UNIVERSIDAD
DE GRANADA

Facultad de Ciencias

E.T.S. de Ingenierías Informática y de Telecomunicación

Doble Grado en Ingeniería Informática y Matemáticas

TRABAJO DE FIN DE GRADO

# Algoritmo de Peterson-Gorenstein-Zierler para códigos cíclicos sesgados

[ 26 de mayo de 2020 a las 20:17 – 0.6 ]

Presentado por

José María Martín Luque

Tutorizado por

Gabriel Navarro Garulo

Curso académico 2019–2020

# RESUMEN

Resumen.

# SUMMARY

Summary.

# ÍNDICE GENERAL


1 PRELIMINARES 11
1.1 Anillos . . . 11
1.2 Cuerpos finitos . . . 14
1.2.1 Anillos de polinomios sobre cuerpos finitos . . . 15
1.2.2 Elementos primitivos . . . 17
1.2.3 Construcción de cuerpos finitos . . . 18
1.2.4 Clases ciclotómicas y polinomios minimales . . . 19
1.3 Automorfismos . . . 20
2 FUNDAMENTOS DE TEORÍA DE CÓDIGOS 21
2.1 Códigos lineales . . . 21
2.1.1 Codificación y decodificación . . . 23
2.1.2 Distancias y pesos . . . 26
2.2 Ejemplos de códigos . . . 28
2.2.1 Códigos de repetición . . . 28
2.2.2 Códigos de control de paridad . . . 28
2.2.3 Códigos de Hamming . . . 29
2.3 Códigos cíclicos . . . 29
2.3.1 Factorización de $x^n - 1$ . . . 30
2.3.2 Construcción de códigos cíclicos . . . 31
2.3.3 Codificación de códigos cíclicos . . . 37
2.4 Idempotentes, multiplicadores y ceros de un código cíclico 38
2.5 Códigos BCH . . . 42
2.6 Algoritmo de Peterson-Gorenstein-Zierler . . . 44
3 ANILLOS DE POLINOMIOS DE ORE 49
A IMPLEMENTACIÓN EN SAGE DEL ALGORITMO PGZ 53

# INTRODUCCIÓN

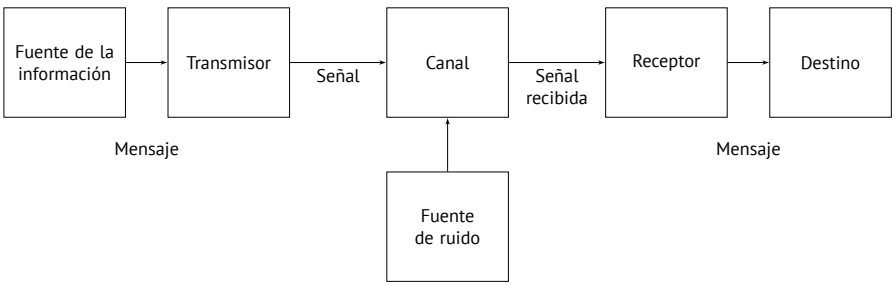
El libro de Claude Shannon *Mathematical Theory of Cryptography* (1945) y su ampliación posterior, *Mathematical Theory of Communication* (1948) dieron a luz a dos disciplinas hoy plenamente establecidas, la teoría de información y la teoría de códigos.

El objetivo principal de la teoría de códigos y de la teoría de la información es el de establecer mecanismos de comunicación que sean eficientes y fiables en ambientes posiblemente hostiles. La eficiencia requiere que la transmisión de la información no necesite de demasiados recursos, sean materiales o temporales. Por otro lado, la fiabilidad requiere que el mensaje recibido en una comunicación sea lo más parecido posible, dentro de unos márgenes de tolerancia, al mensaje original.

La teoría de la información se encarga del estudio tanto de la representación de la información como de la capacidad que tienen los sistemas para transmitir y procesar la información. Por otra parte, la teoría de códigos se basa en los resultados de la teoría de la información para el el diseño y desarrollo de modelos de transmisión de información mediante herramientas algebraicas.

El filósofo inglés Francis Bacon ya afirmó en el año 1623 que únicamente son necesarios dos símbolos para codificar toda la comunicación.

> La transposición de dos letras en cinco emplazamientos bastará para dar 32 diferencias [y] por este arte se abre un camino por el que un hombre puede expresar y señalar las intenciones de su mente, a un lugar situado a cualquier distancia, mediante objetos ... capaces solo de una doble diferencia. (Dyson, 2015, p. 30)




Modelo de comunicación de Shannon

# 1

# PRELIMINARES

La teoría de códigos que vamos a desarrollar en este trabajo se sustenta fundamentalmente en la teoría de cuerpos finitos, de anillos y de polinomios. En este capítulo se detallan algunos conceptos básicos de Álgebra que necesitaremos para poder desarrollar posteriormente la teoría de códigos. Las fuentes principales de este capítulo son (Cohn, 1982, cap. 3 y 6), (Cohn, 1989, cap. 3) y (Lidl & Niederreiter, 1986, cap. 2).

## I.I ANILLOS

Por anillo entendemos un conjunto $R$ junto a dos operaciones binarias: $x+y$, llamada *suma*, y $xy$, llamada *producto*, tales que:

1. $R$ es un grupo abeliano con la suma.
2. $R$ es un monoide con el producto.
3. La suma y el producto están relacionadas mediante la propiedad distributiva:
$$(x+y)z = xz + yz, \qquad x(y+z) = xy + xz.$$

El elemento neutro para la suma se llama *cero* y se escribe 0, mientras que el elemento neutro para el producto se llama *uno* o *la unidad* y se escribe 1. El inverso de la suma para $x$ se denota $-x$.

A continuación vamos a dar una definición completa de anillo, sin depender de remisiones a las definiciones de grupo y monoide.

DEFINICIÓN I.I.I. Un *anillo* es un conjunto junto a dos operaciones: la suma (+) y la multiplicación (·), que verifican las siguientes propiedades.

— Propiedad asociativa:
$$(x+y)+z = x+(y+z), \qquad (xy)z = x(yz).$$

— Propiedad conmutativa para la suma:
$$x+y = y+x.$$

— Existencia de elemento neutro:
$$x+0 = x, \qquad x1 = x.$$

— Existencia de elemento inverso para la suma:

$$x + (-x) = 0.$$

— Propiedad distributiva para la multiplicación sobre la suma:

$$x(y + z) = xy + xz.$$

Si un anillo verifica la propiedad conmutativa para la multiplicación, es decir, $xy = yx$, se dice que es un *anillo conmutativo*.

Comprobamos que en cualquier anillo $R$ se verifica que $0x = x0 = 0$ para todo $x \in R$, ya que $x0 = x(0+0) = x0 + x0$, de donde concluimos que $x0 = 0$ e igualmente, $0x = 0$. Cuando un anillo $R$ tiene solo un elemento es necesario que $1 = 0$. Un anillo de este tipo se denomina *anillo trivial*. Podemos ver que este es el único caso en el que se da la igualdad $1 = 0$. Supongamos que en cualquier otro anillo se da que $1 = 0$. Entonces para cada elemento $x$ del anillo se tiene que $x = x1 = x0 = 0$, luego tiene un único elemento.

Dos anillos son *isomorfos* si hay un *isomorfismo* entre ellos, es decir, existe una biyección que preserva todas las operaciones. Un elemento $a$ de un anillo se dice que es *invertible* si existe un elemento $a'$ en el anillo tal que $aa' = a'a = 1$. A este elemento, que es único, lo llamamos *elemento inverso* de $a$ y lo denotamos por $a^{-1}$. El elemento 0 no puede tener inverso porque ya hemos visto que siempre que se multiplique por él se obtiene el 0. Los anillos en los que todo elemento distinto de 0 es invertible se llaman *anillos de división*.

DEFINICIÓN I.1.2. Un subconjunto $S$ de un anillo $R$ se denomina *subanillo* si $S$ contiene a los elementos neutros de la suma y el producto de $R$, es cerrado bajo dichas operaciones y forma un anillo con ellas.

EJEMPLO I.1.3. Sea $R$ el cuerpo $\mathbb{Q}$ de los racionales. Entonces, el subconjunto $\mathbb{Z}$ de los enteros es un subanillo, pues contiene a los elementos neutros de producto y suma —1 y 0, respectivamente—, la suma de dos enteros es un entero y el producto de dos enteros es, de nuevo, un entero.

DEFINICIÓN I.1.4. Sea $J$ un subconjunto de un anillo $R$.

— Se dice que $J$ es un *ideal por la izquierda* si es un subgrupo aditivo de $R$ y verifica que para todo $j \in J$ y para todo $r \in R$, el elemento $rj \in J$.
— De igual forma, se dice que $J$ es un *ideal por la derecha* si es un subgrupo aditivo de $R$ y verifica que para todo $j \in J$ y para todo $r \in R$, el elemento $jr \in J$.

— Finalmente, se dice que $J$ es un *ideal bilátero* si es ideal tanto por la izquierda como por la derecha.

Notamos que en un anillo conmutativo solo habrá ideales biláteros y por tanto nos referiremos a ellos como ideales a secas. Veamos a continuación un par de ejemplos.

EJEMPLO I.I.5. Continuando el ejemplo 1.1.3, el conjunto $\mathbb{Z}$ de los enteros no es un ideal pues, por ejemplo, $1 \in \mathbb{Z}$, $1/2 \in \mathbb{Q}$, pero $1/2 \cdot 1 = 1/2 \notin \mathbb{Z}$.

EJEMPLO I.I.6. Sea $R$ el anillo de los enteros $\mathbb{Z}$ y consideremos el subconjunto de los números enteros $J = \{2n : n \in \mathbb{Z}\}$. Entonces $J$ es un ideal (bilátero), pues todo producto de un número par es otro número par.

DEFINICIÓN I.I.7. Sea $R$ un anillo. Un ideal por la izquierda $J$ de $R$ se dice que es *principal* si existe un elemento $a \in R$ tal que $J = Ra = \{ra : r \in R\}$. De forma análoga, un ideal por la derecha $J$ de $R$ es *principal* si existe un elemento $a \in R$ tal que $J = aR = \{ar : r \in R\}$.

*Un ideal bilátero será principal si verifica cualquiera de las dos condiciones.*

A continuación vamos a definir algunos otros conceptos relativos a anillos. Decimos que un ideal $J$ es *minimal* si no existe ningún otro ideal entre $\{0\}$ y $J$. Podemos clasificar los elementos de un anillo distintos de cero en dos tipos: *divisores de cero* y *regulares*. Tomemos un elemento $a \neq 0$. Si existe $b \neq 0$ tal que $ab$ o $ba$ es cero, entonces $a$ es un elemento *divisor de cero*, y en caso contrario, un elemento *regular*. A partir de esta clasificación podemos hablar de anillos *enteros* —aquellos que no son triviales y no tienen divisores de cero— y de *dominios de integridad* —anillos enteros conmutativos—.

Una propiedad importante de los elementos regulares —y que es tan natural que estamos plenamente acostumbrados a ella— es la llamada ley de cancelación.

PROPOSICIÓN I.I.8. *Si $c$ es un elemento regular de un anillo $R$ entonces para cada $a, b \in R$, tales que, o bien $ca = cb$, o bien $ac = bc$, se tiene que $a = b$.*

Para cerrar esta sección vamos a introducir dos definiciones: una será una propiedad de los anillos, mientras que la otra será una propiedad que verificarán algunos elementos de los anillos.

DEFINICIÓN I.I.9. Sea $R$ un anillo. La *característica* del anillo es el menor natural $n$ tal que $n1 = 0$. Si no existe tal número, la característica del anillo es 0.

DEFINICIÓN I.I.10. Un elemento $e$ de un anillo tal que $e^2 = e$ se dice que es *idempotente*.

## I.2 CUERPOS FINITOS

En esta sección vamos a introducir el concepto de cuerpo, junto a otros conceptos relevantes, para posteriormente centrarnos en los cuerpos finitos. Estudiaremos además los anillos de polinomios que podemos definir sobre ellos y cómo a partir del cociente de estos podemos construir cuerpos finitos.

DEFINICIÓN I.2.1. Un *cuerpo* es un anillo de división conmutativo. Se dice que un cuerpo es *finito*[1] si tiene un número finito de elementos, al que llamamos *orden* del cuerpo.

Sea $F$ un cuerpo. Un subconjunto $K$ de $F$ que es por sí mismo un cuerpo bajo las operaciones de $F$ se denomina *subcuerpo* de $F$. También podemos referirnos a ello al revés, diciendo que $F$ es una *extensión* de $K$, lo que notaremos como $F/K$. Observamos que $F$ es un espacio vectorial sobre $K$, esto es, los elementos de $F$ pueden ser vistos como vectores sobre el cuerpo de escalares $K$, con las operaciones de suma $\alpha + \beta$, para $\alpha, \beta \in F$ y multiplicación por escalares $a\alpha$, para $a \in K$ y $\alpha \in F$, dadas por las propias operaciones de suma y multiplicación en $K$. Todas las nociones que hemos definido para anillos —como la característica— son válidas para los cuerpos, pues un cuerpo no deja de ser un anillo.

Habitualmente notaremos a los cuerpos finitos por $\mathbb{F}_p$, donde $p$ denota el orden del cuerpo. Trabajaremos habitualmente con cuerpos finitos, pues son los que vamos a utilizar de forma prominente cuando trabajemos con códigos.

TEOREMA I.2.2. *Todo cuerpo $F$ tiene al menos un subcuerpo $P$, el subcuerpo primo de $F$ que está contenido en cada subcuerpo de $F$. O bien $F$ tiene característica $0$ y $P \cong \mathbb{Q}$, o bien $F$ tiene característica $p$, un número primo, y entonces $P \cong \mathbb{F}_p$.*

LEMA I.2.3. *Un espacio vectorial $n$-dimensional sobre $\mathbb{F}_p$ tiene $p^n$ elementos.*

*Demostración.* Sea $V$ un espacio vectorial $n$-dimensional sobre $\mathbb{F}_p$. Si los elementos $u_1, \ldots, u_n$ forman una base de $V$, entonces cada elemento de $V$ se escribe de forma única en la forma $\sum \alpha_i u_i$, donde $\alpha_i \in \mathbb{F}_p$. Como cada coeficiente puede tener hasta $p$ valores distintos, obtenemos un total de $p^n$ elementos. □

Concluimos destacando que en virtud del teorema 1.2.2 todo cuerpo finito $F$ tiene característica $p$ (un primo) y su subcuerpo primo es $\mathbb{F}_p$.

1 Los cuerpos finitos también se suelen conocer como *cuerpos de Galois* en honor a Évariste Galois, uno de los primeros matemáticos en trabajar con ellos.

### 1.2.1 *Anillos de polinomios sobre cuerpos finitos*

Para cualquier anillo $R$ podemos definir un anillo de polinomios en $x$ con coeficientes en $R$, que denotamos $R[x]$ y que está compuesto por el conjunto

$$\{a_0 + a_1x + \cdots + a_nx^n : a_0, a_1, \ldots, a_n \in R\}.$$

Dados $f = a_0 + a_1x + \cdots + a_nx^n$ y $g = b_0 + b_1x + \cdots + b_mx^m$ las operaciones de suma y multiplicación de $R[x]$ (suponiendo $m \leq n$) vienen dadas por:

$$f + g = a_0 + b_0 + (a_1 + b_1)x + \cdots + (a_m + b_m)x^m + \cdots + a_nx^n,$$

$$fg = a_0b_0 + (a_0b_0 + a_1b_0)x + (a_0b_2 + a_1b_1 + a_2b_0)x^2 + \cdots + a_nb_mx^{m+n}.$$

El elemento neutro para la suma es el $0 \in R$, y el del producto, el $1 \in R$. Es fácil comprobar que efectivamente $R[x]$ así definido satisface todas las propiedades de los anillos. Veamos a continuación algunos conceptos básicos sobre polinomios. El *grado* de un polinomio es el mayor grado de cualquier término con coeficiente distinto de cero. El coeficiente del término de mayor grado se denomina *coeficiente líder*. Un polinomio es *mónico* si su coeficiente líder es 1. Sean $f(x)$ y $g(x)$ polinomios en $R[x]$. Decimos que $f(x)$ *divide a* $g(x)$, denotado por $f(x)|g(x)$, si existe un polinomio $h(x) \in \mathbb{R}[x]$ tal que $g(x) = f(x)h(x)$. El polinomio $f(x)$ se llama *divisor* o *factor* de $g(x)$.

