

Algoritmo de Peterson-Gorenstein-Zierler para códigos cíclicos sesgados

José María Martín Luque

Universidad de Granada

9 de julio de 2020 - Curso 2019-2020

Índice

Introducción

Teoría de códigos

Polinomios de Ore y códigos cíclicos sesgados

Algoritmo PGZ para códigos RS sesgados

Implementación en SageMath

Conclusiones

Introducción

Transmisión de la información

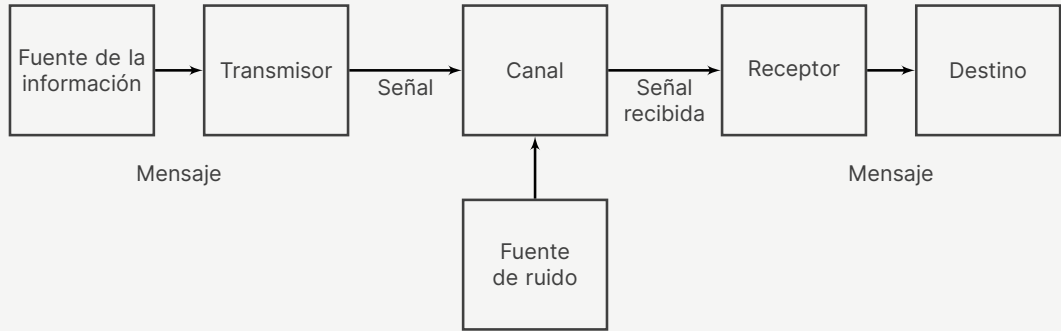


Figura: modelo de comunicación de Shannon

Transmisión de la información

Un código permite expresar información para su transmisión a través de un canal.

El algoritmo de Peterson-Gorenstein-Zierler permite corregir errores producidos en la transmisión de información si se han usado ciertos códigos cíclicos.

Objetivos

1. Exponer y estudiar el algoritmo de Peterson-Gorenstein-Zierler para códigos cíclicos sesgados.

Objetivos

1. Exponer y estudiar el algoritmo de Peterson-Gorenstein-Zierler para códigos cíclicos sesgados.
 - ➔ Estudio de la teoría de códigos lineales.
 - ➔ Estudio de la teoría de polinomios de Ore y sus cocientes.

Objetivos

1. Exponer y estudiar el algoritmo de Peterson-Gorenstein-Zierler para códigos cíclicos sesgados.
 - ➔ Estudio de la teoría de códigos lineales.
 - ➔ Estudio de la teoría de polinomios de Ore y sus cocientes.
2. Implementar sistemas de decodificación en Python usando SageMath.

Contenido

anillos ideales cuerpos finitos cuerpos de descomposición elementos
primitivos clases ciclotómicas anillos de polinomios automorfismos bases
normales **códigos lineales** **códigos cíclicos** códigos BCH PGZ para
códigos BCH códigos RS idempotentes **anillos de polinomios de Ore**
códigos cíclicos sesgados **PGZ para códigos RS sesgados**

Teoría de códigos

¿Qué es un código?

Definición

Un (n, M) código \mathcal{C} sobre el cuerpo \mathbb{F}_q es un subconjunto de tamaño M de \mathbb{F}_q^n .

Los elementos de un código se llaman *palabras código*.

¿Qué es un código?

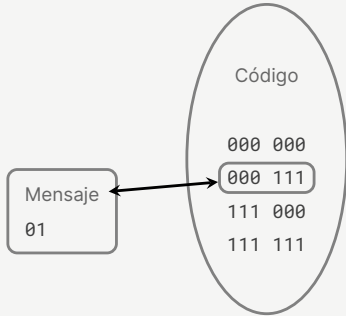


Figura: codificación y decodificación

¿Qué es un código?

