.774**, certificamos que este trabajo cumple con los requisitos exigidos por nuestra Magna Casa de Estudios para optar al título de **Licenciado en Matemática**.

Dra. Irene Santos
Tutor

Dr. Manuel Maia
Jurado

Dr. Mauricio Angel
Jurado

Para Milagros, Aura, Amelia, Alex y Nercy.

Agradecimientos

En el tiempo que llevó realizar este trabajo, fueron muchas las personas que sirvieron de ayuda e inspiración. Las siguientes líneas intentan plasmar (quizás infructuosamente), en palabras, mi gratitud hacia cada una de ellas:

Empecemos por agradecer a *Milagros* y *Alex*, *mis padres*, figuras ejemplares y que me han brindado todo lo que ha estado a su disposición en esta vida (y en todas las demás en las que tuvieron la oportunidad, seguramente).

A la profesora *Irene Santos*, por mostrarme la puerta de los códigos, animarme a atravesarla, y compartir conmigo el viaje, una vez del otro lado.

A *Nercy*, por ser y por estar. Por hacer de fulcro y de palanca, pero nunca hacer las veces del peso.

A un grandioso círculo de amistades que no ha sabido sino apoyarme y animarme a lo largo de toda una vida.

A *Bradbury* y a *Sábato*, por despejar mi mente, o hacerla colapsar, según sea el caso.

A estos y muchos otros que estuvieron ahí, mi más puro agradecimiento.

Índice general

Introducción

Es bien sabido que el mundo de las matemáticas goza de una enorme profundidad y un sinfín de matices; esto gracias a la incesante labor de miles y miles de mentes entusiastas que buscan respuestas a inquietudes, por muy mundanas - o complejas - que puedan parecer. En este trabajo estudiaremos una pequeñísima muestra de las estructuras posibles al adentrarnos en el mundo de los códigos. En específico, nos centraremos en los llamados *códigos perfectos*, desarrollados a mediados del siglo XX por un par de matemáticos, Richard Hamming (USA) y Marcel Golay (Suiza).

Alrededor de 1940 se estaba empezando a desarrollar la teoría de la información como un campo en las ciencias aplicadas, cuando Hamming se percató de la necesidad de corregir errores de transmisión. Si bien ya se estaba usando la idea del *control de paridad*, se debía conseguir un método mas sofisticado de chequear la información. Así surgen los códigos de Hamming. Por otra parte, Golay estaba trabajando en aplicaciones militares de la teoría de información, por lo que se aseguró de desarrollar un código más *complejo* que los que se venían usando. Así, nacen los códigos de Golay. Estos dos son los integrantes insignia de la familia de los códigos perfectos.

Ahora bien. ¿en qué consiste la teoría de códigos? ¿qué hay detrás de ella? Pues, a través de estas páginas veremos que la materia abarca un lecho considerable; y es que, la teoría de códigos, a pesar de ser una rama relativamente nueva, se ha abierto camino en una gran variedad de campos, teóricos (teoría combinatoria, probabilidades, geometría afín) y prácticos (comunicaciones espaciales, procesamiento de audio). Encontraremos aplicación de resultados básicos de álgebra lineal: veremos como los códigos ilustran perfectamente algunas nociones elementales; también se tocarán temas de álgebra abstracta: puntualmente, estudiaremos diseños, estructuras algebráicas que nos permitirán ver ciertos códigos con unos ojos muy geométricos. Por supuesto, nada de esto debe sorprender, pues, como siempre

en las matemáticas, se busca trasladar los resultados a otros campos, para aprovechar las riquezas conocidas de cada área.

En este trabajo empezaremos definiendo algunos elementos básicos que nos permitan comprender la naturaleza y el origen de los códigos. Hablaremos del *alfabeto* y las *palabras*. Definiremos algunos parámetros elementales de los códigos, como la *distancia* y el *peso*. Veremos cómo generar un código y qué se necesita para que este sea *perfecto*. También mostraremos algunos resultados de álgebra de polinomios y grafos, para más adelante emplearlos al definir ciertos tipos de códigos.

En el segundo capítulo nos enfocaremos en tres tipos de códigos: Los códigos de *Hamming*, los códigos de *Golay* y los códigos *cíclicos*. Veremos de dónde surge cada uno, qué propiedades tienen, cómo se estructuran y qué fortalezas poseen y carecen. Ya en el tercer capítulo estudiaremos algunas estructuras de incidencia finita. Definiremos los *diseños* y estudiaremos algunos de ellos; enfocándonos en los *sistemas de steiner*, para posteriormente mostrar que los mismos poseen cualidades fascinantes que nos ayudarán a representar graficamente algunos códigos.

Sin más, ¡empecemos a codificar!

Capítulo 1

Definiciones Básicas Preliminares

1. Códigos

Para empezar, supongamos que tenemos una fuente de información en el extremo A de una larga habitación y que el potencial receptor de dicha información está en el extremo B de la misma. Ahora bien, si se tratase de una habitación completamente libre de ruido, el mensaje puede viajar con cierta facilidad desde A hasta B. Sin embargo, si el espacio de nuestro salón presenta alguna perturbación, los datos podrían verse afectados en su camino, o incluso, podrían no llegar hasta B.

Esto sucedería si nuestro mensaje es enviado sin ningún tipo de estructura que prevenga su distorsión. La teoría de códigos surge de la necesidad de brindar una base que permita la manipulación y transmisión de información desde un punto a otro, con la menor pérdida de datos en el proceso, manteniendo, de esta forma, la fidelidad entre el mensaje enviado y el recibido.

Estudiemos entonces algunas definiciones elementales que nos permitan adentrarnos en la materia en cuestión.

1.1. Definiciones Básicas. En términos generales, se desea transmitir un mensaje a través de un canal con ruido, como se ve en el diagrama:

$$\textit{Mensaje} \dashrightarrow \textit{Codificador} \dashrightarrow \textit{Transmisor} \rightsquigarrow \textit{Receptor} \dashrightarrow \textit{Decodificador} \dashrightarrow \textit{Mensaje decodificado}$$

Donde la flecha ondulada representa un canal con ruido.

Se busca detectar y corregir posibles errores en la transmisión.

El esquema que se plantea es el siguiente:

- Se utiliza el alfabeto binario $\mathbb{Z}_2=\{0,1\}$.
- Se codifica el mensaje original usando palabras binarias (Un error es una confusión entre 0 y 1).
- Se transmite el mensaje, que se corrompe.
- Se corrige el mensaje, de ser posible.

- El decodificador traduce el mensaje.

Observación 1. Hay 2^n palabras binarias de longitud $n \in \mathbb{N}$.

Diremos que un código binario C de longitud n es cualquier subconjunto de $\mathbb{Z}_2^n$.

El modelo será simplificado suponiendo lo siguiente:

- El canal es binario: Los mensajes que se transmiten son cadenas de ceros y unos.
- El canal es simétrico: La probabilidad p de cambiar un 0 por un 1 en la transmisión es la misma que la de cambiar un 1 por un 0.
- El canal no tiene memoria: La probabilidad p de cambiar un 0 por un 1 en la transmisión no depende de mensajes previamente enviados. La probabilidad p es conocida como *probabilidad de error del canal simétrico*.
- No hay errores de sincronización: El número de símbolos recibidos es igual al número de símbolos enviados.
- Enviamos información redundante: Para estar en la capacidad de detectar y corregir errores se utiliza un código de tipo *(n,k)*, n>k, donde un mensaje formado por k símbolos se transforma en una palabra de n símbolos, por lo tanto hay n-k símbolos redundantes.

Observación 2. Un código de tipo *(n,k)* es un subconjunto de $\mathbb{Z}_2^n$.

Ejemplo 1.1. El código $C = \{0000, 1000, 0110, 1011, 1110, 1101, 0011, 0101\}$ es un subconjunto de $\mathbb{Z}_2^4$.

Ejemplo 1.2. El código $C_2 = \{001, 002, 012, 121, 212, 222\}$ es un subconjunto de $\mathbb{Z}_3^3$.

Definición 1.1. Un alfabeto es un conjunto finito $\mathcal{A} = \{a_1, a_2, \ldots, a_q\}$. A los elementos de $\mathcal{A}$ se le llaman símbolos y al número q se le llama raíz de $\mathcal{A}$

Definición 1.2. Una cadena o palabra de longitud n sobre $\mathcal{A}$ es una sucesión de n elementos de $\mathcal{A}$

Se denota por:

- $\mathcal{A}^n$ al conjunto de todas las n-cadenas de longitud n.
- $\mathcal{A}^*$ al conjunto de todas las palabras sobre $\mathcal{A}$.

Sea $\mathcal{A} = \{a_1, a_2, \ldots, a_q\}$. Definimos:

- Un código q-ario sobre $\mathcal{A}$ es un subconjunto $C \subset \mathcal{A}^*$.
- Los elementos de C se llaman palabras código.
- El número M=$\|C\|$ se llama tamaño del código.
- Si todas las palabras tienen longitud fija, n, diremos que C es un código de bloque, de parámetros *[n,M]*, o que C es un $[n, M]_q$-código .
- Si C no es de bloque, diremos que es de longitud variable.

DEFINICIÓN 1.3. Sea C un código de bloque, $x,\ y\ \in C$ dos palabras código. La distancia de Hamming entre $x,\ y$, denotada por $d(x, y)$ se define como el número de coordenadas en que estas difieren.

$$d{:}\mathcal{A}^n \longrightarrow [0, n] \subset \mathbb{N}.$$

DEFINICIÓN 1.4. Sea un código C, la distancia de C, denotada por *d(C)* ó d_c se define:

$$d_c = \min_{x,y \in C, x \neq y} \{d(x,y)\}.$$

DEFINICIÓN 1.5. Sea un código C sobre $\mathbb{Z}_q^n$, el peso de C se define por

$$P(C) = \min_{x \in C, x \neq 0} \{d(x,0)\}.$$

EJEMPLO 1.3. Sea $C = \{0000, 0101, 0011, 0110, 1100, 1110\}$. Veamos que la menor de las distancias entre dos palabras cualesquiera de C es 1:

$$d(x, y) \geq 1 \quad \text{para} \quad x \neq y,\ x,\ y\ \in C.$$

Como $d(1100, 1110) = 1$, tenemos que $d_C = 1$. Sin embargo, $P(C) = 2$.

OBSERVACIÓN 3. Notamos que en el ejemplo anterior $d_C \neq P(C)$, y con esto nos preguntamos, ¿en algún caso se cumple que el *peso* es igual a la *distancia* del código? En la siguiente sección estudiaremos los códigos lineales, y con ello encontraremos una condición suficiente para que se de la igualdad de estos dos parámetros.

1.2. Códigos Lineales. Ahora bien, el siguiente paso sería dotar a nuestro alfabeto de cierta estructura algebráica. Para esto es común considerar a $\mathcal{A} = \mathbb{F}_q$ como un cuerpo finito, valiéndose, de esta forma, de la estructura de espacio vectorial que nos brinda el cuerpo (recordemos que $q = p^r$, para algún pirmo p y $r \in \mathbb{R}$, y si $q = p$, tenemos que $\mathcal{A} = \mathbb{Z}_p$, el cuerpo de enteros módulo p). Así, el conjunto de n-cadenas $\mathcal{A}^n$ es un espacio vectorial sobre $\mathbb{F}_q$ de dimensión n.

Definición 1.6. Un código lineal *q-ario* de longitud n y rango k es un subespacio de $\mathbb{F}_q^n$ de dimensión k. Si d es la distancia de C, decimos que C es un *[n,k,d]-código* con q^k elementos.

Ejemplo 1.4. El código *C1*$= \{000, 011, 101, 110\} \subset \mathbb{Z}_2^3$ es lineal.