Como ya hemos anticipado vamos a centrarnos en el estudio de los anillos de polinomios en cuerpos finitos pues son los que necesitaremos para la teoría de códigos. Siguiendo la notación que hemos establecido, denotaremos el anillo de le los polinomios con coeficientes en $\mathbb{F}_q$ por $\mathbb{F}_q[x]$. Es un anillo conmutativo con las operaciones habituales de suma y multiplicación de polinomios que acabamos de describir. Es, de hecho, un dominio de integridad.

Un polinomio en $\mathbb{F}_q[x]$ viene dado por $f(x) = \sum_{i=0}^{n} a_ix^i$, donde $a_i$ son los coeficientes del término de grado $i$ y pertenecen a $\mathbb{F}_q$. Dados $f(x), g(x) \in \mathbb{F}_q[x]$ el *máximo común divisor* de $f(x)$ y $g(x)$, siendo al menos uno de ellos distinto de cero, es el polinomio mónico de $\mathbb{F}_q[x]$ de mayor grado que divida tanto a $f(x)$ como a $g(x)$. Lo denotamos por $\mathrm{mcd}(f(x), g(x))$. Decimos que dos polinomios son *primos relativos* si su máximo común divisor es 1.

El siguiente resultado nos proporciona la existencia y unicidad de divisores de un polinomio en $\mathbb{F}_q[x]$.

TEOREMA 1.2.4. *Sean $f(x)$ y $g(x)$ polinomios de $\mathbb{F}_q[x]$, siendo $g(x)$ distinto de cero.*

1. *Existen polinomios únicos, $h(x), r(x) \in \mathbb{F}_q[x]$ tales que*

$$f(x) = g(x)h(x) + r(x), \quad \textit{donde } \operatorname{gr} r(x) < \operatorname{gr} g(x) \textit{ o } r(x) = 0.$$

2. *Si* $f(x) = g(x)h(x) + r(x)$, *entonces*

$$\mathrm{mcd}(f(x), g(x)) = \mathrm{mcd}(g(x), r(x)).$$

Podemos utilizar este resultado para hallar el máximo común divisor de dos polinomios. Este procedimiento se conoce como *algoritmo de Euclides* y es muy parecido a su homólogo para números enteros.

TEOREMA I.2.5. (ALGORITMO DE EUCLIDES.) *Sean* $f(x)$ *y* $g(x)$ *dos polinomios en* $\mathbb{F}_q[x]$ *con* $g(x)$ *distinto de cero.*

1. *Realiza los siguientes pasos hasta que* $r_n(x) = 0$ *para algún n:*

$$\begin{aligned}
f(x) &= g(x)h_1(x) + r_1(x), && \text{donde } \operatorname{gr} r_1(x) < \operatorname{gr} g(x),\\
g(x) &= r_1(x)h_2(x) + r_2(x), && \text{donde } \operatorname{gr} r_2(x) < \operatorname{gr} r_1(x),\\
r_1(x) &= r_2(x)h_3(x) + r_3(x), && \text{donde } \operatorname{gr} r_2(x) < \operatorname{gr} r_3(x),\\
&\vdots\\
r_{n-3}(x) &= r_{n-2}(x)h_{n-1}(x) + r_{n-1}(x), && \text{donde } \operatorname{gr} r_{n-1}(x) < \operatorname{gr} r_{n-2}(x),\\
r_{n-2}(x) &= r_{n-1}(x)h_n(x) + r_n(x), && \text{donde } r_n(x) = 0.
\end{aligned}$$

   *Entonces,* $\mathrm{mcd}(f(x), g(x)) = cr_{n-1}(x)$, *donde* $c \in \mathbb{F}_q$ *se escoge para que* $cr_{n-1}(x)$ *sea mónico.*
2. *Existen polinomios* $a(x), b(x) \in \mathbb{F}_q[x]$ *tales que*

$$a(x)f(x) + b(x)g(x) = \mathrm{mcd}(f(x), g(x)).$$

La secuencia de pasos descrita termina porque en cada paso el grado del resto se reduce al menos en 1. A continuación vamos a comentar un par de resultados que nos serán útiles en el futuro.

PROPOSICIÓN I.2.6. *Sean* $f(x)$ *y* $g(x)$ *polinomios en* $\mathbb{F}_q[x]$.

1. *Si* $k(x)$ *es un divisor de* $f(x)$ *y de* $g(x)$, *entonces* $k(x)$ *es divisor del polinomio* $a(x)f(x) + b(x)g(x)$ *para todo* $a(x), b(x) \in \mathbb{F}_q[x]$.
2. *Si* $k(x)$ *es un divisor de* $f(x)$ *y de* $g(x)$ *entonces* $k(x)$ *es divisor del polinomio* $\mathrm{mcd}(f(x), g(x))$.

PROPOSICIÓN I.2.7. *Sea* $f(x)$ *un polinomio en* $\mathbb{F}_q[x]$ *de grado n. Entonces:*

1. *Si* $\alpha \in \mathbb{F}_q$ *es una raíz de* $f(x)$ *entonces* $x - \alpha$ *es un factor de* $f(x)$.
2. *El polinomio* $f(x)$ *tiene al menos n raíces en cualquier cuerpo que contenga a* $\mathbb{F}_q$.

Un polinomio no constante $f(x) \in \mathbb{F}_q[x]$ es *irreducible sobre* $\mathbb{F}_q$ si no es posible factorizarlo como producto de dos polinomios de $\mathbb{F}_q[x]$ de grado menor.

TEOREMA 1.2.8. *Sea $f(x)$ un polinomio no constante. Entonces,*

$$f(x) = p_1(x)^{a_1} p_2(x)^{a_2} \dots p_k(x)^{a_k},$$

*donde cada $p_i(x)$ es irreducible, los polinomios $p_i(x)$ son únicos salvo el orden en el que aparecen y producto por unidades, y los elementos $a_i$ son únicos.*

Este resultado nos dice que $\mathbb{F}_q[x]$ es lo que se conoce como *dominio de factorización única*. Se puede comprobar que es, además, un dominio de ideales principales.

### 1.2.2 *Elementos primitivos*

Vamos a ver que el conjunto $\mathbb{F}_q^*$ —de los elementos de $\mathbb{F}_q$ distintos de cero— es un grupo.

TEOREMA 1.2.9. *Se verifican las siguientes afirmaciones.*

1. *El grupo $\mathbb{F}_q^*$ es cíclico de orden $q-1$ con la multiplicación de $\mathbb{F}_q$.*
2. *Si $\gamma$ es un generador de este grupo cíclico entonces*

$$\mathbb{F}_q = \{0, 1 = \gamma^0, \gamma, \gamma^2, \dots, \gamma^{q-2}\},$$

*y se tiene que $\gamma^i = 1$ si y solo si $(q-1) \mid i$.*

Cada generador $\gamma$ de $\mathbb{F}_q^*$ se llama *elemento primitivo* de $\mathbb{F}_q$. Cuando los elementos distintos de cero de un cuerpo finito se expresan como potencias de $\gamma$ podemos multiplicar de forma sencilla teniendo en cuenta que $\gamma^i \gamma^j = \gamma^{i+j} = \gamma^s$, donde $0 \le s \le q-2$ e $i + j \equiv s \pmod{q-1}$.

TEOREMA 1.2.10. *Los elementos de $\mathbb{F}_q$ son las raíces del polinomio $x^q - x$.*

*Demostración.* Sea $\gamma$ un elemento primitivo de $\mathbb{F}_q$. Entonces, $\gamma^{q-1} = 1$ por definición. Por tanto, $(\gamma^i)^{q-1} = 1$ para todo $i$ tal que $0 \le i \le q-2$. En consecuencia, los elementos de $\mathbb{F}_q^*$ son las raíces de $x^{q-1} - 1 \in \mathbb{F}_p[x]$ y en consecuencia, de $x^q - x$. Como 0 es raíz de $x^q - x$, por la proposición 1.2.7 sabemos que los elementos de $\mathbb{F}_q$ son las raíces de $x^q - x$, como queríamos. □

Un elemento $\xi \in \mathbb{F}_q$ es una raíz enésima de la unidad si $\xi^n = 1$, y es una raíz enésima primitiva de la unidad si además $\xi^s \neq 1$ para todo s tal que $0 < s < n$. Un elemento primitivo $\gamma$ de $\mathbb{F}_q$ es por tanto una raíz $(q-1)$-ésima de la unidad. Se deduce del teorema 1.2.9 que el cuerpo $\mathbb{F}_q$ contiene una raíz enésima primitiva de la unidad si y solo si $n \mid (q-1)$, en cuyo caso $\gamma^{(q-1)/n}$ es dicha raíz.

### 1.2.3 *Construcción de cuerpos finitos*

En esta subsección vamos a construir cuerpos finitos a partir del cociente de anillos de polinomios por polinomios irreducibles. Tomamos por tanto un polinomio $f(x) \in \mathbb{F}_p[x]$ que sea irreducible sobre $\mathbb{F}_p$ y de grado $m$. Consideramos el anillo cociente dado por $\mathbb{F}_p[x]/(f(x))$. Utilizando el algoritmo de Euclides podemos comprobar que es un cuerpo de característica $p$, y de hecho, es un cuerpo finito con $q = p^m$ elementos. Al tratarse de un cociente todo elemento es una clase lateral y será de la forma $g(x) + (f(x))$, donde el polinomio $g(x)$ es único y con grado como mucho $m - 1$.

Escribiremos cada clase lateral como un vector en $\mathbb{F}_p^m$ siguiendo la correspondencia:

$$g_{m-1}x^{m-1} + g_{m-2}x^{m-2} + \cdots + g_1x + g_0 + (f(x)) \iff g_{m-1}g_{m-2}\cdots g_1g_0.$$

Esta notación vectorial nos permite realizar la suma en el cuerpo utilizando la suma habitual de los vectores. Multiplicar es una tarea a priori más complicada. Para multiplicar $g_1(x) + (f(x))$ por $g_2(x) + (f(x))$ primero utilizamos al algoritmo de división para escribir

$$g_1(x)g_2(x) = f(x)h(x) + r(x),$$

donde como sabemos o bien $\operatorname{gr} r(x) \leq m - 1$ o bien $r(x) = 0$. Puesto que estamos en el anillo cociente $\mathbb{F}_p[x]/(f(x))$ nos queda

$$(g_1(x) + (f(x)))(g_2(x) + (f(x))) = r(x) + (f(x)).$$

Esta notación es engorrosa, por lo que habitualmente operaremos en $\alpha$ en vez de en $x$ suponiendo que $f(\alpha) = 0$. Así, $g_1(\alpha)g_2(\alpha) = r(\alpha)$. En consecuencia, multiplicamos los polinomios en $\alpha$ de la forma habitual y utilizamos la ecuación $f(\alpha) = 0$ para reducir las potencias de $\alpha$ de grado mayor a $m - 1$ a polinomios en $\alpha$ de grado menor que $m$.

El conjunto $\{0\alpha^{m-1} + 0\alpha^{m-2} + \cdots + 0\alpha + a_0 \mid a_0 \in \mathbb{F}_p\} = \{a_0 \mid a_0 \in \mathbb{F}_p\}$ es el subcuerpo primo de $\mathbb{F}_q$.

Decimos que obtenemos $\mathbb{F}_q$ a partir de $\mathbb{F}_p$ yuxtaponiendo una raíz $\alpha$ de $f(x)$ a $\mathbb{F}_p$. Esta raíz viene dada formalmente por $\alpha = x + (f(x))$ en el anillo cociente $\mathbb{F}_p[x]/(f(x))$. Por tanto, ya hemos visto antes que $g(x) + (f(x)) = g(\alpha)$ y $f(\alpha) = f(x + (f(x))) = f(x) + (f(x)) = 0 + (f(x))$.

Un polinomio irreducible sobre $\mathbb{F}_p$ de grado $m$ es *primitivo* si tiene una raíz que es un elemento primitivo de $\mathbb{F}_q = \mathbb{F}_{p^m}$. Puede probarse que existen polinomios irreducibles de cualquier grado.

TEOREMA 1.2.11. *Para cualquier primo $p$ y cualquier entero positivo $m$, existe un cuerpo finito, único salvo isomorfismos, con $q = p^m$ elementos.*

### 1.2.4 *Clases ciclotómicas y polinomios minimales*

Sea $\mathbb{F}_{q^t}/\mathbb{F}_q$ una extensión de cuerpos. Por el teorema 1.2.10 cada elemento de $\mathbb{F}_{q^t}$ es raíz del polinomio $x^{q^t} - x$. Existe por tanto un polinomio mónico $M_\alpha$ en $\mathbb{F} - q[x]$ de grado mínimo que tiene a $\alpha$ como raíz. Este polinomio se conoce como *polinomio minimal* de $\alpha$ sobre $\mathbb{F}_q$. En el siguiente teorema vamos a estudiar algunas propiedades de los polinomios minimales.

TEOREMA 1.2.12. *Sea $\mathbb{F}_{q^t}/\mathbb{F}$ una extensión de cuerpos y sea $\alpha$ un elemento de $\mathbb{F}_{q^t}$ cuyo polinomio minimal es $M_\alpha \in \mathbb{F}_q[x]$. Se verifica:*

1. *El polinomio $M_\alpha(x)$ es irreducible sobre $\mathbb{F}_q$.*
2. *Si $g(x)$ es cualquier polinomio en $\mathbb{F}_q[x]$ tal que $g(\alpha) = 0$ entonces $M_\alpha(x) \mid g(x)$.*
3. *El polinomio $M_\alpha(x)$ es único.*

Si partimos de $f(x)$, un polinomio irreducible sobre $\mathbb{F}_q$ de grado $r$, podemos considerar la extensión generada por una de las raíces de $f(x)$ y obtendremos el cuerpo $\mathbb{F}_{q^r}$. De hecho, el siguiente teorema afirma que en ese caso todas las raíces de $f(x)$ estarán en $\mathbb{F}_{q^r}$.

TEOREMA 1.2.13. *Sea $f(x)$ un polinomio irreducible mónico sobre $\mathbb{F}_q$ de grado $r$. Entonces:*

1. *Todas las raíces de $f(x)$ están en $\mathbb{F}_{q^r}$ y en cualquier extensión de cuerpos de $\mathbb{F}_q$ generada por una de sus raíces.*
2. *Podemos expresar $f(x)$ como $f(x) = \prod_{i=1}^{r}(x - \alpha_i)$, donde $\alpha_i \in \mathbb{F}_{q^r}$ para $1 \leq i \leq r$.*
3. *El polinomio $f(x)$ divide a $x^{q^r} - x$.*

*Demostración.* □

En particular este teorema se verifica para los polinomios minimales $M_\alpha(x)$ sobre $\mathbb{F}_q$, pues son mónicos irreducibles.