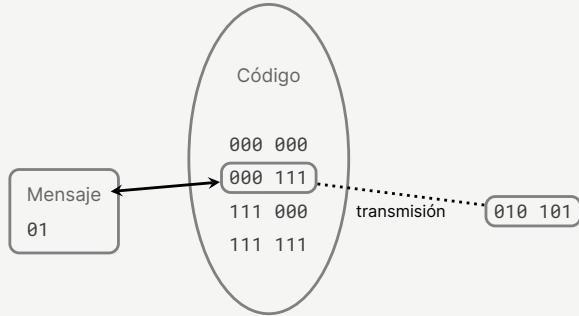


Figura: codificación y decodificación

¿Qué es un código?

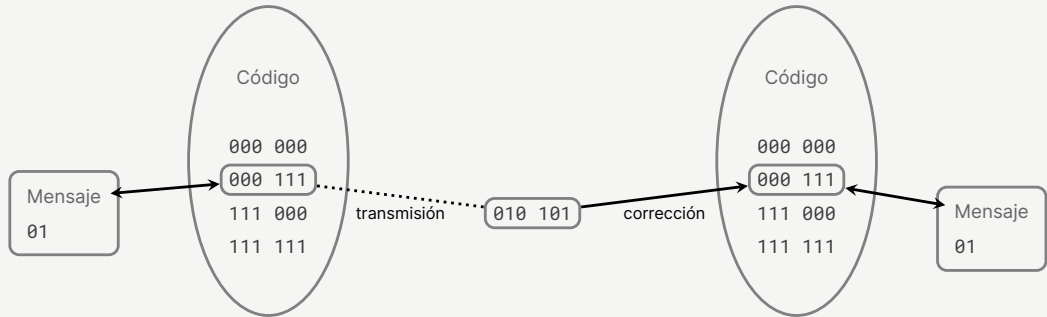


Figura: codificación y decodificación

¿Qué es un código?

Definición

Un (n, M) código \mathcal{C} sobre el cuerpo \mathbb{F}_q es un **subconjunto** de \mathbb{F}_q^n de tamaño M .

Los elementos de un código se llaman *palabras código*.

✗ Es un objeto demasiado sencillo.

Añadiendo estructura

Definición

Un $[n, k]$ código lineal \mathcal{C} de longitud n y dimensión k es un subespacio vectorial de \mathbb{F}_q^n de dimensión k .

Añadiendo estructura

Definición

Un $[n, k]$ código lineal \mathcal{C} de longitud n y dimensión k es un subespacio vectorial de \mathbb{F}_q^n de dimensión k .

✓ Trabajamos con una estructura bien conocida.

Añadiendo estructura

Definición

Un $[n, k]$ código lineal \mathcal{C} de longitud n y dimensión k es un subespacio vectorial de \mathbb{F}_q^n de dimensión k .

- ✓ Trabajamos con una estructura bien conocida.
- ✓ Codificar es sencillo: multiplicar por una matriz (base).

Añadiendo estructura

Definición

Un código lineal \mathcal{C} de longitud n sobre \mathbb{F}_q es *cíclico* si verifica:

$$(c_0, \dots, c_{n-2}, c_{n-1}) \in \mathcal{C} \iff (c_{n-1}, c_0, \dots, c_{n-2}) \in \mathcal{C}$$

Añadiendo estructura

Definición

Un código lineal \mathcal{C} de longitud n sobre \mathbb{F}_q es *cíclico* si verifica:

$$(c_0, \dots, c_{n-2}, c_{n-1}) \in \mathcal{C} \iff (c_{n-1}, c_0, \dots, c_{n-2}) \in \mathcal{C}$$

Podemos asociar los elementos de \mathcal{C} a polinomios mediante una biyección

$$\begin{aligned} v : \mathbb{F}_q^n &\rightarrow \mathbb{F}_q[x]/(x^n - 1) \\ (c_0, c_1, \dots, c_{n-1}) &\mapsto c_0 + c_1x + \dots + c_{n-1}x^{n-1}. \end{aligned}$$

Añadiendo estructura

Definición

Un código lineal \mathcal{C} de longitud n sobre \mathbb{F}_q es *cíclico* si verifica:

$$(c_0, \dots, c_{n-2}, c_{n-1}) \in \mathcal{C} \iff (c_{n-1}, c_0, \dots, c_{n-2}) \in \mathcal{C}$$

