CODIFICACIÓN DE LA INFORMACIÓN (I)

(02A)

CODIFICACIÓN EN CANALES SIN RUIDO

Ir a NIBBEL AUTOMATION

Fin del artículo

Venancio Guntiñas Rodríguez
vguntinas2@gmail.com

Revisado: 23-02-2019

Índice

- Concepto de Código
- Prefijo de una palabra-código
- Códigos instantáneos
- Formas de codificar una fuente
- Códigos de longitud costante
- Teorema fundamental de los códigos de longitud constante
- Códigos de longitud variable
- Código compacto
- Primer teorema de Shannon
- Métodos para obtener códigos compactos
- Método de Huffman
- Método de Shanno-Fano
- Notas sobre los códigos obtenidos por Huffman o Shannon-Fano
- Códigos binarios con longitud constante de los números decimales
- Códigos ponderados
- Código BCD
- Códigos no ponderados
- Código de exceso tres
- Códigos autocomplementarios
- Código de Aiken
- Códigos continuos
- Código de Gray
- Código progresivo de Johnson

CONCEPTO DE CÓDIGO

Un código es una aplicación entre un conjunto de símbolos llamado **alfabeto fuente** y otro conjunto de símbolos llamado **alfabeto código**, de tal manera que, a cada símbolo del alfabeto fuente le corresponde una sucesión de símbolos diferente del alfabeto código.

Se llama **palabra código** a cada una de las sucesiones de símbolos del alfabeto código que corresponde a un símbolo fuente.

Ejemplo:

Sea A el alfabeto fuente: A ={a, b, c, d}
Sea B el alfabeto código: B = {0, 1}
Ejemplos de palabras código:
a => 010
b => 10
c => 101
d => 11

Se llama **longitud de una palabra código** al número de símbolos del alfabeto código que posee dicha palabra.

Los códigos cuyo alfabeto código consiste solamente en el 0 y el 1 se llaman **códigos binarios**.

Para que un **código** sea **útil** ha de cumplir las siguientes propiedades:

- A cada símbolo fuente le corresponderá una única palabra código.
- Dos símbolos fuente distintos han de tener palabras código distintas.
- El código ha de ser **unívocamente decodificable**, es decir, cuando se recibe una serie de palabras código sólo podrá decodificarse de una manera, es decir, sólo podrá obtenerse al decodificar la serie, una única sucesión de símbolos fuente.

Las dos primeras propiedades establecen que la **correspondencia** que define el código ha de ser **biyectiva**.

Ejemplo:
Sea el alfabeto fuente: A = {a_1, a_2, a_3, a_4}
Sea el alfabeto código: B = {0, 1}
Consideremos el código definido por la siguiente correspondencia:

Símbolos fuente	**Palabras código**
a_1	0
a_2	01
a_3	10
a_4	11

Si se recibe la sucesión de símbolos 010, el receptor la puede interpretar de dos maneras:

- Se han emitido los símbolos a_2a_1 (01 + 0)
- Se han emitido los símbolos a_1a_3 (0 + 10)

En consecuencia, el código no es unívocamente decodificable.

Al número de símbolos del alfabeto código se le llama **base del código**.
Al conjunto de todas las palabras código se le llama **libro código**.

PREFIJO DE UNA PALABRA CÓDIGO

Dada una palabra código se llama prefijo de esa palabra código, a toda sub-sucesión de dicha palabra código que comience por el primer símbolo de la izquierda.

EJEMPLO:

Sea la palabra código 101101
Sus prefijos serán: 1, 10, 101, 1011, 10110, 101101.

CÓDIGOS INSTANTÁNEOS

Un código unívocamente decodificable se dice que es instantáneo cuando no existe retardo en el proceso de decodificación, es decir, cuando se pueden decodificar las palabras de una secuencia sin conocer los símbolos que las siguen.

EJEMPLO

Símbolos fuente	Palabras código
a	00
b	01
c	10
d	11

Es un código instantáneo. Si por ejemplo se recibe la secuencia 00111001, se pueden ir decodificando los símbolos según van llegando pues, no existe ninguna ambigüedad. Al llegar el primer 0, no corresponde a ninguna palabra código, al llegar el siguiente 0, 00 es una palabra código y no es prefijo de ninguna otra palabra código, sólo puede corresponder al símbolo 'a' por lo que, se puede decodificar inmediatamente. Luego llega un 1 que no es una palabra código, luego llega otro 1, y 11 si es una palabra código y además no es prefijo de ninguna otra palabra código, por lo que sólo puede corresponder al símbolo 'd' y se puede decodificar inmediatamente. Se procede de modo análogo para el resto de la sucesión.

EJEMPLO

Símbolos fuente	Palabras código
a	0
b	10
c	110
d	1110

Es un código instantáneo.

EJEMPLO:

Símbolos fuente	Palabras código
a	0
b	01
c	011
d	0111

No es un código instantáneo. Pues, si por ejemplo llega la sucesión 01 puede corresponder al símbolo 'b' o ser los primeros dos dígitos de los símbolos 'c' o 'd'.

TEOREMA.- UN CÓDIGO ES INSTANTÁNEO SI Y SÓLO SI NINGUNA DE SUS PALABRAS CÓDIGO ES PREFIJO DE OTRA.
TEOREMA.- TODO CÓDIGO INSTANTÁNEO ES UNÍVOCAMENTE DECODIFICABLE. EL RECÍPROCO NO ES CIERTO, PUEDE HABER CÓDIGOS UNÍVOCAMENTE DECODIFICABLES QUE NO SEAN INSTANTÁNEOS.

FORMAS DE CODIFICAR UNA FUENTE

Para un alfabeto-fuente y un alfabeto-código dados, es posible elaborar más de un código instantáneo. Por este motivo, habrá que adoptar un criterio que permita elegir uno de entre ellos.
La codificación de una fuente de información puede hacerse siguiendo dos criterios distintos:

- **Códigos de longitud constante o códigos bloque**.- Todas las palabras código tienen la misma longitud.

- **Códigos de longitud variable**.- La longitud de las palabras código se elige en función de las probabilidades de emisión de los símbolos de la fuente, de tal manera que, cuanto mayor sea la probabilidad de emisión de un símbolo, menor longitud tendrá la palabra código correspondiente.