TEOREMA 1.2.14. *Sea $\mathbb{F}_{q^t}/\mathbb{F}_q$ una extensión de cuerpos y sea $\alpha$ un elemento de $\mathbb{F}_{q^t}$ con polinomio minimal $M_\alpha$ en $\mathbb{F}_q[x]$. Se verifican las siguientes afirmaciones.*

1. *El polinomio $M_\alpha(x)$ divide a $x^{q^t} - x$.*
2. *El polinomio $M_\alpha(x)$ tiene raíces distintas dos a dos, todas en $\mathbb{F}_{q^t}$.*
3. *El grado de $M_\alpha(x)$ divide a $t$.*
4. *Podemos expresar $x^{q^t} - x = \prod_\alpha M_\alpha(x)$, donde $\alpha$ varía entre los elementos de un subconjunto de $\mathbb{F}_{q^t}$ que enumera los polinomios minimales de todos los elementos de $\mathbb{F}_{q^t}$ una sola vez.*

5. *Podemos expresar* $x^{q^t} - x = \prod_f f(x)$, *donde* $f$ *varía entre todos los mónicos irreducibles sobre* $\mathbb{F}_q$ *cuyo grado divide a* $t$.

*Demostración.* □

Dos elementos de $\mathbb{F}_{q^t}$ que tienen el mismo polinomio minimal en $\mathbb{F}_q[x]$ se llaman *conjugados sobre* $\mathbb{F}_q$. Es importante encontrar todos los conjugados de $\alpha \in \mathbb{F}_q$, es decir, todas las raíces de $M_\alpha(x)$. Sabemos por el teorema 1.2.14 que las raíces de $M_\alpha(x)$ son todas distintas dos a dos y que se encuentran en el cuerpo $\mathbb{F}_{q^t}$. Podemos encontrar estas raíces con ayuda del siguiente teorema.

TEOREMA 1.2.15. *Sea* $f(x)$ *un polinomio en* $\mathbb{F}_q[x]$ *y sea* $\alpha$ *una raíz de* $f(x)$ *en una extensión* $\mathbb{F}_{q^t}/\mathbb{F}_q$. *Entonces se verifican las siguientes afirmaciones.*

1. *Evaluando el polinomio obtenemos que* $f(x^q) = f(x)^q$.
2. *El elemento* $\alpha^q$ *es también una raíz de* $f(x)$ *en* $\mathbb{F}_q$.

*Demostración.* □

Si aplicamos este teorema de forma consecutiva podremos obtener todas las raíces de $M_\alpha(x)$, que serán de la forma $\alpha, \alpha^q, \alpha^{q^2}$, etc.; secuencia que terminará tras $r$ términos, cuando $\alpha^{q^r} = \alpha$.

Supongamos ahora que $\gamma$ es un elemento primitivo de $\mathbb{F}_{q^t}$. Entonces sabemos que $\alpha = \gamma^s$ para algún $s$. Por tanto, $\alpha^{q^r} = \alpha$ si y solo si $\gamma^{sq^r - s} = 1$. Por el teorema 1.2.9 se tiene que $sq^r \equiv s \pmod{q^t - 1}$. Basándonos en esta idea podemos definir la *clase q-ciclotómica de s módulo* $q^t - 1$ como el conjunto

$$C_s = \{s, sq, \dots, sq^{r-1}\} \pmod{q^t - 1},$$

donde $r$ es el menor entero positivo tal que $sq^r \equiv s \pmod{q^t - 1}$. Los conjuntos $C_s$ dividen el conjunto de enteros $\{0, 1, 2, \dots, q^t - 2\}$ en conjuntos disjuntos.

TEOREMA 1.2.16. *Si* $\gamma$ *es un elemento primitivo de* $\mathbb{F}_{q^t}$ *entonces el polinomio minimal de* $\gamma^s$ *sobre* $\mathbb{F}_q$ *es*

$$M_{\gamma^s}(x) = \prod_{i \in C_s} (x - \gamma^i).$$

## 1.3 AUTOMORFISMOS

# 2

# FUNDAMENTOS DE TEORÍA DE CÓDIGOS

En la introducción ya hemos visto cuáles son los objetivos de la teoría de códigos, así como el medio principal del que se sirve: el álgebra. Esta sección vamos a comentar algunos de los conceptos y resultados fundamentales de la teoría de códigos. Daremos la definición más sencilla de código para posteriormente estudiar otras estructuras más complejas, los códigos lineales y los códigos cíclicos, así como resultados básicos asociados a ellos. Finalmente, estudiaremos la versión original del algoritmo de Peterson-Gorenstein-Zierler, diseñado para un tipo de códigos, los BCH.

Las definiciones y los resultados comentados en esta sección seguirán lo descrito en (Huffman & Pless, 2003, cap. 1, 3-5) y (Podestá, 2006).

## 2.1 CÓDIGOS LINEALES

Vamos a comenzar nuestro estudio con los códigos lineales, pues son los más sencillos de comprender. Consideremos el espacio vectorial de todas las $n$-tuplas sobre el cuerpo finito $\mathbb{F}_q$, al que denotaremos en lo que sigue como $\mathbb{F}_q^n$. A los elementos $(a_1, \dots, a_n)$ de $\mathbb{F}_q^n$ los notaremos usualmente como $a_1 \cdots a_n$.

DEFINICIÓN 2.1.1. Un $(n, M)$ *código* $\mathcal{C}$ sobre el cuerpo $\mathbb{F}_q$ es un subconjunto de $\mathbb{F}_q^n$ de tamaño $M$. Si no hay riesgo de confusión lo denotaremos simplemente por $\mathcal{C}$. A los elementos de $\mathcal{C}$ los llamaremos *palabras codificadas*, *palabras código* o *codewords* en inglés. A $n$ se le llama *longitud* del código.

Por ejemplo, un $(5, 4)$ código sobre $\mathbb{F}_2$ puede ser el formado por los siguientes elementos:

$$10101, \qquad 10010, \qquad 01110, \qquad 11111.$$

Como se puede ver realmente un código es un objeto muy sencillo. Concluimos que es necesario añadir más estructura a los códigos para que puedan ser de utilidad. Esto motiva la siguiente definición.

DEFINICIÓN 2.1.2. Decimos que un código $\mathcal{C}$ es un código *lineal de longitud $n$ y dimensión $k$* —abreviado como $[n, k]$-*lineal*, o como $[n, k]_q$-*lineal* en caso de querer informar del cuerpo base— si dicho código es un subespacio vectorial de $\mathbb{F}_q^n$ de dimensión $k$.

NOTA 2.1.3. *Un código lineal $\mathcal{C}$ tiene $q^k$ palabras código.*

Así, hemos pasado de trabajar con un objeto que no tiene estructura alguna a trabajar con espacios vectoriales, cuyas propiedades son ampliamente conocidas y disponemos de numerosas herramientas para tratarlos. Un primer ejemplo de este hecho es la siguiente definición.

DEFINICIÓN 2.1.4. Una *matriz generadora* para un $[n, k]$ código $\mathcal{C}$ es una matriz $k \times n$ cuyas filas conforman una base de $\mathcal{C}$[1].

DEFINICIÓN 2.1.5. Para cada conjunto $k$ de columnas independientes de una matriz generadora $G$ el conjunto de coordenadas correspondiente se denomina *conjunto de información* para un código $\mathcal{C}$. Las $r = n - k$ coordenadas restantes se llaman *conjunto redundante*, y el número $r$, la *redundancia* de $\mathcal{C}$.

Si las primeras $k$ coordenadas de una matriz generadora $G$ forman un conjunto de información entonces el código tiene una única matriz generadora de la forma $[I_k \mid A]$, donde $I_k$ es la matriz identidad $k \times k$ y $A$ es una matriz $k \times r$. Esta matriz generadora se dice que está en *forma estándar*. A partir de cualquier matriz generadora siempre es posible obtener una matriz en forma estándar realizando una permutación adecuada de las coordenadas.

Como un código lineal $\mathcal{C}$ es un subespacio de un espacio vectorial, podemos calcular el ortogonal a dicho subespacio, obteniendo lo que llamaremos el *código dual* (*euclídeo*, si usamos el producto escalar usual) y que denotaremos por $\mathcal{C}^\perp$.

DEFINICIÓN 2.1.6. El *código dual* $\mathcal{C}^\perp$ de un código $\mathcal{C}$ viene dado por

$$\mathcal{C}^\perp = \left\{ x \in \mathbb{F}_q^n : x \cdot c = 0 \quad \forall c \in \mathcal{C} \right\},$$

donde $(\cdot)$ representa el producto escalar usual.

DEFINICIÓN 2.1.7. Sea $\mathcal{C}$ un $[n, k]$ código lineal. Una matriz $H$ se dice que es *matriz de paridad* si es una matriz generadora de $\mathcal{C}^\perp$.

PROPOSICIÓN 2.1.8. *Sea $H$ la matriz de paridad de un $[n, k]$ código lineal $\mathcal{C}$. Entonces,*

$$\mathcal{C} = \left\{ x \in \mathbb{F}_q^n : xH^T = 0 \right\} = \left\{ x \in F_q^n : Hx^T = 0 \right\}.$$

1 Efectivamente la matriz generadora no es única: basta tomar la correspondiente a cualquier otra base del código —que no deja de ser un espacio vectorial— para obtener una distinta. Pero es más, podemos simplemente reordenar las filas de una matriz generadora y en esencia estaremos obteniendo otra distinta.

*Demostración.* Sea $c \in \mathcal{C}$ una palabra código. Sabemos que la podemos expresar como $c = uG$, donde $u \in \mathbb{F}_q^k$ y $G$ es una matriz generadora de $\mathcal{C}$. Tenemos entonces que $cH^T = uGH^T$ y como $GH^T = 0$ —por ser H matriz generadora del subespacio ortogonal $\mathcal{C}$— se tiene que

$$\mathcal{C} \subset S_H = \left\{x \in \mathbb{F}_q^n : Hx^T = 0\right\},$$

que es el espacio solución de un sistema de $n-k$ ecuaciones con $n$ incógnitas y de rango $n-k$. Como $\dim(S_H) = n - (n-k) = k = \dim L$, concluimos que

$$L = S_H = \left\{x \in \mathbb{F}_q^n : Hx^T = 0\right\}. \qquad \square$$

Este último resultado, junto a la definición previa, nos conducen al siguiente teorema.

TEOREMA 2.1.9. *Si $G = [I_k \mid A]$ es una matriz generadora para un $[n,k]$ código $\mathcal{C}$ en forma estándar entonces $H = [-A \mid I_{n-k}]$ es una matriz de paridad para $\mathcal{C}$.*

Como nota final sobre nomenclatura de códigos duales, apuntamos que un código se dice *autoortogonal* cuando $\mathcal{C} \subseteq \mathcal{C}^\perp$, y *autodual* cuando $\mathcal{C} = \mathcal{C}^\perp$.

### 2.1.1 *Codificación y decodificación*

Codificar un mensaje consiste en escribirlo como palabra código de un código. La forma estándar de codificar mensajes con códigos lineales es utilizando una matriz generadora. Dado un mensaje $\mathbf{m} \in \mathbb{F}_q^k$ podemos obtener la palabra código $\mathbf{c}$ en $\mathcal{C}$ realizando la operación $\mathbf{c} = \mathbf{m}G$. Vamos a verlo mejor con un ejemplo.

EJEMPLO 2.1.10. Sea $\mathcal{C}$ $[3,2]$ un código binario lineal y $G$ la matriz generadora dada por

$$G = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix} \in \mathcal{M}_{2\times 3}(\mathbb{F}_2).$$

Dado un mensaje $\mathbf{m} = (x_1, x_2)$, se tiene que

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

y por tanto esta matriz codifica de la forma

$$00 \to 000, \quad 01 \to 011, \quad 10 \to 110, \quad 11 \to 101.$$

Observamos que una matriz generador $G$ define una aplicación lineal de $\mathbb{F}_q^k$ en $\mathbb{F}_q^n$, de forma que el código obtenido es la imagen de dicha aplicación. Podemos comprobar también que es posible codificar en los mismos códigos lineales utilizando distintas matrices generadores, lo que resultará en distintas palabras código para el mismo mensaje. Veamos un ejemplo con el mismo código binario lineal que en el ejemplo anterior pero con distinta matriz generadora.

EJEMPLO 2.1.11. Sea $\mathcal{C}$ un $[3, 2]$ código binario lineal y $G$ la matriz generadora dada por

$$G = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix} \in \mathcal{M}_{2\times 3}(\mathbb{F}_2).$$

Dado un mensaje $\mathbf{m} = (x_1, x_2)$, se tiene que

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

y por tanto esta matriz codifica de la forma

$$00 \to 000, \quad 01 \to 011, \quad 10 \to 101, \quad 11 \to 110.$$

Observamos en este ejemplo que las primeras 2 coordenadas de cada palabra código son iguales a las del mensaje que las genera. Pero en el código anterior también podemos encontrar el mensaje, lo que hay que fijarse en la primera y última coordenada. Cuando un mensaje se encuentra incrustado íntegramente en la palabra código —aunque puede que desordenado— se dice que la codificación seguida es *sistemática*. En caso contario, se dice que es *no-sistemática*. Veamos un ejemplo de codificación no-sistemática con el mismo código binario lineal de antes.

EJEMPLO 2.1.12. Sea $\mathcal{C}$ un $[3, 2]$ código binario lineal y $G$ la matriz generadora dada por

$$G = \begin{pmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \end{pmatrix} \in \mathcal{M}_{2\times 3}(\mathbb{F}_2).$$

Dado un mensaje $\mathbf{m} = (x_1, x_2)$, se tiene que

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

y por tanto esta matriz codifica de la forma

$$00 \to 000, \quad 01 \to 111, \quad 10 \to 101, \quad 11 \to 010.$$

Comprobamos que los mensajes 01 y 11 no están contenidos en las palabras código correspondientes, 111 y 010, respectivamente, luego la codificación es no-sistemática.

Dada una palabra código $\mathbf{c}$ si se desea obtener el mensaje $\mathbf{m}$ a partir del que se obtuvo podemos realizar el procedimiento inverso a la codificación. Para ello tenemos en cuenta que al codificar mediante una matriz generadora $G$ de tamaño $n \times k$ establecemos una correspondencia biyectiva entre mensajes y palabras código. Existe por tanto una matriz $K$ de tamaño $k \times n$ llamada *inversa por la derecha* tal que $GK = I_k$. Así, puesto que $\mathbf{c} = \mathbf{m}G$ podemos obtener el mensaje original calculando $\mathbf{c}K = \mathbf{m}GK = \mathbf{m}$. Veamos un ejemplo de este proceso.

EJEMPLO 2.1.13. Sea $\mathcal{C}$ un $[7, 3]$ código binario lineal y $G$ la matriz generadora dada por

$$G = \begin{pmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 \end{pmatrix} \in \mathcal{M}_{3\times 7}(\mathbb{F}_2).$$

Esta matriz codifica el mensaje $\mathbf{m} = (1, 0, 1)$ como:

$$\mathbf{c} = (1, 0, 1) \begin{pmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 \end{pmatrix} = (1,\ 1,\ 0,\ 1,\ 0,\ 0,\ 1)\,.$$

Para realizar el procedimiento inverso buscamos una matriz $K$ tal que $GK = I_7$. Esta matriz viene dada por

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

y por tanto el mensaje original era

$$\mathbf{m} = (1,\ 1,\ 0,\ 1,\ 0,\ 0,\ 1) \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} = (1, 0, 1).$$

El proceso de decodificación de los mensajes consiste en obtener una palabra código válida a partir de un mensaje recibido[2]. Es una tarea mucho

2 Es importante llamar la atención sobre el hecho de que *decodificar* no es el proceso inverso a *codificar*. Codificar consiste en escribir un mensaje como palabra código y decodificar, en corregir los errores que se hayan podido producir en la transmisión de dicha palabra.

más complicada que los procesos comentados antes, pues como ya se ha mencionado hay que tener en cuenta las posibles interferencias que se hayan podido producir en la comunicación. Los métodos de decodificación de códigos lineales en general se escapan del alcance de este trabajo, pues su objetivo principal es la descripción de un algoritmo de decodificación para un tipo concreto de códigos que veremos más adelante.