Ejemplo 1.5. El código *C2*$= \{001, 010, 012, 021, 100, 101, 120, 221, 222\}$ no es lineal, ya que $010 + 012 = 022 \notin$ *C2*.

Proposición 1.1. *Sea* C *un código lineal. Entonces,*

$$d(C) = P(C).$$

Demostración. Como C es lineal,

$$\begin{aligned} d(C) = P(C) &= \min_{x,y \in C, x \neq y} \{d(x, y)\} \\ &= \min_{x,y \in C, x \neq y} \{P(x - y)\} \\ &= \min_{x \in C, x \neq 0} \{P(x)\} = P(C) \end{aligned}$$

□

Ejemplo 1.6. El código *C1*$= \{000, 011, 101, 110\} \subset \mathbb{Z}_2^3$ es lineal, y por lo tanto,

$$d(C1) = P(C1) = 2.$$

Ejemplo 1.7. El código *C2*$= \{011, 022, 100\} \subset \mathbb{Z}_3^3$ no es lineal. En este caso,

$$d(C2) = 2 \neq P(C2) = 1.$$

DEFINICIÓN 1.7. Sean C y C' códigos lineales. Decimos que C y C' son códigos equivalentes si uno se puede obtener del otro por combinación de operaciones del tipo:

(1) Permutación de posiciones.
(2) Permutación de símbolos en una posición fija.

EJEMPLO 1.8. Los códigos

$$C=\{0000, 0101, 1010, 1111\}$$

y

$$C'=\{0000, 0011, 1100, 1111\}$$

son equivalentes, ya que si permutamos las posiciones 2 y 3 en C, obtenemos C'.

1.3. Matriz Generadora Y Matriz de Paridad. Una vez establecida la estructura de espacio vectorial en nuestro conjunto de n-cadenas, $\mathcal{A}^n$, lo más natural es preguntarse cómo se pueden aprovechar los conceptos elementales de álgebra lineal en este campo. Específicamente, consideremos dos de los conceptos más básicos: el de *Base* y el de *Núcleo* para conseguir nuevas formas de generar y estudiar los códigos lineales.

DEFINICIÓN 1.8. Sea L un $[n, k]_q$-código. Una matriz generadora de L es una matriz $G \in M_{k\times n}(\mathbb{F}_q)$ cuyas filas forman una base de L.

EJEMPLO 1.9. Sea $G= \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix} \in M_{3\times 2}(\mathbb{Z}_2)$. G tiene rango 2 y genera un código L con parámetros $[3, 2]_2$. Además,

$$\begin{pmatrix} x_1 & x_2 \end{pmatrix} \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix} = \begin{pmatrix} x_1, & x_1 + x_2, & x_2 \end{pmatrix}$$

con lo que las palabras código 00, 01, 10, 11 son transformadas en 000, 011, 110, 101 respectivamente. Luego, $L= \{000, 011, 110, 101\}$.

OBSERVACIÓN 4. Si G genera a L, entonces toda matriz equivalente, en la que se realizan operaciones elementales por filas, es también una matriz generadora del mismo código (Sólo se está cambiando la base de L).

DEFINICIÓN 1.9. Un $[n,k]$-código q-ario es sistemático si existen k coordenadas $i_1 \dots i_k$ tal que al restringir las palabras código a estas coordenadas se obtienen todas las q^k palabras de longitud k.

EJEMPLO 1.10. El código L anterior es sistemático en cualquier par de coordenadas.

DEFINICIÓN 1.10. El proceso de agregar una o más coordenadas a las palabras de un código se conoce como *extensión del código*. La forma más común de hacerlo es agregando un dígito de chequeo de paridad total. Si C es un (c, M, d)-código, el *código extendido* $\hat{C}$ se define como:

$$\hat{C} = \{c_1 c_2 \cdots c_n c_{n+1} : c_1 c_2 \cdots c_n \in C \quad \text{y} \quad \sum_{k=1}^{n+1} c_k = 0\}.$$

Así, $\hat{C}$ es un $(\hat{n}, \hat{M}, \hat{d})$-código con

$$\hat{n} = n + 1, \qquad \hat{M} = M, \qquad \hat{d} = d \text{ ó } d + 1.$$

OBSERVACIÓN 5. Si bien el código extendido no mejora las cualidades para corregir errores, sí mejora la capacidad para detectar errores.

EJEMPLO 1.11. Sea $C = \{00, 01, 10, 11\}$. Extendiendo, obtenemos el código

$$\hat{C} = \{000, 011, 101, 110\}.$$

Nótese que $d = 1$, pero $\hat{d} = 2$. Sin embargo, si extendemos nuevamente para obtener el código $\hat{\hat{C}} = \{0000, 0110, 1010, 1100\}$, tenemos que $\hat{\hat{d}} = \hat{d} = 2$.

DEFINICIÓN 1.11. El proceso opuesto a extender un código se denomina *pinchado de un código*. Aquí, una o más coordenadas son extraidas de las palabras códigos. Si C es un (n, M, d)-código, con $d \geq 2$, entonces l *código pinchado* $\dot{C}$ (ó C^*), obtenido pinchado una de las coordenadas de C, tiene parámetros

$$\dot{n} = n - 1, \qquad \dot{M} = M, \qquad \dot{d} = d \text{ ó } d - 1.$$

PROPOSICIÓN 1.2. *Todo [n,k]-código lineal L es sistemático en k coordenadas.*

Demostración. Es directo, pues basta con reducir la matriz generadora de L a una matriz que sea de la forma $(\frac{Id_k}{A})$, donde Id_k es la matriz identidad $k \times k$ y A $\in M_{n-k\times k}$ □

Definición 1.12. Sea G una matriz generadora del código lineal L. Se dice que G está en forma estándar si es de la forma $G = (Id_k|A)$, donde Id_k es la matriz identidad $k \times k$ y A $\in M_{k\times n-k}$.

Observación 6. Si G está en forma estándar, entonces L es sistemático en las primeras k coordenadas.

Proposición 1.3. *Todo código lineal L es equivalente a un código L' cuya matriz generadora está en forma estándar.*

Demostración. Si G es la matriz generadora de L. Sabemos que existen matrices invertibles, de operaciones elementales E y F, tales que $EGF = (I_k|A)$, donde I_k es la matriz identidad de orden k y A es una matriz de orden $n-k \times k$. También sabemos que podemos elegir F como sólo permutación de pivotes. Entonces EG genera el mismo código C, y por lo tanto CF es una permutación equivalente, con la forma que se busca. □

Observación 7. La matriz generadora, además, es una forma eficiente de almacenar un código. Puesto que de ella se pueden obtener todas las palabras del código, basta con almacenar la matriz, sin necesidad de guardar las palabras una a una.

De lo anterior, tenemos que nuestro código se puede ver como la imagen de una matriz, G. Veremos más adelante que el mismo código se puede ver como el kernel de otra matriz.

Definición 1.13. El espacio vectorial $\mathbb{F}_q^n$ tiene un producto interno dado por

$$x \cdot y = x_1y_1 + \ldots + x_ny_n \quad x, y \in \mathbb{F}_q^n.$$

Definición 1.14. Si L es un $[n,k]_q$-código, el código dual de L es el conjunto

$$L^\perp = \{x \in \mathbb{F}_q^n | x \cdot c = 0 \forall c \in L\}.$$

Ejemplo 1.12. Consideremos el código L del ejemplo 1.6. Calculemos su dual. Sea x=$(x_1x_2x_3) \in \mathbb{Z}_2^3$. Luego, para que x pertenezca a $L^\perp$, debe suceder que

$$\begin{pmatrix} x_1 & x_2 & x_3 \end{pmatrix} \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 \end{pmatrix}$$

De aquí obtenemos el sistema

$$\begin{cases} x_1 + x_2 = 0 \\ x_1 + x_3 = 0 \\ x_2 + x_3 = 0 \end{cases} \Longrightarrow \begin{cases} x_1 = -x_3 \\ x_2 = -x_3 \end{cases} \Longrightarrow x = (111)$$

Por lo tanto, $L^{\perp} = \{111\}$.

TEOREMA 1.1. *Sea L un $[n,k]_q$-código.*

(1) *Si G es una matriz generadora de L, entonces*

$$L^{\perp} = \{x \in \mathbb{F}_q^n | G^{\perp} x^{\perp} = 0\} = \{x \in \mathbb{F}_q^n | xG = 0\}.$$

(2) *$L^{\perp}$ es un $[n, n-k]_q$-código.*

(3) *$L^{\perp\perp} = L$.*

DEMOSTRACIÓN.

(1) Por definición, $x \in L^{\perp} \Leftrightarrow x \cdot c = 0, \forall c \in L$.

$$0 = x \cdot c = xc^{\perp} = x(Gu^{\perp}) = (xG)u^{\perp} \text{ para algún } u \in \mathbb{F}_q^k.$$

Si *xG=0*, entonces x $\in L^{\perp}$. Reciprocamente, si x $\in L^{\perp}$ entonces $(xG)u^{\perp} = 0$ para todo u $\in \mathbb{F}_q^k$. En particular, se cumple que para los vectores de la base canónica, $u = e_1, e_2, \ldots, e_k$, $0 = (G^{\perp}x^{\perp})e_i^{\perp} = (xG)^i para 1 \leq i \leq k$. Por lo tanto, xG=0.

(2) Es claro que $L^{\perp}$ es subespacio de $\mathbb{F}_q^n$. Por el item anterior, $L^{\perp} = \{x \in \mathbb{F}_q^n : xg = 0\}$. Esto es, $L^{\perp}$ es el espacio solución de k ecuaciones con n incógnitas. Luego, como G tiene rango k, hay n-k variables libres. Por lo tanto $\dim L^{\perp} = n - k$.

(3) Se tiene que

$$L \subset (L^{\perp})^{\perp} = \{e \in \mathbb{F}_q^n : x \cdot c' = 0\} \ \forall c' \in L^{\perp}.$$

Pero

$$\dim(L^{\perp})^{\perp} = n - (n-k) = k = \dim L,$$

Por lo tanto, $L = (L^{\perp})^{\perp}$.

□

Hemos encontrado otra forma de generar nuestro código, o, al menos, de diferenciar cuáles palabras del alfabeto pertenecen al código y cuáles no, de una forma directa y (computacionalmente) sencilla.

DEFINICIÓN 1.15. Sea L un $[n,k]_q$-código. Una matriz H se dice matriz de paridad de L si es una matriz generadora de $L^\perp$.

OBSERVACIÓN 8. Se cumple que,

(1) La matriz H siempre existe y $H \in M_{n-k\times n}(\mathbb{F}_q)$.
(2) $HG = 0$. En efecto, sean c $\in L$ y c' $\in L^\perp$, entonces $c = Gu^\perp$ y c'=$Hw^\perp$ para ciertos u $\in \mathbb{F}_q^k$, w $\in \mathbb{F}_q^{n-k}$. Luego, $c \cdot c' = 0$ si y sólo si se cumple
$$0 = Hw^\perp \cdot Gu^\perp = wHGu^\perp,$$
lo que a su vez sucede si y sólo si HG=0, ya que $e_i(HG)e_j^\perp = (HG)_{ij}$.
(3) Si $H \in M_{n-k\times n}(\mathbb{F}_q)$ y $HG = 0$, entonces H genera a $L^\perp$. Por lo tanto, H es matriz de paridad de L.
(4) Si G es una matriz generadora de L, entonces G es una matriz de paridad de $L^\perp$.