CÓDIGOS DE LONGITUD CONSTANTE

Veamos algunas de sus características:

INCONVENIENTES

En general, son más largos que los códigos de longitud variable, por lo que, su transmisión es menos eficiente, es decir, para transmitir una misma información requieren más tiempo y equipos más complejos.

VENTAJAS

La codificación puede hacerse sin necesidad de conocer la estadística de la fuente.

APLICACIONES

Se utilizan en aquellos casos en que la longitud de la palabra código ha de ser fija. Por ejemplo en telefonía automática.

TEOREMA FUNDAMENTAL DE LOS CÓDIGOS INSTANTÁNEOS DE LONGITUD CONSTANTE

SEA 'N' EL NÚMERO DE SÍMBOLOS FUENTE, SEA 's' EL NÚMERO DE SÍMBOLOS DEL ALFABETO CÓDIGO, SEA 'L' LA LONGITUD DE LAS PALABRAS CÓDIGO, SE CUMPLE QUE:

$$L \geq LOG_s N$$

ES DECIR, NO ES POSIBLE ENCONTRAR UN CÓDIGO INSTANTÁNEO DE LONGITUD CONSTANTE, EN EL CUAL, LA LONGITUD DE SUS PALABRAS CÓDIGO SEA MENOR QUE $LOG_s N$.

EJEMPLO:

¿Se puede construir un código binario instantáneo de longitud constante igual a 3, para una fuente de ocho símbolos?
En este caso tenemos: n=8, s=2, L=3, por lo tanto: $\log_s n = \log_2 8 = 3$.
Por lo tanto, la respuesta es 'si se puede construir' pues $3 \geq \log_s n = \log_2 8$.

CÓDIGOS DE LONGITUD VARIABLE

Al ser variable la longitud de cada palabra código, se define el concepto de **Longitud media del código,** de la siguiente manera:
Sea A = {a1, a2, ..., an} el alfabeto fuente.
Sea P(A) = {p(a1), p(a2), ..., p(an)} las probabilidades de emisión de cada símbolo fuente.
Si el código utilizado asigna a:

a_1	Una palabra de longitud L_1
a_2	Una palabra de longitud L_2
……	
a_n	Una palabra de longitud L_n

Se define la **longitud media del código** y se representa por L_m a la siguiente expresión:

Lm = Σ p(ai) . Li símbolos-código/palabra-código

Como cada palabra código corresponde a un símbolo fuente, la unidad puede tomarse como símbolos-código/símbolos-fuente (sc/sf).
Para que una codificación sea eficaz, la longitud media de las palabras código debe ser lo menor posible.

EJEMPLO:

Sea A = { a, b, c, d} el alfabeto fuente, P(A) = {1/2, 1/4, 1/8, 1/8} la probabilidad de emisión de los símbolos fuente y { 1, 4, 2, 4} las longitudes de las palabras código correspondientes. Calcular la longitud media del código:
L_m= ½ . 1 + ¼ . 4 + 1/8 . 2 + 1/8 . 4 = ½ + 1 + ¼ + ½ = 9/4 símbolos-código/símbolos-fuente.

El valor de Lm puede hacerse pequeño si se asignan las palabras código más largas a los signos menos probables. Este fue el procedimiento que se siguió para elaborar el **código Morse**, en el cual, se asignan los códigos más cortos a las letras del alfabeto más frecuentes en un texto escrito en inglés.
Si en el ejemplo anterior las longitudes fuesen (1, 2, 4, 4) =>
L_m = ½ . 1 + ¼ . 2 + 1/8 . 4 + 1/8 . 4 = ½ + ½ + ½ + ½ =2 sc/sf.

CÓDIGO COMPACTO

Un código compacto es un código instantáneo cuya longitud media es mínima.
El empleo de estos códigos conduce a una mayor eficacia en la transmisión de la información, en el sentido de ocupar el canal el menor tiempo posible para una cantidad de información dada. Además, si se utilizan durante el procesamiento de la información se consigue una simplificación en el equipo.

TEOREMA: LA LONGITUD MEDIA DE UN CÓDIGO INSTANTÁNEO ES MÍNIMA SI Y SÓLO SI SE ESCOGEN LAS LONGITUDES DE CADA PALABRA CÓDIGO IGUAL AL LOGARITMO EN BASE 'S' (SIENDO 'S' LA BASE DEL CÓDIGO) DE LA INVERSA DE LA PROBABILIDAD DE EMISIÓN DEL SÍMBOLO FUENTE CORRESPONDIENTE.
DE FORMA MATEMÁTICA: L_m **ES MÍNIMA** ⇔ $L_i = LOG_s\,(1/P(A_i))$

Ejemplo: Sea A= {a, b, c, d} el alfabeto fuente, sea P(A) = {1/2, ¼, 1/8, 1/8} las probabilidades de emisión de dichos símbolos. Quiere obtenerse un código binario que sea compacto.
Para que el código sea compacto ha de ser instantáneo y de longitud mínima.
La longitud de la palabra código que asignaremos a cada símbolo será la siguiente:

$L(a) = \log_2(1/(1/2)) = \log_2(2) = 1$ sc/sf
$L(b) = \log_2(1/(1/4)) = \log_2(4) = \log_2(2^2) = 2$ sc/sf
$L(c) = L(d) = \mathrm{Log}_2(1/(1/8)) = \log_2(8) = \log_2(2^3) = 3$ sc/sf

Para que el código sea instantáneo ninguna palabra código ha de ser prefijo de otra, en consecuencia, un posible código compacto es:

Símbolos fuente	Palabra código
a	0
b	10
c	110
d	111

PRIMER TEOREMA DE SHANNON

PARA CUALQUIER FUENTE SIEMPRE ES POSIBLE ENCONTRAR UN CÓDIGO COMPACTO.

En general, será necesario extender la fuente hasta un orden 'r' para obtener un código compacto.

MÉTODOS PARA OBTENER CÓDIGOS COMPACTOS

Dos métodos para obtener un código compacto de una fuente con probabilidades arbitrarias son: el método de Huffman y el método de Shanon-Fano.

