Lección 12: Codificación de Canal. Parte II

Gianluca Cornetta, Ph.D.
Dep. de Ingeniería de Sistemas de Información y Telecomunicación
Universidad San Pablo-CEU

Contenido

- Códigos de Bloque Lineales
- Detección y Corrección de Error

Códigos de Bloque Lineales

- Un <u>código de bloque lineal</u> es una clase de códigos con control de paridad de tipo (n, k):
  - La operación de codificación consiste en transformar un bloque de k dígitos de un alfabeto dado en un bloque codificado de n dígitos
  - Cuando el alfabeto consiste de sólo dos elementos $\{0, 1\}$, <u>el código se denomina binario</u>
- Los mensajes de k bits (dentro de un conjunto de 2^k secuencias) se denominan k-*tuplas*
- Los bloques de n bits forman 2^n posibles secuencias denominadas n-*tuplas*
- <u>La operación de codificación asigna de forma unívoca a cada una de las 2^k k-*tuplas* una de las posibles 2^n n-*tuplas*</u>

Códigos de Bloque Lineales

- El conjunto V_n de las 2^n *n-tuplas* define un espacio vectorial sobre el campo binario de dos elementos $\{0, 1\}$
- El campo binario tiene asociadas dos operaciones: suma módulo 2 y multiplicación

Addition	*Multiplication*
$0 \oplus 0 = 0$	$0 \cdot 0 = 0$
$0 \oplus 1 = 1$	$0 \cdot 1 = 0$
$1 \oplus 0 = 1$	$1 \cdot 0 = 0$
$1 \oplus 1 = 0$	$1 \cdot 1 = 1$

- La suma módulo 2 se indica indistintamente con el símbolo $\oplus$ o con $+$
- Un subconjunto S de V_n es un subespacio si cumple con las siguientes condiciones:
  - El vector formado por todos ceros pertenece a S
  - La suma de dos vectores de S es también un elemento de S(es decir si $V_i, V_j \in S$ $V_k = V_i \oplus V_j \Rightarrow \in S$ –<u>*propiedad de clausura*</u>)
- Un conjunto V_n de 2^k *n-tuplas* es un código de bloque lineal si y solo si es un subespacio del espacio vectorial de todas las *n-tuplas*
- El objetivo de diseño de un código es:
  - <u>Minimizar el número de bits de redundancia</u> para reducir el exceso de banda
  - <u>Aumentar la distancia entre los elementos del conjunto</u> (es decir, el número de 0 y 1 que difieren) para aumentar la tolerancia a errores de transmisión

Códigos de Bloque Lineales

- En general, un código (n, k) se puede generar fácilmente de forma tabular (*lookup table*) con una ROM en las que cada entrada se forma añadiendo a los k bits de mensaje, $(n\text{-}k)$ bits de redundancia procurando cumplir con las propiedades del vector nulo y de clausura
- Cuando k es elevado este enfoque no es práctico y es preferible generar los códigos en vez de almacenarlos en una memoria
- Un conjunto $\{\mathbf{U}\}$ de 2^k palabras de código es un subespacio vectorial k-dimensional del espacio vectorial binario n-dimensional por lo que es siempre posible encontrar una base (es decir, un conjunto n-*tuplas* de linealmente independientes) de k n-*tuplas* $\mathbf{V}_1, \mathbf{V}_2, \ldots, \mathbf{V}_k$ que puede generar todos los elementos de $\{\mathbf{U}\}$, es decir:

$$\boxed{\mathbf{U} = m_1\mathbf{V}_1 + m_2\mathbf{V}_2 + \ldots + m_k\mathbf{V}_k}$$

- Donde los $m_i = \{0,1\}$ (con $i = 1, \ldots, k$) son los dígitos del mensaje

Códigos de Bloque Lineales

- Es posible definir una matriz $k \times n$ $\mathbf{G}$ *generadora del código*:

$$\mathbf{G} = \begin{bmatrix} \mathbf{V}_1 \\ \mathbf{V}_2 \\ \vdots \\ \mathbf{V}_k \end{bmatrix} = \begin{bmatrix} v_{11} & v_{12} & \cdots & v_{1n} \\ v_{21} & v_{22} & \cdots & v_{2n} \\ \vdots & & & \\ v_{k1} & v_{k2} & \cdots & v_{kn} \end{bmatrix}$$

- Por otro lado, el mensaje $\mathbf{m}$ es una matriz $1 \times k$, es decir, $\mathbf{m}=m_1,m_2,\ldots, m_k$
- Codificar un mensaje $\mathbf{m}$ generando un código $\mathbf{U}$ equivale a realizar la siguiente operación: $\mathbf{U}=\mathbf{mG}$
- Por ejemplo, dada la siguiente matriz generadora:

$$\mathbf{G} = \begin{bmatrix} \mathbf{V}_1 \\ \mathbf{V}_2 \\ \mathbf{V}_3 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$