PROPOSICIÓN 1.4. *Sea H la matriz de paridad de un $[n,k]q$-código L. Entonces,*
$$L=\{x \in \mathbb{F}_q^n | Hx^\perp = 0\} = \{x \in \mathbb{F}_q^n | xH^\perp = 0\}.$$

DEMOSTRACIÓN. Si c $\in L$ entonces $c = Gx^\perp$, donde $x \in \mathbb{F}_q^k$ y G es la matriz generadora de L. Luego $Hc^\perp = HGx^\perp = 0$ y por lo tanto L $\subset S_H = \{x \in \mathbb{F}_q^n : Hx^\perp = 0\}$, el espacio solución de un sistema de n-k ecuaciones con n incógnitas y rango n-k. Como $\dim S_H = n - (n-k) = k = \dim L$, tenemos que $L = \{x \in \mathbb{F}_q^n : Hx^\perp = 0\}$. □

OBSERVACIÓN 9. Para obtener H a partir del código L, se buscan las ecuaciones implícitas del código. Una forma rápida es utilizar la matriz generadora G. Si $G = (\frac{Id_k}{A})$ está en forma estándar, entonces $H = (-A|I_{n-k})$ es una matriz de paridad de L.

TEOREMA 1.2. *Sea L un $[n,k,d]_q$-código y H una matriz de paridad de L. Entonces H tiene d columnas linealmente dependientes, pero cualquier conjunto de d-1 de conlumnas es linealmente independiente.*

DEMOSTRACIÓN. Sean $H_1, \ldots, H_n$ las columnas de H.

$$c \in L \subset \mathbb{F}_q^n \Leftrightarrow Hc^\perp = 0$$
$$\Leftrightarrow H^1c_1 + \cdots + H^nc_n = 0.$$

Pero si $c \in L$, c tiene peso r si y sólo si hay un conjunto de r columnas linealmente dependiente en H. Como $r \geq d$ no puede haber $d-1$ columnas linealmente dependientes en H. □

Observación 10. Este teorema se puede usar para construir códigos lineales con distancia d prefijada.

Con esto complementamos la idea básica que veniamos desarrollando implicitamente: los códigos lineales se pueden ver a través de transformaciones lineales. Es decir, tener un código lineal q-ário de longitud n y rango k es equivalente a tener una sucesión de la forma

$$0 \to \mathbb{F}_q^k \xrightarrow{R_G} \mathbb{F}_q^n \xrightarrow{R_{H^\perp}} \mathbb{F}_q^{n-k} \to 0$$

donde G y H son matrices $k \times n$ y $n-k \times n$ respectivamente.

1.4. Esferas y códigos Perfectos. Hemos dotado a nuestro alfabeto de una estructura algebráica. Seguidamente, pasamos a generar nuestro código a través de transformaciones lineales. Entonces, ¿estos códigos que generamos a través de transformacioens lineales tienen alguna estructura? ¿cómo se comportan las palabras código entre ellas?
La respuesta a esta pregunta no es trivial. Sin embargo, para solventar de alguna forma esta inquietud, podemos enfocarnos en un tipo de códigos en específico: Los códigos perfectos. Estos son un género muy especial de códigos, que *aprovechan* completamente el espacio en el que viven. Existe una gran simetría en la distribución de sus palabras, y poseen algunas otras cualidades distintivas que desarrollaremos a lo largo de esta sección.

Empecemos con algunas definiciones previas.

Definición 1.16. Dado x $\in \mathcal{A}^n$ con $\|\mathcal{A}\| = q$ y r ≥ 0, se define la esfera de radio r, centrada en x como

$$S_q(x,r) = \{y \in \mathcal{A}^n : d(x,y) = r\}$$

y la bola de radio r centrada en x como

$$B_q(x,r) = \{y \in \mathcal{A}^n : d(x,y) \leq r\} = \bigcup_{i=0}^{r} S_q(x,i)$$

Definición 1.17. El volumen, $V_q(n,r)$ es el cardinal de cualquier bola de radio r en $\mathcal{A}^n$. Luego,

$$V_q(n,r) = |B_q(x,r)| = \sum_{i=0}^{r} |S_q(x,r)| = \sum_{i=0}^{r} \tbinom{n}{i}(q-1)^i.$$

LEMA 1.1. *Si C es un código con distancia mínima $d_C = 2t+1$ ó $d_C = 2t+2$, entonces*

$$B_q(c,t) \bigcap B_q(c',t) = \emptyset \qquad \forall c, c' \in C, c \neq c'.$$

DEMOSTRACIÓN. Sea $x \in B_q(c,t)$ con $c \in C$. Entonces, $x \notin B_q(c',t)$ para todo $c' \in C$, con $c \neq c'$. De no ser así, por desigualdad triangular, se tendría:

$$d(c,c') \leq d(c,x) + d(x,c') \leq t + t = 2t < 2t+1 = d_C,$$

lo cual es absurdo. □

PROPOSICIÓN 1.5. *(Cota de Hamming). Si C es un $(n,M,d)_q$-código con $d = 2t$ ó $d = 2t+2$, entonces:*

$$M \cdot \sum_{i=0}^{t} \binom{n}{i}(q-1)^i \leq q^n.$$

DEMOSTRACIÓN. Como las bolas de radio $t = \lfloor \frac{d-1}{2} \rfloor$ son disjuntas, y cada bola $B_q(c,t)$, con $c \in C$ contiene $V_q(n,t)$ palabras de $\mathcal{A}^n$, el resultado es directo. □

EJEMPLO 1.13. La cota de Hamming da una cota superior para el tamaño M que un código de longitud n y distancia d puede tener. Por ejemplo, si C es un $[6,k,3]$-código binario, entonces, como C es lineal y $t = 1$,

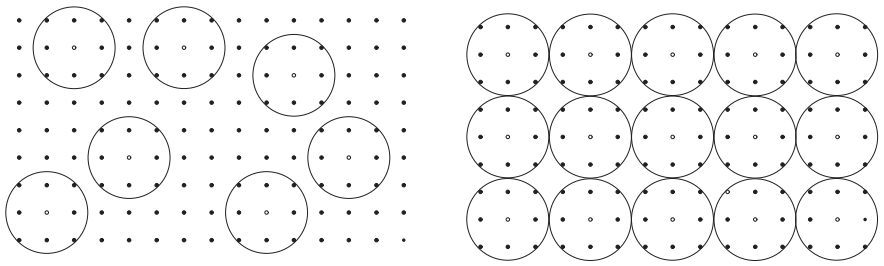
$$M = 2^k \leq \frac{2^6}{1 + \binom{6}{1}} = \frac{64}{7} < 10.$$

Por lo tanto, $k \leq 3$.Es decir, no existen códigos lineales con parámetros $[6,k,3]$ y $4 \leq k \leq 6$.

DEFINICIÓN 1.18. Un código $C \subset \mathcal{A}^n$ se dice perfecto si existe un r tal que las bolas de radio r centradas en las palabras códigos son todas disjuntas entre sí y cubren todo el espacio, es decir,

$$\mathcal{A}^n = \bigcup_{c \in C} B_q(c,r).$$

Esta definición formaliza el comentario hecho enel texto introductorio de esta sección. Los códigos perfectos *abarcan* todo el espacio que los contiene. Veamos la siguiente gráfica para ilustrar el principio:



A la izquiera observamos la distribución no homogénea de las palabras de un código no-perfecto, mientras que a la derecha observamos como las palabras de un código perfecto se distribuyen de manera uniforme y las bolas que las contienen cubren todo el espacio.

TEOREMA 1.3. *(Condición de empaquetamiento) Sea C un $(n, M, d)_q$-código. Entonces C es perfecto si y sólo si $d = 2t + 1$ y*

$$M \cdot \sum_{k=0}^{t} \binom{n}{k} (q-1)^k = q^n.$$

DEMOSTRACIÓN. Supongamos que C es perfecto. Si $d = 2t+2$ es par, las esferas de radio $r \leq t$ son disjuntas pero no cubren $\mathcal{A}^n$. Por otra parte, las esferas de radio $r = t + 1$ cubren $\mathcal{A}^n$ pero no son disjuntas. Luego $d = 2t + 1$ es impar. Las esferas de radio t son disjuntas y cubren $\mathcal{A}^n$, luego se cumple la igualdad en la cota de Hamming y, por lo tanto, vale la igualdad que se quiere probar. Recíprocamente, si vale la igualdad en la cota de Hamming para un $(n, M, 2t + 1)$-código, como las esferas de radio t son disjuntas y cubren todo $\mathcal{A}^n$, entonces C es perfecto.

□

OBSERVACIÓN 11. El teorema dice que un código es perfecto si y sólo si la distancia es impar y se alcanza la igualdad en la cota de Hamming.

OBSERVACIÓN 12. La existencia de números n, M y t que satisfagan la igualdad no implica la existencia de un código perfecto con parámetros $(n, M, 2t + 1)$.

EJEMPLO 1.14. Aquí algunas familias de parámetros que satisfacen la condición de empaquetamiento:

- $(n, q^n, 1)$ - Códigos $\mathcal{A}^n$ tales que $|\mathcal{A}| = q$.
- $(n, 1, 2n + 1)$ - Códigos perfectos triviales.

- $(2m+1, 2, 2m+1)$ - Códigos de repetición $Rep_2(2m+1)$.
- $(\frac{q^r-1}{q-1}, q^{n-r}, 3)$ - Hamming - Golay pinchados.
- $(23, 2^{11}, 7)$ - Hamming - Golay pinchados.
- $(11, 3^6, 5)$ - Hamming - Golay pinchados.

Observación 13. Existen códigos perfectos no lineales. Más adelante se mostrará cómo construir códigos no-lineales con parámetros de Hamming y, por lo tanto, perfectos.

Observación 14. Por el momento, los códigos perfectos no están clasificados.

2. Álgebra de Polinomios

DEFINICIÓN 1.19. Sea $\mathbb{F}$ un cuerpo. *Un álgebra lineal sobre el cuerpo* $\mathbb{F}$ es un espacio vectorial $\mathcal{A}$ con una operación adicional, llamada multiplicación de vectores, que asocia a cada par de vectores $\alpha, \beta \in \mathcal{A}$ un vector $\alpha\beta \in \mathcal{A}$ llamado *el producto de* α *y* β, de forma que:

(1) La multiplicación es asociativa

$$\alpha(\beta\gamma) = (\alpha\beta)\gamma,$$

(2) la multiplicación es asociativa con respecto a la suma

$$\alpha(\beta + \gamma) = \alpha\beta + \alpha\gamma \quad \text{y} \quad (\alpha + \beta)\gamma = \alpha\gamma + \beta\gamma,$$

(3) para cada escalar $c \in \mathbb{F}$,

$$c(\alpha\beta) = (c\alpha)\beta = \alpha(c\beta).$$

Si existe un elemento $1 \in \mathcal{A}$, tal que $1\alpha = \alpha 1 = \alpha$ para cada $\alpha \in \mathcal{A}$, decimos que $\mathcal{A}$ es un álgebra lineal con identidad sobre $\mathbb{F}$. El álgebra $\mathcal{A}$ es conmutativo si $\alpha\beta = \beta\alpha$ para todo $\alpha,\ \beta \in \mathcal{A}$.