MÉTODO DE HUFFMAN (CÓDIGOS DE HUFFMAN)

Veamos el **caso de los códigos binarios**:

Sea la fuente (con símbolos A{a1,………,an} y probabilidades P{ p(a1),………,p(an)})

Se realizarán los siguientes pasos:

Paso 1.- ordenamos los símbolos de la fuente por probabilidades decrecientes.

Paso 2.- Se reduce el número de símbolos de la fuente en uno, combinando los dos símbolos de probabilidades más pequeñas en un solo símbolo, su probabilidad será la suma de las probabilidades de los dos símbolos que sustituye. A la nueva fuente se le llama fuente reducida.

Paso 3.- Se vuelven a reordenar los símbolos de la fuente reducida según probabilidades decrecientes.

Paso 4.- Se reducen los dos símbolos de menor probabilidad a uno solo. Se obtiene así una nueva fuente reducida.

Paso 5.- Se continúa el proceso, obteniéndose una secuencia de fuentes, cada una con un símbolo menos que la anterior, hasta llegar a una fuente de sólo dos símbolos.

Paso 6.- El código compacto de la última fuente reducida será el formado por las palabras código 0 y 1.

Paso 7.- El código compacto de una de las fuentes reducidas de la secuencia, se establece a partir del de la fuente reducida inmediatamente siguiente, según la siguiente regla:

"Cada fuente reducida se distingue de la siguiente en un único símbolo. Los símbolos iguales tendrán la misma palabra-código y los dos símbolos s_{j0} y s_{j1} que dan origen a s_j, se les asignará una palabra-código que se obtiene añadiendo un 0 y un 1 respectivamente a la palara-código de s_j".

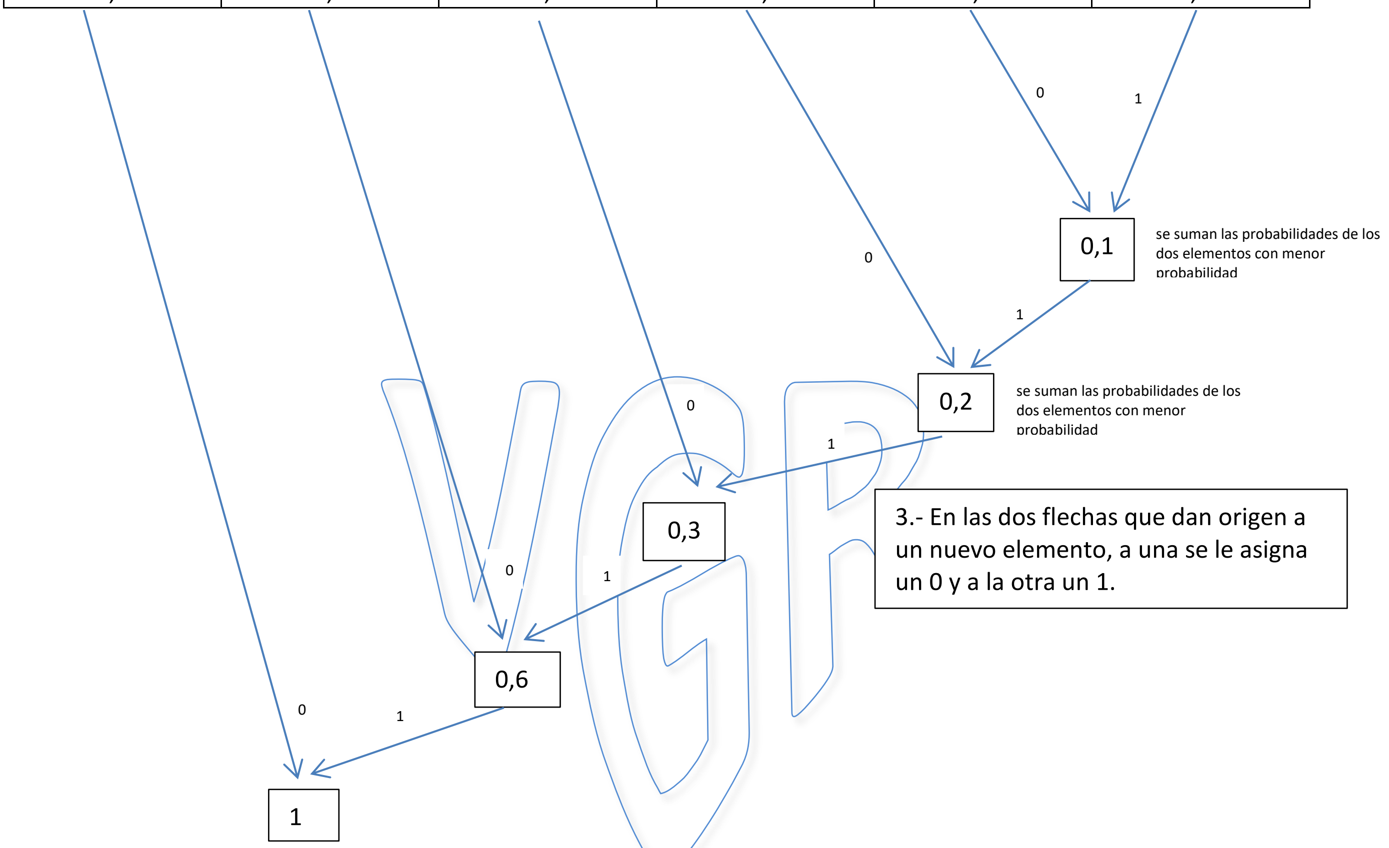
EJEMPLO

Sea la fuente A = {a1, a2, a3, a4, a5, a6} con probabilidades P={0,4/0,3/0,1/0,1/0,06/0,04}. Otener un código binario compacto.

1.- Se ordenan los símbolos según probabilidades decrecientes:

2.- Se van agrupando los dos de menores probabilidades en cada nivel.

a1	a2	a3	a4	a5	a6
0,4	0,3	0,1	0,1	0,06	0,04



3.- En las dos flechas que dan origen a un nuevo elemento, a una se le asigna un 0 y a la otra un 1.