Podemos **asociar los elementos de \mathcal{C} a polinomios** mediante una biyección

$$\begin{aligned} v : \mathbb{F}_q^n &\rightarrow \mathbb{F}_q[x]/(x^n - 1) \\ (c_0, c_1, \dots, c_{n-1}) &\mapsto c_0 + c_1x + \dots + c_{n-1}x^{n-1}. \end{aligned}$$

✓ Más estructura: los códigos cíclicos son ideales del cociente $\mathbb{F}_q[x]/(x^n - 1)$.

Añadiendo estructura

Definición

Un código lineal \mathcal{C} de longitud n sobre \mathbb{F}_q es *cíclico* si verifica:

$$(c_0, \dots, c_{n-2}, c_{n-1}) \in \mathcal{C} \iff (c_{n-1}, c_0, \dots, c_{n-2}) \in \mathcal{C}$$

Podemos **asociar los elementos de \mathcal{C} a polinomios** mediante una biyección

$$\begin{aligned} v : \mathbb{F}_q^n &\rightarrow \mathbb{F}_q[x]/(x^n - 1) \\ (c_0, c_1, \dots, c_{n-1}) &\mapsto c_0 + c_1x + \dots + c_{n-1}x^{n-1}. \end{aligned}$$

- ✓ Más estructura: los códigos cíclicos son ideales del cociente $\mathbb{F}_q[x]/(x^n - 1)$.
- ✓ Codificar es sencillo: multiplicar por un polinomio (generador del ideal).

Distancia de un código

Definición

La *distancia de Hamming* de un código \mathcal{C} es el menor número de coordenadas que distinguen a una palabra código de otra.

Polinomios de Ore y códigos cíclicos sesgados

Polinomios de Ore

Definición

Sea \mathbb{F}_q un cuerpo finito y σ un automorfismo de \mathbb{F}_q . Entonces, el anillo $\mathbb{F}_q[x; \sigma]$ de los polinomios usuales de $\mathbb{F}_q[x]$ cuyo producto verifica la relación:

$$xa = \sigma(a)x, \quad a \in \mathbb{F}_q,$$

es un *anillo de polinomios de Ore* o *anillos de polinomios sesgados*.

Polinomios de Ore

Es un anillo no conmutativo.

Pero es un DIP y la aritmética funciona igual, teniendo en cuenta siempre que las operaciones se realizarán a izquierda o a derecha.

Códigos cíclicos sesgados

Códigos cíclicos pero usando polinomios de Ore: la biyección es $\mathfrak{v} : \mathbb{F}_q^n \rightarrow \mathbb{F}_q[x; \sigma]/(x^n - 1)$, donde n es el orden de σ .

Códigos cíclicos sesgados

Códigos cíclicos pero usando polinomios de Ore: la biyección es $\mathfrak{v} : \mathbb{F}_q^n \rightarrow \mathbb{F}_q[x; \sigma]/(x^n - 1)$, donde n es el orden de σ .

Definición

Un *código cíclico sesgado* sobre \mathbb{F}_q es un subespacio vectorial $\mathcal{C} \subseteq \mathbb{F}_q^n$ tal que:

$$(a_0, \dots, a_{n-2}, a_{n-1}) \in \mathcal{C} \iff (\sigma(a_{n-1}), \sigma(a_0), \dots, \sigma(a_{n-2})) \in \mathcal{C}$$


Códigos cíclicos sesgados

Códigos cíclicos pero usando polinomios de Ore: la biyección es $\mathfrak{v} : \mathbb{F}_q^n \rightarrow \mathbb{F}_q[x; \sigma]/(x^n - 1)$, donde n es el orden de σ .

Definición

Un *código cíclico sesgado* sobre \mathbb{F}_q es un subespacio vectorial $\mathcal{C} \subseteq \mathbb{F}_q^n$ tal que:

$$(a_0, \dots, a_{n-2}, a_{n-1}) \in \mathcal{C} \iff (\sigma(a_{n-1}), \sigma(a_0), \dots, \sigma(a_{n-2})) \in \mathcal{C}$$

 Los códigos cíclicos sesgados son ideales del cociente $\mathbb{F}_q[x; \sigma]/(x^n - 1)$.