Sea $\mathbb{F}$ un cuerpo y S el conjuntode los enteros no negativos. Es fácil ver que el conjunto

$$\mathbb{F}^{\infty} = \{f \mid f : \mathbb{F} \mapsto S\}$$

es un espacio vectorial. Los vectores en $\mathbb{F}^{\infty}$ son secuencias infinitas $f = (f_0, f_1, \ldots)$ de escalares $f_i \in \mathbb{F}$. Si $g = (g_0, g_1, \ldots)$, $g_i \in \mathbb{F}$, y a, b son escalares en $\mathbb{F}$, $af + bg$ es la secuencia infinita dada por

$$af + bg = (af_0 + bg_0, af_1 + bg_1, \ldots).$$

Definimos el producto en $\mathbb{F}^{\infty}$ asociando a cada par de vectores $f,\ g$ en $\mathbb{F}^{\infty}$ el vector fg dado por

$$(fg)_n = \sum_{i=0}^{n} f_i g_{n-i}, \quad n = 0, 1, 2, \ldots$$

Por lo tanto,

$$fg = (f_0g_0, f_0g_1 + f_1g_0, f_0g_2 + f_1g_1 + f_2g_0)$$

y, como se cumple que

$$(gf)_n = \sum_{i=0}^{n} g_i f_{n-i} = \sum_{i=0}^{n} f_i g_{n-i} = (fg)_n, \text{ para } n = 0, 1, 2, \ldots$$

tenemos que la multiplicación es conmutativa.

Si h también pertenece a $\mathbb{F}^\infty$ entonces,

$$\begin{aligned}
[(fg)h]_n &= \sum_{i=0}^{n} (fg)_i h_{n-i} \\
&= \sum_{i=0}^{n} \left(\sum_{j=0}^{i} f_j g_{i-j} \right) h_{n-i} \\
&= \sum_{i=0}^{n} \sum_{j=0}^{i} f_j g_{i-j} h_{n-i} \\
&= \sum_{j=0}^{n} f_i \sum_{i=0}^{n-j} g_i h_{n-i-j} \\
&= \sum_{j=0}^{n} f_i (gh)_{n-j} = [f(gh)]_n \quad para\ n = 0, 1, 2, \ldots
\end{aligned}$$

Con esto se cumple (1) de la definición anterior. Demostremos (2):

$$\begin{aligned}
[f(g+h)]_n &= \sum_{i=0}^{n} f_i (g+h)_{n-i} \\
&= \sum_{i=0}^{n} f_i (g_{n-i} + h_{n-i}) \\
&= \sum_{i=0}^{n} f_i g_{n-i} + f_i h_{n-i} \\
&= \sum_{i=0}^{n} f_i g_{n-i} + \sum_{i=0}^{n} f_i h_{n-i} = (fg)_n + (fh)_n \quad para\ n = 0, 1, 2, \ldots
\end{aligned}$$

Con lo que obtenemos el resultado. Ahora, si c es un escaalar en $\mathbb{F}$, vemos que 3 se cumple. En efecto,

$$\begin{aligned}
[c(fg)]_n &= c\sum_{i=0}^{n}(f_i g_{n-i} \\
&= \sum_{i=0}^{n} c f_i g_{n-i} = (cf)g_n \\
&= \sum_{i=0}^{n} f_i c g_{n-i} \\
&= f(cg) \quad para\ n = 0, 1, 2, \ldots
\end{aligned}$$

Con esto, y considerando el vector $1 = (1, 0, 0, \ldots)$ como el vector identidad en $\mathbb{F}^\infty$, hemos probado que $\mathbb{F}^\infty$, con la multiplicación definida anteriormente, es un álgebra lineal conmutativa con identidad sobre el cuerpo $\mathbb{F}$.

DEFINICIÓN 1.20. Sea $\mathbb{F}[x]$ el subespacio de $\mathbb{F}^\infty$ generado por $1, x, x^2, \ldots$. Un elemento de $\mathbb{F}[x]$ es llamado *polinomio sobre* $\mathbb{F}$.

DEFINICIÓN 1.21. Sea $f(x) \in \mathbb{F}[x]$. Si $f(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^n$ y $a_n \neq 0$ etonces decimos que el *grado de f*, denotado por $\deg f(x)$, es n.

LEMA 1.2. *El conjunto de todos los polinomios sobre un cuerpo dado* $\mathbb{F}$ *es un álgebra lineal conmutativa con identidad sobre* $\mathbb{F}$.

TEOREMA 1.4. *Algoritmo de la División: Sean* $f(x) = a_n x^n + a_{n-1} x^{n-1} + \ldots + a_0$ *y* $g(x) = b_m x^m + b_{m-1} x^{m-1} + \ldots + b_0$ *dos elementos de* $\mathbb{F}[x]$*, tales que* $\deg f(x) = n$ *y* $\deg g(x) = m \geq 0$*. Entonces, existen* $q(x), r(x) \in \mathbb{F}[x]$*, únicos, tales que:*

$$f(x) = g(x)q(x) + r(x),$$

donde $\deg r(x) = 0$ *ó* $\deg r(x) \leq \deg g(x)$.

DEMOSTRACIÓN. Consideremos el conjunto $S = \{f(x) - g(x)s(x) \mid s(x) \in \mathbb{F}[x]\}$. Si $0 \in S$, entonces existe un $s(x) \in \mathbb{F}[x]$, tal que

$$f(x) - g(x)s(x) = 0 \ \Rightarrow f(x) = g(x)s(x).$$

Tomando $q(x) = s(x)$ y $r(x) = 0$ obtenemos el resultado.

Si $0 \notin S$, sea $r(x)$ un elemento de grado minimal en S. Entonces,

$$f(x) = g(x)q(x) + r(x)$$

para algún $q(x) \in \mathbb{F}[x]$.

Debemos probar que $\deg r(x) \leq m$. Sea

$$r(x) = c_t x^t + c_{t-1}x^{t-1} + \ldots + c_0 \text{ , con } c_t \neq 0,$$

y supongmos que $t \geq m$. Entonces,

$$f(x) - g(x)q(x) - \left(\frac{c_t}{b_m}x^{t-m}\right) g(x) = r(x) - \left(\frac{c_t}{b_m}x^{t-m}\right) g(x), \tag{1.1}$$

donde el elemento a la derecha de la igualdad es de la forma

$$r(x) - (c_t x^t + \text{ elementos de grado menor }),$$

lo que constituye un polinómio de grado menor a t. Sin embargo, el polinomio de la ecuación 1.1 se puede escribir de la forma

$$f(x) - g(x)\left[q(x) + \frac{c_t}{b_m}x^{t-m}\right],$$

por lo tanto, está en S, lo que contradice que $r(x)$ sea de grado minimal en S.

Así $\deg r(x) \leq \deg g(x)$.

Para probar la unicidad, sea

$$f(x) = g(x)q_1(x) + r_1(x) \text{ y } f(x) = g(x)q_2(x) + r_2(x)$$

Restando,

$$g(x)[q_1(x) - q_2(x)] = r_2(x) - r_1(x)$$

Como $\deg r(x) = 0$ ó $\deg r(x) \leq \deg g(x)$, tenemos que la igualdad se alcanza sólo si $q_1(x) - q_2(x) = 0$, lo que implica que $q_1(x) = q_2(x)$, y así, debe pasar que $r_1(x) - r_2(x) = 0$, de donde $r_1(x) = r_2(x)$.

□

3. Grafos

Definición 1.22. Un Grafo H es un par $H = (V(H), E(H))$, donde $V(H)$ es un conjunto no vacío de puntos, llamado *conjunto de vértices* y $E(H) \subset V(H) \times V(H)$ es un conjunto de pares de vértices distintos, llamados lados de H.

Definición 1.23. Sea un grafo $H = (V(H), E(H))$, la cardinalidad de $V(H)$ es llamada *orden de* H y la denotaremos por $|V(H)|$. El número de lados es denotado por $e(H)$.

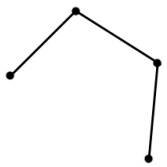
Definición 1.24. El número de lados incidentes sobre un vértice, $x \in V(H)$ se denomina *grado de* x y se denota por $d(x)$. Un grafo se dice *m-regular* si todos sus vértices tienen grado m.

Definición 1.25. Sea $H = (V(H), E(H))$ un grafo con $|V(H)| = n$ y sean $x,\ y \in V(H)$.

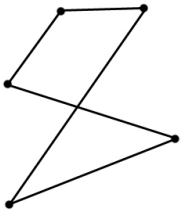
- Un *camino de* x *a* y es una secuencia alternada de vértices y lados que empieza en x y termina en y, donde los vértices no se repiten; todo lado es incidente al vértice que le precede, y al cértice que le sigue.

- Un *ciclo* es un camino que empieza y termina en x.

Ejemplo 1.15. Camino

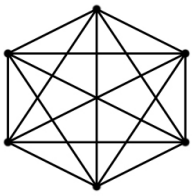


Ejemplo 1.16. Ciclo

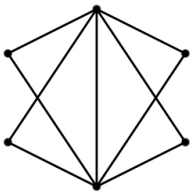


DEFINICIÓN 1.26. Un grafo $H = (V(H), E(H))$ se dice *completo* si cada par de vértices está conectado por un lado. El grafo completo con m vértices se denota por K_m

EJEMPLO 1.17. Completo

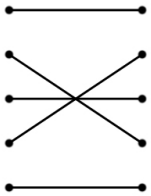


EJEMPLO 1.18. No completo:



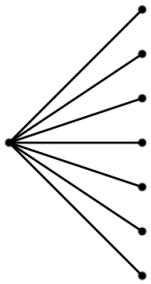
Definición 1.27. Un *matching* es un grafo tal que sus lados no tienen vértices en común. Un matching es perfecto si contiene todos los lados del grafo.

Ejemplo 1.19. Matching



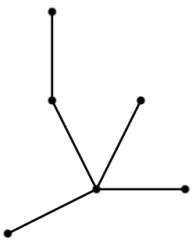
Definición 1.28. Se dice que un grafo $H = (V(H), E(H))$ es una *estrella* (o *m-estrella*) si consta de m lados, y uno de sus vértices, x, se relaciona con los vértices restantes, sin que estos se relacionen entre si.

Ejemplo 1.20. Estrella



Definición 1.29. un grafo H se dice que es un *arbol* si no tiene ciclos y conecta todos los vértices. En un grafo con n vértces, un árbol tiene $n - 1$ lados.

Ejemplo 1.21. Árbol



Capítulo 2

Códigos Lineales

1. Códigos de Hamming

Como hemos visto en el capítulo anterior, existen códigos *especiales*, que satisfacen propiedades algebráicas interesantes, y además poseen ciertas cualidades geométricas que benefician su empaquetamiento. Estos son los *códigos perfectos*. Dentro de esta categoría encontramos algunos códigos clásicos, como lo son los *Códigos de Hamming* y los *Códigos de Golay*. Ambos, aunque definidos hace más de medio siglo, siguen siendo usados comunmente al día de hoy.

Los códigos de Hamming fueron descubiertos independientemente por Marcel Golay (1949) y por Richard Hamming (1950). Antes de definirlos, consideremos la matriz H, cuyas columnas son las $2^3 - 1 = 7$ palabras no nulas de $\mathbb{F}_2^3$:

$$H = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix} \in Mat_{3\times 7}(\mathbb{F}_2)$$

Ahora, pensemos H como la matriz de paridad de un código lineal C con parámetros $[7, 4, d]$. Usando resultados conocidos, podemos calcular la distancia mínima de C. Como cualquier par de columnas de H son linealmente independientes, pero H tiene 3 columnas liealmente dependientes, entonces C tiene una distancia $d = 3$. Resolviendo las ecuaciones de paridad determinadas por H conseguimos las $2^4 = 16$ palabras código de C:

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

Obtenemos una base de C tomando a x_3, x_5, x_6 *y* x_7 como variables libres. Es decir, $c_1 = 1110000,\ c_2 = 1101100,\ c_3 = 0101010$ *y* $c_4 = 11010001$. Luego, el código C se obtiene haciendo todas las sumas posibles entre estas cuatro palabras. Así, hemos obtenido el código de Hamming binario de longitud 7, denotado por $\mathcal{H}_2(3)$.