4.- Empezando por el final, a cada símbolo se le asigna la sucesión de unos y ceros que se encuentran en el camino para llegar a él.

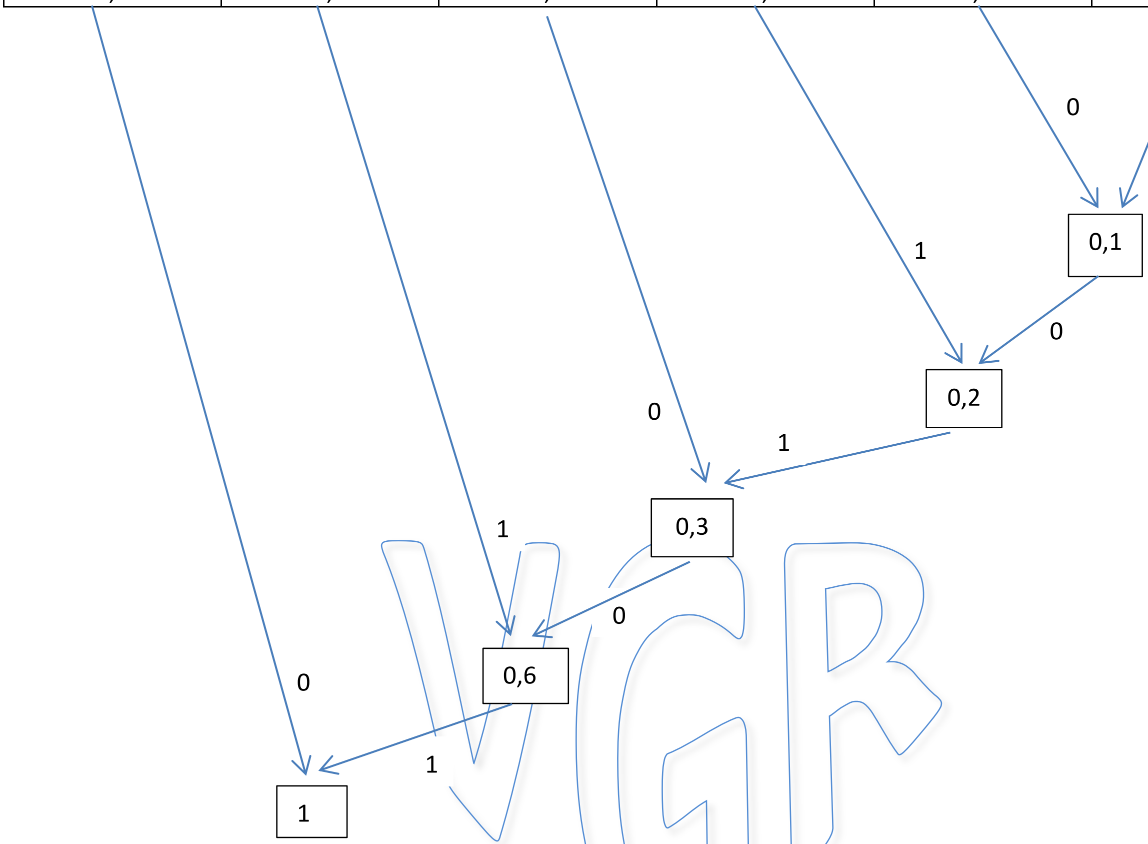
a1 -> 0
a2 -> 10
a3 -> 110
a4-> 1110
a5-> 11110
a6-> 11111

Ninguna palabra-código es prefijo de otra => es instantáneo.
L_m= 0,4 . 1 + 0,3 . 2 + 0,1 . 3 +0,1 . 4 + 0,06 . 5 + 0,04 . 5 = 2,2 sc/sf

La solución no es única pues, la asignación de unos y ceros a cada par de flechas es arbitraria en cada asignación. En unas se puede asignar el 0 a la flecha de la izquierda y el 1 a la flecha de la derecha y en otras al revés.

En el ejemplo anterior se podría hacer lo siguiente:

a1	a2	a3	a4	a5	a6
0,4	0,3	0,1	0,1	0,06	0,04



Se obtendría el siguiente código compacto:

a1-> 0
a2-> 11
a3->100
a4-> 1011
a5->10100
a6- 10101

Ninguna palabra-código es prefijo de otra => es instantáneo.
L_m= 0,4 . 1 + 0,3 . 2 + 0,1 . 3 +0,1 . 4 + 0,06 . 5 + 0,04 . 5 = 2,2 sc/sf

La misma longitud media que antes.

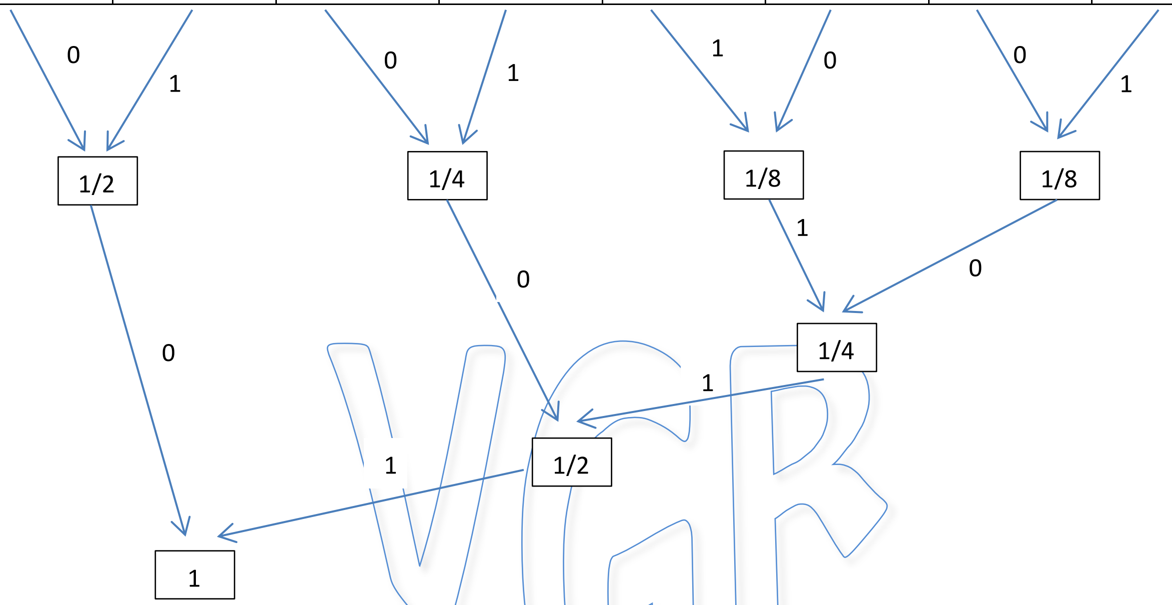
Existen muchos otros que son instantáneos y tienen una longitud media de 2,2 sc/sf , también serán códigos compactos, pero ninguno que sea instantáneo tendrá una longitud media inferior a 2,2 sc/sf.