### 2.1.2 *Distancias y pesos*

Códigos distintos poseen distintas propiedades, lo que implica que sus capacidades de corrección difieran. En este apartado vamos a estudiar dos propiedades de los códigos muy relacionadas con esta idea.

DEFINICIÓN 2.1.14. La *distancia de Hamming* $\mathrm{d}(\boldsymbol{x}, \boldsymbol{y})$ entre dos vectores $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{F}_q^n$ se define como el número de coordenadas en las que difieren $\boldsymbol{x}$ e $\boldsymbol{y}$.

TEOREMA 2.1.15. *La función de distancia* $\mathrm{d}(\boldsymbol{x}, \boldsymbol{y})$ *verifica las siguientes propiedades.*

1. *No negatividad:* $\mathrm{d}(\boldsymbol{x}, \boldsymbol{y}) \geq 0$ *para todo* $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{F}_q^n$.
2. *La distancia* $\mathrm{d}(\boldsymbol{x}, \boldsymbol{y}) = 0$ *si y solo si* $\boldsymbol{x} = \boldsymbol{y}$.
3. *Simetría:* $\mathrm{d}(\boldsymbol{x}, \boldsymbol{y}) = \mathrm{d}(\boldsymbol{y}, \boldsymbol{x})$ *para todo* $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{F}_q^n$.
4. *Desigualdad triangular:* $\mathrm{d}(\boldsymbol{x}, \boldsymbol{z}) \leq \mathrm{d}(\boldsymbol{x}, \boldsymbol{y}) + \mathrm{d}(\boldsymbol{y}, \boldsymbol{z})$ *para todo elemento* $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z} \in \mathbb{F}_q^n$.

*Demostración.* □

La *distancia* (*mínima*) de un código $C$ es la menor distancia posible entre dos palabras código distintas. Si la distancia mínima $d$ de un $[n, k]$ código es conocida, nos referiremos a él como un $[n, k, d]$ código. Este valor es importante pues nos ayuda a determinar la capacidad de corrección de errores del código $C$, como ilustra el siguiente teorema.

TEOREMA 2.1.16. *Sea* $C$ *un* $[n, k, d]$ *código. Entonces* $C$ *tiene capacidad de corrección de errores*

$$t = \left\lfloor \frac{d-1}{2} \right\rfloor.$$

Efectivamente, a mayor distancia mínima, mayor número de errores en el código se pueden corregir. Otra medida interesante es el *peso de Hamming*.

DEFINICIÓN 2.1.17. El *peso de Hamming* $\mathrm{wt}(\boldsymbol{x})$ de un vector $\boldsymbol{x}$ es el número de coordenadas distintas de cero de $\boldsymbol{x}$.

El siguiente teorema nos ilustra la relación existente entre los conceptos de peso y distancia.

TEOREMA 2.1.18. *Si* $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{F}_q^n$, *entonces* $\mathrm{d}(\boldsymbol{x}, \boldsymbol{y}) = \mathrm{wt}(\boldsymbol{x} - \boldsymbol{y})$. *Si* $C$ *es un código lineal, la distancia mínima es igual al peso mínimo de las palabras código de* $C$ *distintas de cero.*

*Demostración.* □

Como consecuencia de este teorema —para códigos lineales— la distancia mínima también se llama *peso mínimo* del código.

DEFINICIÓN 2.1.19. Sea $A_i(C)$ —que abreviaremos $A_i$— el número de palabras código de peso $i$ en $C$. Para cada $0 \leq i \leq n$, la lista $A_i$ se denomina *distribución de peso* o *espectro de peso* de $C$.

EJEMPLO 2.1.20. Sea $C$ el código binario con matriz generadora

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

Dado $(x_1, x_2, x_3)$, se tiene que

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

y por tanto podemos obtener las palabras código de la forma

$$000 \to 000000, \quad 001 \to 000011, \quad 010 \to 001100, \quad 011 \to 001111,$$

$$100 \to 110000, \quad 101 \to 110011, \quad 110 \to 111100, \quad 111 \to 111111.$$

Luego la distribución de peso de $C$ es $A_0 = A_6 = 1$ y $A_2 = A_4 = 3$. Usualmente solo se listan los $A_i$ que son distintos de cero.

TEOREMA 2.1.21. *Sea* $C$ *un* $[n, k, d]$ *código sobre* $\mathbb{F}_q$. *Entonces,*

1. $A_0(C) + A_1(C) + \cdots + A_n(C) = q^k$.
2. $A_0(C) = 1$ *y* $A_1(C) = A_2(C) = \cdots = A_{d-1}(C) = 0$.

*Demostración.* □

TEOREMA 2.1.22. *Sea* $C$ *un código lineal con matriz de paridad* $H$. *Si* $\boldsymbol{c} \in C$, *las columnas de* $H$ *que se corresponden con coordenadas no nulas de* $\boldsymbol{c}$ *son linealmente independientes. Recíprocamente, si entre* $w$ *columnas de* $H$ *existe una relación de dependencia lineal con coeficientes no nulos, entonces hay una palabra código en* $C$ *de peso* $w$ *cuyas coordenadas no nulas se corresponden con dichas columnas.*

*Demostración.* □

COROLARIO 2.1.23. *Un código lineal tiene peso mínimo $d$ si y solo si su matriz de paridad tiene un conjunto de $d$ columnas linealmente dependientes pero no tiene un conjunto de $d-1$ columnas linealmente dependientes.*

*Demostración.* □

## 2.2 EJEMPLOS DE CÓDIGOS

En esta sección vamos a describir someramente algunas familias de códigos relevantes: los códigos de repetición, los códigos de control de paridad y los códigos de Hamming.

### 2.2.1 *Códigos de repetición*

Los códigos de repetición son una de las familias de códigos más sencillas. Dado un mensaje $\mathbf{m} = (m_1, m_2, \dots, m_n) \in \mathbb{F}_q^n$ lo que hacemos para codificarlo es repetir cada elemento $m_i$ de la tupla $k$ veces:

$$\mathbf{c} = (m_{11}, m_{12}, \dots, m_{1k}, m_{21}, m_{22}, \dots, m_{2k}, \dots, m_{n1}, m_{n2}, \dots, m_{nk}).$$

A la hora de decodificar un mensaje cada bloque de $k$ elementos se fija al valor del elemento que más se repita. Los más utilizados son los códigos de repetición binarios, es decir, los que se definen sobre $\mathbb{F}_2$. No son códigos lineales.

### 2.2.2 *Códigos de control de paridad*

Los $[n, n-1]$-códigos lineales que tienen matriz de paridad

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

se llaman *códigos de control de paridad* o *códigos de peso par*. Por la proposición 2.1.8 las palabras código $\mathbf{c}$ de este tipo de códigos han de cumplir que

$$\mathbf{c}H^T = c_1 + c_2 + \cdots + c_n = 0,$$

es decir, el número de 1 en la palabra código ha de ser par —de ahí el nombre—. La codificación de mensajes se realiza entonces añadiendo un *bit* de paridad al final del mensaje cuyo valor se fija para que el número de 1 en el mismo sea par. Estos códigos tienen distancia 2 y pueden corregir un solo error.

### 2.2.3 *Códigos de Hamming*

Consideremos una matriz $r \times (2^r - 1)$ cuyas columnas son los números $1, 2, 3, \ldots, 2^{r-1}$ escritos en binario. Dicha matriz es la matriz de paridad de un $[n = 2^{r-1}, k = n - r]$ código binario. A los códigos de esta forma los llamaremos códigos de Hamming de longitud $n = 2^{r-1}$ y los denotamos por $\mathcal{H}_r$ o $\mathcal{H}_{2,r}$.

Como las columnas son distintas y no nulas, la distancia es al menos 3 por el corolario 2.1.23. Además, como las columnas correspondientes a los números $1, 2, 3$ son linealmente independientes, la distancia mínima es 3 por el mismo corolario. Podemos decir por tanto que los códigos de Hamming $\mathcal{H}_r$ son $[2^{r-1}, 2^{r-1-r}, 3]$ códigos binarios.

Podemos generalizar esta definición y definir los códigos de Hamming $\mathcal{H}_{q,r}$ sobre un cuerpo finito arbitrario $\mathbb{F}_q$. Para $r \geq 2$ un código $\mathcal{H}_{q,r}$ tiene matriz de paridad $H_{q,r}$, cuyas columnas están compuestas por un vector no nulo por cada uno de los subespacios de dimensión 1 de $\mathbb{F}_q^r$. Hay $(q^r - 1)/(q - 1)$ subespacios de dimensión 1, por lo que $\mathcal{H}_{q,r}$ tiene longitud $n = (q^r - 1)/(q - 1)$, dimensión $n - r$ y redundancia $r$. Como todas las columnas son independientes unas de otras, $\mathcal{H}_{q,r}$ tiene peso mínimo al menos 3. Si sumamos dos vectores no nulos de dos subespacios unidimensionales distintos obtenemos un vector no nulo de un tercer subespacio unidimensional, por lo que $\mathcal{H}_{q,r}$ tiene peso mínimo 3. Cuando $q = 2$, $\mathcal{H}_{2,r}$ es el código $\mathcal{H}_r$.

## 2.3 CÓDIGOS CÍCLICOS

En esta sección vamos a estudiar los aspectos fundamentales de la familia de los códigos cíclicos, que representan la base del objeto de estudio de este trabajo.

DEFINICIÓN 2.3.1. Un código lineal $\mathcal{C}$ de longitud $n$ sobre $\mathbb{F}_q$ es *cíclico* si para cada vector $\mathbf{c} = c_0 \ldots c_{n-2} c_{n-1}$ en $\mathcal{C}$, el vector $c_{n-1} c_0 \ldots c_{n-2}$ —obtenido a partir de $\mathbf{c}$ desplazando cíclicamente las coordenadas, llevando $i \mapsto i + 1$ mód $n$— también está en $\mathcal{C}$.

Al trabajar con códigos cíclicos pensamos en la posición de las coordenadas de forma cíclica, pues al llegar a $n-1$ se comienza de nuevo en 0. Al hablar de «coordenadas consecutivas» siempre tendremos en cuenta esta ciclicidad. Representaremos las palabras código de los códigos cíclicos como polinomios, pues podemos definir de forma natural una biyección entre el vector $\mathbf{c} = c_0 c_1 \ldots c_{n-1}$ en $\mathbb{F}_q$ y los polinomios de la forma $c(x) = c_0 + c_1 x + \ldots c_{n-1} x^{n-1}$ en $\mathbb{F}_q[x]$ de grado al menos $n - 1$. Obtenemos así un isomorfismo entre $\mathbb{F}_q$-espacios vectoriales. Obsérvese que dado un polinomio $c(x)$ descrito como

antes, el polinomio $xc(x) = c_{n-1}x^n + c_0x + c_1x^2 + \cdots + c_{n-2}x^{n-1}$ equivale a representar la palabra código $\mathbf{c}$ desplazada una posición a la derecha, siempre que $x^n$ fuese igual a 1.

Formalmente, el hecho de que un código $\mathcal{C}$ sea invariante bajo un desplazamiento cíclico implica que si $c(x)$ está en $\mathcal{C}$, también ha de estar $xc(x)$, siempre que multipliquemos módulo $x^n - 1$. Esto nos sugiere que el contexto adecuado para estudiar códigos cíclicos es el anillo cociente $\mathcal{R}_n = \mathbb{F}_q[x]/(x^n - 1)$.

Por tanto, bajo la correspondencia vectores-polinomios que hemos descrito antes, los códigos cíclicos son ideales de $\mathcal{R}_n$, y los ideales de $\mathcal{R}_n$ son códigos cíclicos. En consecuencia, el estudio de los códigos cíclicos en $\mathbb{F}_q^n$ es equivalente al estudio de los ideales en $\mathcal{R}_n$, que va a depender de la factorización de $x^n - 1$ y por tanto lo vamos a abordar a continuación.

### 2.3.1 *Factorización de $x^n - 1$*

A la hora de factorizar $x^n - 1$ existen dos posibilidades, pues dicha factorización puede tener factores irreducibles repetidos o no. Vamos a asumir que $q$ y $n$ son primos relativos y por tanto $x^n - 1$ no tiene factores repetidos; en caso contrario el anillo cociente sería semisimple, lo que no nos aportaría nada: los ideales generados por los polinomios con factores repetidos no aumentan la distancia, pues es la misma que la del subcódigo sin factores repetidos.

Para factorizar $x^n - 1$ sobre $\mathbb{F}_q$ necesitamos considerar la extensión de cuerpos $\mathbb{F}_{q^t}$ de $\mathbb{F}_q$ que contenga todas sus raíces. El cuerpo $\mathbb{F}_{q^t}$ debe contener una enésima raíz primitiva de la unidad, que por el teorema 1.2.9 sabemos que ocurre cuando $n \mid (q^t - 1)$. Definimos el *orden* $\mathrm{ord}_n(q)$ de $q$ módulo $n$ como el menor entero positivo $a$ tal que $q^a \equiv 1 \bmod n$. Si $t = \mathrm{ord}_n(q)$ entonces $\mathbb{F}_{q^t}$ contiene una enésima raíz primitiva de la unidad $\alpha$ pero no hay una extensión de cuerpos más pequeña de $\mathbb{F}_q$ que la contenga. Como todos los $\alpha^i$ son distintos dos a dos para $0 \le i < n$ y $(\alpha^i)^n = 1$, entonces $\mathbb{F}_{q^t}$ contiene todas las raíces de $x^n - 1$. Por tanto, $\mathbb{F}_{q^t}$ es lo que se conoce como *cuerpo de descomposición* de $x^n - 1$ sobre $\mathbb{F}_q$.

Los factores irreducibles de $x^n - 1$ sobre $\mathbb{F}_q$ deben ser el producto de los distintos polinomios minimales de las enésimas raíces de la unidad en $\mathbb{F}_{q^t}$. Supongamos que $\gamma$ es un elemento primitivo de $\mathbb{F}_{q^t}$. Entonces $\alpha = \gamma^d$ es una enésima raíz primitiva de la unidad, donde $d = (q^t - 1)/n$. Las raíces del polinomio $M_{a^s}(x)$ son $\{\gamma^{ds}, \gamma^{dsq}, \gamma^{dsq^2}, \ldots, \gamma^{dsq^{r-1}}\} = \{\alpha^s, \alpha^{sq}, \alpha^{sq^2}, \ldots, \alpha^{sq^{r-1}}\}$, donde $r$ es el menor entero positivo tal que $dsq^r \equiv ds \bmod q^t - 1$ por el teorema *. Pero $dsq^r \equiv ds \bmod q^t - 1$ si y solo si $sq^r \equiv s \bmod n$.

Todo esto nos lleva a extender la definición de clases $q$-ciclotómicas que hemos introducido en la sección *. Sea $s$ un entero tal que $0 \leq s < n$. La *clase $q$-ciclotómica de $s$ módulo $n$* es el conjunto

$$C_s = \{s, sq, \ldots, sq^{r-1}\} \text{ mód } n,$$

donde $r$ es el menor entero positivo tal que $sq^r \equiv s$ mód $n$. Se deduce entonces que $C_s$ es la órbita de la permutación $i \mapsto iq$ mód $n$ que contiene a $s$. Las distintas clases $q$-ciclotómicas módulo $n$ dividen el conjunto de enteros $\{0, 1, 2, \ldots, n-1\}$. En la sección * estudiamos el caso particular en el que $n = q^t - 1$. Obsérvese que $\text{ord}_n(q)$ es el tamaño de la clase $q$-ciclotómica $C_1$ módulo $n$.