Podemos hacer la construcción anterior para cualquier $r \geq 2$. Esto es, si dormamos la matriz H cuyas columndas son las $2^r - 1$ palabras no nulas de $\mathbb{F}_2^r$, tenemos una matriz $r \times n$, con $n = 2^r - 1$, en donde cualquier par de columnas son linealmente independientes, pero hay 3 columndas linealmente dependientes. Luego, H es la mariz de paridad e un código lineal denotado por $\mathcal{H}_2(r)$ con parámetros $n = 2^r - 1,\ k = n - r,\ d = 3$. Este es el llamado *código de Hamming binario de orden r.*

Ejemplo 2.1. $\mathcal{H}_2(4)$ tiene parámetros $[15, 11, 3]$, luego codifica $2^{11} = 2048$ mensajes y corrige 1 error.

Observación 15. Utilizando el procedimiento anterior, los códigos de Hamming binarios pueden generalizarse a cualquier alfabeto $\mathbb{F}_q$.Para cada r, queremos construir una matriz $H_{q,r} \in M_{r\times n}(\mathbb{F}_q)$, con el mayor número de columnas, de modo que cualquier par de columnas sea linealmente independiente, pero que algún conjunto de 3 columnas sea linealmente dependiente.

Para cada r fijo, construimos la matriz $H_{q,r}$ de la siguiente manera: Elegimos cualquier columna no nula de $c_1 \in V_1 = \mathbb{F}_q^r$. Luego, elegimos cualquier columna no nula

$$c_2 \in V_2 = V_1 \setminus \{\alpha c_1 : \alpha \in \mathbb{F}_q\}$$

.

Continuamos eligiendo columnas no nulas de esta forma y descartamos los múltiplos escalares de las columnas elegidas hasta agotar todas las columnas de $\mathbb{F}_q^r$. Como cada columna $c \in \mathbb{F}_q^r$ tiene $q-1$ múltiplos escalares no nulos $\alpha c, \alpha \in \mathbb{F}_q$, tenemos que la matriz $H_{q,r}$, formada por las columnas c_i tiene $\frac{(q^r-1)}{(q-1)}$ columnas.

Así, definimos lo siguiente.

DEFINICIÓN 2.1. La matriz $H_{q,r} \in Mat_{r\times n}(\mathbb{F}_q)$, con $n = \frac{(q^r-1)}{(q-1)}$, se llama *matriz de Hamming de orden r* y es la matriz de paridad de un código lineal q-ario con parámetros

$$n = \frac{(q^r-1)}{(q-1)}, \quad k = n - r, \quad d = 3$$

denotado por $\mathcal{H}_q(r)$ y llamado *Código de Hamming q-ario de orden r.*

OBSERVACIÓN 16. Los códigos de Hamming satisfacen la igualdad en la cota de Hamming. En efecto,

$$q^k \sum_{i=0}^{t1} \binom{n}{i} (q-1)^i = q^k(1 + n(q-1)) = q^{k+r} = q^n$$

.

Por lo tanto, los códigos de Hamming son códigos perfectos, y además, corrigen 1 error.

OBSERVACIÓN 17. Una forma de construir una matriz $H_{q,r}$ es tomar todos los vectores de $\mathbb{F}_q^r$ cuya primera coordenada no nula es 1. En efecto, hay $q^r - 1$ vectores no nulos y el primer elemento no nulo puede ser $1, 2, \ldots, q-1$. Luego, hay $\frac{(q^r-1)}{(q-1)}$ vectores cutya primer coordenada no nula comienza con 1.

EJEMPLO 2.2. El código $\mathcal{H}_3(3)$ tiene parámetros $[13, 10, 3]$ con matriz de paridad

$$H_{3,3} = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 2 & 2 & 2 \\ 1 & 0 & 1 & 2 & 0 & 1 & 2 & 0 & 1 & 2 & 0 & 1 & 2 \end{pmatrix}$$

Observación 18. Notemos que la selección de las columnas no es única y por lo tanto hay muchos códigos de Hamming con los mismos parámetros. Sin embargo, todos los códigos de Hamming de igual tamaño son múltiplo escalar equivalentes. Más aún, todo código lineal con los parámetros de Hamming es múltiplo escalar equivalente a un código de Hamming.

Ejemplo 2.3. Consideremos una variación de $\mathcal{H}_3(3)$, tomando las palabras no nulas con la primera cordenada igual a 2.

$$\hat{H_{3,3}} = \begin{pmatrix} 0 & 0 & 0 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 2 & 2 & 2 \\ 1 & 0 & 1 & 2 & 0 & 1 & 2 & 0 & 1 & 2 & 0 & 1 & 2 \end{pmatrix}$$

Esta es la matriz de paridad de un $[13, 10, 3]$-código de Hamming distinto, pero equivalente al $\mathcal{H}_3(3)$ visto anteriormente.

Observación 19. Hay códigos no lineales con los parámetros de Hamming. Luego, hay códigos perfectos no lineales.

2. Códigos de Golay

Los códigos de Golay fueron introducidos por Marcel Golay en 1949. Los mismos son 4, $\mathcal{G}_{24}$, $\mathcal{G}_{23}$, $\mathcal{G}_{12}$ *y* $\mathcal{G}_{11}$.

El código de Golay $\mathcal{G}_{24}$ es el código lineal binario definido por la matriz generadora

$$G = (Id_{12}|A) \in M_{12\times 24}(\mathbb{F}_2),$$

donde

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

Notamos que A posee una estructura cíclica. Veremos que $\mathcal{G}_{24}$ tiene parámetros $[24, 12, 8]$ y, por lo tanto, corrige 3 errores. Sólo falta ver que $d = 8$.

TEOREMA 2.1. *El código de Golay $\mathcal{G}_{24}$ tiene las siguientes propiedades*

(1) *$\mathcal{G}_{24}$ es autudual, es decir, $\mathcal{G}_{24} = \mathcal{G}_{24}^{\perp}$.*
(2) *$\mathcal{G}_{24}$ está generado por la matriz $G = (A|Id_{12})$.*
(3) *Si $c \in \mathcal{G}_{24}$, entonces $4|\omega(c)$.*
(4) *$\mathcal{G}_{24}$ no tiene palabras de peso 4.*
(5) *$\mathcal{G}_{24}$ es un $[24, 12, 8]$-código.*

DEMOSTRACIÓN. Sea G la matriz generados de $\mathcal{G}_{24}$ mostrada anteriormente.

(1) Como las filas de G son ortogonales, todo par de palabras código en $\mathcal{G}_{24}$ son ortogonales. Luego, $\mathcal{G}_{24} \subset \mathcal{G}_{24}^{\perp}$. Pero como $\dim \mathcal{G}_{24}^{\perp} = \dim \mathcal{G}_{24}$, entonces se cumple que $\mathcal{G}_{24} = \mathcal{G}_{24}^{\perp}$.
(2) Notemos que $A = A^{\perp}$. Luego, por (1), se obtiene el resultado.

(3) El peso de las filas de G es 8 o12, por lo tanto disvisible por 4. Si x e y son filas de G, entonces

$$\omega(x+y) = \omega(x) + \omega(y) - 2\omega(x \cap y).$$

Pero $\omega(x \cap y) \equiv x \cdot y = 0 \ mod(2)$ y, por lo tanto, $x+y$ tiene peso par. Por inducción, el peso de la suma de cualquier número de filas de G es múltiplo de 4.

(4) Usaremos las dos matrices generadoras $G_1 = (Id_{12}|A)$ y $G_2 = (A|Id_{12})$ de $\mathcal{G}_{24}$. Si $c \in \mathcal{G}_{24}$ escribimos $c = c_L c_R$ donde $c_L, c_R \in \mathbb{F}_2^{12}$ son la parte izquierda y derecha de c respectivamente. Supongamos que $\omega(c) = 4$. Cualquier combinación lineal de las filas de G_1 tiene parte izquierda con peso mayor que 1. Usando G_2, la parte derecha tiene peso mayor que 1. Entonces $\omega(c_L) \geq 1$ y, también, $\omega(c_R) \geq 1$. Ahora, si $\omega(c_L) = 1$, entonces c es una fila de G y por lo tanto $\omega(c) \neq 4$. Luego, $\omega(c_L) \geq 2$ y, analogamente, $\omega(c_R) \geq 2$. porlo tanto, la única posibilidad es $\omega(c_L) = \omega(c_R) = 2$. Luego, c es la suma de dos filas x e y de G_1. lo cual es absurdo, pues $\omega(x+y) \neq 4$ para todo par de filas x,y de G_1. Luego, G_{24} no tiene palabras de peso 4.

(5) Por los dos items anteriores tenemos que $\omega_{\mathcal{G}_{24}} \geq 8$, y la fila 2 de G tiene peso 8, luego $d_{\mathcal{G}_{24}} = 8$.

□

Observación 20. Pinchando el código de Golay en la última coordenada se obtiene el *código de Golay pinchado* $\mathcal{G}_{23} = \mathcal{G}_{24}$, con parámetros $[23, 12, 7]$. Por lo tanto, este código es perfecto. Se puede ver que pinchando el código $\mathcal{G}_{24}$ en cualquier coordenada se obtienen códigos equivalentes. Extendiendo el código $\mathcal{G}_{23}$ con un dígito extra de paridad podemos recuperar $\mathcal{G}_{24}$.

El *código ternario de Golay* $\mathcal{G}_{12}$ es el código definido por la matiz generadora

$$G = (Id_6|B) \in M_{6\times 12}(\mathbb{F}_3),$$

donde,

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

Notemos que B es de naturaleza cíclica.

OBSERVACIÓN 21. Al igual que los códigos binarios de Golay, pinchando el código $\mathcal{G}_{12}$ en cualquier coordenada se obtienen los códigos equivalentes, y agregando un dígito extra de paridad a cualquiera de estos se obtiene $\mathcal{G}_{12}$.

3. Códigos Cíclicos

Supongamos que n y q son coprimos. En particular, si $q = 2$, entonces n es impar.

DEFINICIÓN 2.2. Un código lineal $C \subset \mathbb{F}_q^n$ es cíclico si

$$c_0c_1 \cdots c_{n-1} \in C \;\; \Rightarrow \;\; c_{n-1}c_0 \cdots c_{n-2} \in C.$$

Notemos que, por definición, un código lineal C es cíclico si es cerrado por el *desplazamiento cíclico*

$$c_0c_1 \cdots c_{n-1} \mapsto c_{n-1}c_0 \cdots c_{n-2}.$$

En este caso, C es cerrado por todos los desplazamientos cíclicos

$$c_0c_1 \cdots c_{n-1} \mapsto c_k \cdots c_{n-1}c_0 \cdots c_{nk-1}.$$

Si $C \subset \mathbb{F}_q^n$ es un código lineal q-ario, a cada palabra código podemos asignarle un polinomio como sigue:

$$\phi : C \to \mathbb{F}_q[x], \quad c_0c_1 \cdots c_{n-1} \mapsto c_0 + c_1x + \cdots + c_{n-1}x^{n-1}.$$

ϕ es un isomorfimo de espacio vectorial de C sobre $\phi(C)$. Luego, se puede simplemente, pensar las palabras código como polinomios.

El cociente

$$R_n = \frac{\mathbb{F}_q[x]}{\langle x^n - 1\rangle}$$

es el álgebra de polinomios de grado menor que n, con la suma usual de polinomios y el producto de polinomios seguido de reducción módulo $x^n - 1$.

