Codage et compression des données

1. Incertitude et information d'un évènement

1.1. Introduction

Considérons une expérience possédant le propriété suivante : quelque soit sa nature, quelque soit l'endroit, son résultat est inconnu avant sa réalisation. Par contre, nous pouvons décrire l'ensemble de ces résultats possibles. Cet ensemble est un système complet d'évènements.

Soit une expérience A dont les résultats possibles sont un nombre fini d'évènements $a_1,\ldots, a_n$.

Posons $p_1,\ldots, p_n$ les probabilités de réalisation de ces évènements, nous avons donc :

$$\forall 1 \le k \le n \qquad p_k \ge 0 \text{ et } \sum_{k=1}^{n} p_k = 1$$

Notre expérience met en évidence un certain espace probabilisé $\{A, a_k, p_k\}$ et par conséquent un certain schéma :

$$A = \begin{cases} a_1 \rightarrow p_1 \\ \ldots \\ a_n \rightarrow p_n \end{cases}$$

Problème : étant donné que nous ne savons pas d'avance le résultat de l'expérience A, il s'ensuit qu'elle contient une certaine incertitude qui ne pourra être levée qu'après la réalisation :

- *comment mesurer cette incertitude sur A ?*
- *comment calculer la dimension de cette incertitude ?*
- *quel est l'intérêt de mesurer l'incertitude ?*

Remarque :

- *une telle possibilité de mesurer l'incertitude nous permettrait de comparer deux incertitudes de deux expériences différentes*
- *l'incertitude dépend de la probabilité de réalisation des évènements*

1.2. Mesure de l'incertitude

Soit {A, a_k, p_k} un espace probabilisé et $a_k \in A$ un évènement. On se propose de définir une présence de l'incertitude H(a) liée à l'évènement a de telle sorte que :

- H(a) soit d'autant plus grande que P(a) est petite ; on peut poser alors $H(a) = f\left(\frac{1}{P(a)}\right)$ avec f une fonction croissante
- H(a) = 0 quand P(a) = 1 ; il n'y a aucune incertitude quant à la réalisation d'un évènement certain donc f(1) = 0
- si a et b sont deux évènements indépendants H(a ∩ b) = H(a) + H(b) il vient $f\left(\frac{1}{P(a \cap b)}\right) = f\left(\frac{1}{P(a) * P(b)}\right) = f\left(\frac{1}{P(a)}\right) + f\left(\frac{1}{P(b)}\right)$

⇨ on cherche une fonction f telle que :

- $f : [1 ; +\infty [\rightarrow R^+$
- *f est croissante*
- *f(1) = 0*
- *f(x.y) = f(x) + f(y)*

En dérivant f(x.y) = f(x) + f(y) par rapport à x on obtient y.f'(x.y)=f'(x)

Posons $y = \frac{1}{x}$ => $\frac{1}{x}.f'(1) = f'(x)$ avec f'(1) > 0

La solution est de la forme α.ln(x) = f(x) avec α > 0

Posons $\alpha = \frac{1}{\ln \beta}$ avec β > 1 ; il vient $f(x) = \frac{\ln x}{\ln \beta}$

Unités : choisir une unité d'incertitude revient à fixer la base β. On définit alors :

- *le NAT (natural unit) pour β = e*
- *le DECIT (decimal unit) pour β = 10*
- *le bit (binary unit) pour β = 2*

la base 2 étant sous-entendue dans toute la suite.

Dans ce cas $H(a) = \frac{\ln \frac{1}{P(a)}}{\ln 2} = \log_2\left(\frac{1}{P(a)}\right)$

d'où l'incertitude $H(a) = -\log_2 P(a)$

Remarque : binary unit (bit) et binary digit (chiffre)

- *le bit désigne une unité d'incertitude*
- *le digit désigne un signe*

⇨ le bit est la quantité d'incertitude fournie par le choix entre deux évènements équiprobables.

Entropie

La mesure de l'incertitude sur tout l'espace probabilisé (c'est-à-dire l'expérience A) s'appelle d'après Claude E. Shannon (1948) l'entropie de l'expérience A et est donnée par $H(A) = -\sum_{k=1}^{n} p_k \log_2 p_k$

1.3. Mesure de l'information

Considérons deux évènements a et b ; la probabilité conditionnelle P(a/b) définie par $P(a/b) = \frac{P(a \cap b)}{P(b)}$ peut être interprétée comme le changement de la probabilité P(a) de l'évènement a lorsqu'on reçoit l'information « l'évènement b est réalisé ». Dans le cas où $a \subset b$ on a $a \cap b = a$ et $P(a \cap b) = P(a)$ donc $P(a/b) = \frac{P(a)}{P(b)} \geq P(a)$.

Il s'ensuit que l'information « b réalisé » augmente la probabilité de l'évènement a ou diminue l'incertitude sur a. En effet :

$$H(a/b) = -\log P(a/b) = -\log(P(a) + P(b)) = H(a) - H(b)$$

Donc savoir que b est réalisé diminue l'incertitude H(a) de la quantité H(b). On a apporté la quantité d'information H(b). En particulier si a = b on a H(a/a) = 0 qui exprime simplement que l'incertitude sur a est supprimée dès que l'évènement a est réalisé.

⇨ L'incertitude sur un évènement et l'information apportée sur la réalisation de cet évènement sont des notions équivalentes ; on a donc :

quantité d'information de A = $H(A) = -\sum_{k=1}^{n} p_k \log_2(p_k)$

Interprétation numérique : H(A) correspond au nombre moyen d'éléments binaires nécessaires à la représentation des différents évènements de l'expérience.