- Generar un código $(6, 3)$ para el mensaje $\mathbf{m}=[1\ 1\ 0]$ equivale a:

$$\mathbf{G} = \begin{bmatrix} 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{V}_1 \\ \mathbf{V}_2 \\ \mathbf{V}_3 \end{bmatrix} = 1 \cdot \mathbf{V}_1 + 1 \cdot \mathbf{V}_2 + 0 \cdot \mathbf{V}_3 = 110100 + 011010 + 000000 = 101110$$

Códigos de Bloque Lineales

- Un *código de bloque lineal sistemático* (n, k) es una correspondencia de un espacio k-dimensional a uno n-dimensional tal que parte de la secuencia generada coincide con los k bits del mensaje. Los $(n\text{-}k)$ bits restantes son los bits de paridad
- La matriz generadora de un código sistemático es:

$$\mathbf{G} = [\mathbf{P} \quad | \quad \mathbf{I}_k] = \begin{bmatrix} p_{11} & p_{12} & \cdots & p_{1,(n-k)} & 1 & 0 & \cdots & 0 \\ p_{21} & p_{22} & \cdots & p_{2,(n-k)} & 0 & 1 & \cdots & 0 \\ \vdots & & & & & & \vdots & \\ p_{k1} & p_{k2} & \cdots & p_{k,(n-k)} & 0 & 0 & \cdots & 1 \end{bmatrix}$$

- $\mathbf{P}$ (de dimensiones $k\times(n\text{-}k)$) es la matriz de paridad, $\mathbf{I}_k$ es la matriz identidad $k\times k$ y $p_{ij}=\{0,1\}$
- Un código formado por la *n-tupla* $\mathbf{U}=u_1,u_2,\ldots,u_n$ que codifica un mensaje formado por la *k-tupla* $\mathbf{m}=m_1,m_2,\ldots,m_k$ se obtiene de la siguiente manera:

$$u_1,u_2,\ldots u_n = [m_1 \quad m_2 \quad \cdots \quad m_k] \times \begin{bmatrix} p_{11} & p_{12} & \cdots & p_{1,(n-k)} & 1 & 0 & \cdots & 0 \\ p_{21} & p_{22} & \cdots & p_{2,(n-k)} & 0 & 1 & \cdots & 0 \\ \vdots & & & & & & \vdots & \\ p_{k1} & p_{k2} & \cdots & p_{k,(n-k)} & 0 & 0 & \cdots & 1 \end{bmatrix}$$

- Donde:

$$u_i = \begin{cases} m_1 p_{1i} + m_2 p_{1i} + \cdots + m_k p_{ki} & \text{for } i = 1,\ldots,(n-k) \\ m_{i-n+k} & \text{for } i = (n-k+1),\ldots,n \end{cases}$$

Códigos de Bloque Lineales

- Un vector de código sistemático puede expresarse como:

$$\mathbf{U} = \underbrace{p_1 \quad p_2 \quad \cdots \quad p_{n-k}}_{\text{parity bits}} \underbrace{m_1 \quad m_2 \quad \cdots \quad m_k}_{\text{message bits}}$$

- Donde:

$$\begin{aligned}
p_1 &= m_1 p_{11} + m_2 p_{21} + \cdots + m_k p_{k1} \\
p_2 &= m_1 p_{12} + m_2 p_{22} + \cdots + m_k p_{k2} \\
&\vdots \\
p_{n-k} &= m_1 p_{1,(n-k)} + m_2 p_{2,(n-k)} + \cdots + m_k p_{k,(n-k)}
\end{aligned}$$

- Por ejemplo, para el código un código (6, 3) del ejemplo anterior se obtiene:

$$\mathbf{U} = \begin{bmatrix} m_1 & m_2 & m_3 \end{bmatrix} \left[\underbrace{\begin{matrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{matrix}}_{\mathbf{P}} \;\; \underbrace{\begin{matrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{matrix}}_{\mathbf{I}_3} \right] = \overbrace{\underbrace{m_1 + m_3}_{u_1} \quad \underbrace{m_1 + m_2}_{u_2} \quad \underbrace{m_2 + m_3}_{u_3}}^{\text{parity bits}} \quad \overbrace{\underbrace{m_1}_{u_4} \quad \underbrace{m_2}_{u_5} \quad \underbrace{m_3}_{u_6}}^{\text{message bits}}$$