Códigos RS sesgados

Consideremos una base normal $\{\alpha, \sigma(\alpha), \dots, \sigma^{n-1}(\alpha)\}$ de \mathbb{F}_q y $\beta = \alpha^{-1}\sigma(\alpha)$.

El monomio $x - \beta$ divide a $x^n - 1$ por la derecha, y de hecho

$$x^n - 1 = [x - \beta, x - \sigma(\beta), \dots, x - \sigma^{n-1}(\beta)]_i.$$

Códigos RS sesgados

Consideremos una base normal $\{\alpha, \sigma(\alpha), \dots, \sigma^{n-1}(\alpha)\}$ de \mathbb{F}_q y $\beta = \alpha^{-1}\sigma(\alpha)$.

El monomio $x - \beta$ divide a $x^n - 1$ por la derecha, y de hecho

$$x^n - 1 = [x - \beta, x - \sigma(\beta), \dots, x - \sigma^{n-1}(\beta)]_i.$$

i A los elementos de $\{\beta, \sigma(\beta), \dots, \sigma^{n-1}(\beta)\}$ los llamamos β -raíces.

Códigos RS sesgados

Definición

Un *código RS sesgado* de distancia mínima diseñada δ es un código cíclico sesgado \mathcal{C} tal que $\mathfrak{v}(\mathcal{C})$ está generado por

$$g = [x - \sigma^r(\beta), x - \sigma^{r+1}(\beta), \dots, x - \sigma^{r+\delta-2}(\beta)]_i$$

para algún $r \geq 0$.

Códigos RS sesgados

Definición

Un *código RS sesgado* de distancia mínima diseñada δ es un código cíclico sesgado \mathcal{C} tal que $\mathfrak{v}(\mathcal{C})$ está generado por

$$g = [x - \sigma^r(\beta), x - \sigma^{r+1}(\beta), \dots, x - \sigma^{r+\delta-2}(\beta)]_i$$

para algún $r \geq 0$.

- Tienen distancia $\delta = n - k + 1$, máxima distancia posible para un código de esta longitud y dimensión.

Norma

En un anillo $\mathbb{F}_q[x; \sigma]$ definimos la *norma i -ésima* de un elemento $\gamma \in \mathbb{F}_q$ como

$$N_i(\gamma) = \sigma(N_{i-1}(\gamma))(\gamma) = \sigma^{i-1}(\gamma) \dots \sigma(\gamma)\gamma \quad \text{para } i > 0 \quad \text{y } N_0(\gamma) = 1.$$

Norma

En un anillo $\mathbb{F}_q[x; \sigma]$ definimos la *norma i -ésima* de un elemento $\gamma \in \mathbb{F}_q$ como

$$N_i(\gamma) = \sigma(N_{i-1}(\gamma))(\gamma) = \sigma^{i-1}(\gamma) \dots \sigma(\gamma)\gamma \quad \text{para } i > 0 \quad \text{y } N_0(\gamma) = 1.$$

 Es el equivalente a evaluar un polinomio en álgebra conmutativa.

Norma

En un anillo $\mathbb{F}_q[x; \sigma]$ definimos la *norma i -ésima* de un elemento $\gamma \in \mathbb{F}_q$ como

$$N_i(\gamma) = \sigma(N_{i-1}(\gamma))(\gamma) = \sigma^{i-1}(\gamma) \dots \sigma(\gamma)\gamma \quad \text{para } i > 0 \quad \text{y } N_0(\gamma) = 1.$$

i Es el equivalente a evaluar un polinomio en álgebra conmutativa.