Exemple : on considère une expérience A qui consiste à extraire au hasard une carte d'un jeu de 32 cartes → calculer l'entropie de A.
Chaque carte a la même probabilité d'être extraite ; il vient :

$$H(A) = -\sum_{k=1}^{32} p_k \log_2 p_k = -[p_1 \log_2 p_1 + ... + p_{32} \log_2 p_{32}]$$

$$H(A) = -\left[\frac{1}{32}\log_2 \frac{1}{32} + ... + \frac{1}{32}\log_2 \frac{1}{32}\right] = -\log_2 \frac{1}{32} = \log_2 32 = 5 bits$$

Interprétation : pour savoir quelle carte a été extraite il faut :

- *1 bit pour la couleur (0 pour rouge, 1 pour noir)*
- *1 bit pour faire la différence cœur/carreau ou pique/trèfle*
- *1 bit pour savoir si la carte appartient ou groupe {7,8,9,10} ou {valet,dame,roi,as}*
- *1 bit pour savoir à quel sous-groupe constitué de deux cartes elle appartient*
- *1 bit pour connaître la carte tirée parmi les deux cartes*

1.4. Propriétés de l'entropie

- *H(A) ≥ 0 (1)*
- *si pour un indice i nous avons $p_i = 1$ alors H(A) = 0 (2)*
- *pour toute répartition $p_1,...,p_n$ nous avons* $\boxed{H(p_1,...,p_n) \le H\left(\frac{1}{n},...,\frac{1}{n}\right) = \log_2 n}$ *(3)*

Interprétation
L'entropie est maximale quand toutes les probabilités sont égales entre elles. Cette propriété de l'entropie correspond à l'image que nous avons de l'incertitude puisqu'il est clair que l'expérience de plus grande incertitude parmi toutes les expériences ayant n résultats possibles est celle dont les résultats ont la même probabilité. Aucun résultat n'a alors un résultat préférentiel (équiprobabilité).

1.5. Entropie liée à un couple de variables

Soit deux expériences A et B et admettons les schémas suivants :

$$A = \begin{cases} a_1 \to p_1 \\ ... \\ a_n \to p_n \end{cases} \qquad B = \begin{cases} b_1 \to p_1 \\ ... \\ b_m \to p_m \end{cases}$$

L'évènement qui consiste en la réalisation conjointe des évènements a_k et b_l sera noté ($a_k \cap b_l$) ou (a_k,b_l) et sa probabilité par Π_{kl} avec $1 \le k \le n$ et $1 \le l \le m$. On a alors avec $1 \le k \le n$:

$$\Pi_{kl} = P(a_k, b_l) = P(a_k) * P(b_l / a_k) = p_k * P(b_l / a_k)$$

Premier cas : A et B sont indépendantes, dans ce cas $P(b_l / a_k) = p(b_l) = q_l$ et $\Pi_{kl}=p_k * q_l$

Question : *quelle relation y'a t'il entre l'entropie H(A,B) sur l'expérience produit (A,B) et les informations H(A) et H(B) ?*

Propriété : si A et B sont indépendantes alors $\boxed{H(A,B) = H(A) + H(B)}$ (4)
Preuve :

$$H(A,B) = -\sum_{k=1}^{n}\sum_{l=1}^{n} \Pi_{kl} \log_2 \Pi_{kl} = -\sum_{k=1}^{n}\sum_{l=1}^{n} p_k q_l \log_2 p_k q_l = -\sum_{k=1}^{n}\sum_{l=1}^{n} p_k q_l (\log_2 p_k + \log_2 q_l)$$

$$H(A,B) = -\sum_{k=1}^{n}\sum_{l=1}^{n} p_k q_l \log_2 p_k - \sum_{k=1}^{n}\sum_{l=1}^{n} p_k q_l \log_2 q_l$$

$$H(A,B) = -\sum_{k=1}^{n} p_k \log_2 p_k \sum_{l=1}^{n} q_l - \sum_{l=1}^{n} q_l \log_2 q_l \sum_{k=1}^{n} p_k = H(A) + H(B)$$

Deuxième cas : A et B sont quelconques

Dans ce cas, l'expérience B est en général conditionnée par les résultats de A, c'est-à-dire B est une autre expérience mettant en évidence m évènements $b_1,\ldots,b_m$ dont la réalisation est directement influencée par le résultat de A.
Nous ne pouvons alors plus parler de la probabilité des évènements $b_1,\ldots,b_m$ soit des probabilités q_l avec $1 \leq l \leq m$. Donc pour chaque évènement b_l nous avons n possibilités $q_{1l},\ldots,q_{nl}$ avec $1 \leq l \leq m$ et $\sum_{k=1}^{n} q_{kl} = 1$ pour tout $1 \leq k \leq n$.
q_{kl} représente la probabilité de réalisation de l'évènement b_l en supposant que l'expérience A nous ait fourni l'évènement a_k. Si nous considérons l'expérience produit (A,B) alors la probabilité Π_{kl} s'écrit $\Pi_{kl} = P(a_k, b_l) = P(a_k) * q_{kl} = p_k * q_{kl}$.
Si dans l'expérience A l'évènement a_k s'est réalisé alors pour l'expérience B nous avons le schéma suivant :

$$B = \begin{cases} b_1 \rightarrow q_{k1} \\ \ldots \\ b_m \rightarrow q_{km} \end{cases}$$

C'est la répartition conditionnelle de B quand l'évènement a_k est donné et comme A a n évènements possibles il s'ensuit que pour B nous aurons n répartitions conditionnelles possibles :

$$B(a_1) = \begin{cases} b_1 \rightarrow q_{11} \\ \ldots \\ b_m \rightarrow q_{1m} \end{cases} \quad \ldots \quad B(a_n) = \begin{cases} b_1 \rightarrow q_{n1} \\ \ldots \\ b_m \rightarrow q_{nm} \end{cases}$$

Pour chacune de ces n répartitions nous avons :

$$\boxed{H_k(B) = -\sum_{l=1}^{m} q_{kl} \log_2 q_{kl}}$$

Définition : $H_k(B)$ s'appelle l'entropie (information) conditionnelle de B quand l'évènement a_k est donné.

Définition : $H_A(B) = \sum_{k=1}^{n} p_k * H_k(B)$ s'appelle l'entropie conditionnelle de B quand l'expérience A est donnée.

Remarque : si A et B sont indépendantes nous avons $H_k(B) = H(B)$; il s'ensuit que :

$$H_A(B) = \sum_{k=1}^{n} p_k * H_k(B) = \sum_{k=1}^{n} p_k * H(B) = H(B) * \sum_{k=1}^{n} p_k = H(B)$$

Propriété : soit A et B quelconques alors nous avons $\boxed{H(A,B) = H(A) + H_A(B)}$ (5)

Interprétation :
La propriété (5) nous montre que pour deux expériences quelconques l'incertitude sur l'expérience produit (A,B) est égale à l'incertitude sur une expérience augmentée de l'incertitude donnée sur l'autre quand la première est donnée.
Si A et B sont indépendantes, on retrouve bien H(A,B) = H(A) + H(B) (4).

Propriété : si A et B sont quelconques alors nous avons $\boxed{H_A(B) \leq H(B)}$ (6)
l'égalité n'étant possible que si A et B sont indépendantes.