Códigos de Bloque Lineales

- En el proceso de decodificación se utiliza una *matriz de control de paridad* $\mathbf{H}$
- Para cada matriz $\mathbf{G}$ $k\times n$ existe una matriz $\mathbf{H}$ $(n\text{-}k)\times n$ tal que las filas de $\mathbf{G}$ son ortogonales a las filas de $\mathbf{H}$, es decir $\mathbf{GH}^T=\mathbf{0}$
- Donde:
  - $\mathbf{0}$ es la matriz nula $k\times(n\text{-}k)$
  - $\mathbf{H}^T$ es la matriz transpuesta de $\mathbf{H}$ con dimensiones $n\times(n\text{-}k)$

$$\mathbf{H}=\left[\mathbf{I}_{n-k} \quad | \quad \mathbf{P}^T\right] \Rightarrow \begin{bmatrix} \mathbf{I}_{n-k} \\ --- \\ \mathbf{P}^T \end{bmatrix} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & & \vdots & \\ 0 & 0 & \cdots & 1 \\ p_{11} & p_{12} & \cdots & p_{1,(n-k)} \\ p_{21} & p_{22} & \cdots & p_{2,(n-k)} \\ \vdots & & & \\ p_{k1} & p_{k2} & \cdots & p_{k,(n-k)} \end{bmatrix}$$

- $\mathbf{U}$ es un código generado por $\mathbf{G}$ si y sólo si $\mathbf{UH}^T=\mathbf{0}$, de hecho:

$$\mathbf{UH}^T = p_1+p_1 \quad p_2+p_2 \quad \cdots \quad p_{n-k}+p_{n-k} = \mathbf{0}$$

- Donde:

$$\begin{aligned} p_1 &= m_1 p_{11} + m_2 p_{21} + \cdots + m_k p_{k1} \\ p_2 &= m_1 p_{12} + m_2 p_{22} + \cdots + m_k p_{k2} \\ &\vdots \\ p_{n-k} &= m_1 p_{1,(n-k)} + m_2 p_{2,(n-k)} + \cdots + m_k p_{k,(n-k)} \end{aligned}$$

Códigos de Bloque Lineales

- Sea $\mathbf{r}=r_1,r_2,\ldots,r_n$ el vector recibido (es decir, una de las posibles 2^n n-*tuplas*) resultante de la transmisión de $\mathbf{U}=u_1,u_2,\ldots,u_n$ (es decir, una de las posibles 2^k n-*tuplas*), es decir $\mathbf{r}=\mathbf{U}+\mathbf{e}$
- $\mathbf{e}=e_1,e_2,\ldots,e_n$ es el patrón de error introducido por el canal, es decir, $e_i=1$ si y sólo si $r_i \neq u_i$ (lo que se recibe es distinto de lo que se ha transmitido)
  - Existen pues 2^n-1 patrones de error potenciales en el espacio de 2^n n-*tuplas*
- El *síndrome* $\mathbf{S}$ de $\mathbf{r}$ es definido como $\mathbf{S}=\mathbf{rH}^T$
- Por consiguiente el síndrome es el resultado de un control de paridad realizado sobre $\mathbf{r}$ para determinar si el vector recibido pertenece al conjunto de códigos utilizados
  - Si el control tiene éxito positivo $\mathbf{S}=\mathbf{0}$
  - Si $\mathbf{r}$ contiene errores detectables $\mathbf{S}$ tiene algunos valores no nulos
  - Si $\mathbf{r}$ contiene errores corregibles $\mathbf{S}$ tiene algunos valores no nulos que permiten identificar el patrón de error

Códigos de Bloque Lineales

- Sabiendo que $\mathbf{r}=\mathbf{U}+\mathbf{e}$ se obtiene $\mathbf{S}=(\mathbf{U}+\mathbf{e})\mathbf{H}^T=\mathbf{U}\mathbf{H}^T+\mathbf{e}\mathbf{H}^T=\mathbf{0}+\mathbf{e}\mathbf{H}^T=\mathbf{e}\mathbf{H}^T$
  - El test del síndrome da el mismo resultado tanto si se realiza sobre el vector corrupto recibido $\mathbf{r}$ como si se realiza sobre el patrón de error $\mathbf{e}$ que lo causado
- Es interesante evidenciar dos importantes propiedades de la matriz $\mathbf{H}$:
  - Ninguna columna de $\mathbf{H}$ puede ser nula, <u>en caso contrario un error en la posición correspondiente a la columna nula no afecta $\mathbf{S}$ y por tanto es indetectable</u>
  - Todas las columnas de $\mathbf{H}$ deben ser únicas, en caso contrario los error en las posiciones correspondientes a las columnas idénticas serían indetectables