TEOREMA 2.3.2. *Sea $n$ un entero positivo primo relativo con $q$. Sea $t = \text{ord}_n(q)$. Sea $\alpha$ una raíz enésima primitiva de la unidad en $\mathbb{F}_{q^t}$. Se verifican las siguientes afirmaciones.*

1. *Para cada entero $s$ tal que $0 \leq s < n$ el polinomio minimal de $\alpha^s$ sobre $\mathbb{F}_q$ es*
$$M_{\alpha^s}(x) = \prod_{i \in C_s} (x - \alpha^i),$$
*donde $C_s$ es la clase $q$-ciclotómica de $s$ módulo $n$.*
2. *Los conjugados de $\alpha^s$ son los elementos $\alpha^i$ con $i \in C_s$.*
3. *Se tiene que*
$$x^n - 1 = \prod_s M_{\alpha^s}(x)$$
*es la factorización de $x^n - 1$ en factores irreducibles sobre $\mathbb{F}_q$, donde $s$ varía en un conjunto de representantes de las clases $q$-ciclotómicas módulo $n$.*

*Demostración.* □

TEOREMA 2.3.3. *El tamaño de cada clase $q$-ciclotómica es un divisor de $\text{ord}_n(q)$. Además, el tamaño de $C_1$ es $\text{ord}_n(q)$.*

### 2.3.2 *Construcción de códigos cíclicos*

Una vez factorizado $x^n - 1$ vamos a ver que hay una correspondencia biyectiva entre sus polinomios divisores mónicos y los códigos cíclicos en $\mathcal{R}_n$. El siguiente teorema es el resultado fundamental de códigos cíclicos que nos va a permitir describirlos.

TEOREMA 2.3.4. *Sea $C$ un ideal de $\mathcal{R}_n$, es decir, un código cíclico de longitud $n$. Entonces:*

1. *Existe un único polinomio mónico $g(x)$ de grado mínimo en $C$.*

2. *El polinomio descrito en (1) genera $\mathcal{C}$, es decir, $\mathcal{C} = \langle g(x)\rangle$.*
3. *El polinomio descrito en (1) verifica que $g(x) \mid x^n - 1$.*

*Sea $k = n - \operatorname{gr} g(x)$ y sea $g(x) = \sum_{i_0}^{n-k} g_i x^i$, donde $g_{n-k} = 1$. Entonces:*

4. *Se verifica que*
$$\mathcal{C} = \langle g(x)\rangle = \{f(x)g(x) : \operatorname{gr} f(x) < k\}.$$
5. *El conjunto $\{g(x), xg(x), \dots, x^{k-1}g(x)\}$ es una base de $\mathcal{C}$ y $\mathcal{C}$ tiene dimensión $k$.*
6. *La matriz $G$ dada por*
$$G = \begin{bmatrix} g_0 & g_1 & g_2 & \cdots & \cdots & g_{n-k} & 0 & 0 & \dots & 0 \\ 0 & g_0 & g_1 & g_2 & \cdots & \cdots & g_{n-k} & 0 & \dots & 0 \\ 0 & 0 & g_0 & g_1 & g_2 & \cdots & \cdots & g_r & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & & & \ddots & 0 \\ 0 & 0 & \dots & 0 & g_0 & g_1 & g_2 & \cdots & \cdots & g_{n-k} \end{bmatrix},$$
*donde cada fila es un desplazamiento cíclico de la fila previa, es una matriz generadora de $\mathcal{C}$.*
7. *Si $\alpha$ es una enésima raíz primitiva de la unidad en alguna extensión de cuerpos de $\mathbb{F}_q$ entonces*
$$g(x) = \prod_s M_{\alpha^s}(x),$$
*siendo dicho producto sobre un subconjunto de representantes de las clases $q$-ciclotómicas módulo $n$.*

*Demostración.* Veamos la demostración apartado por apartado.

1. Supongamos que $\mathcal{C}$ contiene dos polinomios mónicos distintos, $g_1(x)$ y $g_2(x)$, ambos de grado mínimo $r$. Entonces, $g_1(x) - g_2(x)$ es un polinomio no nulo de grado menor que $r$, lo que es absurdo. Existe por tanto un único polinomio de grado mínimo $r$ en $\mathcal{C}$, como queríamos.
2. Como $g(x) \in \mathcal{C}$ y $\mathcal{C}$ es un ideal, tenemos que $\langle g(x)\rangle \subset \mathcal{C}$. Por otra parte, dado $p(x) \in \mathcal{C}$ el algoritmo de división nos da elementos $q(x), r(x)$ tales que $p(x) = q(x)g(x) + r(x)$, de forma que o bien $r(x) = 0$ o bien $\operatorname{gr} r(x) < \operatorname{gr} g(x)$. Como podemos expresar $r(x)$ de la forma $r(x) = p(x) - q(x)g(x) \in \mathcal{C}$ y tiene grado menor que $\operatorname{gr} g(x)$, al ser este último de grado mínimo necesariamente ha de darse que $r(x) = 0$. Por tanto, $p(x) = q(x)g(x) \in \langle g(x)\rangle$ y $\mathcal{C} \subset \langle g(x)\rangle$. En consecuencia, $\langle g(x)\rangle = \mathcal{C}$.
3. Por el algoritmo de división, al dividir $x^n - 1$ por $g(x)$ tenemos que $x^n - 1 = q(x)g(x) + r(x)$. De nuevo, o bien $r(x) = 0$ o bien $\operatorname{gr} r(x) < \operatorname{gr} g(x)$. Como en $\mathcal{R}_n$ se tiene que $x^n - 1 = 0 \in \mathcal{C}$, necesariamente $r(x) \in \mathcal{C}$. Esto supone una contradicción, a menos que $r(x) = 0$. En consecuencia, $g(x) \mid x^n - 1$.

4. El ideal generado por $g(x)$ es $\langle g(x)\rangle = \{f(x)g(x) : f(x) \in \mathcal{R}_n\}$. Queremos ver que podemos restringir los polinomios $f(x)$ a aquellos que tengan grado menor que $k$. Por (3) sabemos que $x^n - 1 = h(x)g(x)$ para algún polinomio $h(x)$ que tenga grado $k = n - \operatorname{gr} g(x)$. Dividimos entonces $f(x)$ por este polinomio $h(x)$ y por el algoritmo de división obtenemos $f(x) = q(x)h(x) + r(x)$, donde $\operatorname{gr} r(x) < \operatorname{gr} h(x) = k$. Entonces, tenemos

$$\begin{aligned} f(x)g(x) &= q(x)h(x)g(x) + r(x)g(x) \\ &= q(x)(x^n - 1) + r(x)g(x), \end{aligned}$$

   luego $f(x)g(x) = r(x)g(x)$, y puesto que antes ya hemos visto que $\operatorname{gr} r(x) < k$, hemos obtenido lo que buscábamos.
5. A partir de (4) tenemos que el conjunto

$$\{g(x), xg(x), \dots, x^{k-1}g(x)\}$$

   genera $\mathcal{C}$, y como es linealmente independiente, forma una base de $\mathcal{C}$. Esto demuestra también que la dimensión de $\mathcal{C}$ es $k$.
6. La matriz $G$ es matriz generadora de $\mathcal{C}$ pues

$$\{g(x), xg(x), \dots, x^{k-1}g(x)\}$$

   es una base de $\mathcal{C}$.
7. TODO.

□

Este teorema nos proporciona una forma de obtener los códigos cíclicos de longitud $n$ a partir de los divisores del polinomio $x^n - 1$ así como describir una matriz generadora de dichos códigos a partir de ellos. Vamos a ver a continuación que el polinomio mónico divisor de $x^n - 1$ que genera a un código cíclico $\mathcal{C}$ es único.

COROLARIO 2.3.5. *Sea $\mathcal{C}$ un código cíclico en $\mathcal{R}_n$ distinto de cero. Son equivalentes:*

1. *El polinomio $g(x)$ es el polinomio mónico de menor grado en $\mathcal{C}$.*
2. *Podemos expresar $\mathcal{C}$ como $\mathcal{C} = \langle g(x)\rangle$, $g(x)$ es mónico y $g(x) \mid (x^n - 1)$.*

*Demostración.* Que (1) implica (2) ya lo hemos probado en el teorema 2.3.4. Veamos que partiendo de (2) obtenemos (1). Sea $g_1(x)$ el polinomio mónico de menor grado en $\mathcal{C}$. Por el teorema 2.3.4, $g_1(x) \mid g(x)$ en $\mathbb{F}_q[x]$ y $\mathcal{C} = \langle g_1(x)\rangle$. Como $g_1(x) \in \mathcal{C} = \langle g(x)\rangle$, podemos expresarlo como $g_1(x) = g(x)a(x)$ mód $x^n - 1$, luego tenemos que $g_1(x) = g(x)a(x) + (x^n - 1)b(x)$

en $\mathbb{F}_q[x]$. Por otro lado, como $g(x) \mid (x^n - 1)$, tenemos que $g(x) \mid g(x)a(x) + (x^n - 1)b(x)$, o lo que es lo mismo, que $g(x) \mid g_1(x)$. En consecuencia, como $g_1(x)$ y $g(x)$ son ambos mónicos y dividen el uno al otro en $\mathbb{F}_q[x]$, son necesariamente iguales. □

A este polinomio $g(x)$ lo llamamos *polinomio generador* del código cíclico $\mathcal{C}$. Por el corolario anterior, este polinomio es tanto el polinomio mónico en $\mathcal{C}$ de grado mínimo como el polinomio mónico que divide a $x^n - 1$ y genera a $\mathcal{C}$. Existe por tanto una correspondencia biunívoca entre los códigos cíclicos distintos de cero y los divisores de $x^n - 1$ distintos de él mismo. Para extender dicha correspondencia entre todos los códigos cíclicos en $\mathcal{R}_n$ y todos los divisores mónicos de $x^n - 1$ definimos como polinomio generador del código cíclico $\{\mathbf{0}\}$ el polinomio $x^n - 1$. Esta correspondencia biyectiva nos conduce al siguiente corolario.

COROLARIO 2.3.6. *El número de códigos cíclicos en $\mathcal{R}_n$ es $2^m$, donde $m$ es el número de clases $q$-ciclotómicas módulo $n$. Es más, las dimensiones de los códigos cíclicos en $\mathcal{R}_n$ son todas sumas de tamaños de las clases $q$-ciclotómicas módulo $n$.*

*Demostración.* □

Ahora mismo todo este desarrollo puede parecer demasiado abstracto. Vamos a ver un ejemplo exhaustivo para entender cómo podemos obtener los polinomios generadores de los códigos cíclicos de una longitud arbitraria y cómo éstos son generados a partir de ellos.

EJEMPLO 2.3.7. Vamos a describir todos los códigos cíclicos binarios de longitud 7. Sobre $\mathbb{F}_2$ podemos descomponer $x^7 - 1$ como

$$x^7 - 1 = (x + 1)(x^3 + x + 1)(x^3 + x^2 + 1).$$

Así, los 8 polinomios generadores son todos los divisores de $x^7 - 1$, a saber:

1. $1$
2. $(x + 1)$
3. $(x^3 + x + 1)$
4. $(x^3 + x^2 + 1)$
5. $(x + 1)(x^3 + x + 1) = x^4 + x^3 + x^2 + 1$
6. $(x + 1)(x^3 + x^2 + 1) = x^4 + x^2 + x + 1$
7. $(x^3 + x + 1)(x^3 + x^2 + 1) = x^6 + x^5 + x^4 + x^3 + x^2 + x + 1$
8. $(x + 1)(x^3 + x + 1)(x^3 + x^2 + 1) = x^7 - 1$

Vamos a ver qué códigos generan estos polinomios:

1. La dimensión del código es $k = 7 - 0 = 7$, luego el código generado es un $[7, 7]$-código lineal, que es evidentemente $\mathbb{F}_2^7$. La matriz generadora que nos proporciona el teorema 2.3.4(6) es

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

2. La dimensión del código es $k = 7 - 1 = 6$, luego el código generado es un $[7, 6]$-código lineal. La matriz generadora que nos proporciona el teorema 2.3.4(6) es

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

   Comprobamos que la matriz de paridad es

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

   y por tanto el código obtenido es un código de control de paridad.

3. La dimensión del código es $k = 7 - 3 = 4$, luego el código generado es un $[7, 4]$-código lineal. La matriz generadora que nos proporciona el teorema 2.3.4(6) es

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

   La matriz de paridad en este caso es

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

   y por tanto el código generado es un $\mathcal{H}_3$ código de Hamming.

4. La dimensión del código es $k = 7 - 3 = 4$, luego el código generado es un $[7, 4]$-código lineal. La matriz generadora que nos proporciona el teorema 2.3.4(6) es

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

La matriz de paridad en este caso es

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

y por tanto el código generado es un $\mathcal{H}_3$ código de Hamming.

5. La dimensión del código es $k = 7 - 4 = 3$, luego el código generado es un $[7, 3]$-código lineal. La matriz generadora que nos proporciona el teorema 2.3.4(6) es

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

La matriz de paridad en este caso es

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

y por tanto el código generado es un $\mathcal{H}_4$ código de Hamming.

6. La dimensión del código es $k = 7 - 4 = 3$, luego el código generado es un $[7, 3]$-código lineal. La matriz generadora que nos proporciona el teorema 2.3.4(6) es

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

La matriz de paridad en este caso es

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

y por tanto el código generado es un $\mathcal{H}_4$ código de Hamming.

7. La dimensión del código es $k = 7 - 6 = 1$, luego el código generado es un $[7, 1]$-código lineal. La matriz generadora que nos proporciona el teorema 2.3.4(6) es

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

   por lo que concluimos que el código generado es el código de repetición de longitud 7.
8. La dimensión del código es $k = 7 - 7 = 0$, luego el código generado es $\{\mathbf{0}\}$.

Finalmente, el siguiente resultado nos muestra la relación entre dos polinomios generadores cuando un código es subcódigo de otro.

COROLARIO 2.3.8. *Sean $\mathcal{C}_1$ y $\mathcal{C}_2$ códigos cíclicos sobre $\mathbb{F}_q$ con polinomios generadores $g_1(x)$ y $g_2(x)$, respectivamente. Entonces, $\mathcal{C}_1 \subset \mathcal{C}_2$ si y solo si $g_2(x) \mid g_1(x)$.*

### 2.3.3 *Codificación de códigos cíclicos*

Vamos a ver a continuación tres tipos de codificación de códigos cíclicos. Consideraremos un código cíclico $\mathcal{C}$ de longitud $n$ sobre $\mathbb{F}_q$ con polinomio generador $g(x)$ de grado $n - k$, por lo que $\mathcal{C}$ tiene dimensión $k$.

CODIFICACIÓN NO-SISTEMÁTICA Esta forma de codificación está basada en la técnica natural de codificación que describimos en la sección ( ). Sea $G$ la matriz generadora obtenida a partir de los desplazamientos de $g(x)$ descrita en el teorema 2.3.4. Dado el mensaje $\mathbf{m} \in \mathbb{F}_q^k$, lo codificamos como la palabra código $\mathbf{c} = \mathbf{m}G$. De igual forma, si $m(x)$ y $c(x)$ son los polinomios en $\mathbb{F}_q[x]$ asociados a $\mathbf{m}$ y $\mathbf{c}$, entonces $c(x) = m(x)g(x)$.

CODIFICACIÓN SISTEMÁTICA El polinomio $m(x)$ asociado a un mensaje $\mathbf{m}$ tendrá como mucho grado $k - 1$. Por tanto, el polinomio $n^{n-k}m(x)$ tendrá como mucho grado $n - 1$ y sus primeros $n - k$ coeficientes son nulos. Por tanto, el mensaje está contenido en los coeficientes de $x^{n-k}, x^{n-k+1}, \ldots, x^{n-1}$. Por el algoritmo de división tenemos que

$$x^{n-k}m(x) = g(x)a(x) + r(x), \qquad \text{donde } \operatorname{gr} r(x) < n - k \text{ o } r(x) = 0.$$

Sea $c(x) = x^{n-k}m(x) - r(x)$. Como $c(x)$ es múltiplo de $g(x)$, $c(x) \in \mathcal{C}$. El polinomio $c(x)$ difiere de $x^{n-k}m(x)$ en los coeficientes de $1, x, \ldots, x^{n-k-1}$ ya que $\operatorname{gr} r(x) < n - k$. Por tanto, $c(x)$ contiene el mensaje $\mathbf{m}$ en los coeficientes de los términos de grado al menos $n - k$.