Interprétation : si nous effectuons l'expérience B seule son résultat nous fournit une quantité d'information H(B).

Question : *supposons que nous effectuons l'expérience A, quelle serait la quantité d'information que nous allons recevoir sur l'expérience B ?*
La propriété (6) nous dit alors que si nous effectuons A, l'incertitude sur B diminue et devient $H_A(B)$ telle que $H_A(B) \leq H(B)$. Il s'ensuit qu'en effectuant A, la quantité d'information reçue sur B est égale à l'incertitude de B qui s'élimine si on effectue A c'est-à-dire

$$\boxed{I(A,B) = H(B) - H_A(B)}$$

C'est la quantité d'information apportée par A sur B.

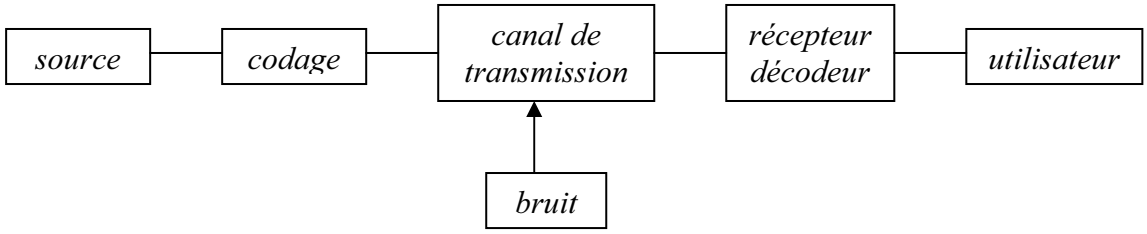
Remarque :
- *si A et B sont indépendantes alors $H_A(B) = H(B)$ donc $I(A,B) = 0$*
- *on montre que* ***I(A,B) = I(B,A)***

Propriétés : soit A et B quelconques, nous avons alors :
- *$H(A,B) \leq H(A) + H(B)$, l'égalité n'étant possible que si A et B sont indépendantes*
- *$H_B(A) = H_A(B) + H(A) - H(B)$*

Remarque :
- *si A détermine complètement B alors sa réalisation fait disparaître toute incertitude sur B ; dans ce cas $H_A(B) = 0$ et $I(A,B) = H(B)$*
- *si A et B sont liées sans que A détermine complètement B alors $0 \leq I(A,B) \leq H(B)$*

Transmission de l'information

1. Source



Soit X un ensemble fini de signaux x avec lesquels nous travaillons et soit P(x) la probabilité de transmission du signal x. La probabilité P(x) est établie à la suite d'une recherche statistique en évaluant la fréquence relationnelle d'utilisation de chaque signal dans la transmission du signal considéré.
Donc transmettre une certaine quantité d'information revient à connaître la probabilité d'utilisation de chaque signal x de X.

Définition : l'espace probabilisé {X,x,P(x)} s'appelle source discrète.

Définition : une source X est stationnaire ssi les signaux ont la même loi de probabilité.

Définition : on dira que la source X a une mémoire d'ordre m ssi

$P(x_i/x_1,x_2,\ldots,x_n) = P(x_i/x_1,x_2,\ldots,x_{i-m})$
si m = 0 la source est dite sans mémoire
si m = 1 la source est dite de Markov

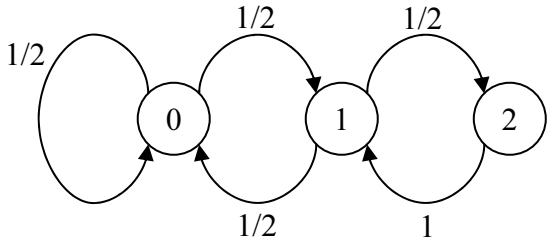
Définition : on définira $H_\infty(X)$ d'une source de mémoire d'ordre m par

$H_\infty(X) = H(x_i/x_{i-1},x_{i-2},\ldots x_{i-m}) = H_A(x_i)$ avec $A = (x_{i-1},x_{i-2},\ldots x_{i-m})$
soit $H_\infty(X) = H_A(B)$ avec $B = (x_i)$

Définition : soit D_S le nombre de symboles finis pour la source en 1 seconde ; le débit d'information noté H'(X) est

$H'(X) = D_S * H_\infty(X)$

Exemple : considérons une source ternaire d'alphabet X = {0,1,2} ; supposons que X est une source de Markov de graphe :



$$T = \begin{pmatrix} 1/2 & 1/2 & 0 \\ 1/2 & 0 & 1/2 \\ 0 & 1 & 0 \end{pmatrix}$$

La source étant d'ordre 1, on a alors :

$H(x_i/x_{i-1} = 0) = -1/2 \log_2 1/2 - 1/2 \log_2 1/2 = 1$ bit
$H(x_i/x_{i-1} = 1) = -1/2 \log_2 1/2 - 1/2 \log_2 1/2 = 1$ bit
$H(x_i/x_{i-1} = 2) = 0$ bit

$$H(x_i / x_{i-1}) = H_{x-i}(x_i) = \sum_{l=0}^{2} P(x_l) * H(x_i / x_{i-1} = x_l)$$

$$H(x_i / x_{i-1}) = P(0) * H(x_i / x_{i-1} = 0) + P(1) * H(x_i / x_{i-1} = 1) + P(2) * H(x_i / x_{i-1} = 2)$$

Pour calculer $\begin{pmatrix} P(0) \\ P(1) \\ P(2) \end{pmatrix}$ on cherche la loi stationnaire c'est-à-dire $\begin{pmatrix} P(0) \\ P(1) \\ P(2) \end{pmatrix} = T^t * \begin{pmatrix} P(0) \\ P(1) \\ P(2) \end{pmatrix}$

avec $P(0) + P(1) + P(2) = 1$

La résolution de ce système conduit à : $P(0) = P(1) = 1/3$ et $P(2) = 1/3$
d'où $H_\infty(X) = H(x_i/x_{i-1}) = 1/3 * 1 + 1/3 * 1 + 1/3 * 0 = 0{,}66$ bit

$$\begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} = \begin{pmatrix} 1/2 & 1/2 & 0 \\ 1/2 & 0 & 1 \\ 0 & 1/2 & 0 \end{pmatrix} * \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix}$$