Códigos de Bloque Lineales

- Existe una correspondencia unívoca entre código corrupto y patrón de error, por tanto, <u>el test del síndrome no sólo permite detectar errores sino también corregirlos</u>
- Es posible organizar las 2^n *n-tuplas* que representan los vectores recibidos en forma matricial
- Esta matriz se denomina <u>*matriz estándar*</u> y tiene las características siguientes:
  - La primera fila contiene todos los códigos empezando por el código nulo $\mathbf{U}_1$
  - La primera columna contiene todos los posibles patrones de error ($\mathbf{U}_1$ es el código nulo pero puede considerarse también como el patrón de error $\mathbf{e}_1$, el patrón que representa la recepción sin errores, es decir $\mathbf{r}=\mathbf{U}$)
  - La matriz contiene todas las 2^n *n-tuplas* del espacio V_n, cada *n-tupla* aparece sólo en una posición (ninguna falta y ninguna es replicada)

Coset leaders

$$\overbrace{\begin{bmatrix} \mathbf{U}_1 & \mathbf{U}_2 & \cdots & \mathbf{U}_j & \cdots & \mathbf{U}_{2^k} \\ \mathbf{e}_2 & \mathbf{U}_2+\mathbf{e}_2 & \cdots & \mathbf{U}_j+\mathbf{e}_2 & \cdots & \mathbf{U}_{2^k}+\mathbf{e}_2 \\ \mathbf{e}_3 & \mathbf{U}_2+\mathbf{e}_3 & \cdots & \mathbf{U}_j+\mathbf{e}_3 & \cdots & \mathbf{U}_{2^k}+\mathbf{e}_3 \\ \vdots & \vdots & & & & \vdots \\ \mathbf{e}_j & \mathbf{U}_2+\mathbf{e}_j & \cdots & \mathbf{U}_j+\mathbf{e}_j & \cdots & \mathbf{U}_{2^k}+\mathbf{e}_j \\ \vdots & \vdots & & & & \vdots \\ \mathbf{e}_{2^{n-k}} & \mathbf{U}_2+\mathbf{e}_{2^{n-k}} & \cdots & \mathbf{U}_j+\mathbf{e}_{2^{n-k}} & \cdots & \mathbf{U}_{2^k}+\mathbf{e}_{2^{n-k}} \end{bmatrix}}^{2^k\ n\text{-tuplas}} \; \Bigg\} \; 2^n/2^k=2^{n-k}\ \textit{cosets}$$

Códigos de Bloque Lineales

- El algoritmo de decodificación prueba a sustituir un código corrupto recibido (una *n-tupla* cualquiera excepto las de la primera fila) con un código valido que se encuentra en la primera posición de la columna correspondiente
  - Por ejemplo si se recibe $\mathbf{U}_i + \mathbf{e}_j$ y el error $\mathbf{e}_j$ causado por el canal es líder del *coset*, el vector recibido será decodificado correctamente como $\mathbf{U}_i$
  - <u>Un *coset* es un conjunto de números que tienen una característica en común, *el mismo síndrome*</u>
    - Cada *coset* es identificado unívocamente por su síndrome $\mathbf{S}$, por lo que <u>$\mathbf{S}$ puede utilizarse para estimar el patrón de error</u>
- Si es el líder del *coset* (es decir, el patrón de error de los elementos del *j*-ésimo *coset*), entonces $\mathbf{U}_i + \mathbf{e}_j$ es una *n-tupla* de este *coset*
- El síndrome de esta *n-tupla* puede escribirse como:

$$\boxed{\mathbf{S} = \left(\mathbf{U}_i + \mathbf{e}_j\right)\mathbf{H}^T = \mathbf{U}_i\mathbf{H}^T + \mathbf{e}_j\mathbf{H}^T = \mathbf{0} + \mathbf{e}_j\mathbf{H}^T = \mathbf{e}_j\mathbf{H}^T}$$

Códigos de Bloque Lineales

- Para corregir un error se sigue el procedimiento siguiente:
  - Calcular el síndrome de $\mathbf{r}$ utilizando $\mathbf{S}=\mathbf{r}\mathbf{H}^T$
  - Encontrar el líder del *coset* $\mathbf{e}_j$ cuyo síndrome es igual a $\mathbf{r}\mathbf{H}^T$
    - Se asume que $\mathbf{e}_j$ es el patrón de error generado por el canal
  - Obtener el código esperado realizando la operación $\mathbf{U} = \mathbf{r} + \mathbf{e}_j$
    - En aritmética módulo dos suma y resta son equivalentes por lo que esta operación equivale a restar el error $\mathbf{e}_j$ al mensaje $\mathbf{r}$ recibido
- En el caso de un código $(6, 3)$ el conjunto V_n tiene 64 *n-tuplas*,por lo que, una vez acotados los 6 posibles patrones de error sencillo, todavía queda la posibilidad de poder corregir un patrón adicional