EJEMPLO 2.3.9. Sea $\mathcal{C}$ un código cíclico de longitud 15 con polinomio generador $g(x) = (1+x+x^4)(1+x+x^2+x^3+x^4)$. Supongamos que queremos codificar el mensaje $m(x) = 1 + x^2 + x^5$. Vamos a ver su codificación con los dos métodos descritos. Como la longitud de $\mathcal{C}$ es 15 y el grado de su polinomio generador es 8, la dimensión del código es $15 - 8 = 7$. Escribimos el mensaje $m(x)$ en forma de vector: $\mathbf{m} = (1, 0, 1, 0, 0, 1, 0)$. Una matriz generadora del código $\mathcal{C}$ es:

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

1. Codificación no-sistemática. Simplemente multiplicamos $\mathbf{m}$ por $G$, obteniendo:

$$\mathbf{c} = \mathbf{m}G = (1,\ 0,\ 1,\ 0,\ 1,\ 1,\ 0,\ 1,\ 0,\ 0,\ 1,\ 1,\ 1,\ 1,\ 0)\,.$$

2. Codificación sistemática. Calculamos el cociente de $x^{n-k}m(x)$ por $g(x)$ para obtener el resto $r(x) = x^6 + x + 1$. Entonces, la palabra código viene dada por $c(x) = x^{n-k}m(x) - r(x) = x^{13} + x^{10} + x^8 + x^6 + x + 1$, que en forma de vector resulta $\mathbf{c} = (1,\ 1,\ 0,\ 0,\ 0,\ 0,\ 1,\ 0,\ 1,\ 0,\ 1,\ 0,\ 0,\ 1,\ 0)$. Observamos que la codificación es efectivamente sistemática: nuestro mensaje $m$ está contenido íntegramente en las últimas 7 coordenadas.

## 2.4 IDEMPOTENTES, MULTIPLICADORES Y CEROS DE UN CÓDIGO CÍCLICO

Cada código cíclico en $\mathcal{R}_n$ contiene un único idempotente que genera el ideal. Este elemento se denomina *idempotente generador* del código cíclico. En el siguiente teorema probaremos este hecho y mostraremos además un método para determinar el idempotente generador de un código cíclico.

TEOREMA 2.4.1. *Sea $\mathcal{C}$ un código cíclico en $\mathcal{R}_n$. Entonces:*

1. *Existe un único idempotente $e(x) \in \mathcal{C}$ tal que $\mathcal{C} = \langle e(x) \rangle$.*
2. *Si $e(x)$ es un idempotente no nulo en $\mathcal{C}$, entonces $\mathcal{C} = \langle e(x) \rangle$ si y solo si $e(x)$ es la unidad de $\mathcal{C}$.*

*Demostración.* Si $\mathcal{C}$ es el código cero, entonces el idempotente es el cero, con lo que (1) está claro y (2) no se aplica a este caso. Veamos entonces la demostración por apartados suponiendo que $\mathcal{C}$ es distinto de cero.

1. Supongamos primero que $e(x)$ es una unidad en $\mathcal{C}$. Entonces, $\langle e(x)\rangle \subset \mathcal{C}$, ya que $\mathcal{C}$ es un ideal. Si $c(x) \in \mathcal{C}$, entonces $c(x)e(x) = c(x)$ en $\mathcal{C}$. En consecuencia, $\langle e(x)\rangle = \mathcal{C}$. Por otro lado, supongamos que $e(x)$ es un idempotente distinto de cero y tal que $\mathcal{C} = \langle e(x)\rangle$. Entonces, cada elemento $c(x)$ lo podemos escribir como $c(x) = f(x)e(x)$. Pero se tiene que $c(x)e(x) = f(x)(e(x))^2 = f(x)e(x) = c(x)$, luego $e(x)$ es la unidad de $\mathcal{C}$.
2. Tenemos que probar la existencia y la unicidad. Comenzamos con la existencia. Sea $g(x)$ el polinomio generador dde $\mathcal{C}$. Entonces, sabemos que $g(x) \mid (x^n - 1)$ por el teorema 2.3.4. Tomemos $h(x) = (x^n - 1)/g(x)$. Sabemos que $\mathrm{mcd}(g(x), h(x)) = 1$ en $\mathbb{F}_q[x]$, ya que $x^n - 1$ tiene todas sus raíces distintas. En consecuencia, el algoritmo de Euclides nos proporciona los polinomios $a(x), b(x) \in \mathbb{F}_q[x]$ tales que $a(x)g(x) + b(x)h(x) = 1$. Llamemos $e(x) \equiv a(x)g(x) \bmod x^n - 1$, que será el representante de dicha clase de equivalencia en $\mathcal{R}_n$. Entonces, en $\mathcal{R}_n$,

$$\begin{aligned} e(x)^2 &\equiv (a(x)g(x))(1 - b(x)h(x)) \bmod x^n - 1 \\ &\equiv a(x)g(x) - a(x)g(x)b(x)h(x) \bmod x^n - 1 \\ &\equiv a(x)g(x) - a(x)b(x)(x^n - 1) \bmod x^n - 1 \\ &\equiv a(x)g(x) \bmod x^n - 1 \\ &\equiv e(x) \bmod x^n - 1. \end{aligned}$$

   Por tanto, este elemento $e(x)$ es idempotente. Veamos ahora que si $c(x) \in \mathcal{C}$, entonces $c(x) = f(x)g(x)$, luego

$$\begin{aligned} c(x)e(x) &= f(x)g(x)(1 - b(x)h(x)) \\ &\equiv f(x)g(x) \bmod x^n - 1 \\ &\equiv c(x) \bmod x^n - 1, \end{aligned}$$

   por lo que $e(x)$ es la unidad en $\mathcal{C}$. En consecuencia, podemos deducir la existencia a partir de (2). Veamos ahora la unicidad. Por (2), si tenemos dos elementos idempotentes $e_1(x)$ y $e_2(x)$ que generan $\mathcal{C}$, ambos han de ser unidades, y en consecuencia se tiene que $e_1(x) = e_1(x)e_2(x) = e_2(x)$, con lo que podemos deducir la unicidad.

□

Deducimos por tanto que un método para encontrar el idempotente generador $e(x)$ de un código cíclico $\mathcal{C}$ a partir del polinomio generador $g(x)$ es resolver la ecuación

$$1 = a(x)g(x) + b(x)h(x)$$

para $a(x)$ utilizando el algoritmo de Euclides, donde $h(x) = (x^n - 1)/g(x)$. Entonces, reduciendo $a(x)g(x)$ módulo $x^n - 1$ obtenemos el idempotente $e(x)$ que buscamos. De hecho, el siguiente teorema nos muestra además que podemos obtener $g(x)$ a partir de $e(x)$.

TEOREMA 2.4.2. *Sea $\mathcal{C}$ un código cíclico sobre $\mathbb{F}_q$ con idempotente generador $e(x)$. Entonces, el polinomio generador de $\mathcal{C}$ es $g(x) = \operatorname{mcd}(e(x), x^n - 1)$, calculado en $\mathbb{F}_q[x]$.*

*Demostración.* Sea $d(x) = \operatorname{mcd}(e(x), x^n - 1)$ en $\mathbb{F}_q[x]$ y sea $g(x)$ el polinomio generador de $\mathcal{C}$. Como $d(x) \mid e(x)$, podemos expresarlo como $e(x) = d(x)k(x)$ para algún $k(x) \in \mathbb{F}_q[x]$. Por tanto cada elemento de $\mathcal{C} = \langle e(x) \rangle$ es también múltiplo de $d(x)$, por lo que $\mathcal{C} \subset \langle d(x) \rangle$. Por el teorema 2.3.4 tenemos que en $\mathbb{F}_q[x]$, $g(x) \mid (x^n - 1)$ y que $g(x) \mid e(x)$, ya que $e(x) \in \mathcal{C}$. Luego, por la proposición (TODO: proposición, ejercicio 158) tenemos que $g(x) \mid d(x)$ y en consecuencia $d(x) \in \mathcal{C}$. Por tanto, $\langle d(x) \rangle \subseteq \mathcal{C}$ y deducimos entonces que $\mathcal{C} = \langle d(x) \rangle$. Como $d(x)$ es divisor mónico de $x^n - 1$ y genera a $\mathcal{C}$, necesariamente $d(x) = g(x)$ por el corolario 2.1.23. □

El siguiente teorema nos muestra —igual que para el polinomio generador— que el idempotente generador y sus primeros $k - 1$ desplazamientos cíclicos forman una base de un código cíclico.

TEOREMA 2.4.3. *Sea $\mathcal{C}$ un $[n, k]$ código cíclico con idempotente generador $e(x) = \sum_{i=0}^{n-1} e_i x^i$. Entonces, la matriz $k \times n$*

$$\begin{pmatrix} e_0 & e_1 & e_2 & \cdots & e_{n-2} & e_{n-1} \\ e_{n-1} & e_0 & e_1 & \cdots & e_{n-3} & e_{n-2} \\ & & & \vdots & & \\ e_{n-k+1} & e_{n-k+2} & e_{n-k+3} & \cdots & e_{n-k-1} & e_{n-k} \end{pmatrix}$$

*es una matriz generadora de $\mathcal{C}$.*

*Demostración.* Probar este resultado equivale a probar que el conjunto $\{e(x), xe(x), \ldots, x^{k-1}e(x)\}$ es una base de $\mathcal{C}$. Entonces, solo hay que probar que si $a(x) \in \mathbb{F}_q[x]$ tiene grado menor que $k$, tal que $a(x)e(x) = 0$, se tiene que $a(x) = 0$. Sea $g(x)$ el polinomio generador de $\mathcal{C}$. Si $a(x)e(x) = 0$, entonces $0 = a(x)e(x)g(x) = a(x)g(x)$, tal que $e(x)$ es la unidad de $\mathcal{C}$ según el teorema 2.4.1, y por tanto, si $a(x)$ no es cero estaríamos contradiciendo el teorema 2.3.4. □

Estamos ya en disposición de describir un conjunto especial de idempotentes, denominados *idempotentes primitivos*. Estos elementos, una vez conocidos, nos permitirán obtener todos los idempotentes en $\mathcal{R}_n$, y en consecuencia, todos los códigos cíclicos.

Sea $x^n - 1 = f_1(x) \cdots f_s(x)$, donde cada polinomio $f_i(x)$ es irreducible sobre $\mathbb{F}_q$ para $1 \leq i \leq s$. Todos los $f_i(x)$ son distintos, pues $x^n - 1$ tiene raíces distintas. Sea $\widehat{f}_i(x) = (x^n - 1)/f_i(x)$. En el teorema * veremos que los ideales $\langle \widehat{f}_i(x) \rangle$ de $\mathcal{R}_n$ son los ideales minimales de $\mathcal{R}_n$. Al idempotente generador de $\langle \widehat{f}_i(x) \rangle$ lo denotaremos por $\widehat{e}_i(x)$. Los elementos idempotentes $\widehat{e}_1(x), \ldots, \widehat{e}_s(x)$ se denominan *primitivos idempotentes* de $\mathcal{R}_n$.

TEOREMA 2.4.4. *En $\mathcal{R}_n$ se verifican las siguientes afirmaciones.*

1. *Los ideales $\langle \widehat{f}_i(x) \rangle$ para cada $1 \leq i \leq s$ son todos los ideales minimales de $\mathcal{R}_n$.*
2. *$\mathcal{R}_n$ es el espacio vectorial suma directa de todos los $\langle \widehat{f}_i(x) \rangle$ para $1 \leq i \leq s$.*
3. *Si $i \neq j$ entonces $\widehat{e}_i(x)\widehat{e}_j(x) = 0$ en $\mathcal{R}_n$.*
4. *La suma $\sum_{i=1}^{s} \widehat{e}_i(x) = 1$ en $\mathcal{R}_n$.*
5. *Los únicos idempotentes en $\langle \widehat{f}_i(x) \rangle$ son $0$ y $\widehat{e}_i(x)$.*
6. *Si $e(x)$ es un idempotente no nulo en $\mathcal{R}_n$, entonces existe un subconjunto $T$ de $\{1, 2, \ldots, s\}$ tal que $e(x) = \sum_{i \in T} \widehat{e}_i(x)$ y $\langle e(x) \rangle = \sum_{i \in T} \langle \widehat{f}_i(x) \rangle$.*

*Demostración.* □

El teorema * nos muestra que los ideales minimales del teorema * son extensiones de cuerpos de $\mathbb{F}_q$.

TEOREMA 2.4.5. *Sea $\mathcal{M}$ un ideal minimal de $\mathcal{R}_n$. Entonces $\mathcal{M}$ es una extensión de cuerpos de $\mathbb{F}_q$.*

*Demostración.* □

A continuación vamos a describir una permutación que lleva idempotentes de $\mathcal{R}_n$ en idempotentes de $\mathcal{R}_n$. Sra $a$ un entero tal que $\operatorname{mcd}(a, n) = 1$. La función $\mu_a$ definida sobre $\{0, 1, \ldots, n-1\}$ por $i\mu_a \equiv ia \bmod n$ es una permutación de las posiciones de coordenadas $\{0, 1, \ldots, n-1\}$ de un código cíclico de longitud $n$ y se denomina *multiplicador*. Dado que los códigos cíclicos de longitud $n$ se representan como ideales de $\mathcal{R}_n$, para $a > 0$ es conveniente interpretar que $\mu_a$ actúa sobre $\mathcal{R}_n$ como

$$f(x)\mu_a \equiv f(x^a) \bmod x^n - 1. \tag{2.1}$$

Esta ecuación es consistente con la definición original de $\mu_a$ pues $x^i\mu_a = x^{ia} = x^{ia+jn}$ en $\mathcal{R}_n$ para un entero $j$ tal que $0 \leq ia + jn$, pues $x^n = 1$ en $\mathcal{R}_n$. En otras palabras, $x^i\mu_a = x^{ia \bmod n}$. Si $a < 0$ podemos dar significado a $f(x^a)$ en $\mathcal{R}_n$ definiendo $x^i\mu_a = x^{ia \bmod n}$, donde $0 \leq ia \bmod n < n$. Con esta interpretación la ecuación (2.1) es consistente con la definición original de $\mu_a$ cuando $a < 0$.

Como vimos en la sección 2.3.1, si $t = \operatorname{ord}_n(q)$ entonces $\mathbb{F}_{q^t}$ es un cuerpo de descomposición de $x^n - 1$. Por tanto, $\mathbb{F}_{q^t}$ contiene una enésima raíz primitiva de la unidad $\alpha$, y $x^n - 1 = \prod_{i=0}^{n-1}(x - \alpha^i)$ es la factorización de $x^n - 1$ sobre $\mathbb{F}_{q^t}$. De hecho, $x^n - 1 = \prod_s M_{\alpha^s}(x)$ es la factorización de $x^n - 1$ en factores irreducibles sobre $\mathbb{F}_q$, donde $s$ varía en un conjunto de representantes de las clases $q$-ciclotómicas módulo $n$.