$y_1 = 1/2 * y_1 + 1/2 * y_2$
$y_2 = 1/2 * y_1 + 1/2 * y_3$
$y_3 = y_2$
et $y_1 + y_2 + y_3 = 1$

2. Codage de source

Soit X une source de signaux x et une certaine information exprimée à l'aide de ces signaux. Soit une autre variable A de signaux a.
Coder l'information à l'aide des signaux a de A revient à réaliser une correspondance entre X et l'ensemble des séquences de signaux de A, donc à chaque signal x de X on associe une séquence de signaux de A.
Ainsi le codage est une fonction f définie sur X et à valeurs dans l'ensemble A* de toutes les séquences de signaux de A, c'est-à-dire :

$f : X \rightarrow A^*$
$x \rightarrow f(x)$

on notera Q la taille de A c'est-à-dire Card(A) = Q et n(x) la longueur du mot code du signal x soit le nombre de symboles nécessaires à la représentation de x :

X = {A,B,C} Q = 2 A = {0,1}
A → 0 0
n(A) = 2

Question : *pourquoi* a-t-on besoin du codage de l'information ?
- *nature du système de communication : il faut absolument s'adapter*
- *présence de bruit dans la voie de communication et comme le bruit a pour effet la perte d'information, il est nécessaire de pouvoir disposer d'un outil permettant de récupérer l'information perdue*
- *compression de l'information sans perte*

Théorème : compression sans perte d'information
Si on se donne une source X d'alphabet à N symboles et d'entropie $H_\infty(X)$, alors son contenu est compréhensible sans perte d'information si

$\boxed{H_\infty(X) < \log_2 N}$ soit si son entropie n'est pas maximale.

On dit alors que la source possède de la redondance et l'opération du codage de source va consister à réduire en partie/totalité la redondance.

Définition : on peut mesurer la redondance par :

$$\boxed{r = 1 - \frac{H_\infty(X)}{\log_2 N}} \quad en\ \%$$

2.1. Code préfixé

Exemple : soit X une source d'alphabet {A,B,C,D} et supposons les mots codes suivants :
A → 1 ; B → 10 ; C → 00 ; D → 01

Si on reçoit le code 10001 on constate qu'elle peut être interprétée de deux façons : ACD ou BCA. On dit alors que le code n'est pas uniquement déchiffrable. Par conséquent une propriété importante d'un code est son caractère uniquement déchiffrable.

Théorème : une condition suffisante pour qu'un code soit uniquement déchiffrable est qu'il soit préfixé c'est-à-dire qu'il ne comporte aucun mot code qui soit le début d'un autre mot code.

Question : à quelle condition peut-on transformer un code quelconque en un code préfixé ?

Théorème : une condition nécessaire et suffisante pour qu'un code C puisse être transformé en un code préfixé équivalent (possédant la même distribution de longueur des mots) est que l'inégalité suivante (inégalité de Kraft) soit satisfaite :

$$\sum_{a \in C} Q^{-n(a)} \leq 1$$

Exemple : A → 1 ; B → 10 ; C → 00 ; D → 01 et Q = 2
$2^{-1} + 2^{-2} + 2^{-2} + 2^{-2} = 1/2 + 3/4 = 5/4 > 1$ donc le code n'est pas préfixé.

Théorème : un code C uniquement déchiffrable vérifie l'inégalité de Kraft.

2.2. Performances du codage

Soit X une source discrète de taille N et x_2 de X un signal de probabilité $P(x_i)$.

Définition : la longueur moyenne $\bar{n}$ des séquences de codage est définie par

$$\bar{n} = \sum_{i=1}^{N} P(x_i) * n(x_i)$$

avec $n(x_i)$ le nombre de symboles nécessaires à la représentation du signal x_i.

Objectif : un codage est d'autant plus économique qu'il fait correspondre aux signaux des fréquences d'émission plus courtes ; il en résulte que nous sommes intéressés par des codages dont la longueur moyenne $\bar{n}$ des séquences est la plus petite.

Problèmes :

1. *comment peut-on évaluer la plus petite longueur moyenne $\bar{n}$ possible ; peut-on donner une borne inférieure pour $\bar{n}$?*
2. *existe-t-il un codage correspondant à cette longueur moyenne minimale $\bar{n}$; existe-t-il un codage le plus économique et comment peut-on le réaliser ?*

Théorème (1) : pour pouvoir effectuer un codage en utilisant un ensemble A de Q signaux à l'aide desquels il faut transmettre une quantité d'information $H_\infty(X)$, il est nécessaire que

$$\boxed{\bar{n} \geq \frac{H_\infty(X)}{\log_2 Q} = \bar{n}^*}$$

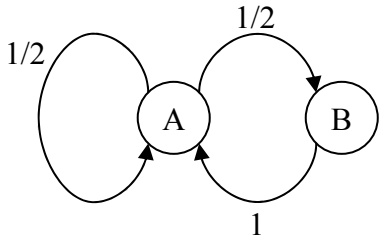
$\bar{n}^*$ (longueur moyenne optimale) est une limite inférieure pour $\bar{n}$ en dessous de laquelle il est impossible de descendre.

Théorème de Shannon (2) : soit X une source à l'aide de laquelle nous transmettons une quantité d'information $H_\infty(X)$. Supposons que l'on effectue le codage de cette information à l'aide de l'ensemble de Q signaux simples en attachant des séquences de codage non pas à chaque signal x de X mais directement aux blocs de M signaux x_i.
On obtient une nouvelle source appelée extension d'ordre M :

$$\boxed{\frac{H_\infty(X)}{\log_2 Q} \leq \bar{n} = \frac{\bar{n}_M}{M} \leq \frac{H_\infty(X)}{\log_2 Q} + \frac{1}{M}}$$

où $\bar{n}_M$ est la longueur moyenne correspondant aux blocs de M signaux.