000000	110100	011010	101110	101001	011101	110011	000111
000001	110101	011011	101111	101000	011100	110010	000110
000010	110110	011000	101100	101011	011111	110001	000101
000100	110000	011110	101010	101101	011001	110111	000011
001000	111100	010010	100110	100001	010101	111011	001111
010000	100100	001010	111110	111001	001101	100011	010111
100000	010100	111010	001110	001001	111101	010011	100111
010001	100101	001011	111111	111000	001100	100010	010110

Códigos de Bloque Lineales

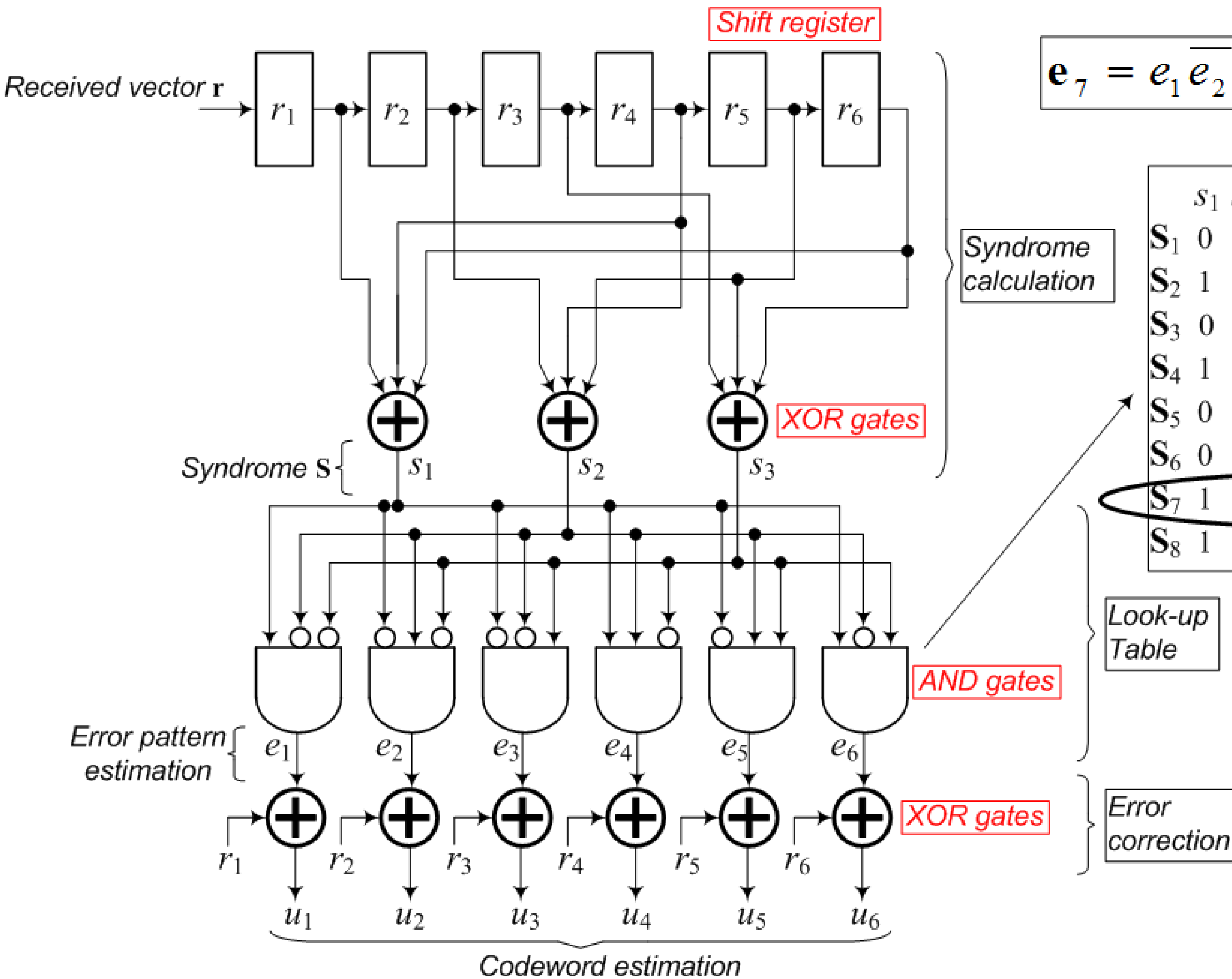
- Dada una matriz generadora $\mathbf{G}$ de un código (n, k), la tabla que permite estimar el patrón de error a partir del síndrome se construye a partir de la relación $\mathbf{S}_j=\mathbf{e}_j\mathbf{H}^T$ con $j=1,\ldots,2^{n\text{-}k}$
- Cada fila de la tabla contiene el síndrome $\mathbf{S}_j$ y su respectivo patrón de error $\mathbf{e}_j$
  - El decodificador calcula el síndrome $\mathbf{S}=\mathbf{r}\mathbf{H}^T$ relativo al vector recibido $\mathbf{r}$
  - El decodificador busca en la tabla un $\mathbf{S}_j$ tal que $\mathbf{S}_j=\mathbf{S}$ para determinar el patrón de error $\mathbf{e}_j$ que lo ha generado
  - $\mathbf{e}_j=\hat{\mathbf{e}}$ es una estimación del error
  - El decodificador suma $\hat{\mathbf{e}}$ a $\mathbf{r}$ (una suma módulo 2 equivale a una resta) para obtener una estimación $\hat{\mathbf{U}}$ del código transmitido:

$$\boxed{\hat{\mathbf{U}}=\mathbf{r}+\hat{\mathbf{e}}=(\mathbf{U}+\mathbf{e})+\hat{\mathbf{e}}=\mathbf{U}+(\mathbf{e}+\hat{\mathbf{e}})}$$

  - Si el error estimado es igual al error real (es decir, $\mathbf{e}=\hat{\mathbf{e}}$) entonces $\hat{\mathbf{U}}=\mathbf{U}$; en caso contrario la estimación es equivocada y el error indetectable
  - En el caso de códigos pequeños como el $(6, 3)$ la implementación del decodificador es muy sencilla. Por ejemplo, el código considerado en los ejemplos anteriores tiene un síndrome $\mathbf{S}=\mathbf{r}\mathbf{H}^T$ igual a:

$$\boxed{\mathbf{S}=\begin{bmatrix} r_1 & r_2 & r_3 & r_4 & r_5 & r_6 \end{bmatrix}\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix} \Rightarrow \begin{cases} s_1 = r_1 + r_4 + r_6 \\ s_2 = r_2 + r_4 + r_5 \\ s_3 = r_3 + r_5 + r_6 \end{cases}}$$

Códigos de Bloque Lineales



$$\mathbf{e}_7 = e_1\bar{e}_2\bar{e}_3\bar{e}_4\bar{e}_5\bar{e}_6 \Rightarrow e_1 = s_1\bar{s}_2\bar{s}_3$$

	s_1	s_2	s_3	e_1	e_2	e_3	e_4	e_5	e_6	
$\mathbf{S}_1$	0	0	0	0	0	0	0	0	0	$\mathbf{e}_1$
$\mathbf{S}_2$	1	0	1	0	0	0	0	0	1	$\mathbf{e}_2$
$\mathbf{S}_3$	0	1	1	0	0	0	0	1	0	$\mathbf{e}_3$
$\mathbf{S}_4$	1	1	0	0	0	0	1	0	0	$\mathbf{e}_4$
$\mathbf{S}_5$	0	0	1	0	0	1	0	0	0	$\mathbf{e}_5$
$\mathbf{S}_6$	0	1	0	0	1	0	0	0	0	$\mathbf{e}_6$
$\mathbf{S}_7$	1	0	0	1	0	0	0	0	0	$\mathbf{e}_7$
$\mathbf{S}_8$	1	1	1	0	1	0	0	0	1	$\mathbf{e}_8$

Detección y Corrección de Error

- No todos los códigos recibidos pueden ser detectados correctamente
- Se define *peso de Hamming* $w(\mathbf{U})$ del código $\mathbf{U}$ el número de elementos de $\mathbf{U}$ no nulos
- Se define *distancia de Hamming* $d(\mathbf{U},\mathbf{V})$ entre dos vectores $\mathbf{U}$ y $\mathbf{V}$ el número de elementos por los que difieren, es decir:

$$\boxed{\begin{matrix}\mathbf{U} = 100101101 \\ \mathbf{V} = 011110100\end{matrix} \Rightarrow d(\mathbf{U},\mathbf{V}) = 6}$$

- Observe que $\mathbf{U}+\mathbf{V}=111011001$ es un vector tal que los elementos en las posiciones en las que $\mathbf{U}$ y $\mathbf{V}$ difieren valen 1
  - Por construcción resulta que $d(\mathbf{U},\mathbf{V})= w(\mathbf{U}+\mathbf{V})$
- Asimismo resulta que $w(\mathbf{U}) =d(\mathbf{U},\mathbf{0})$
- La mínima distancia de Hamming de un conjunto de códigos caracteriza la robustez del mismo (es decir su capacidad de detectar y corregir errores)
- Si $\mathbf{U}$ y $\mathbf{V}$ son palabras de código de un espacio $V_n \Rightarrow$ también está $\mathbf{W}=\mathbf{U}+\mathbf{V}$ en V_n, por tanto: $d(\mathbf{U},\mathbf{V})= w(\mathbf{U}+\mathbf{V}) = w(\mathbf{W})$
  - La distancia mínima d_{min} puede calcularse simplemente observando el peso de cada elemento del conjunto (excepto el vector nulo)