Teorema

Si $f(x) = \sum_{i=0}^n a_i x^{n-i} \in \mathbb{F}_q[x; \sigma]$ y $\gamma \in \mathbb{F}_q$ entonces el resto de dividir $f(x)$ por $(x - \gamma)$ por la derecha es

$$\sum_{i=0}^n a_i N_i(\gamma).$$

Norma

Sea N la matriz formada por las normas de las β -raíces:

$$N = \begin{pmatrix} N_0(\beta) & N_0(\sigma(\beta)) & \cdots & N_0(\sigma^{n-1}(\beta)) \\ N_1(\beta) & N_1(\sigma(\beta)) & \cdots & N_1(\sigma^{n-1}(\beta)) \\ \vdots & \vdots & & \vdots \\ N_{n-1}(\beta) & N_{n-1}(\sigma(\beta)) & \cdots & N_{n-1}(\sigma^{n-1}(\beta)) \end{pmatrix}.$$

Multiplicando los coeficientes de un polinomio por N , $(f_1, \dots, f_{n-1})N$ obtenemos todos los restos de dividir dicho polinomio por cada una de las β -raíces.

Algoritmo PGZ para códigos RS sesgados

Recepción de mensajes

Un mensaje recibido puede expresarse como

$$y(x) = c(x) + e(x)$$

donde $c(x)$ es el mensaje codificado original y $e(x)$ es el error que se ha producido durante la transmisión, que es de la forma

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

Recepción de mensajes

Un mensaje recibido puede expresarse como

$$y(x) = c(x) + e(x)$$

donde $c(x)$ es el mensaje codificado original y $e(x)$ es el error que se ha producido durante la transmisión, que es de la forma

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

- ➡ El algoritmo encuentra el error $e(x)$ para códigos RS sesgados.
- ➡ Puede corregir hasta $t = \left\lfloor \frac{d-1}{2} \right\rfloor$ errores.

Algoritmo

Entrada: el código \mathcal{C} , el mensaje recibido $y = (y_0, \dots, y_{n-1}) \in \mathbb{F}_q^n$ con no más de t errores

Salida: el error $e = (e_0, \dots, e_{n-1})$ tal que $y - e \in \mathcal{C}$

// Paso 1: calcular síndromes

```
1 para  $0 \leq i \leq 2t - 1$  hacer
2   |  $s_i \leftarrow \sum_{j=0}^{n-1} y_j N_j(\sigma^i(\beta))$ 
3 fin
4 si  $s_i = 0$  para todo  $0 \leq i \leq 2t - 1$  entonces
5   | devolver 0
6 fin
```

Algoritmo

Entrada: el código \mathcal{C} , el mensaje recibido $y = (y_0, \dots, y_{n-1}) \in \mathbb{F}_q^n$ con no más de t errores

Salida: el error $e = (e_0, \dots, e_{n-1})$ tal que $y - e \in \mathcal{C}$

// Paso 1: calcular síndromes

```
1 para  $0 \leq i \leq 2t - 1$  hacer  
2   |  $s_i \leftarrow \sum_{j=0}^{n-1} y_j N_j(\sigma^i(\beta))$   
3 fin  
4 si  $s_i = 0$  para todo  $0 \leq i \leq 2t - 1$  entonces  
5   | devolver 0  
6 fin
```

Algoritmo

// Paso 2: hallar polinomio localizador y las coordenadas de error

$$7 \ S^t \leftarrow \left(\sigma^{-j}(s_{i+j}) \sigma^i(\alpha) \right)_{0 \leq i \leq t, 0 \leq j \leq t-1}$$

8 Calcular

$$\text{mepc}(S^t) = \left(\begin{array}{c|c} I_\mu & \\ \hline a_0 \cdots a_{\mu-1} & \\ \hline H' & \end{array} \middle| 0_{(t+1) \times (t-\mu)} \right)$$

$$9 \ \rho = (\rho_0, \dots, \rho_\mu) \leftarrow (-a_0, \dots, -a_{\mu-1}, 1) \text{ y } \rho N \leftarrow (\rho_0, \dots, \rho_\mu, 0, \dots, 0)N$$

$$10 \ \{k_1, \dots, k_v\} \leftarrow \text{coordenadas igual a cero de } \rho N$$