Application : on considère une source X de mémoire d'ordre 1 (de Markov) :



$$T = \begin{pmatrix} 1/2 & 1/2 \\ 1 & 0 \end{pmatrix}$$

La loi stationnaire est donnée par $\begin{pmatrix} P(A) \\ P(B) \end{pmatrix} = T^t * \begin{pmatrix} P(A) \\ P(B) \end{pmatrix} = \begin{pmatrix} 1/2 & 1 \\ 1/2 & 0 \end{pmatrix} * \begin{pmatrix} P(A) \\ P(B) \end{pmatrix}$

$$\Rightarrow \begin{cases} P(A) = 1/2 * P(A) + P(B) \\ P(B) = 1/2 * P(A) \\ P(A) + P(B) = 1 \end{cases} \Rightarrow \begin{cases} P(A) = 2/3 \\ P(B) = 1/3 \end{cases}$$

$H_\infty(X) = P(A)*H_{Xn-1=A}(X_n) + P(B) * H_{Xn-1=B}(X_n)$
$H_{Xn-1=A}(X_n) = -1/2 \log_2 1/2 - 1/2 \log_2 1/2 = 1$ bit
$H_{Xn-1=B}(X_n) = -1 \log_2 1 = 0$
d'où $H_\infty(X) = 2/3 * 1 + 1/3 * 0 = 2/3$ soit environ 0,67 bits

Premier cas : *on code la source X par A → 0 et B → 1, donc* $\bar{n}$ *= 1 bit*

on a $\bar{n} \geq \frac{H_\infty(X)}{\log_2 Q} = 0{,}67 bit$

Deuxième cas : *on considère l'extension d'ordre 2*

$P(AA) = P(X_{n-1}=A \cap X_n=A) = P(A) * P(X_n=A/X_{n-1}=A) = 2/3 * 1/2 = 1/3$
$P(AB) = P(X_{n-1}=A \cap X_n=B) = P(A) * P(X_n=B/X_{n-1}=A) = 2/3 * 1/2 = 1/3$
$P(BA) = P(X_{n-1}=B \cap X_n=A) = P(B) * P(X_n=A/X_{n-1}=B) = 1/3 * 1 = 1/3$
$P(BB) = P(X_{n-1}=B \cap X_n=B) = P(B) * P(X_n=B/X_{n-1}=B) = 1/3 * 0 = 0$

$AA \rightarrow 1$; $AB \rightarrow 00$; $BA \rightarrow 01$
$\bar{n}_2 = 1/3 * 1 + 1/3 * 2 + 1/3 * 2 = 1,66$
donc $\bar{n} = \frac{\bar{n}_2}{2} = \frac{1,66}{2} = 0,83 bit$
$1 – 0,67 = 33\ \%$
$1 – 0,83 = 17\ \%$

Canaux de transmission

Définition : on définit un canal de transmission avec :
- *un alphabet d'entrée $\{x_1,x_2,...,x_n\}$ qui correspond aux valeurs possibles d'une variable aléatoire X*
- *un alphabet de sortie $\{y_1,y_2,...,y_m\}$ qui correspond aux valeurs possibles d'une variable aléatoire Y*
- *une matrice de transition dont les termes t_{ij} sont les probabilités de transition $X \rightarrow Y$, c'est-à-dire $t_{ij} = P(Y=y_i/X=x_j)$: la probabilité conditionnelle de recevoir le signal y_i lorsque l'on a émis le signal x_j*

Définition : un canal est dit discret si :
- *X et Y ne comprennent qu'un nombre fini de symboles*
- *un canal sans mémoire est un canal pour lequel Y_k ne dépend que de X_k avec k l'échelle du temps*

1. Bruit et capacité d'un canal

Supposons que $P(Y=y_i/X=x_j)$ prenne seulement les valeurs 0 et 1 : $\forall i \in \{1,2,...,m\}$ et $\forall j \in \{1,2,...,n\}$, considérons alors que le signal $x_l \in X$ est fixé.
Puisque $P(Y=y_i/X=x_j)$ est une probabilité sur Y nous avons alors :

$$\boxed{\sum_{i=1}^{m} P(Y = y_i / X = x_l) = 1} \text{ (a)}$$

et comme $P(Y=y_i/X=x_l)$ ne peut prendre que les valeurs 0 ou 1, il s'ensuit de (a) qu'il existe un seul signal $y_p \in Y$ tel que $P(Y=y_p/X=x_l) = 1$ et pour le reste des signaux y_i avec $i \neq p$ nous avons $P(Y=y_i/X=x_l) = 0$.

Conclusion : à chaque signal émis correspond un seul signal reçu (probabilité conditionnelle de 0 ou 1).

Définition : si $P(Y=y_i/X=x_j)$ ne prend pas seulement les valeurs 0 ou 1, $\forall (y_i, x_j) \in Y \times X$ nous dirons qu'il y a du bruit sur la voie.

On sait que $H(X/Y) = H_Y(X)$ représente l'incertitude sur l'entrée lorsque la sortie est connue. Comme elle dépend de la probabilité conditionnelle $P(X=x_i/Y=y_j)$ qui à son tour est déterminée par $P(Y=y_j/X=x_i)$ alors H(X/Y) apparaît comme le bruit.

Conclusion : H(X/Y) représente la quantité moyenne d'information qui, pendant la transmission, se perd dans la voie à cause du bruit.

Finalement si l'on transmet la quantité d'information H(X) de la source, il s'ensuit qu'il n'arrivera seulement qu'une quantité d'information $I(X,Y) = H(X) - H(X/Y)$ à l'utilisateur.
Si nous considérons la valeur maximale de I(X,Y), pour toutes les probabilités d'émission possibles, on met ainsi en évidence la quantité maximale d'information qui peut être acheminée par le canal. Cette valeur maximale peut donc être appelée capacité du canal.

Définition : on appelle capacité d'un canal de transmission la valeur :

$$C = \max_{P(X)}(I(X,Y))$$

Exemples : calculs de capacité

- *canal parfait*

$$\begin{array}{ccc} X & & Y \\ A & \xrightarrow{1} & A' \\ B & \xrightarrow{1} & B' \\ C & \xrightarrow{1} & C' \\ D & \xrightarrow{1} & D' \end{array}$$

I(X,Y) = H(X) – H(X/Y)
or H(X/Y) = 0 car connaître Y revient à connaître X
donc maximiser I(X,Y) $\Leftrightarrow$ maximiser H(X)
soit max(I(X,Y)) = $\log_2 4$ = 2 bits = C.