Detección y Corrección de Error

- Para detectar cuál es el código recibido, el detector aplica el algoritmo de máxima verosimilitud (*maximum likelihood*) sobre el vector recibido $\mathbf{r}$:

$$\boxed{P(\mathbf{r} \mid \mathbf{U}_i) = \max_{\forall \mathbf{U}_j} P(\mathbf{r} \mid \mathbf{U}_j)}$$

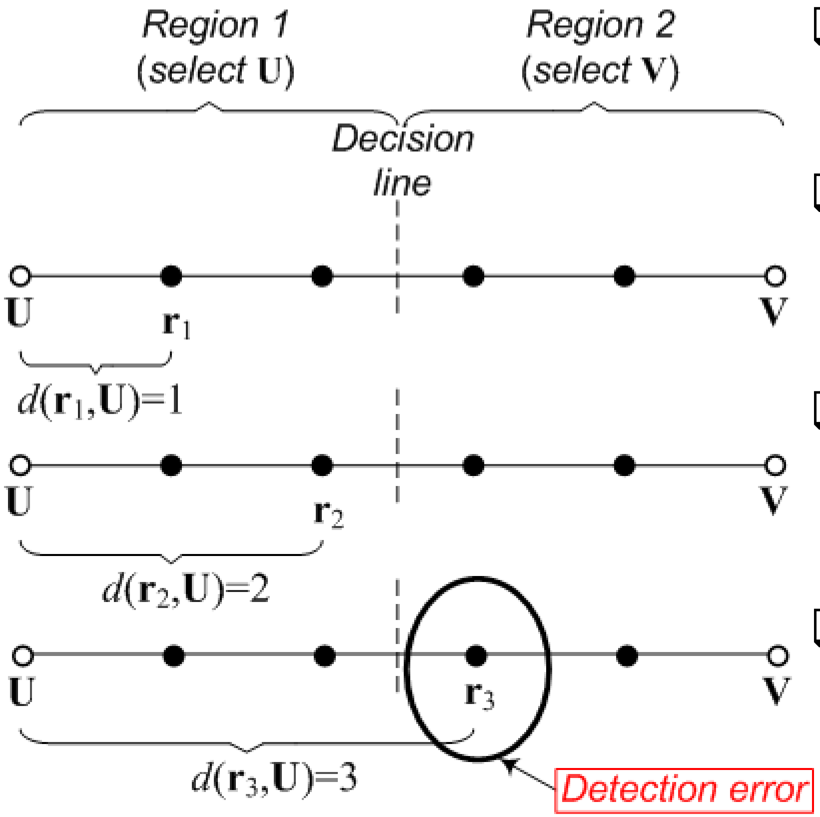
- Para un canal binario simétrico (*Binary Symmetric Channel* –BSC) la verosimilitud $P(\mathbf{r}|\mathbf{U}_i)$ es inversamente proporcional a la distancia de Hamming entre $\mathbf{r}$ y $\mathbf{U}_i$
- El decodificador escoge $\mathbf{U}_i$ si y sólo si:

$$\boxed{d(\mathbf{r}, \mathbf{U}_i) = \min_{\forall \mathbf{U}_j} d(\mathbf{r}, \mathbf{U}_j)}$$

  - El decodificador determina la distancia de Hamming entre $\mathbf{r}$ y todos los posibles códigos $\mathbf{U}_j$ y selecciona $\mathbf{U}_i$ tal que:

$$\boxed{d(\mathbf{r}, \mathbf{U}_i) \le d(\mathbf{r}, \mathbf{U}_j) \quad \text{for } i, j = 1, \ldots, M \quad \text{and} \quad i \ne j}$$

Detección y Corrección de Error



- Transmito $\mathbf{U}$ y recibo $\mathbf{r}_1$. El vector recibido cae en la región de decisión de $\mathbf{U}$ por lo que el decodificador detecta este código
- Si $\mathbf{r}_1$ es el resultado de un error sencillo la estimación realizada es correcta, pero si $\mathbf{r}_1$ es el resultado de un error cuádruple la estimación es incorrecta
- *Un error quíntuple hace que el símbolo estimado es* $\mathbf{V}$ (es decir, un símbolo correcto) *por lo que el error es indetectable*
- En este ejemplo la distancia entre los dos códigos válidos ($\mathbf{U}$ y $\mathbf{V}$) es d_{min}=5, y es posible detectar y corregir hasta errores dobles