Algoritmo

// Paso 2: hallar polinomio localizador y las coordenadas de error

$$7 \ S^t \leftarrow \left(\sigma^{-j}(s_{i+j}) \sigma^j(\alpha) \right)_{0 \leq i \leq t, 0 \leq j \leq t-1}$$

8 Calcular

$$\text{mepc}(S^t) = \left(\begin{array}{c|c} I_\mu & \\ \hline a_0 \cdots a_{\mu-1} & \\ \hline H' & \end{array} \begin{array}{c} \\ \\ 0_{(t+1) \times (t-\mu)} \end{array} \right)$$

$$9 \ \rho = (\rho_0, \dots, \rho_\mu) \leftarrow (-a_0, \dots, -a_{\mu-1}, 1) \text{ y } \rho N \leftarrow (\rho_0, \dots, \rho_\mu, 0, \dots, 0)N$$

$$10 \ \{k_1, \dots, k_v\} \leftarrow \text{coordenadas igual a cero de } \rho N$$

Algoritmo

// Paso 2: hallar polinomio localizador y las coordenadas de error

$$7 \ S^t \leftarrow \left(\sigma^{-j}(s_{i+j}) \sigma^j(\alpha) \right)_{0 \leq i \leq t, 0 \leq j \leq t-1}$$

8 Calcular

$$\text{mepc}(S^t) = \left(\begin{array}{c|c} I_\mu & \\ \hline a_0 \cdots a_{\mu-1} & \\ \hline H' & \end{array} \middle| 0_{(t+1) \times (t-\mu)} \right)$$

$$9 \ \rho = (\rho_0, \dots, \rho_\mu) \leftarrow (-a_0, \dots, -a_{\mu-1}, 1) \text{ y } \rho N \leftarrow (\rho_0, \dots, \rho_\mu, 0, \dots, 0)N$$

$$10 \ \{k_1, \dots, k_v\} \leftarrow \text{coordenadas igual a cero de } \rho N$$

Algoritmo

11 **si** $\mu \neq v$ **entonces**

12 Calcular

$$M_\rho \leftarrow \begin{pmatrix} \rho_0 & \rho_1 & \dots & \rho_\mu & 0 & \dots & 0 \\ 0 & \sigma(\rho_0) & \dots & \sigma(\rho_{\mu-1}) & \sigma(\rho_\mu) & \dots & 0 \\ 0 & \dots & 0 & \sigma^{n-\mu-1}(\rho_0) & \dots & \dots & \sigma^{n-\mu-1}(\rho_\mu) \end{pmatrix}_{(n-\mu) \times n}$$

13 $N_\rho \leftarrow M_\rho N$

14 $H_\rho \leftarrow \text{mepf}(N_\rho)$

15 $H' \leftarrow$ la matriz obtenida al eliminar las filas de H_ρ distintas de ε_i para algún i

16 $\{k_1, \dots, k_v\} \leftarrow$ las coordenadas de las columnas igual a cero de H'

17 **fin**

Algoritmo

11 **si** $\mu \neq v$ **entonces**

12 Calcular

$$M_\rho \leftarrow \begin{pmatrix} \rho_0 & \rho_1 & \dots & \rho_\mu & 0 & \dots & 0 \\ 0 & \sigma(\rho_0) & \dots & \sigma(\rho_{\mu-1}) & \sigma(\rho_\mu) & \dots & 0 \\ 0 & \dots & 0 & \sigma^{n-\mu-1}(\rho_0) & \dots & \dots & \sigma^{n-\mu-1}(\rho_\mu) \end{pmatrix}_{(n-\mu) \times n}$$

13 $N_\rho \leftarrow M_\rho N$

14 $H_\rho \leftarrow \text{mepf}(N_\rho)$

15 $H' \leftarrow$ la matriz obtenida al eliminar las filas de H_ρ distintas de ε_i para algún i

16 $\{k_1, \dots, k_v\} \leftarrow$ las coordenadas de las columnas igual a cero de H'