EJEMPLO:

Dada la fuente de símbolos A={a1, a2, a3, a4, a5, a6, a7, a8} y probabilidades P={1/4, 1/4, 1/8, 1/8, 1/16, 1/16, 1/16, 1/16}. Obtener un código compacto por el método de Huffman.

Una posible solución sería:

a1	a2	a3	a4	a5	a6	a7	a8
1/4	1/4	1/8	1/8	1/16	1/16	1/16	1/16



Cuando hay más de dos elementos que tienen la menor probabilidad de los que quedan se cogen arbitrariamente dos cualesquiera de ellos. La asignación de ceros y unos a cada par de flechas también es arbitraria en el sentido de que el cero se puede asignar a la flecha de la derecha o a la de la izquierda y la asignación es independiente en cada par de flechas.

En la asignación realizada se obtendría el siguiente código compacto:

a1-> 00
a2-> 01
a3-> 100
a4-> 101
a5-> 1111
a6-> 1110
a7-> 1100
a8-> 1101

Ninguna palabra código es prefijo de las restantes.

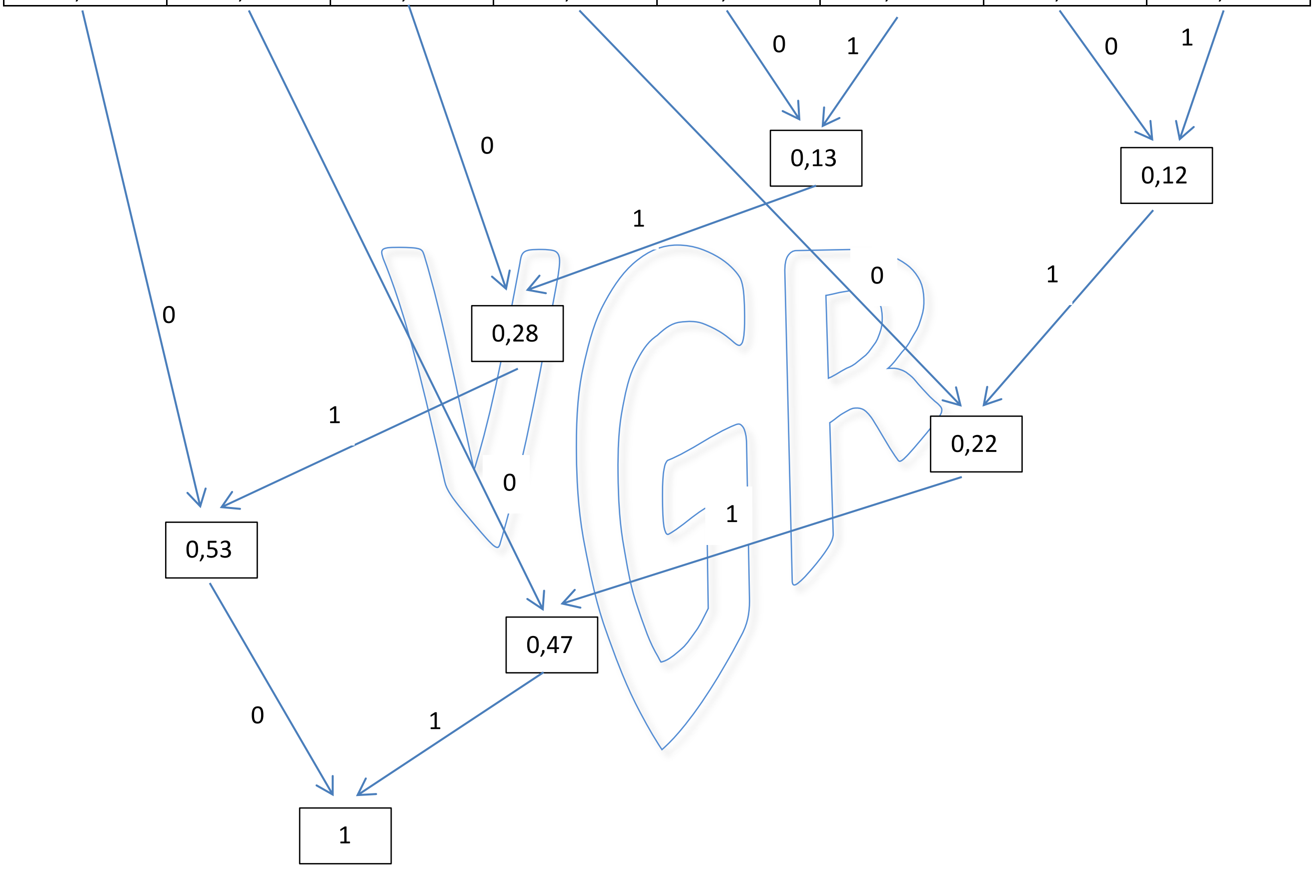
Lm= ¼ . 2 + ¼ . 2 + 1/8 . 3 + 1/8 . 3 + 1/16 . 4 + 1/16 . 4 + 1/16 . 4 + 1/16 .4 = 2,375 sc/sf

EJEMPLO

Dada una fuente de 8 símbolos A={a, x, e, z, b, c, d, u} y probabilidades respectivas P={0,25/0,06/0,065/0,06/0,25/0,15/0,10/0,065}. Obtener un código compacto por el método de Huffman.