OBSERVACIÓN 22. Un código C es cíclico si y sólo si $\phi(C)$ es un ideal en $\mathbb{F}_q[x]/\langle x^n - 1\rangle$. En efecto, si $c_0c_1\cdots c_{n-1} \in C$, entonces

$$\begin{aligned} x(c_0 + c_1x + \cdots + c_{n-1}x^{n-1}) &= c_0x + c_1x^2 + \cdots + c_{n-1}x^n \quad \text{mód } (x^n - 1) \\ &= c_{n-1}x + c_0x + \cdots + c_{n-2}x^{n-1}. \end{aligned}$$

TEOREMA 2.2. *Sea C un ideal de R_n, es decir, un código cíclico de longitud n. Entonces:*

(1) *Existe un único polinomio mónico $g(x)$ de grado mínimo en C. Además, este polinomio genera C, es decir $C = \langle g(x)\rangle$.*
(2) *$g(x)|x^n - 1$.*
(3) *Si $\deg(g(x)) = r$, entonces, C tiene dimensión $n - r$. Más aún,*

$$C = \langle g(x)\rangle = \{r(x)g(x) : deg(g(x)) \leq n - r\}.$$

(4) *Si $g(x) = g_0 + g_1x + \cdots + g_rxr$, entonces $g_0 \neq 0$ y, además, C tiene matíz generadora*

$$G = \begin{pmatrix} g_0 & g_1 & g_2 & \cdots & \cdots & g_r & 0 & 0 & \cdots & 0 \\ 0 & g_0 & g_1 & g_2 & \cdots & \cdots & g_r & 0 & \cdots & 0 \\ 0 & 0 & g_0 & g_1 & g_2 & \cdots & \cdots & g_r & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & \cdots & 0 & g_0 & g_1 & g_2 & \cdots & \cdots & g_r \end{pmatrix}$$

donde cada fila de G es un desplazamiento cíclico de la fila previa.

DEMOSTRACIÓN.

Supongamos que C contiene 2 polinomios mónicos distintos, $g_1(x)$ y $g_2(x)$, de grado mínimo r. Entonces, $g_1(x) - g_2(x)$ es un polinomio no nulo de grado menor que r, lo cual es absurdo. Luego, existe un único polinomio mónico de grado mínimo

r en C. Como $g(x) \in C$ y C es un ideal, tenemos que $\langle g(x)\rangle \subset C$. Por otra parte, supongamos que $p(x) \in C$. Existen $q(x),\ r(x))$ tales que:

$$p(x) = q(x)g(x) + r(x), \qquad 0 \leq \deg(r(x)) \leq r.$$

Luego, como $r(x) = p(x) - q(x)g(x) \in C$, y tiene grado menor que r, necesariamente $r(x) = 0$. Así, $p(x) = q(x)g(x) \in \langle g(x)\rangle\ yC \subset \langle g(x)\rangle$. Por lo tanto, $\langle g(x)\rangle = C$.

(2) Dividiendo $x^n - 1$ por $g(x)$ tenemos:

$$x^n - 1 = q(x)g(x) + r(x),\ \text{con}\ 0 \leq \deg(r(x)) \leq r.$$

Como en R_n se tiene que $x^n - 1 = 0 \in C$, vemos que $r(x) \in C$ y por lo tanto, $r(x) = 0$.

(3) El ideal generado por $g(x)$ es $\langle g(x)\rangle = \{f(x)g(x) : f(x) \in R_n\}$. Queremos ver que basta restringir $f(x)$ a polinomios de grado menor que $n - r$. Sabemos que $x^n - 1 = h(x)g(x)$ para algún polinomio $h(x)$ de grado $n - r$. Dividiendo, $f(x) = q(x)h(x) + r(x),\ \text{con}\ \deg(r(x)) \leq n - r$. Entonces,

$$f(x)g(x) = q(x)h(x)g(x) + r(x)g(x)) = q(x)(x^n - 1) + r(x)g(x),$$

y así, $f(x)g(x) = r(x)g(x)enR_n$, que es lo que se quería ver. También queda demostrado que el conjunto

$$\{g(x), xg(x), \ldots, x^{n-r+1}g(x)\}$$

genera a C, y como es linealmente independiente, forma una base de C. Luego, $\dim C = n - r$.

(4) Si $g_0 = 0$, entonces $g(x) = xg_1(x)$ con $\deg(g_1(x)) \leq r$. Pero entonces tenemos,

$$g_1(x) = 1 \cdot g_1(x) \equiv x^n g_1(x) = x^{n-1}g(x) \in C,$$

lo que es absurdo, pues $g_1 \neq 0$ tiene grado menor que $g(x)$. Porl o tanto $g_0 = 0$. Por último, G es la maatriz generadora de C, pues $\{g(x), xg(x), \ldots x^{n-r+1}g(x)\}$ es una base de C.

□

Observación 23. Un código cíclico puede estar generado por otros polinomios además del polinomio generador.

Hasta el momento, hemos estudiado tres códigos clásicos: los de Hamming, lo de de Golay y los cíclicos. Cada uno se genera de una forma distinta a los demás y cada uno tiene parámetros distintivos. Pero, ¿podemos decir que alguno es *mejor* que los demás? ¿acaso alguno es *más completo* que el resto?

Las respuestas a estas preguntas, tal como uno esperaría, no son cerradas. Cada código tiene sus ventajas y deventajas, desde su generación, hasta sus capacidades de codificación.

En resumen, los códigos de Hamming son lineales y perfectos, al igual que los de Golay. Por otra parte, los códigos cíclicos no son necesariamente perfectos.

Los códigos de Golay son pocos (sólo se han definido 4), mientras que el conjunto de los códigos cíclicos es extenso, al igual que el de los códigos de Hamming (puesto que hemos visto que todo código de Hamming es equivalente a otro con los mismos parámetros).

Por una parte, los códigos cíclicos tienen su base en el álgebra de polinomios, lo que les brinda una estructura bastante flexible. Por otro lado, los códigos de Hamming y Golay estan netamente basados en el álgebra lineal. Especificamente, los de Hámming se pueden generar con matrices relativamente sencillas de construir, lo que facilita la rápida creación de uno de ellos; en cambio, para generar un código de Golay, es necesaria la existencia de matrices específicas, y con estructuras no tan intuitivas.

Todo lo anterior muestra la riqueza y amplitud (valga el uso coloquial de la palabra) algebráica del mundo de los códigos.

Tabla Comparativa de Códigos

Códigos de Hamming	**Códigos de Golay**	**Códigos Cíclicos**
+ Son lineales.	+ Son lineales.	+ Son lineales.
+ Son perfectos.	+ Son únicos.	+ Definición más algebraica.
+ Codifica bastantes palabras con matrices no tan grandes.	+ Codifican más palabras que los de Hamming con matrices no mucho más grandes.	+ Tenemos una expresión explícita para la matriz generadora.
+ La matriz H es fácil de construir.	+ $\mathcal{G}_{24} = \mathcal{G}_{24}^{\perp}$	+ Se generan de forma recursiva.
+ Todos los códigos de haming de igual tamaño son equivalentes.	+ Se puede obtener un código perfecto a partir de uno de Golay (Código Pinchado de Golay).	+ La distancia mínima puede ser grande, lo que permite detectar y corregir más errores.
- Existen códigos con parámetros de Hamming que no son lineales.	- Las matrices generadoras son particulares y poco intuitivas.	- No son perfectos.
- Corrigen un solo error.	- Son pocos.	- Son menos intuitivos.

Capítulo 3

Sistemas de Steiner

1. Estructuras de Incidencia Finita

Una Estructura de Incidencia Finita es una tripla $\mathbb{S} = (\mathcal{P}, \mathcal{B}, \mathcal{I})$, donde $\mathcal{P}$ y $\mathcal{B}$ son conjuntos, e $\mathcal{I} \subset \mathcal{P} \times \mathcal{B}$. A los elementos de de $\mathcal{B}$ se les llama bloques. Si $(p, B) \in \mathcal{I}$, decimos que p es incidente con B, o que B contiene a p.

Definición 3.1. Una estructura de incidencia $\mathcal{D} = (\mathcal{P}, \mathcal{B}, \mathcal{I})$ es un $t - (v, k, \lambda)$ diseño, con $t, v, k, \lambda \in \mathbb{N}$, si

- $|\mathcal{P}| = v$,
- todo bloque contiene exactamente a k puntos,
- todo conjunto de t puntos distintos es incidente con exactamente λ bloques.

Ejemplo 3.1. Sea $\mathcal{P} = X$ un conjunto cualquiera, $\mathcal{B} \in \mathbb{P}(X)$ y definimos la relación de incidencia por

$$(p, B) \ \in \ I \Leftrightarrow \ p \in B.$$

Este par, $(X, \mathbb{P}(X))$, resulta un $t - (v, k, \lambda)$ diseño si $|X| = v, \mathcal{B} \in \mathbb{P}_k(X) = \{B \subset X : |B| = k\}$ y, para todo $T \subset P$, con $|T| = t$ se tiene que $\#\{B \in \mathcal{B} : T \subset B\} = \lambda$.

2. Tipos de Diseños

Básicos Generales:

- Un 1-deseño se suele llamar **configuración táctica**. Nótese que en el caso especial $t = \lambda = 1$, tenemos una partición de $\mathcal{P}$ en un subconjunto de cardinal k. Un 2-diseño con $k < v$ se dice **BIBD** (balanced incomplete block design).
- Si $k = 2$, un $t - (v, 2, \lambda)$ diseño es un grafo $G = (V, E)$ (no dirigido, sin lazos), donde $\mathcal{P} = V$ son los vértices y $\mathcal{B} = E$ son los lados. Si $t \geq 1$, el grafo es conexo

y r-regular, donde r es el número de bloques incidentes con un punto $p \in \mathcal{P}$. Si $t = 2$, entonces $\mathcal{D}$ es el grafo completo K_v.

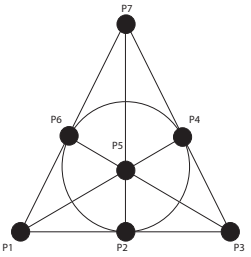
Sistemas de Steiner:

- Un $t-(v,k,1)$ diseño es un **sistema de Steiner**, denotado por $S(t,k,v)$. Es una colección de k-subconjuntos de un v-conjunto tal que todo t-subconjunto está en exactamente un bloque.
- Si $v = 2$ y $k = 3$, tenemos un **sistema triple de Steiner**, $S(2,3,v)$. Es decir, una colección de triplas, tal que todo par de puntos está en exactamente una tripla.

Planos Proyectivos:

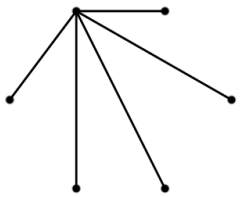
- Un $2-(v,k,1)$ diseño es un **plano proyectivo finito**. Nótese que es tanto un BIBD con $\lambda = 1$, como un sistema de Steiner $S(2,k,v)$. Los k-subconjuntos son las líneas y todo par de puntos está en exactamente una línea.
- Todo $2-(v,k,1)$ diseño es un triple de Steiner $S(2, n+1, n^2+n+1)$ y se lo llama plano proyectivo de orden n.
- Existe un $S(2,3,v)$ si y sólo si $v \equiv 1$ ó $3(\mod 6)$.

Acá ilustramos un plano de Fano (o diagrama de Fano), un cásico ejemplo de sistema de Steiner:

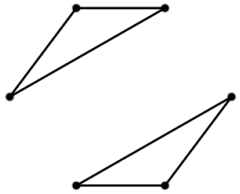


Es un $2-(7,3,1)$ sistema, ya que consta de *siete* bloques, cada bloque (línea) consta de *tres* puntos y por cada *dos* puntos, el tercero está definido por estos dos primeros. Más adelante veremos que este diagrama está fuertemente relacionado con los códigos de Hamming.