Sea $\mathcal{C}$ un código cíclico en $\mathcal{R}_n$ con polinomio generador $g(x)$. Por los teoremas * y 2.3.4(7), $g(x) = \prod_s M_{\alpha^s}(x) = \prod_s \prod_{i \in C_s}(x - \alpha^i)$, donde $s$ de nuevo varía en un conjunto $C_s$ de representantes de las clases $q$-ciclotómicas módulo $n$. Sea $T = \cup_s C_s$ la unión de estas clases $q$-ciclotómicas. Las raíces de la unidad $\mathcal{Z} = \{\alpha^i \mid i \in T\}$ se denominan los *ceros* del código cíclico $\mathcal{C}$ y los elementos $\{\alpha^i \mid i \notin T\}$, los *elementos no nulos* de $\mathcal{C}$. El conjunto $T$ se denomina *conjunto característico* de $\mathcal{C}$. Se deduce del teorema 2.3.4 que $c(x)$ pertenece a $\mathcal{C}$ si y solo si $c(\alpha^i) = 0$ para cada $i \in T$. Es importante observar que $T$, y por ello tanto el conjunto de ceros como el de elementos distintos de cero, determinan por completo el polinomio generador $g(x)$. Del teorema 2.3.4 concluimos que la dimensión de $\mathcal{C}$ es $n - |T|$, pues $|T|$ es el grado de $g(x)$.

TEOREMA 2.4.6. *Sea $\alpha$ una raíz primitiva de la unidad en una extensión de cuerpos de $\mathbb{F}_q$ y sea $\mathcal{C}$ un código cíclico de longitud $n$ sobre $\mathbb{F}_q$ con conjunto característico $T$ y polinomio generador $g(x)$. Se verifica que:*

1. *El polinomio generador se puede expresar como $g(x) = \prod_{i \in T}(x - \alpha^i)$.*
2. *Una palabra código $c(x) \in \mathcal{R}_n$ está en $\mathcal{C}$ si y solo si $c(\alpha^i) = 0$ para todo $i \in T$.*
3. *La dimensión de $\mathcal{C}$ es $n - |T|$.*

## 2.5 CÓDIGOS BCH

En esta sección vamos a estudiar los códigos BCH, un tipo de códigos cíclicos que permiten ser diseñados con una capacidad de corrección concreta. Como ya sabemos, para cualquier tipo de código es importante determinar la distancia mínima si queremos determinar su capacidad de corrección de errores. A este respecto es útil disponer de cotas en la distancia mínima, especialmente cotas inferiores, pues son las que maximizan la capacidad de corrección. Existen varias cotas conocidas para la distancia mínima de un código cíclico, pero nos vamos a centrar en la llamada *cota de Bose-Ray-Chaudhuri-Hocquenghem*, usualmente abreviada como *cota BCH*. Esta cota es esencial para comprender la definición de los códigos BCH que estudiamos en esta sección. La cota BCH va a depender de los ceros del código, concretamente en la posibilidad de encontrar cadenas de ceros «consecutivos».

En lo que sigue vamos a consdierar un código cíclico $\mathcal{C}$ de longitud $n$ sobre $\mathbb{F}_q$ y $\alpha$ una enésima raíz primitiva de la unidad en $\mathbb{F}_{q^t}$, donde $t = \mathrm{ord}_n(q)$. Recordemos que $T$ es un conjunto característico de $\mathcal{C}$ siempre y cuando los ceros de $\mathcal{C}$ sean $\{\alpha^i \mid i \in T\}$. Por tanto $T$ ha de ser una unión de clases $q$-ciclotómicas módulo $n$. Decimos que $T$ contiene un conjunto de $s$ *elementos consecutivos* si existe un conjunto $\{b, b+1, \dots, b+s-1\}$ de $s$ enteros consecutivos tal que

$$\{b, b+1, \dots, b+s-1\} \text{ mód } n = S \subseteq T.$$

Antes de proceder con la cota BCH vamos a enunciar un lema —que será utilizado en la demostración de dicha cota— sobre el determinante de una matriz de Vandermonde. Sean $\alpha_1, \dots, \alpha_s$ elementos de un cuerpo $\mathbb{F}$. La matriz $s \times s$ $V = (v_{i,j})$, onde $v_{i,j} = \alpha_j^{i-1}$ se denomina *matriz de Vandermonde*. Observamos que la transpuesta de una matriz de Vandermonde es otra matriz de Vandermonde.

LEMA 2.5.1. *El determinante de una matriz Vandermonde $V$ viene dado por* $\det V = \prod_{1 \le i < j \le s} (\alpha_j - \alpha_i)$. *En particular, $V$ es no singular si los elementos $\alpha_1, \dots, \alpha_s$ son todos diferentes dos a dos.*

Estamos ya en condiciones de presentar y demostrar el teorema de la cota BCH.

TEOREMA 2.5.2. (COTA BCH.) *Sea $\mathcal{C}$ un código cíclico de longitud $n$ sobre $\mathbb{F}_q$ con conjunto característico $T$. Supongamos que $\mathcal{C}$ tiene peso mínimo $d$. Asumamos que $T$ contiene $\delta - 1$ elementos consecutivos para algún entero $\delta$. Entonces, $d \ge \delta$.*

*Demostración.* □

Los códigos BCH son códigos cíclicos diseñados para aprovechar la cota BCH. Nos gustaría poder construir un código cíclico $\mathcal{C}$ de longitud $n$ sobre $\mathbb{F}_q$ que tenga a la vez un peso mínimo grande y una dimensión grande. Tener un peso mínimo grande, basándonos en la cota BCH, se puede conseguir escogiendo un conjunto característico para $\mathcal{C}$ que tenga un gran número de elementos consecutivos.

Como la dimensión de $\mathcal{C}$ es $n - |T|$ por el teorema 2.4.6, nos gustaría que $|T|$ fuese tan pequeño como sea posible. Por tanto, si quisiesemos que $\mathcal{C}$ tenga distancia mínima de al menos $\delta$, podemos escoger un conjunto característico tan pequeño como sea posible que sea una unión de clases $q$-ciclotómicas con $\delta - 1$ elementos consecutivos.

Sea $\delta$ un entero tal que $2 \le \delta \le n$. Un *código BCH* $\mathcal{C}$ sobre $\mathbb{F}_q$ de longitud $n$ y *distancia mínima prevista* $\delta$ es un código cíclico con conjunto característico

$$T = C_b \cup C_{b+1} \cup \cdots \cup C_{b+\delta-2}, \tag{2.2}$$

donde $C_i$ es la clase $q$-ciclotómica módulo $n$ que contiene a $i$. Por la cota BCH este código tiene distancia mínima prevista al menos $\delta$.

TEOREMA 2.5.3. *Un código BCH de distancia mínima prevista $\delta$ tiene peso mínimo de al menos $\delta$.*

*Demostración.* El conjunto característico 2.2 tiene al menos $\delta - 1$ elementos. El resultado se deduce de la cota BCH. □

Al variar el valor de $b$ obtenemos distintos códigos con distancias mínimas y dimensiones diferentes. Cuando $b = 1$ el código $\mathcal{C}$ se dice que es un código BCH *en sentido estricto*. Como con cualquier código cíclico, si $n = q^t - 1$ entonces $\mathcal{C}$ es un código BCH *primitivo*. En la sección siguiente vamos a estudiar un algoritmo de decodificación que permite aprovechar las ventajas de los códigos BCH.

## 2.6 ALGORITMO DE PETERSON-GORENSTEIN-ZIERLER

El algoritmo de Peterson-Gorenstein-Zierler —de ahora en adelante, algoritmo PGZ— es un algoritmo de decodificación de códigos BCH que permite corregir hasta $t = \lfloor (\delta - 1)/2 \rfloor$ errores. Fue desarrollado originalmente en 1960 por Peterson (Peterson, 1960) para decodificar códigos BCH binarios, pero fue generalizado poco después por Gorenstein y Zierler para códigos no binarios (Gorenstein & Zierler, 1961).

El objetivo es obtener el mensaje original $c(x)$ a partir de un mensaje recibido $y(x)$, para lo que hay que hallar primero los errores $e(x)$ que se han producido en la transmisión, de forma que $c(x) = y(x) - e(x)$. El vector de errores ha de tener peso $v \leq t$, ya que no podemos corregir más errores de los que el código permite. Vamos a considerar que los errores se han producido en coordenadas desconocidas $k_1, k_2, \ldots, k_v$, de forma que el vector de errores lo podemos expresar como

$$e(x) = e_{k_1} x^{k_1} + e_{k_2} x^{k_2} + \cdots + e_{k_v} x^{k_v}.$$

Como nuestro objetivo es determinar $e(x)$ tenemos que hallar:

— las *coordenadas de error* $k_j$;
— las *magnitudes de error* $e_{k_j}$.

Vamos a estudiar a continuación el desarrollo teórico y la justificación del funcionamiento del algoritmo para después dar una versión del mismo esquematizada en pseudocódigo. Comenzamos observando que por el teorema 2.4.6 un elemento $c(x) \in \mathcal{C}$ si y solo si $c(\alpha^i) = 0$ para todo

$i \in T$, y en nuestro caso particular, dado que $t = \lfloor (\delta - 1)/2 \rfloor$ y $T$ contiene a $\{1, 2, \dots, \delta - 1\}$, se tiene que

$$y(\alpha^i) = c(\alpha^i) + e(\alpha^i) = e(\alpha^i)$$

para todo $1 \leq i \leq 2t$. Estas ecuaciones van a ser fundamentales para encontrar el error $e(x)$. Vamos a llamar *síndrome* $s_i$ de $y(x)$ al elemento de $\mathbb{F}_q^m$ dado por $s_i = y(\alpha^i)$. El primer paso del algoritmo es encontrar los síndromes para todo $1 \leq i \leq 2t$. Estos síndromes nos conducen a un sistema de ecuaciones en el que se encuentran las coordenadas de error $k_j$ y las magnitudes de error $e_{k_j}$. Desarrollando lo anterior podemos expresar los síndromes como

$$s_i = y(\alpha^i) = \sum_{j=1}^{v} e_{k_j} (\alpha^i)^{k_j} = \sum_{j=1}^{v} e_{k_j} (\alpha^{k_j})^i \tag{2.3}$$

para todo $1 \leq i \leq 2t$. A fin de simplificar la notación, para $1 \leq j \leq v$ definimos:

— $E_j = e_{k_j}$, que llamaremos *magnitud de error en la coordenada* $k_j$, y
— $X_j = \alpha^{k_j}$, que llamaremos *número de coordenada de error correspondiente a la coordenada* $k_j$.

Observamos que al conocer $X_j$ conocemos de forma unívoca la coordenada de error $k_j$, ya que si $\alpha^i = \alpha^k$ para $i$ y $k$ entre 0 y $n - 1$, entonces $i = k$. Con la notación que hemos descito la igualdad (2.3) la podemos escribir como

$$S_i = \sum_{j=1}^{v} E_j X_j^i, \quad \text{para } 1 \leq i \leq 2t, \tag{2.4}$$

lo que nos conduce al sistema de ecuaciones:

$$\begin{cases} S_1 = E_1 X_1 + E_2 X_2 + \cdots + E_v X_v, \\ S_2 = E_1 X_1^2 + E_2 X_2^2 + \cdots + E_v X_v^2, \\ S_3 = E_1 X_1^3 + E_2 X_2^3 + \cdots + E_v X_v^3, \\ \quad \vdots \\ S_1 = E_1 X_1^{2t} + E_2 X_2^{2t} + \cdots + E_v X_v^{2t}. \end{cases} \tag{2.5}$$

De este sistema desconocemos tanto los valores de los $X_j$ como los de los $E_j$, pero es que además no es lineal para los $X_j$. Como no podemos resolverlo directamente vamos a tratar de encontrar otra forma con la que calcular los valores $X_j$ y utilizarlos para resolver el sistema lineal que forman los $E_j$. Para ello vamos a buscar un sistema lineal que dependa de otras variables $\sigma_1, \dots, \sigma_v$ que nos conduzca a los valores $X_j$. Definimos el *polinomio localizador de errores* $\sigma(x)$ como

$$\sigma(x) = (1 - xX_1)(1 - xX_2) \dots (1 - xX_v) = 1 + \sum_{i=1}^{v} \sigma_i x^i.$$

Como vemos inmediatamente por su definición, las raíces de $\sigma(x)$ son los inversos de los números de coordenadas de error. Por tanto,

$$\sigma(X_j^{-1}) = 1 + \sigma_1 X_j^{-1} + \sigma_2 X_j^{-2} + \cdots + \sigma_v X_j^{-v} = 0$$

para $1 \leq j \leq v$. Si multiplicamos a ambos lados de la expresión por $E_j X_j^{i+v}$ obtenemos

$$E_j X_j^{i+v} + \sigma_1 E_j X_j^{i+v-1} + \cdots + \sigma_v E_j X_j^{i} = 0$$

para todo $i$. Si sumamos para todo $j$ en $1 \leq j \leq v$ tenemos

$$\sum_{j=1}^{v} E_j X_j^{i+v} + \sigma_1 \sum_{j=1}^{v} E_j X_j^{i+v-1} + \cdots + \sigma_v \sum_{j=1}^{v} E_j X_j^{i} = 0.$$

Lo que hemos obtenido en estas sumas son los síndromes descritos en (2.4), ya que $1 \leq i$ y $i + v \leq 2t$. Como $v \leq t$ la expresión anterior se convierte en

$$S_{i+v} + \sigma_1 S_{i+v-1} + \sigma_2 S_{i+v-2} + \cdots + \sigma_v S_i = 0,$$

que equivale a

$$\sigma_1 S_{i+v-1} + \sigma_2 S_{i+v-2} + \cdots + \sigma_v S_i = -S_{i+v},$$

para todo $1 \leq i \leq v$. Por tanto podemos encontrar los $\sigma_k$ si resolvemos el sistema de ecuaciones dado por:

$$\begin{pmatrix} S_1 & S_2 & S_3 & \dots & S_{v-1} & S_v \\ S_1 & S_2 & S_3 & \dots & S_{v-1} & S_v \\ S_1 & S_2 & S_3 & \dots & S_{v-1} & S_v \\ & & \vdots & & & \\ S_1 & S_2 & S_3 & \dots & S_{v-1} & S_v \end{pmatrix} \begin{pmatrix} \sigma_v \\ \sigma_{v-1} \\ \sigma_{v-2} \\ \vdots \\ \sigma_1 \end{pmatrix} = \begin{pmatrix} -S_{v+1} \\ -S_{v+2} \\ -S_{v+3} \\ \vdots \\ -S_{2v} \end{pmatrix}.$$

La dificultad de este paso es que desconocemos el valor de $v$ (el número de errores), por lo que vamos a realizar un procedimiento iterativo. Suponemos que nuestro número de errores es $\mu = t$, que es el máximo que podemos corregir. Tenemos que quedarnos con el menor valor de $v$ que sea posible. Para ello tenemos en cuenta que la matriz

$$M_\mu = \begin{pmatrix} S_1 & S_2 & \dots & S_\mu \\ S_2 & S_3 & \dots & S_{\mu+1} \\ & & \vdots & \\ S_\mu & S_{\mu+1} & \dots & S_{2\mu-1} \end{pmatrix}$$

será no singular si $\mu = v$ y singular si $\mu > v$. Así, si $M_\mu$ es singular reducimos el valor de $\mu$ en 1, $\mu = \mu - 1$ y probamos de nuevo si $M_\mu$ es singular. Repetimos hasta encontrar una matriz que no sea singular. Ese valor $\mu$ será

el número de errores $v$. Conocido el tamaño podemos resolver el sistema y obtener los valores $\sigma_k$. Ahora solo tenemos que deshacer el camino que hemos recorrido hasta ahora. Conocidos los $\sigma_k$ podemos determinar $\sigma(x)$ y con él, sus raíces, utilizando el procedimiento que queramos, usualmente, calculando reiteradamente $\sigma(\alpha^i)$ para $0 \leq i < n$ hasta encontrarlas. Como ya dijimos, si las invertimos hallaremos los valores de $X_j$, y con ellos ya podemos resolver el sistema (2.5), obteniendo así los valores de los $E_j$. Conocidos todos los valores de $X_j$ y $E_j$ podemos obtener los de $k_j$ y $e_{k_j}$, con los que podemos determinar el vector de error $e(x)$. Ya solo queda restar $y(x) - e(x)$ para obtener el mensaje original.