Detección y Corrección de Error

- La *máxima capacidad de corrección de error t* es definida como:

$$\boxed{t = \left\lfloor \frac{d_{\min} - 1}{2} \right\rfloor}$$

- Un código de bloque lineal (n, k) con capacidad de corregir errores de clase t puede corregir también un único error de clase $t+1$
- <u>Un código de bloque lineal que puede corregir sólo hasta errores de clase *t* se denomina *código perfecto*</u>
- La probabilidad P_M de error de mensaje en un canal binario simétrico con probabilidad de transición (es decir, probabilidad de error de bit p) es:

$$\boxed{P_M \le \sum_{j=t+1}^{n} \binom{n}{j} p^j (1-p)^{n-j}}$$

- Se trata de un valor limitado superiormente que <u>se vuelve igual al límite superior en el caso de decodificadores que sólo detectan hasta errores de clase *t*</u> (*bounded distance decoders*)
- La probabilidad de error de bit P_B es:

$$\boxed{P_B \approx \frac{1}{n} \sum_{j=t+1}^{n} j \binom{n}{j} p^j (1-p)^{n-j}}$$

Detección y Corrección de Error

- Un código de bloque con distancia mínima de Hamming d_{min} garantiza la capacidad de detectar hasta errores de clase e:

$$\boxed{e = d_{\min} - 1}$$

- En un espacio de 2^n *n-tuplas* existen 2^n-1 posibles patrones de error
- Si el vector recibido es igual a uno de los posibles códigos válidos el error es indetectable, por tanto existen 2^k-1 posibles patrones de errores indetectables
- El número de errores detectables es $(2^n\text{-}1)$ - $(2^k\text{-}1) = 2^n\text{-}2^k$
- Se define con A_j el número de palabras de código de peso j de un código de bloque lineal (n, k)
- $A_0, A_1, \ldots, A_n$ representan las distribuciones de pesos del código
- Si el código se utiliza para detectar errores en canales binarios simétricos, la probabilidad de error indetectable es:

$$\boxed{P_{nd} = \sum_{j=1}^{n} A_j p^j (1-p)^{n-j}}$$

- Si la distancia mínima del código es d_{min}, los valores $A_1, \ldots, A_{d_{min}\text{-}1}$ son nulos

Detección y Corrección de Error

- Es posible buscar una solución de compromiso entre el máximo t de un código y su capacidad de corregir errores
- Un código de bloque lineal con distancia mínima de Hamming d_{min} puede corregir simultáneamente α errores y detectar simultáneamente β errores ($\beta \geq \alpha$) si:

$$\boxed{d_{\min} \geq \alpha + \beta + 1}$$

- Cuando ocurre un número de errores menor o igual a t, el código puede detectarlos y corregirlos
- Cuando ocurre un número de errores superior a t e inferior a $e + 1$, el código puede detectarlos pero sólo puede corregir un subconjunto de ellos
- Por ejemplo, si $d_{min} = 7 \Rightarrow \alpha + \beta = 6$, por tanto existen las siguientes posibilidades:

Detect (α)	Correct (β)
3	3
4	2
5	1
6	0

Detección y Corrección de Error

- Un detector puede ser diseñado con la capacidad de cancelar la recepción de un símbolo en el caso de ambigüedad o de malfuncionamiento transitorio del canal de comunicación
- Estos tipos de canal se caracterizan por un alfabeto de entrada de Q símbolos y por uno de salida de $Q + 1$ símbolos
  - El símbolo adicional es el <u>indicador de cancelación</u> (*erasure flag*)
- Cuando un detector realiza un error de símbolo, necesita dos parámetros para corregirlo: <u>la posición del error y el valor correcto del símbolo (en el caso de símbolos binarios sólo la posición)</u>
- Si el detector declara un símbolo como cancelado, aunque el valor correcto del símbolo no se conoce a priori, la posición del símbolo cancelado es sabida
- La decodificación de códigos cancelados puede ser más sencilla de la corrección de errores
- Asumiendo que sólo haya ρ símbolos cancelados, un código con distancia mínima de Hamming d_{min} puede corregir los símbolos cancelados si :

$$\boxed{d_{\min} \geq \rho + 1 \Rightarrow \rho \leq d_{\min} - 1}$$

  - Esta condición es mucho más ventajosa de la de corrección de error: $t \leq (d_{min} - 1)/2$
- Asumiendo que haya γ símbolos cancelados y α símbolos erróneos, un código con distancia mínima de Hamming d_{min} puede corregir todos los símbolos cancelados y erróneos si :

$$\boxed{d_{\min} \geq 2\alpha + \gamma + 1}$$

- El algoritmo de corrección es el siguiente:
  - Se sustituyen las posiciones γ canceladas con ceros y se decodifica el código
  - Se sustituyen las posiciones γ canceladas con unos y se decodifica el código
  - De los dos valores decodificados se elige el que lleva a un mínimo número de correcciones en las posiciones restantes