Una posible solución será la siguiente:
Lo primero será ordenar los símbolos fuente según sus probabilidades decrecientes:

a	b	c	d	e	u	x	z
0,25	0,25	0,15	0,10	0,065	0,065	0,06	0,06



El código será:

a -> 00
b -> 10
c -> 010
d -> 110
e -> 0110
u -> 0111
x -> 1110
z -> 1111

L_m = 025 . 2 + 0,25 . 2 + 0,15 .3 + 0,1 . 3 + 0,065 . 4 + 0,065 . 4 + 0,06 . 4 + 0,06 . 4 = 2,75 sc/sf

MÉTODO DE SHANNON-FANO

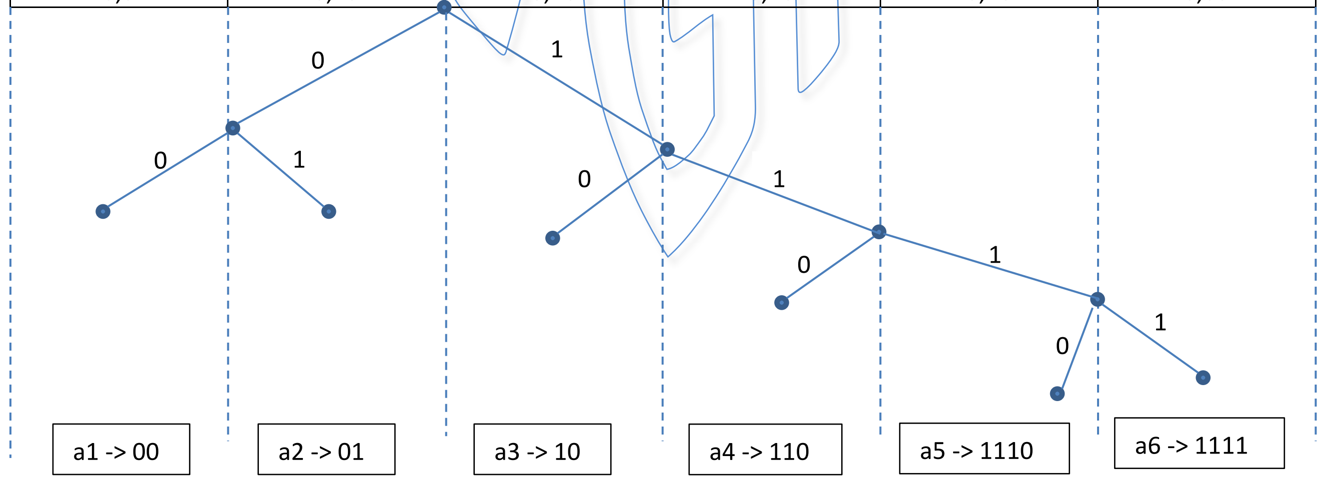
Se realizará el siguiente procedimiento:

Paso 1.- Dado el alfabeto-fuente A={a1, a2,........,an}
con sus correspondientes probabilidades P={p(a1), p(a2),.........,p(an)},
se ordenan de izquierda a derecha según probabilidades decrecientes.
Paso 2.- Las probabilidades se dividen en dos grupos de tal manera que, la suma de las probabilidades de cada grupo se aproxime lo más posible a 0,5; un grupo por encima y l otro por debajo.
Paso 3.- Se asigna un 0 como primer bit a los elementos de un grupo y un 1 como primer bit a los elementos del otro grupo.
Paso 4.- A continuación se divide cada grupo en otros dos, de tal manera que, la suma de las probabilidades de cada grupo sea lo más próxima a 0,25.
Paso 5.- Los elementos de uno de los subgrupos llevarán como segunda cifra un 0 y los elementos del otro subgrupo llevarán un 1.
Paso 6.- Se continúa el proceso hasta que en cada grupo sólo quede un símbolo de la fuente.
Paso 7.- El código correspondiente a cada símbolo fuente se establece siguiendo la siguiente regla:
Empezando por el vértice superior del árbol que se forma, a cada símbolo fuente le corresponde la sucesión de símbolos código (ceros y unos) correspondientes a la trayectoria que hay que seguir para llegar a él.

EJEMPLO

Sea la fuente de alfabeto A={a1, a2, a3, a4, a5, a6} y probabilidades P={0,3/0,25/0,20/0,10/0,10/0,05}
Codificarla en binario de una manera compacta mediante el procedimiento de Shannon-Fano.

a1	a2	a3	a4	a5	a6
0,3	0,25	0,20	0,10	0,10	0,05



Ninguna palabra código es prefijo de las restantes.

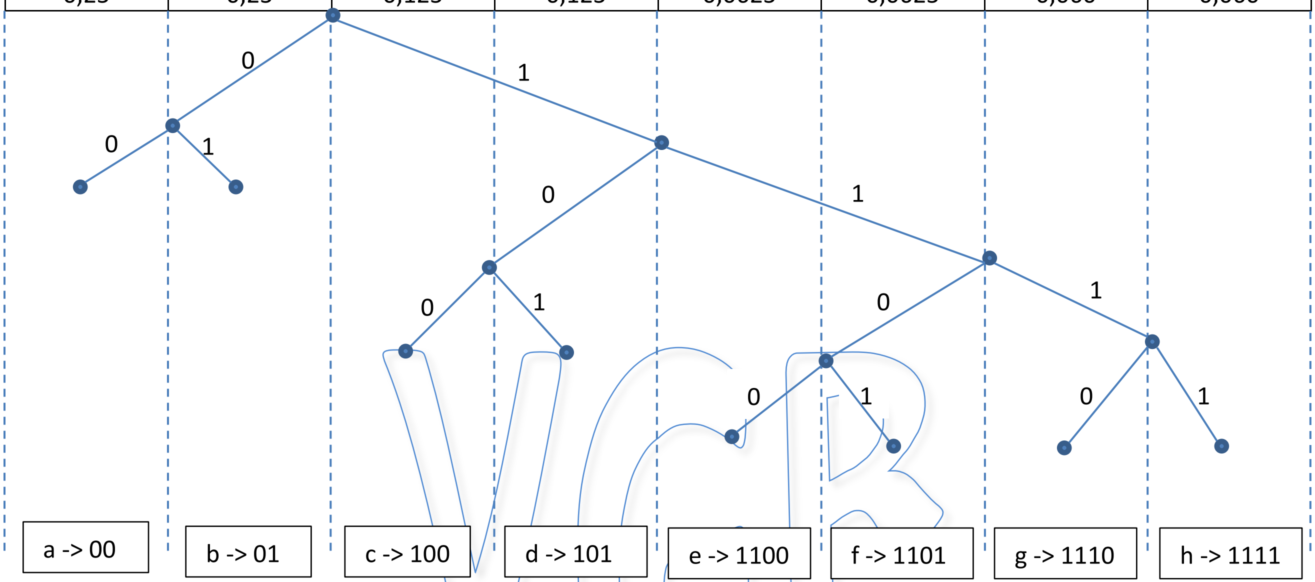
L_m= 0,3 . 2 + 0,25 . 2 + 0,20 . 2 + 0,10 . 3 + 0,10 . 4 + 0,05 . 4 = 2,4 sc/sf