- *canal quelconque*

$$\begin{array}{ccc} X & & Y \\ A & \xrightarrow{1} & A' \\ B & \xrightarrow{1} & A' \\ C & \xrightarrow{1} & B' \\ D & \xrightarrow{1} & B' \end{array}$$

I(X,Y) = H(X) – H(X/Y) ou I(Y,X) = H(Y) – H(Y/X)
or H(Y/X) = 0 car avec X on connaît Y
d'où I(Y,X) = H(Y) ; calculons la loi de Y :
P(Y=A') = P(Y=A'/X=A)*P(A) + P(Y=A/X=B)*P(B) = P(A) + P(B) = $p_A + p_B$
P(Y=B') = 1 – P(Y=A') = 1 – ($p_A + p_B$)

posons p = $p_A + p_B$, il vient :
I(X,Y) = H(Y) = -p*$\log_2$ p – (1-p)*$\log_2$ (1-p) = f(p)
cherchons le maximum de f(p), soit les valeurs pour lesquelles sa dérivée est nulle :

$$f'(p) = \log_2 \frac{1-p}{p}$$

$\forall p \in [0,1] : \log_2 \frac{1-p}{p} = 0 \Leftrightarrow \log_2 \frac{1-p}{p} = \log_2 1 \Leftrightarrow \frac{1-p}{p} = 1 \Leftrightarrow 1-p = p \Leftrightarrow p = \frac{1}{2}$

$C = \max_P(I(X,Y)) = 1$ quand p = 1/2 c'est-à-dire quand $p_A + p_B = 1/2$.

2. Canal symétrique

Définition : un canal est dit symétrique $\Leftrightarrow$ il existe une partition de l'alphabet de sortie telle que pour tout élément de cette partition, la sous-matrice de translation vérifie :

- *toutes les lignes sont identiques à des permutations près*
- *toutes les colonnes sont identiques à des permutations près*

Exemple : soit un canal de matrice de transition :

		D	E	F	G	$\leftarrow$	Y
A	(	0,1	0,5	0,1	0,3	)	
B	(	0,5	0,1	0,1	0,3	)	
C	(	0,1	0,1	0,5	0,3	)	
$\uparrow$							
X							

$\Rightarrow$ en conservant la colonne G et en permutant à la fois (A,D) avec (A,E) et (C,D) avec (C,F), on obtient des lignes/colonnes identiques, donc le canal représenté est symétrique.

Théorème : pour un canal symétrique, la capacité est atteinte pour une loi uniforme sur l'alphabet d'entrée :

$\boxed{I(X,Y) = H(X) - H(X/Y) = H(Y) - H(Y/X)}$

$\Rightarrow$ il suffit donc de calculer l'information mutuelle entre l'entrée et la sortie avec une loi uniforme sur l'alphabet d'entrée.

Exemple : calculer la capacité du canal binaire symétrique suivant :

$0 \xrightarrow{1-p} 0$
$0 \xrightarrow{p} 1$
$1 \xrightarrow{p} 0$
$1 \xrightarrow{1-p} 1$

avec p la probabilité qu'un symbole soit changé

La matrice de transition est :

$$\begin{array}{cccc} & 0 & 1 & \leftarrow Y \\ 0 & (\; 1-p & p &) \\ 1 & (\; p & 1-p &) \\ \uparrow & & & \\ X & & & \end{array}$$

Calculons I(X,Y) = H(Y) – H(Y/X) pour X tel que P(X=0) = P(X=1) = 1/2 :

P(Y=0) = P(Y=0 ∩ X=0) + P(Y=0 ∩ X=1)
= P(Y=0/X=0) * P(X=0) + P(Y=0/X=1) * P(X=1)
= (1-p) * 1/2 + p * 1/2 = 1/2
d'où P(Y=1) = 1 – P(Y=0) = 1/2
comme les évènements sont équiprobables, il vient : H(Y) = - $\log_2$ 1/2 = 1 bit

Pour calculer H(Y/X) il faut évaluer H(Y/X=0) et H(Y/X=1) ; en effet :
H(Y/X=0) = -p * $\log_2$ p – (1-p) * $\log_2$ (1-p)
H(Y/X=1) = -p * $\log_2$ p – (1-p) * $\log_2$ (1-p)

Posons $H_2(p)$ = -p * $\log_2$ p – (1-p) * $\log_2$ (1-p) ; il vient alors :
H(Y/X) = 1/2 * $H_2(p)$ + 1/2 * $H_2(p)$ = $H_2(p)$

Finalement : C(p) = 1 – $H_2(p)$; le tracé de C(p) est une parabole centrée en 1/2.

Interprétation :
- *on constate qu'il y a une symétrie par rapport à l'axe p = 1/2*
- *si p ≥ 1/2, cela signifie que partant de 0 (respectivement 1) on parvient plus souvent à 1 (respectivement 0) ; on a donc intérêt à échanger les lettres de l'alphabet de sortie*
- *si p = 0 ou 1, le canal est parfait ; le lien entre les entrées et les sorties est déterministe : la connaissance de Y permet de déterminer X*
- *si p = 1/2, C = 0 ; dans ce cas X et Y sont indépendants : la sortie n'apporte aucune information sur l'entrée*

2.1. Théorème de Shannon (2)

On considère :
- *une source S d'entropie $H_\infty(S)$ délivrant ses symboles au débit D_S*
- *un canal de capacité C utilisé au débit D_S*

Question : sous quelle condition peut-on transmettre de façon satisfaisante le contenu de la source par le biais d'un canal ?

Théorème : si le débit d'entropie H'(S) = $H_\infty(S)$ * D_S d'une source S est inférieur à la capacité par unité de temps C' = C * D_C d'un canal alors $\forall \varepsilon > 0$ il existe un code pour transmettre le contenu de la source sur le canal tel que P(erreur après décodage) < ε .

Interprétation : cela signifie que si H' < C' alors on peut transmettre le contenu de la source sur le canal avec une probabilité d'erreur aussi petite que souhaitée.

Exemple : soit une source binaire sans mémoire telle que :

- *P(S=0) = 0,98*
- *P(S=1) = 0,02*
- D_S *= 600 Kbits/s*

On dispose d'un canal symétrique de probabilité d'erreur $p = 10^{-3}$ avec $D_c = 450$ Kbits/s

1) Peut-on transmettre le contenu de S sur le canal ?

$H_\infty(S) = 0,1414$ bit

$H'(S) = H_\infty(S) * D_S = 0,1414 * 600 * 10^3$ bits/s = 84864 bits/s

$C(p) = 1 - H_2(p)$

$C(10^{-8}) = 0,9886$

$C' = 0,9886 * 450 * 10^3$ bits/s = 444870 bits/s

$\Rightarrow$ H'(S) < C' donc on peut adapter la source au canal.