En resumen, el algoritmo consiste en:

1. Determinar los síndromes del mensaje recibido.
2. Encontrar el polinomio localizador.
3. Hallar las raíces del polinomio localizador e invertirlas para obtener las coordenadas de error $k_j$.
4. Utilizar estos inversos para resolver el sistema de ecuaciones formado por los síndromes, obteniendo así las magnitudes de error $e_{k_j}$.
5. Hallar el vector de error $e(x)$ y restárselo al mensaje $y(x)$.

Hemos expresado en el algoritmo 1 el algoritmo PGZ en pseudocódigo siguiendo este esquema. A partir de él se ha realizado una implementación en el sistema Sage que puede consultarse en el anexo A.

3 Aquí estamos describiendo una matriz por sus entradas, las filas varían en $i$, y las columnas, en $j$.

**Entrada:** el código $\mathcal{C}$, el mensaje recibido $y(x)$
**Salida:** el mensaje decodificado $c(x)$

$\delta \longleftarrow$ distancia designada de $\mathcal{C}$
$t \longleftarrow \lfloor (\delta - 1)/2 \rfloor$
$g \longleftarrow$ polinomio generador de $\mathcal{C}$
$\alpha \longleftarrow$ raíz primitiva del cuerpo de descomposición usado para generar el conjunto característico de $\mathcal{C}$
// Paso 1: calcular síndromes
**para** $1 \leq i \leq 2t$ **hacer**
  $S_i \longleftarrow y(\alpha^i)$
**fin**
// Paso 2: hallar polinomio localizador
$\mu \longleftarrow t$
$M_\mu \longleftarrow (S_{i+j-1})_{i,j}{}^{3}, 1 \leq i, j \leq \mu$
**mientras** $M_\mu$ *es no singular* **hacer**
  $\mu \longleftarrow \mu - 1$
  $M_\mu \longleftarrow (S_{i+j-1})_{i,j}, 1 \leq i, j \leq \mu$
**fin**
$v \longleftarrow \mu$
$\sigma \longleftarrow (\sigma_{v-i+1})_i, 1 \leq i \leq v$
$b_\mu \longleftarrow (-S_{v+i})_i, 1 \leq i \leq v$
$\sigma_k \longleftarrow$ soluciones del sistema $M_\mu \sigma = b_\mu$
$\sigma(x) \longleftarrow 1 + \sum_{i=1}^{v} \sigma_i x^i$
// Paso 3: obtener las coordenadas de error
$r_k \longleftarrow$ raíces de $\sigma(x)$
$X_j \longleftarrow r_j^{-1}$
$k_j \longleftarrow \log_\alpha(X_j)$
// Paso 4: obtener las magnitudes de error
$M_S \longleftarrow (X_j^i)_{i,j}, 1 \leq i, j \leq v$
$E \longleftarrow (E_i)_i, 1 \leq i \leq v$
$b_S \longleftarrow (S_i)_i, 1 \leq i \leq v$
$E_k \longleftarrow$ soluciones del sistema $M_S E = b_S$
// Paso 5: calcular el mensaje original
$e(x) \longleftarrow \sum_{j=1}^{v} E_i x^{k_i}$
$c(x) = y(x) - e(x)$

**Algoritmo 1:** Peterson-Gorenstein-Zierler para códigos cíclicos.

# 3

# ANILLOS DE POLINOMIOS DE ORE

En esta sección vamos a hablar sobre los anillos de polinomios de Ore, que serán la base de los códigos cíclicos sesgados. [Introducir nota histórica] Primero vamos a dar la definición general, sin detenernos a justificar su construcción, pues acto seguido vamos a centrarnos en el caso que nos va a ocupar cuando trabajemos con códigos cíclicos sesgados. Las definiciones y el desarrollo teórico seguidos en esta sección proceden de (citas).

DEFINICIÓN 3.0.1. Sea $R$ un anillo, $\sigma$ un endomorfismo de $R$ y $\delta$ una $\sigma$-*derivación* de $R$, es decir, $\delta$ es un homomorfismo de grupos abelianos tal que para $a, b \in R$ se verifica que

$$\delta(ab) = (\sigma a)(\delta b) + (\delta a)b.$$

Entonces, el anillo $R[t; \sigma, \delta]$ de los polinomios en $R[t]$ de la forma

$$a_0 + a_1 t + \cdots + a_n t^n,$$

donde $a_i \in R$, con la igualdad y suma usuales, y en el que la multiplicación verifica la relación

$$ta = (\sigma a)t + \delta a, \qquad a \in R,$$

se conoce como *anillo de polinomios de Ore* o *anillos de polinomios torcidos*.

Para comprobar que $R[t; \sigma, \delta]$ es un anillo tendríamos que ver que efectivamente con las operaciones que hemos dado se verifican todas las propiedades de los anillos. Puesto que hemos usado la suma usual de los polinomios, bastaría probar que se verifica la propiedad asociativa para la multiplicación que hemos definido. No vamos a entrar en detalle, pues no es el objetivo de este trabajo el estudio de los anillos de polinomios de Ore en general.

Trabajar con códigos cíclicos sesgados requiere del estudio del anillo $\mathbb{F}_q[x, \sigma]$, con $\sigma$ un automorfismo. Por tanto, nos vamos a centrar los anillos de polinomios de Ore en los que $R = \mathbb{F}_q$ —cuerpo finito de $q$ elementos—, hemos llamado $x$ a $t$, $\sigma$ es un automorfismo y $\delta = 0$. En estos anillos la multiplicación verifica la relación

$$xa = (\sigma a)x, \qquad a \in R.$$

Es este caso particular el que sí vamos a estudiar en profundidad, justificando que, como ya hemos adelantado, se trata de un anillo.

Vamos a ver por inducción que, como podemos intuir, $x^n a = (\sigma^n a)x^n$. Estudiado el caso base anterior y supuesto que se verifica la igualdad para $n-1$, para $n$ tenemos que

$$x^n a = xx^{n-1}a = x(\sigma^{n-1}a)x^{n-1} = \sigma(\sigma^{n-1}a)x^{n-1}x = (\sigma^n a)x^n.$$

Ahora definimos

$$(ax^n)(bx^m) = a(\sigma^n b)x^{n+m},$$

con lo que, junto a la propiedad distributiva podemos definir el producto de polinomios en $x$ como

$$(\sum a_n x^n)(\sum b_m x^m) = \sum (a_n x^n)(b_m x^m).$$

Para comprobar que $\mathbb{F}_q[x,\sigma]$ es un anillo, como ya hemos comentado en el caso general, necesitamos comprobar que se verifica la propiedad asociativa para la multiplicación. Comprobar esta afirmación directamente es tedioso, por lo que Jacobson propone (cita) demostrarlo utilizando una representación matricial de los elementos.

A continuación vemos que, partiendo de que $\mathbb{F}_q$ es en particular un anillo de división, $\mathbb{F}_q[x,\sigma]$ es un dominio de integridad no conmutativo. Por tanto, $\mathbb{F}_q[x,\sigma]$ no tiene divisores de cero distintos del cero, por lo que es un dominio de integridad no conmutativo, como habíamos afirmado.

Podemos definir algoritmos de división en $\mathbb{F}_q[x,\sigma]$ tanto a la izquierda como a la derecha (cita) —descritos en los algoritmos 2 y 3—, de forma que para cada $f(x), g(x) \in \mathbb{F}_q[x,\sigma]$ —con $g(x) \neq 0$— existen elementos $q(x), r(x)$ únicos, con $\mathrm{gr}(r) < \mathrm{gr}(g)$ tales que al dividir por la izquierda obtenemos

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

y al dividir por la derecha,

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

Cuando dividimos por la izquierda (respectivamente por la derecha) el polinomio $g(x)$ se le llama *cociente por la izquierda* (*derecha*) y a $r(x)$, *resto por la izquierda* (*derecha*).

**Entrada:** polinomios $f, g \in \mathbb{F}_q[x, \sigma]$ con $g \neq 0$
**Salida:** polinomios $q, r \in \mathbb{F}_q[x, \sigma]$ tales que $f = qg + r$, y $\operatorname{gr}(r) < \operatorname{gr}(g)$

$q \longleftarrow 0$
$r \longleftarrow f$
**mientras** $\operatorname{gr}(g) \leq \operatorname{gr}(r)$ **hacer**
  $a \longleftarrow \operatorname{cl}(r)\sigma^{\operatorname{gr}(r)-\operatorname{gr}(g)}(\operatorname{cl}(g)^{-1})$
  $q \longleftarrow q + ax^{\operatorname{gr}(r)-\operatorname{gr}(g)}$
  $r \longleftarrow r - ax^{\operatorname{gr}(r)-\operatorname{gr}(g)}g$
**fin**

**Algoritmo 2:** División por la izquierda en $\mathbb{F}_q[x, \sigma]$

**Entrada:** polinomios $f, g \in \mathbb{F}_q[x, \sigma]$ con $g \neq 0$
**Salida:** polinomios $q, r \in \mathbb{F}_q[x, \sigma]$ tales que $f = gq + r$, y $\operatorname{gr}(r) < \operatorname{gr}(g)$

$q \longleftarrow 0$
$r \longleftarrow f$
**mientras** $\operatorname{gr}(g) \leq \operatorname{gr}(r)$ **hacer**
  $a \longleftarrow \sigma^{-\operatorname{gr}(g)}(\operatorname{cl}(g)^{-1}\operatorname{cl}(r))$
  $q \longleftarrow q + ax^{\operatorname{gr}(r)-\operatorname{gr}(g)}$
  $r \longleftarrow r - gax^{\operatorname{gr}(r)-\operatorname{gr}(g)}$
**fin**

**Algoritmo 3:** División por la derecha en $\mathbb{F}_q[x, \sigma]$

# A

# IMPLEMENTACIÓN EN SAGE DEL ALGORITMO DE PETERSON-GORENSTEIN-ZIERLER

En el listado siguiente se describe una Implementación del algoritmo de Peterson-Gorenstein-Zierler en Sage, utilizando las definiciones de códigos cíclicos y códigos BCH que proporciona.

```
1  DEBUG = false
2
3  def logger(s):
4      if DEBUG:
5          print(s)
6
7  def xentries_m_mu(i, j, s):
8      return s[i+j]
9
10 def xentries_m_syn(i, j, X):
11     return X[j]^(i+1)
12
13 def PGZ(C, y):
14     delta = C.designed_distance()
15     t = floor((delta - 1)/2)
16     g = C.generator_polynomial()
17     a = C.primitive_root()
18
19     logger('raíz primitiva: ' + str(a))
20
21     # Paso 1: calcular los síndromes
22     s = []
23     for i in range(0, 2*t):
24         s.append(y(a^(i+1)))
25
26     logger('síndromes: ' + str(s))
27
28     # Paso 2: probar mu partiendo de t hasta encontrar
           matriz m_mu
29     # que sea no singular. Resolver el sistema para
           encontrar sigma(x)
```

```
30      mu = t
31
32      m_mu = matrix(mu, lambda i, j: xentries_m_mu(i, j, s))
33
34      while m_mu.determinant() == 0:
35          mu = mu - 1
36          m_mu = matrix(mu, lambda i, j: xentries_m_mu(i, j,
                s))
37
38      logger('tamaño de m_mu: ' + str(mu))
39      logger('matriz m_mu: \n' + str(m_mu))
40
41      if m_mu == matrix():
42          print("Se produjeron más de ", t, " errores: no
                podemos decodificar")
43          return
44
45      b_mu = []
46
47      for i in [mu..2*mu-1]:
48          b_mu.append(-s[i])
49
50      b_mu = vector(b_mu)
51
52      logger('vector b_mu: ' + str(b_mu))
53
54      sol_mu = m_mu.solve_right(b_mu)
55
56      logger('matriz de soluciones de m_mu*S = b_mu: ' + str
            (sol_mu))
57
58      sigma = 1
59      l = len(sol_mu) - 1
60      for i in [0..l]:
61          sigma = sigma + sol_mu[len(sol_mu)-1-i]*x^(i+1)
62
63      logger('polinomio localizador sigma(x): ' + str(sigma)
            )
64
65      # Paso 3: encontrar las raíces de sigma(x) e
            invertirlas para
```

```
66      # encontrar los números de posición de error X_j
67      # Obtenemos también los k_j
68      r = sigma.roots()
69
70      logger('raíces de sigma(x): ' + str(r))
71
72      if r == []:
73          print('sigma(x) no tiene raíces, no podemos seguir
                ')
74          return
75
76      l = len(r) - 1
77      X = []
78      for i in [0..l]:
79          X.append(r[i][0]^-1)
80
81      logger('X_j: ' + str(X))
82
83      k = []
84      for i in [0..l]:
85          k.append(log(X[i], a))
86
87      logger('k_j: ' + str(k))
88
89      # Paso 4: Resolvemos las primeras mu ecuaciones del
            sistema de los
90      # síndromes para hallar las magnitudes de error E_j
91
92      m_syn = matrix(mu, lambda i, j: xentries_m_syn(i, j, X
            ))
93
94      b_syn = []
95      for i in [0..mu-1]:
96          b_syn.append(s[i])
97      b_syn = vector(b_syn)
98
99      E = m_syn.solve_right(b_syn)
100
101     # Paso final: hallamos el error y se lo restamos al
            mensaje recibido
102
```

```
103     e = 0
104     for i in [0..mu-1]:
105         e = e + E[i]*x^(k[i])
106
107     logger('error e: ' + str(e))
108
109     c = y - e
110
111     return c
```

**Listado A.1:** Implementación del algoritmo de Peterson-Gorenstein-Zierler en Sage

# BIBLIOGRAFÍA


Dyson, G. (2015). *La catedral de Turing: los orígenes del universo digital*. OCLC: 904326706. Barcelona: Debate.

Cohn, P. M. (1982). *Algebra* (2nd ed.). New York: Wiley.

Cohn, P. M. (1989). *Algebra Vol. 2* (2nd ed., reprint with corr.). OCLC: 832519027. New York: Wiley.

Lidl, R. & Niederreiter, H. (1986). *Introduction to finite fields and their applications*. Cambridge [Cambridgeshire] ; New York: Cambridge University Press.

Huffman, W. C. & Pless, V. (2003). *Fundamentals of error-correcting codes*. doi:10.1017/CBO9780511807077

Podestá, R. (2006). *Introducción a la Teoría de Códigos Autocorrectores*. Universidad Nacional de Córdoba. Recuperado desde https://www.famaf.unc.edu.ar/documents/940/CMat35-3.pdf

Peterson, W. (1960, septiembre). Encoding and error-correction procedures for the bose-chaudhuri codes. *IEEE Transactions on Information Theory*, *6*(4), 459-470. doi:10.1109/TIT.1960.1057586

Gorenstein, D. & Zierler, N. (1961). A class of error-correcting codes in $p^m$ symbols. *J. SIAM*, *9*, 207-214.

## COLOFÓN

Este trabajo comenzó a escribirse en Granada en octubre de 2019 y fue terminado en Rincón de la Victoria en junio de 2020 en unas circunstancias complicadas, pues el mundo se encontraba inmerso en la pandemia de la covid-19.

Este trabajo ha sido compuesto utilizando LaTeX con el estilo del paquete `classicthesis`, desarrollado por André Miede e Ivo Pletikosić e inspirado en el del trascendental libro de Robert Bringhurst, «*The Elements of Typographic Style*». Puede obtenerse una copia de dicho paquete en

https://bitbucket.org/amiede/classicthesis/

Las tipografías utilizadas han sido *EBGaramond* para el cuerpo, *Garamond Math* para las matemáticas, *Open Sans* para las leyendas y *Go Mono* para el código.