Pueden obtenerse otros códigos compactos pero, todos tendrán una longitud media de 2,4sc/sf.

EJEMPLO

Dada la fuente de alfabeto A={a,b,c,d,e,f,g,h}
con probabilidades P={0,25/0,25/0,125/0,125/0,0625/0,0625/0,060/0,60}
codificarla en binario de una manera compacta siguiendo el método de Shannon-Fano.
Solución:

a	b	c	d	e	f	g	h
0,25	0,25	0,125	0,125	0,0625	0,0625	0,060	0,060



Ninguna palabra-código es prefijo de otra.
L_m = 0,25 . 2 + 0,25 . 2 + 0,125 . 3 + 0,125 . 3 + 0,0625 . 4 + 0,0625 . 4 +0,060 . 4 + 0,060 . 4 = 2,73 sc/sf

Todos los códigos compactos que se obtengan, tendrán una longitud media de 2,73 sc/sf.

NOTAS SOBRE LOS CÓDIGOS OBTENIDOS POR LOS MÉTODOS DE HUFFMAN O SHANNON-FANO

Nota 1.- Los métodos de Huffman y de Shannon-Fano poseen una redundancia igual a cero y ocupan un espacio mínimo pro, se necesita más tiempo para la decodificación pues, el algoritmo que la realice, necesita explorar el árbol correspondiente en cada caso.

Nota 2.- El código generado por ambos métodos, generalmente no es único, sino que pueden otenerse varios equivalentes. Por ejemplo, obtenido un código, si se cambian todos los ceros por unos y todos los unos por ceros se obtiene también un código compacto.

Nota 3.- Estos tipos de códigos son muy sensibles a los errores. Para decodificar un mensaje codificado con ellos que se reciba bit a bit, se examina el primer bit y se compara para saber si es una palabra código. Si no lo es, se examinan y comparan los dos primeros bits, sino los tres primeros y así sucesivamente hasta que se reconoce una palabra código. A continuación se repite el proceso empezando por el siguiente bit. En consecuencia, si un bit que se recibe es erróneo, el receptor no encuentra los límites correctos de cada palabra-código.
Para mejorar la fiabilidad de los mensajes codificados con estos métodos, se divide el total de bits que forman el mensaje, en grupos de n bits y cada uno de estos grupos se codifica mediante un código de Hamming.

EJERCICIO

Dada la fuente de alfabeto A={a,b,c,d,e,f,g,h,i,j} con probabilidades
P={0,2/0,15/0,12/0,1/0,1/0,08/0,075/0,075/0,05/0,05}
Codificarla en binario de una forma compacta utilizando los métodos de Huffman y de Shannon-Fano.
Realizada la solución, se obtiene que todos los códigos compactos tienen una longitud media igual a 3,23 sc/sf

CÓDIGOS BINARIOS CON LONGITUD CONSTANTE DE LOS NÚMEROS DECIMALES

Una manera de clasificarlos es la siguiente:

CÓDIGOS PONDERADOS

En cada palabra-código un bit tiene un valor distinto dependiendo de la posición que ocupe, es decir, a cada posición de la palabra-código se le asigna un peso y el número decimal equivalente se obtiene sumando los pesos de las posiciones que poseen los bits que valen 1.

CÓDIGO BCD

El alfabeto fuente son las cifras decimales: {0; 1; 2; 3; 4; 5; 6; 7; 8; 9}.
El alfabeto código es: {0; 1}.
La aplicación que define el código está definida por la siguiente tabla:

Cifras decimales	Palabras-código			
	Peso de los bits			
	8	4	2	1
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1

Para **pasar de un número escrito en decimal a su equivalente en BCD**, en la sucesión de cifras decimales se sustituye cada cifra decimal por su palabra-código correspondiente.

EJEMPLO:

3	7	4	,	2	5	decimal
0011	0111	0100	,	0010	0101	BCD

Para **pasar de un número escrito en BCD a su equivalente en decimal**, a partir de la coma decimal se forman grupos de 4 bits y se sustituye cada grupo por su cifra decimal correspondiente.

Nota.- No toda sucesión de unos y ceros (número escrito en base 2) puede ser un número escrito en BCD. Tanto la parte entera como la parte decimal han de poseer un número de bits que sea múltiplo de 4 y cada grupo de cuatro bits ha de ser una palabra-código de las existentes en la tabla anterior.

EJEMPLO:

Sea 01010110,1001 un número escrito en BCD, su equivalente en decimal será:
A partir de la coma se forman los grupos de 4 bits: 0101| 0110,1001
Se sustituye cada palabra-código por su cifra decimal correspondiente: 56,9 será el número decimal correspondiente.

CÓDIGOS NO PONDERADOS

En cada palabra-código no hay ningún peso asignado a las diferentes posiciones.