Codage de canal

Exemple
On dispose d'une source binaire X = {0,1} sans mémoire telle que P(1) = q et P(0) = 1-q.
Le contenu de cette source doit être transmis par un canal symétrique de probabilité p :

$$0 \xrightarrow{1-p} 0$$
$$0 \xrightarrow{p} 1$$
$$1 \xrightarrow{p} 0$$
$$1 \xrightarrow{1-p} 1$$

Objectif : on se propose de calculer la probabilité d'erreur résultant d'une transmission directe (sans codage) et d'une transmission avec codage.

1. Transmission sans codage

On relie directement la source au canal ; soit α = {on commet une erreur} :

on a alors P(α) = P(α ; X=0) + P(α ; X=1)
or P(α ; X=0) = P(Y=1 ; X=0) = P(Y=1 / X=0)*P(X=0) = p*(1-q)
on obtient de même P(α ; X=1) = p*q
et donc :

$$\boxed{P(\alpha) = p*(1-q) + p*q = p}$$

On code les éléments binaires à transmettre en les répétant deux fois :

$$0 \rightarrow 000$$
$$1 \rightarrow 111$$

La règle de décodage consiste à choisir pour symbole émis l'élément binaire qui figure au moins deux fois dans le mot reçu constitué de trois éléments binaires :

on a toujours P(α) = P(α ; X=0) + P(α ; X=1)
or P(α ; X=0) = P(α / X=0)*P(X=0)
or P(α / X=0) = P(au moins deux « 1 » détectés / X=0)
= P(deux « 1 » détectés / X=0) + P(trois « 1 » détectés / X=0)
= $3*p^2*(1-p) + p^3$
de même P(α ; X=1) = $3*p^2*(1-p) + p^3$
d'où P(α) = $(3*p^2*(1-p) + p^3)*(1-p) + (3*p^2*(1-p) + p^3)*q = -2*p^3 + 3*p^2$

$$\boxed{P(\alpha) = -2*p^3 + 3*p^2}$$

La représentation sur un même graphe des deux P(α) en fonction de p∈ [0,1] donne :

- une droite « identité » ($P_1(\alpha) = p$)
- une courbe avec point d'inflexion en p = 1/2 ($P_2(\alpha) = -2*p^3 + 3*p^2$)

Interprétation
- pour $p \leq 1/2$, $P_1(\alpha) \geq P_2(\alpha)$: le codage diminue la probabilité d'erreur
- pour $p \geq 1/2$, $P_1(\alpha) \leq P_2(\alpha)$: le codage augmente la probabilité d'erreur

$\Rightarrow$ Ceci engendre un paradoxe qui s'explique par le fait que si $p \geq 1/2$, alors on a intérêt à permuter les lettres de l'alphabet de sortie.

2. Principe du codage

L'information transmise par le canal est I(X,Y) = H(X) – H(X/Y) = H(Y) – H(Y/X).
Supposons que l'on effectue un codage binaire optimum de X ; dans ce cas H(X) représente le nombre moyen de chiffres 0 et 1 assignés à chaque symbole source.
H(X/Y) mesure la perte d'information due au bruit ; c'est le nombre de bits qui nous manque à la réception. Ce nombre représente l'information requise pour supprimer l'ambiguïté.

Le principe du codage consiste à former des mots codes constitués d'éléments binaires d'information et d'éléments binaires de contrôle (ou de redondance).

Remarque : dans tout ce qui suit on supposera les mots codes formés de m bits d'information et k bits de contrôle :

$$\boxed{C = (c_1 \ldots c_m\ c_{m+1} \ldots c_n)} \qquad \text{avec } n = m + k$$

3. Principe du décodage

Les mots codes C sont transmis sur un canal symétrique de probabilité d'erreur $p < 1/2$. A la réception on détecte un mot Y et on se propose de décider du mot code émis.
Il s'agit d'un problème de décision statistique qui peut être résolu en appliquant le principe du maximum de vraisemblance, c'est-à-dire on choisit l'hypothèse (mot code) qui rend l'observation (mot reçu Y) la plus probable. En effet on cherche le mot code C tel que P(Y/C) soit maximum.
On doit donc étudier la fonction P(Y/C) ; soit C un mot code qui diffère de Y en un nombre l de rangs, on a $P(Y/C) = f(l) = p^l*(1-p)^{n-l}$:

soit $\ln f(l) = l*\ln p + (n-l)*\ln (1-p)$
il vient $\frac{d \ln f}{df} = \ln p - \ln(1-p) = \ln \frac{p}{1-p}$
comme $p < 1/2$ on a $\frac{p}{1-p} < 1 \Rightarrow \frac{d \ln f}{df} < 0$
donc ln f est décroissante ; f est aussi décroissante
d'où maximiser f revient à trouver l le plus petit possible

$\Rightarrow$ Finalement choisir un mot code C tel que P(Y/C) soit maximum équivaut à chercher C tel que Y diffère d'un nombre minimum de rangs.

4. Notions

Soit A un alphabet de C un code sur A.

Définitions
|x| est la longueur du mot $x = (x_1,x_2,\ldots,x_n) \in C$ avec $x_i \in A$ le nombre de symboles dans x
A^n est l'ensemble de tous les mots de longueur n qui peuvent être construits à l'aide de A
poids de Hamming : noté w(x), c'est le nombre de bits à 1 d'un mot $x \in C$
distance de Hamming : notée $d_H(x,y)$, c'est le nombre total d'indice i tel que $x_i \neq y_i$ (ex : $d_H(1011,1110) = 2$)
poids minimum : c'est le poids du mot le plus léger que l'on puisse trouver mis à part (00…0)
distance minimum : $d_{min}(C) = \min\{d_H(x,y) ; x \neq y\}$

4.1. Détection et correction d'erreur

Théorème : un code C de longueur n sur un alphabet A permet de détecter t erreurs ssi

$$\boxed{d_{min}(C) > t}$$

Théorème : un code C de longueur n sur un alphabet A permet de corriger t erreurs ssi

$$\boxed{d_{min}(C) > 2t}$$

4.2. Décodage par distance de Hamming

Soient C un code de longueur n, M = Card(C), x^i un des mots M émis du code et y un mot reçu.