17 **fin**

Algoritmo

11 si $\mu \neq v$ entonces

12 Calcular

$$M_\rho \leftarrow \begin{pmatrix} \rho_0 & \rho_1 & \dots & \rho_\mu & 0 & \dots & 0 \\ 0 & \sigma(\rho_0) & \dots & \sigma(\rho_{\mu-1}) & \sigma(\rho_\mu) & \dots & 0 \\ 0 & \dots & 0 & \sigma^{n-\mu-1}(\rho_0) & \dots & \dots & \sigma^{n-\mu-1}(\rho_\mu) \end{pmatrix}_{(n-\mu) \times n}$$

13 $N_\rho \leftarrow M_\rho N$

14 $H_\rho \leftarrow \text{mepf}(N_\rho)$

15 $H' \leftarrow$ la matriz obtenida al eliminar las filas de H_ρ distintas de ε_i para algún i

16 $\{k_1, \dots, k_v\} \leftarrow$ las coordenadas de las columnas igual a cero de H'

17 fin

Algoritmo

Cálculo de las magnitudes de error

Teorema

Las magnitudes de error (e_1, \dots, e_v) son las soluciones del sistema de ecuaciones lineales

$$\underbrace{X \begin{pmatrix} \sigma^{k_1}(\alpha) & \sigma^{k_1+1}(\alpha) & \dots & \sigma^{k_1+v-1}(\alpha) \\ \sigma^{k_2}(\alpha) & \sigma^{k_2+1}(\alpha) & \dots & \sigma^{k_2+v-1}(\alpha) \\ \vdots & \vdots & & \vdots \\ \sigma^{k_v}(\alpha) & \sigma^{k_v+1}(\alpha) & \dots & \sigma^{k_v+v-1}(\alpha) \end{pmatrix}}_{(\Sigma^{v-1})^T} = (\alpha s_0, \sigma(\alpha)s_1, \dots, \sigma^{v-1}(\alpha)s_{v-1}).$$

Algoritmo

// Paso 3: resolver el sistema de los síndromes, obteniendo las magnitudes de error

18 Encontrar (x_1, \dots, x_v) tal que $(x_1, \dots, x_v)(\Sigma^{v-1})^T = (\alpha s_0, \sigma(\alpha)s_1, \dots, \sigma^{v-1}(\alpha)s_{v-1})$

// Paso 4: construir el error y devolverlo

19 devolver (e_0, \dots, e_{n-1}) con $e_i = x_i$ para $i \in \{k_1, \dots, k_v\}$, cero en otro caso

Algoritmo

// Paso 3: resolver el sistema de los síndromes, obteniendo las magnitudes de error

18 Encontrar (x_1, \dots, x_v) tal que $(x_1, \dots, x_v)(\Sigma^{v-1})^T = (\alpha s_0, \sigma(\alpha)s_1, \dots, \sigma^{v-1}(\alpha)s_{v-1})$

// Paso 4: construir el error y devolverlo

19 **devolver** (e_0, \dots, e_{n-1}) con $e_i = x_i$ para $i \in \{k_1, \dots, k_v\}$, cero en otro caso

Obtención del mensaje original

Obtenido el error podemos restárselo al mensaje recibido, de forma que

$$c(x) = y(x) - e(x) \in \mathcal{C}$$

es el mensaje decodificado.

Implementación en SageMath

Clases desarrolladas

Se han implementado en SageMath:

- Un decodificador para códigos BCH usando el algoritmo PGZ.
- Los códigos cíclicos sesgados y los códigos RS sesgados.
- Un decodificador para códigos RS sesgados usando el algoritmo PGZ.

Aprovechan la estructura de códigos de Sage y su uso es similar a las incluidas.

Ejemplo

Implementación en SageMath

Conclusiones

Conclusiones

- Objetivos:
 - ✓ Estudio de polinomios de Ore, códigos cíclicos sesgados y del algoritmo PGZ.
 - ✓ Implementación en SageMath.
- Posible trabajo futuro: completar implementación códigos cíclicos sesgados y contruibuir lo desarrollado al proyecto SageMath.

Gracias por su atención