EJEMPLO 3.2. Sea $K_6 = (V, E)$ el grafo completo de 6 vértcices mostrado anteriormente. Sea $\mathcal{P}$ el conjunto de los lados de K_6 más un lado adicional, que llamaremos ∞, esto es, $\mathcal{P} = E \cup \{\infty\}$. Los bloques que contienen a ∞ constan de ∞ y los 5 lados de las 5-estrellas



Luego, hay 6 bloques que contienen a ∞. Los bloques que no contienen a ∞ son los subgrafos e K_6 que forman dos triángulos disjuntos, como por ejemplo



De donde, fijando un vértive, hay $\binom{5}{2} = 10$ formas de elegir 2 vértices, de los 5 restantes, para armar un triángulo. Así, hay 10 bloques que no contienen a ∞.

Notemos que dos lados cualesquiera de K_6 están en exactamente dos bloques (la 5-estrella y en los dos triángulos disjuntos). Esto da un $2-(16,6,2)$ diseño.

EJEMPLO 3.3. Sea $\mathcal{P} = \{(i,j) : 1 \leq i,j \leq 4\}$. Para cada $(i,j) \in \mathcal{P}$ definimos el bloque B_{ij} que contiene a los 6 puntos de la fila y columna que pasan por (i,j) distintos (i,j),

$$B_{ij} = \{(i,k) : k \neq j\} \cup \{(k,j) : k \neq i\}.$$

Luego, $|\mathcal{P}| = |\mathcal{B} = 16$. Por ejemplo, $B_{14} = \{(1,1),(2,1),(3,1),(4,2),(4,3),(4,4)\}$. Lo representamos por

$$\begin{array}{cccc} \bullet & \circ & \circ & \circ \\ \bullet & \circ & \circ & \circ \\ \bullet & \circ & \circ & \circ \\ \circ & \bullet & \bullet & \bullet \end{array}$$

Todo par de puntos es incidente con exactamente 2 bloques. Los puntos(i,j) y (k,l) pertenecen a $B_{k,j}$ y $B_{i,l}$. Luego, $(\mathcal{P},\mathcal{B})$ es un $2-(16,6,2)$ diseño. Además, es un $1-(16,6,6)$ diseño.

OBSERVACIÓN 24. Nótese que una construcción similar a la del ejemplo anterior, pero con una matriz $n \times n$, con $n \neq 4$, sólo daría un 1-diseño, pero no un 2-diseño.

TEOREMA 3.1. *Sea $\mathcal{D} = (\mathcal{P}, \mathcal{B})$ un $t-(v,k,\lambda)$ diseño. Para cada s, con $0 \leq s \leq t$, λ_s está dado por*

$$\lambda_s = \lambda \frac{\binom{v-s}{t-s}}{\binom{k-s}{t-s}} = \lambda \frac{(v-s)(v-s-1)\cdots(v-t+1)}{(k-s)(k-s-1)\cdots(k-t+1)}.$$

En particular, $\mathcal{D}$ es un $s-(v,k,\lambda_s)$ diseño para todo $1 \leq s \leq t$.

DEMOSTRACIÓN. Sea A un subconjunto de $\mathcal{P}$, y supongamos que

$$\lambda_s = \#\{B \in \mathcal{B} : A \subset B\}.$$

Veamos que λ_s no depende de A. Contemos el número de pares (D, B), con D un t-subconjunto de $\mathcal{P}$ y $A \subset D \subset B$, son B un bloque.

Hay $\binom{v-s}{t-s}$ formas de elegir $D \supset A$ y hay λ bloques B que contienen a D. Por otro lado, hay λ_s bloques $B \supset A$ y $\binom{k-s}{v-s}$ formas de elegir D.

Luego,

$$\lambda \binom{v-s}{t-s} = \lambda_s \binom{k-s}{t-s}$$

de donde se obtiene que λ_s no depende de s. Además, todo $t-(v,k,\lambda)$ diseño es un $s-(v,k,\lambda_s)$ diseño para todo $0 \leq s \leq t$.

□

COROLARIO 3.1. *Sea $\mathcal{D} = (\mathcal{P}, \mathcal{B})$ un $t-(v,k,\lambda)$ diseño y r el número de bloques que contienen a un punto en $\mathcal{P}$.*

(1) *Todo punto de $\mathcal{P}$ está contenido en* r *bloques con*

$$r = \lambda_1 = \lambda \frac{\binom{v-1}{t-1}}{\binom{k-1}{t-1}}$$

(2) *$\mathcal{D}$ tiene b bloques con*

$$b = \lambda_0 = \lambda \frac{\binom{v}{t}}{\binom{k}{t}}$$

(3) *Para* $0 \leq s \leq t-1$, λ_s *se obtiene recursivamente de*

$$\lambda_s = \frac{(v-s)}{(k-s)}\lambda_{s+1}$$

(4) *Los parámetros satisfacen*

$$rv = bk, \text{ y para } t = 2, r(k-1) = \lambda(v-1)$$

Demostración. (1) y (2) se obtienen tomando $s = 1$ y $s = 0$, respectivamente. Para (3),

$$\lambda = \frac{\binom{k-s}{t-s}}{\binom{v-s}{t-s}}\lambda_s = \frac{\binom{k-(s+1)}{t-(s+1)}}{\binom{v-(s+1)}{t-(s+1)}}\lambda_{s+1}$$

de donde obtenemos que $(k-s)\lambda_s = (v-s)\lambda_{s+1}$. Para (4), por la recursión,

$$b = \lambda_0 = \frac{v-s}{k-s}\lambda_1 = \frac{v-s}{k-s}r.$$

Ahora bien, si $s = 0$, entonces $bk = vr$, y si $s = 1,\ t = 2$, entonces $b(k-1) = (v-1)\lambda$.

□

3. Códigos Perfectos y Diseños

Dados $x, y \in \mathbb{F}_2^n$, *el* ***soporte*** *de* x *es*

$$sopx = \{1 \leq i \leq n : x_1 \neq 0\}$$

y decimos que x ***cubre*** *a* y *si sop* $y \subset$ *sop* x. *Notemos que sop* $0 \subset$ *sop* $x \subset$ *sop* 1 *para todo* x. *Por ejemplo* $x = 011101$ *tiene soporte* $\{2,3,4,6\}$ *y cubre a* $y = 010100$, *que tiene soporte* $\{2,4\}$.

Sea $\mathcal{C}$ *un código binario de longitud* n, S_ω *es el conjunto de palabras de* $\mathcal{C}$ *de peso* ω, *es decir,*

$$S_\omega = \{c \in \mathcal{C} : w(c) = \omega\},\ |S_\omega| = A_\omega.$$

Decimos que S_ω *satisface un* $t-(n,\omega,\lambda)$ *diseño si el conjunto de soportes de palabras en* S_ω,

$$sop\, c : c \in S_\omega$$

es el conjunto $\mathcal{B}$ *de bloques de un* $t-(n,\omega,\lambda)$ *diseño. Es decir, si para cada conjunto* $T \subset \{1,\ldots,n\}$, *con* $|T| = t$ *existen exactamente* λ *palabras código de peso* ω *con unos en todas las coordenadas indexadas por* T.

TEOREMA 3.2. *Sea $\mathcal{C}$ un $(n, M, d = 2t+1)$-código binario perfecto. Entonces, el conjunto de palabras de peso mínimo S_d satisface un $t+1-(n, d, 1)$ diseño, es decir, un sistema de Steiner $S(t+1, d, n)$. En otras palabras, para todo $T = \{j_1, \ldots, j_{t+1}\} \subset \{1, \ldots, n\}$ existe un único $c \in \mathcal{C}$ con $w(c) = d$ y $T \subset sop\ c$.*

DEMOSTRACIÓN. Como $\mathcal{C}$ es perfecto, las colas $B(c, t)$, $c \in \mathcal{C}$, son disjuntas y cubren $\mathbb{F}_2^n$. Sea $x \in \mathbb{F}_2^n$ con $w(x) = t+1$. Luego $x \notin \mathcal{C}$ y $x \in B(c, t)$ para un único $c \in \mathcal{C}$. Luego, $d(x, c) \leq t$, y

$$w(c) \leq d(c, x) + d(x, 0) \leq t + t + 1 = 0$$

de donde, $c \in S_d$.
Como $w(x) = t+1,\ w(c) = 2t+1$ y $d(x, c) \leq t$, tenemos

$$2w(x \cap c) = w(c) + w(c) - d(x, c) \geq 2t + 2.$$

Luego, $w(x \cap c) \geq t+1 = w(x)$, entonces $sop\, c \supset sop\, x \cap c \supset sop\, x$. El resultado se obtiene tomando $T = sop\, x$. □

COROLARIO 3.2. *Sea $\mathcal{C}$ un $(n, M, d = 2t+1)$-código binario perfecto. Entonces,*

$$A_d = \frac{\binom{n}{t+1}}{\binom{d}{t+1}} \in \mathbb{Z}.$$

DEMOSTRACIÓN. Por corolario y teorema anterior, S_d satisface un Sistema de Steiner $S(t+1, d, n)$, es decir, un $t+1-(n, d, 1)$ diseño $\mathcal{D}$. Luego,

$$A_d = |S_d| = \text{número de bloques de } \mathcal{D} = b = \frac{\binom{n}{t+1}}{\binom{d}{t+1}}.$$

□

COROLARIO 3.3. *Sea $\mathcal{C}$ un $(n, M, d = 2t+1)$-código binario perfecto. Entonces,*

$$\lambda_s = \frac{\binom{n-s}{t+1-s}}{\binom{2t+1-s}{t+1-s}} \in \mathbb{Z},\ 1 \leq s \leq t.$$

DEMOSTRACIÓN. Basta sustituir los parámetros del Sistema de Steiner que se muestran en el teorema anterior, en la expresión ya conocida, $\lambda_s = \lambda \frac{\binom{v-s}{t-s}}{\binom{k-s}{t-s}}$. □

TEOREMA 3.3. *Sea $\mathcal{C}$ un código binario perfecto de longitud n y distancia $d = 2t + 1$. Sea $\hat{\mathcal{C}}$ el código extendido de $\mathcal{C}$. Entonces, las palabras codificadas de peso $2t+2$ en $\hat{\mathcal{C}}$ forman un diseño $S(t + 2, 2t + 2, n + 1)$.*

DEMOSTRACIÓN. Queremos probar que todo vector $\bar{u} \in \mathbb{F}_2^{n-1}$ de peso $t + 2$ es cubierto por una única palabra codificada en $\hat{\mathcal{C}}$ de peso $2t + 2$. Supongamos $\bar{c}$ y $\bar{c'}$ dos palabras codificadas de peso $2t + 2$ en $\mathcal{C}$ que cubren al vector $\bar{u} \in \mathbb{F}_2^{n+1}$ de peso $t + 2$. Por propiedad mostrada anteriormente, tenemos que:

$$d(\bar{c}, \bar{u}) = w(\bar{c}) + w(\bar{u}) - 2w(c \bar{*} u) = (2t + 2) + (t + 2) - 2(t + 2) = t.$$

Entonces $\bar{u}$ está contenida en la bola $B(\bar{c}, t)$. Analogamente, $\bar{u}$ estará en $B(\bar{c'}, t)$: como nuestro código es perfecto, con esto se consigue que $\bar{c} = \bar{c'}$ (i.e. la unicidad de la palabra codificada de peso $2t + 2$).