CÓDIGO DE EXCESO 3

Es un código bloque de longitud 4 en el cual la palabra-código correspondiente a cada cifra decimal se obtiene escribiendo con 4 bits el número binario que corresponde al número decimal que se obtiene al sumarle 3 a la cifra decimal correspondiente. Las palabras-código están definidas en la siguiente tabla:

Cifras decimales	Palabras-código	
0	0011	Número en base 2 correspondiente a (0+3)
1	0100	Número en base 2 correspondiente a (1+3)
2	0101	Número en base 2 correspondiente a (2+3)
3	0110	.
4	0111	.
5	1000	.
6	1001	.
7	1010	.
8	1011	.
9	1100	Número en base 2 correspondiente a (9+3)

Ejemplo: 405,7 en decimal será 011100111000,1010 en código exceso 3.

CÓDIGOS AUTO-COMPLEMENTARIOS

Dada una cifra decimal "n", la palabra-código correspondiente al complemento a 9 de dicha cifra decimal se obtiene hallando el complemento a 1 de la palabra-código correspondiente a "n".
Nota.- el complemento a 9 de una cifra decimal se obtiene restando de 9 dicha cifra decimal. El complemento a 9 de de 3 será: (9-3)=6.
Nota.- el complemento a 1 de un número binario se obtiene cambiando los ceros por unos y los unos por ceros. El complemento a 1 de 10110 será: 01001.
Los códigos auto-complementarios pueden ser no ponderados o ponderados.

CÓDIGOS AUTOCOMPLEMENTARIOS NO PONDERADOS

El código exceso 3 es auto-complementario.
Por ejemplo el 3 decimal tiene como complemento a 9 el número 6.
La palabra-código en exceso 3 del número 3 es 0110.
La palabra-código en exceso 3 del número 6 es 1001.
El complemento a 1 de 0110 es 1001.
Es decir, se cumplen las condiciones para ser auto-complementario. Lo mismo ocurre con el resto de las cifras decimales.

CÓDIGOS AUTOCOMPLEMENTARIOS PONDERADOS

CÓDIGO DE AIKEN

Para obtener las palabras-código se escriben con cuatro bits en binario desde el 0 hasta el cuatro y las restantes del 5 al 9 se obtienen de las anteriores complementando a 1 las palabras-código de las cifras complementarias a 9. La siguiente tabla define todas las palabras-código:

Cifras decimales	Pesos 2421
0	0000
1	0001
2	0010
3	0011
4	0100
5	1011
6	1100
7	1101
8	1110
9	1111

Otros códigos auto-complementarios ponderados:
Los códigos con pesos 5211; 4311; 4221; 3321 son auto-complementarios.

CÓDIGOS CONTINUOS

En estos códigos, las palabras-código correspondientes a cifras decimales consecutivas son adyacentes, es decir, difieren en un solo bit.
Un **código cíclico** es un código continuo en el que la última palabra-código es adyacente a la primera palabra-código.

CÓDIGO DE GRAY

Es un código binario continuo y cíclico que también se llama **código reflejado** pues la formación de las palabras-código de longitud “n” se consigue partiendo de las palabras-código de longitud “n-1” repitiendo **simétricamente con respecto a un eje horizontal** estas últimas palabras-código y añadiendo por la izquierda un nuevo bit: un 0 para las $2^{(n-1)}$ primeras palabras-código y un 1 para las $2^{(n-1)}$ palabras-código siguientes.

Nº decimal	Palabras de 1 bit	Palabras de 2 bits	Palabras de 3 bits	Palabras de 4 bits
0	0	0-0	0-00	0-000
1	1	0-1	0-01	0-001
2		1-1	0-11	0-011
3		1-0	0-10	0-010
4			1-10	0-110
5			1-11	0-111
6			1-01	0-101
7			1-00	0-100
8				1-100
9				1-101
10				1-111
11				1-110
12				1-010
13				1-011
14				1-001
15				1-000

PASO DE GRAY A BINARIO

Dado un número decimal escrito en el código de GRAY para obtener el mismo número escrito en binario se realizarán los pasos siguientes:

- El primer bit por la izquierda del número escrito en GRAY se coloca también en binario.
- Se van observando sucesivamente los siguientes bits del número escrito en GRAY.

- Si existe un 1, el bit correspondiente en binario cambia su valor con respecto al bit anterior en binario, es decir, si la cifra anterior en binario era un 0 se pone 1 y si la cifra anterior en binario era 1 se pone un 0.
- Si existe un 0, el bit correspondiente en binario se mantiene igual a la cifra anterior en binario.

Ejemplos:

1011000110	En GRAY	1001100101	En GRAY
1101111011	En binario	1110111001	En binario

PASO DE BINARIO A GRAY

Dado un número decimal escrito en binario para obtener el mismo número escrito en código GRAY se realizarán los pasos siguientes:

- El primer bit por la izquierda del número escrito en binario se coloca también en GRAY.
- Se van observando sucesivamente los siguientes bits del número escrito en binario.
  - Si el siguiente bit en binario cambia con respecto al bit anterior en binario se colocará un 1 en el correspondiente bit en GRAY.
  - Si el siguiente bit en binario no cambia con respecto al bit anterior en binario se colocará un 0 en el correspondiente bit en GRAY.

Ejemplos:

1101111011	En binario	1110111001	En binario
1011000110	En GRAY	1001100101	En GRAY

Otra manera de obtener los bits en GRAY es aplicar la fórmula:

$$g_i = b_i \boxed{+} b_{i-1}$$

en donde: gi es un bit en GRAY, bi es el correspondiente bit en binario y $\boxed{+}$ es la suma directa bit a bit.

Nota.- La suma directa está definida por la siguiente tabla:

0 + 0 =	0
0 + 1 =	1
1 + 0 =	1
1 + 1 =	0

CÓDIGO PROGRESIVO DE JOHNSON

Es un código continuo y cíclico. El principal inconveniente es que con palabras-código de "n" bits sólo se pueden codificar "2n" símbolos.
Las palabras-código correspondientes a las cifras decimales se definen en la tabla siguiente:

Decimal	Johnson
0	00000
1	00001
2	00011
3	00111
4	01111
5	11111
6	11110
7	11100
8	11000
9	10000

Inicio