Algorithme

1. *s'il existe i tel que $y = x^i$ avec $i \in \{1,2,\ldots,M\}$ alors xi est émis*
2. *si pour tout $i \in \{1,2,\ldots,M\}$ $y \neq x^i$ on cherche le mot $z \in C$ le plus proche et y c'est-à-dire $d_H(y,z) \leq d_H(y,x^i)$ pour tout $i \in \{1,2,\ldots,M\}$:*
    - *s'il existe un seul z alors l'erreur est corrigée et z est émis*
    - *s'il existe plusieurs z l'erreur est détectée mais pas corrigée*

Codes linéaires

Définition : un code linéaire C(n,k) est un sous-espace vectoriel de $\{0,1\}^n$ de dimension k ; le nombre d'éléments (mots) différents d'un tel code est $M - 2^k$.

Propriété : la distance minimale d'un code linéaire est le poids minimum du code c'est-à-dire $d_{min}(C) = w_{min}(C)$.

1. Matrice génératrice

Définition : une matrice génératrice G d'un code linéaire C(n,k) est une matrice k*n sur A={0,1} dont les lignes forment une base de C(n,k) et rang(G) = k (rang : nombre de lignes indépendantes).

Propriété : toute matrice k*n sur {0,1} de rang k est une matrice génératrice de C(n,k).

2. Codage par les codes linéaires

Soit G une matrice génératrice d'un code linéaire C(n,k).

Définition : le code C(n,k) est l'ensemble des mots de la forme C = i*G avec $i = \{i_1,i_2,\ldots,i_k\}$ de $\{0,1\}^k$.

Définition : deux codes seront dits équivalents si l'un d'eux est obtenu à partir de l'autre en appliquant sur chaque mot une même permutation des composantes.

Propriété : toute matrice G' obtenue à partir de G au moyen de combinaisons des opérations suivantes donne un code équivalent :

- *permutation de lignes/colonnes*
- *multiplication d'une ligne/colonne par un scalaire*
- *substitution d'une ligne/colonne par la somme de celle-ci et d'une ligne/colonne parallèle*

3. Code systématique

Définition : on dit qu'une matrice génératrice G d'un code C(n,k) est normalisée (ou canonique) si la matrice formée par les k premières colonnes est la matrice unité c'est-à-dire $G = (I_k,N_{k,n-k})$; dans ce cas on dit que le code est systématique.

4. Matrice de contrôle

Définition : une matrice H est dite de contrôle ou de test si et ssi pour x un mot code on a $H * x^t = \vec{0}$:

$$\boxed{C = Ker(H) = \left\{ x \in \{0,1\}^n / H * x^t = \vec{0} \right\}}$$

On peut construire une matrice H dans le cas où le code est systématique ; en effet si $G = (I_k, N_{k,n-k})$ alors $\boxed{H = (N^t_{n-k,k}, I_{n-k})}$

5. Décodage des codes linéaires

Pour contrôler l'appartenance au code d'un mot reçu y, il suffit d'effectuer le produit $S = H * y^t$:

- *si S = 0 alors y appartient au code*
- *si S ≠ 0 alors le mot reçu a subi des modifications*

Définition : on appelle syndrome le vecteur $S = H * y^t$ avec y un mot reçu ; l'ordre de S est r, soit r = n-k.

Propriété : soient C(n,k) un code linéaire et x un mot code. Supposons que pendant la transmission certains bits de x soient accidentellement complémentés. Dans ce cas on peut exprimer le mot reçu y comme

$$\boxed{y = x + e}$$

Le mot erreur e contient des « 1 » à l'emplacement des bits erronés de y, les autres étant égaux à « 0 ».

Exemple : x = (11011) et y = (10001)

$$\Rightarrow\ y^t = \begin{pmatrix}1\\1\\0\\1\\1\end{pmatrix} + \begin{pmatrix}0\\1\\0\\1\\0\end{pmatrix} = x + e$$

Puisque $H * x^t = \vec{0}$ (car x appartient à C(n,k))
il vient $S = H * y^t = H(x^t + e^t) = H * x^t + H * e^t \ \Rightarrow\ S = H * e^t$

Notons $h_1 h_2 \ldots h_n$ les colonnes de H et $e = (e_1, e_2, \ldots, e_n)$
alors $S = H * e^t \Rightarrow S = h_1 * e_1 + h_2 * e_2 + \ldots + h_n * e_n$
Il apparaît donc que le syndrome S est la somme des colonnes de H correspondant aux bits erronés. Nous pouvons alors exploiter ce résultat dans le cas où l'on est sûr que le mot reçu ne contient pas plus d'une erreur.

Théorème : s'il y a au plus une erreur dans le mot reçu alors le syndrome S recopiera la colonne de H qui correspond à l'endroit du bit erroné.

Exercice : examen 2002

On considère la matrice H de vérification du code de Hamming C(7,4) :

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

1) Trouver les équations exprimant les bits de contrôle en fonction des bits d'information.

Les colonnes 1,2,4 forment la matrice identité I ; ce sont les bits correcteurs :

$$x = (c_1 c_2 e_1 c_3 e_2 e_3 e_4) \Rightarrow x^t = \begin{pmatrix} c_1 \\ c_2 \\ e_1 \\ c_3 \\ e_2 \\ e_3 \\ e_4 \end{pmatrix}$$

$\Rightarrow$ il faut trouver la relation entre c_1,c_2,c_3 et e_1,e_2,e_3,e_4.

$$H * x^t = \vec{0} \Rightarrow \begin{pmatrix} c_3 + e_2 + e_3 + e_4 \\ c_2 + e_1 + e_2 + e_3 \\ c_1 + e_1 + e_2 + e_4 \end{pmatrix} = 0 \Rightarrow \begin{cases} c_3 = e_2 + e_3 + e_4 \\ c_2 = e_1 + e_2 + e_3 \\ c_1 = e_1 + e_2 + e_4 \end{cases}$$

2) Déduire la matrice génératrice sous la forme systématique

On trouve la matrice G grâce à l'équation : $(e_1e_2e_3e_4c_1c_2c_3) = (e_1e_2e_3e_4) * G$
Donc les équations de c_1,c_2,c_3 en fonction de e_1,e_2,e_3,e_4 (ex : $c_1 = 1*e_1 + 1*e_2 + 0*e_3 + 1*e_4$) pour chaque colonne 5,6,7 ; les autres étant une matrice identité 4*4.

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

3) Décoder le mot 0100110 1100110 1101110 1100111

$$S = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix} * \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

or $\begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$ peut correspondre à h_1, $h_2 + h_3$, $h_4 + h_5$:

- *on calcule le mot pour chaque cas*
- *le mot code est celui dont la distance de Hamming par rapport au mot original est la plus faible*