En vista de que $\hat{\mathcal{C}}$ es un código extendido, supongamos que se obtuvo a partir de $\mathcal{C}$, agragando la $(n+1)$-ésima coordenada. Sea $\bar{u} \in \mathbb{F}_2^{n+1}$ un vector de peso $t+2$ con unos en las posiciones $i_1, i_2, \ldots, i_{t+2}$. Sea $\bar{u'}$ el vector que se obtiene al suprimir la última coordenada. Veamos que esta palabra existe en $\mathcal{C}$:
Si $i_{t+2} = n + 1$, entonces $\bar{u'}$ es de peso $t + 1$. Como $\mathcal{C}$ es perfecto,, y corrige t errores, existe una palabra $\bar{c} \in \mathcal{C}$ tal que $d(\bar{u'}, \bar{c}) \leq t$. Así, se tiene que

$$t \geq d(\bar{u'}, c) = w(\bar{c'}) + w(\bar{u'}) - 2w(\bar{u'} * \bar{c}) \geq w(\bar{c}) + (t + 1) - 2w(\bar{u'} * \bar{c}),$$

por lo tanto,

$$2w(\bar{u'} * \bar{c}) \geq w(\bar{c}) + 1 \geq 2t + 2.$$

Luego, se cumplen ambas igualdades y $\bar{c}$ cubre a $\bar{u}$. Además, se obtiene que $w(\bar{c}) = 2t + 2$. Entonces, la palabra extendida, $|\bar{c}|1|$ está en $\mathcal{C}$, cubre a $\bar{u}$ y es de peso $2t + 2$.

Del mismo modo, si $i_{t+2} \leq n + 1$, podemos ver que alguna palabra $\bar{c} \in \mathcal{C}$, de peso $2t + 1$ ó $2t + 2$, cubre a $\bar{u'}$. Al agregarle el chqueo de paridad total (coordenada de extensión), se obtiene el resultado.

□

COROLARIO 3.4. *El número de palabras codificadas de peso 4 en $\hat{\mathcal{H}}_r$ es*

$$\frac{\binom{2^r}{3}}{\binom{4}{3}} = \frac{2^{r-2}(2^r - 1)(2^{r-1} - 1)}{3}.$$

4. Interpretación Geométrica

En la sección anterior definimos los diseños, entre los que mencionamos al plano proyectivo finito. Describamos una relación fascinante entre los códigos perfectos y el Plano de Fano.

Consideremos el siguiente código de Hamming y veamos de qué forma puede ayudarnos el plano de Fano a decodificar e identificar palabras código, corrigiendo hasta un error:

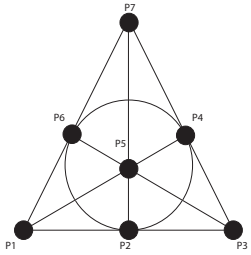
Empezamos con la siguiente matriz de paridad,

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

que nos ayudará a obtener todas las palabras de nuestro código:

$$\mathcal{H}_2(3) = \left\{ \begin{array}{ll} 0000000 & 1111111 \\ 1110000 & 0001111 \\ 1001100 & 0110011 \\ 1000011 & 0111100 \\ 0101010 & 1010101 \\ 0100101 & 1011010 \\ 0011001 & 1100110 \\ 0010110 & 1101001 \end{array} \right\}.$$

Le damos un nombre a cada punto de nuestro plano de fano,



A cada palabra, c, le asignamos un conjunto de puntos, $D(c)$, donde P_i pertenece a $D(c)$, si y sólo si, c tiene un 1 en la i-ésima coordenada.

Nótese que las palabras código constituyen una línea (diremos que una línea es la unión de tres puntos colineales) o el complemento de una línea. Esto contribuye grandiosamente al proceso de correción de errores, puesto que basta con observar la representación geométrica de la palabra obtenida para determinar si pertenece o no al código, y en caso negativo, para conseguir la palabra más próxima.

Veamos qué sucede con cada una de las palabras, según su peso:

Peso cero: No es necesario ningún análisis.

Peso uno: Las palabras de peso *uno* van directamente a la palabra nula.

Peso dos: Para las palabras de peso *dos* adjuntamos el tercer punto colineal para formar una línea.

Peso tres: Toda palabra de peso *tres* forma una línea, o un triángulo. En el caso del triángulo, agregamos el cuarto punto necesario para formar un cuadrilátero.

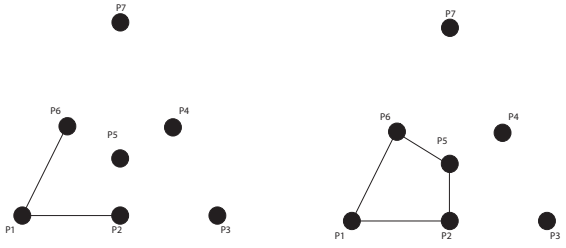
Peso cuatro: Las palabras de peso *cuatro* forman un cuadrilátero (en cuyo caso, ya son palbras código) o son la unión de una línea con un punto aislado. En este caso extraemos el punto aislado.

Peso cinco: El trato con las palabras de peso *cinco* es similar. Primero es necesario notar que una palabra de peso cinco es la unión de dos líneas. Para obtener una palbra código, es necesario quitar el elemento intersección de ambas líneas.

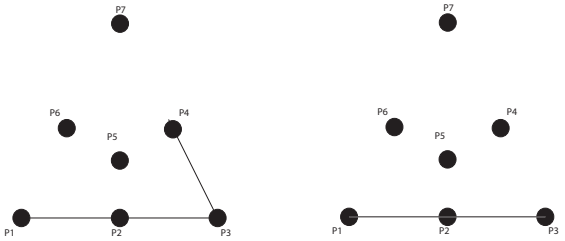
Peso seis: Análogo a las de peso uno, estas palabras van a la palabra con todas las coordenadas iguales a 1.

Peso siete: No es necesario ningún análisis.

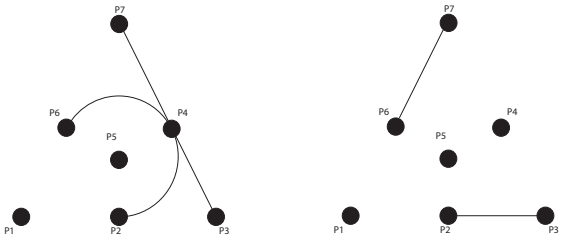
Ejemplo 3.4. Si se recibe la palabra 1100010, se forma un triángulo con los puntos $P_1, P_2,$ y P_6. Completamos el cuadrilatero agregando el punto P_5, para conseguir lapalabra código 1100110.



Ejemplo 3.5. Si recibimos la palabra 1111000, notamos que podemos trazar la línea $\overline{P_1}P_3$ y unirla con P_4. Anulamos P_4 y nuestra palabra viene dada por la línea $\overline{P_1}P_3$. Con esto obtenemos la palabra código 110000.



Ejemplo 3.6. Consideremos la palabra 0111011. Es la unión de las líneas $\overline{P_2P_4P_6}$ y $\overline{P_3P_4P_7}$. Quitando el punto de intersección, P_4, obtenemos la palabra código 0110011, representada por el segmento $\overline{P_2P_3}$, unido con $\overline{P_6P_7}$.



Nótese que el orden de los vectores que constituyen la matriz de paridad, H, determinó las palabras de nuestro código (ya hemos visto que existen múltiples códigos de Hamming equivalentes). A su vez, las palabras de nuestro código determinan el orden de los puntos de nuestro diagrama de Fano. La idea, por supuesto, sería conseguir una forma general de plasmar nuestro código en el diagrama. De lograrse, podríamos trasladar nuestro diagrama a un K_7 (grafo completo de 7 vértices).

En los ejemplos dados podemos percibir rastros de grafos: en el primer ejemplo notamos que la palabra que no pertece al código forma un *camino* de tres puntos, para luego, al ser corregida, transformarse en un *ciclo* de cuatro puntos. En el segundo ejemplo notamos que tenemos un *camino* de cuatro puntos, para luego convertirlo en uno de tres. En el último, obtenemos una *estrella*, para corregirla y convertirla en un *matching*.

Queda claro, entonces, que la relación con el grafo completo está allí, presente, pero sin ser aún genralizada. Este es un problema que se deja abierto, e intenta alentar al lector a seguir la línea de este trabajo.

Conclusiones

La teoría de la información es un área de las matemáticas que aporta cada vez más y más al mundo en el que vivimos, debido al increíble alcance de la tecnología y las comunicaciones. Sabemos que cada día se transmite más información, y por consiguiente, se requieren mejores formas de empaquetar y analizar la data.

Nuestra aproximación a la teoría de códigos nos deja un agradable sabor de boca, al ver que no es un área que corresponde únicamente a temas computacionales, sino que contiene un amplio contenido algebráico. Nos vimos inmersos en diversas estructuras conocidas, como lo son los espacios vectoriales. Notamos que los códigos pueden vivir en espacios que creíamos estaban alejados de la materia, y sin embargo, los albergan sin ningún problema. Por otra parte, también observamos que algunas etructuras algebráicas como los grafos y los sistemas de steiner modelan exactamente un código perfecto.

Vale recalcar que lo estudiado en estas páginas abarca una pequeña porción de lo que es la *codificación* y la *detección de errores.* A la hora de *decodificar* se consediran muchos resultados que escapan de este trabajo. Para abordar el tema de la decodificación, se recomiendan algunos trabajos, como [**1**], [**4**].

Hemos basado el estudio de los códigos en tres parámetros básicos d, M y n, que parecieran tener todos el mismo peso (valga el juego de palabras) a la hora de caracterizar al código. De hecho, la gran mayoría de las propiedades se las adjudicamos a alguna de estas tres variables. Así pues, si estos prámetros son tan vitales para definir y caracterizar un código, cabe entonces la pregunta: ¿hay directrices generales para crear y describir códigos cualesquiera, dado(s) alguno(s) de estos parámetros?

La respuesta, como es de imaginarse, es negativa. Quedan muchos problemas abiertos en el área que involucran a nuestras tres letras clave. Veamos algunos que no tienen solución aún, excepto en casos específicos:

- Dados d y n mayores a 1. ¿Cuál es el menor k, mayor a 1, para el cual existe un código C, de longitud n, dimensión k y distancia mínima d?
- Dados d y n mayores a 1. ¿Cuál es el menor M, mayor a 1, para el cual existe un código C, de tamaño M y distancia mínima d?
- Dados k y n mayores a 1. ¿Cuál es el mayor d , para el cual existe un código lineal C, de longitud n, dimensión k y distancia mínima d?

Como vemos, son problemas que surgen de preguntas naturales, sin alejarnos de los elementos básicos de los códigos, lo que ilsutra la complejidad y riqueza de la materia. Es por esto que se alienta al lector a seguir el estudio de la misma.

¡Continuemos codificando!

Bibliografía

[1] R. Hill. *A First Course in Coding Theory*. Oxford Applied Mathematics and Computing Science Series.1996.

[2] Herstein.*Algebra Moderna*.

[3] K. Hoffman y R. Kunze. *Algebra lineal*. Prentice Hall Inc.

[4] R. Podestá. *Introducción a la Teoría de Códigos Autocorrectores*. Notas del curso dado en el ENAIII, Vaquerias, agosto 2006.

[5] R. Podestá. *Algunos aspectos combinatorios de la Teoría de Códigos*. Notas del curso dado en la Primera Escuela Puntana de Combinatoria, San Luis, julio 2012.

[6] W. A. Schmid and A. Plagne. *An application of coding theory to estimating Davenport constant*. Des. Codes Cryptogr. (2011) 61: 105-118.

[7] N. Aydin. *An Introduction to Coding Theory via Hamming Codes*. A Computational Science Module. Kenyon College. 2007.

[8] P. Fernández. *Códigos Correctores de Errores, Quinielas y Empaquetamientos*.

[9] J. Gutierrez. *Códigos Lineales y Geometrías Finitas*. 1997.

[10] A. Tavera *Geometría Logicamente*. Propiedades de la configuración de Desargues y árboles semánticos. 2004.