.

COURS ELE112
BASES DE COMMUNICATIONS NUMERIQUES

Didier LE RUYET

Département Electronique Automatique et Systèmes (EASY)
CNAM Paris

didier.le_ruyet@cnam.fr

PROGRAMME

- Cours 1 : Introduction, paradigme de Shannon, rappels de probabilités, notion d'entropie, Information mutuelle
- Cours 2 : théorème fondamental du codage de source, inégalité de Kraft, codage d'Huffman, débit d'entropie, codage LZ
- Cours 3 : quantification, loi A et mu, codage pour source analogique
- Cours 4 : Codes en ligne, NRZ
- Cours 5 : filtrage adapté,
- Cours 6 calcul du taux d'erreurs
- Cours 7 : critère de Nyquist

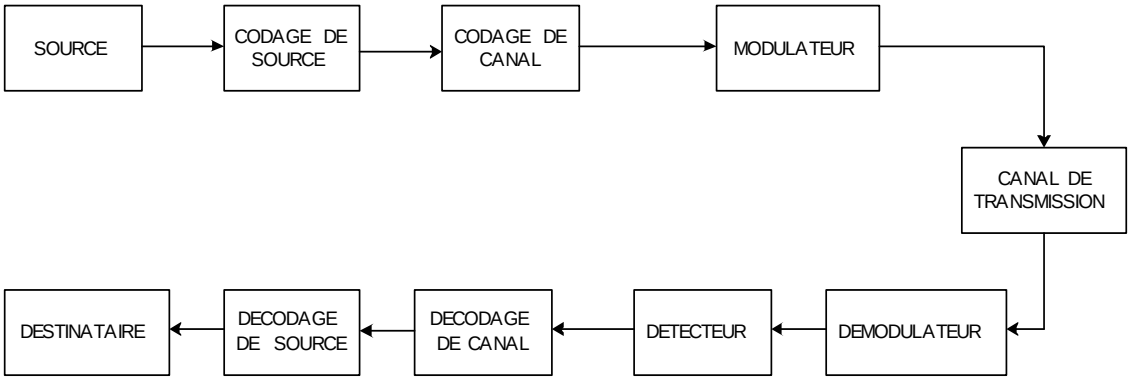
PROGRAMME

- Cours 8 : capacité d’un canal de transmission
- Cours 9 : introduction codes en bloc
- Cours 10 : Décodage des codes en bloc (méthode du syndrome, Viterbi, ...)
- Cours 11 : Critère MAP/ML décodage à entrées pondérées
- Cours 12 : Codes cycliques , BCH, RS
- Cours 13 : codes convolutifs
- Cours 14 : introduction aux modulations numériques

COURS 1

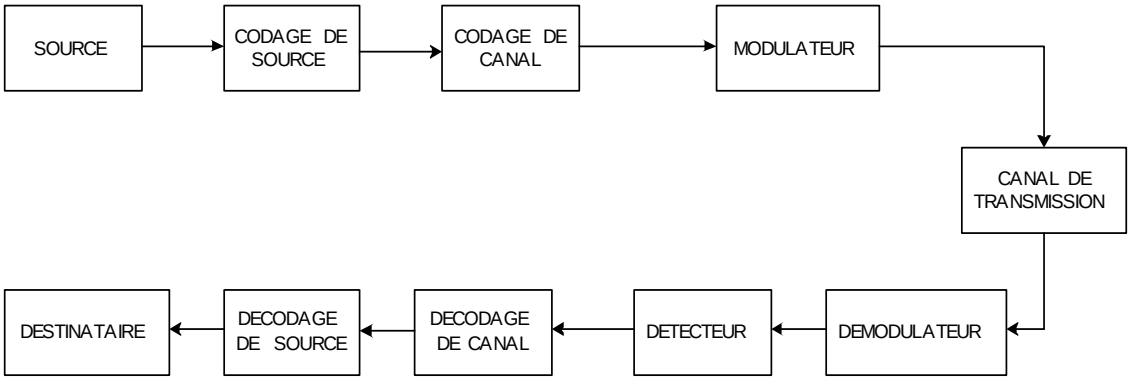
Introduction, paradigme de Shannon, rappels de probabilités notion d’entropie, Information mutuelle

PARADIGME DE SHANNON



- L'objectif du codeur de source est de représenter le message avec le moins de bits possibles. Pour ce faire, il cherche à éliminer toute la redondance contenue dans le message de la source.
- Le rôle du codage de canal est de protéger le message des perturbations du canal de transmission en ajoutant de la redondance au message compressé.
- La modulation consiste à effectuer un codage dans l'espace euclidien. Pour une modulation M-aire, on associe à chaque mot de g bits un signal $\phi_i(t), i = 1, \ldots, M$ de durée T choisi parmi les $M = 2^g$ signaux.

PARADIGME DE SHANNON



- Le rôle du démodulateur est d'extraire les échantillons tout en maximisant le rapport signal à bruit
- L'objectif du détecteur est de décider en faveur des symboles les plus probablement émis
- La qualité d'un système de transmission est évaluée en calculant ou en mesurant la probabilité d'erreurs par bit d'information (ou par bloc d'information).

RAPPELS DE PROBABILITES

Soit une variable aléatoire X ayant pour alphabet ou espace de réalisations $A_X = \{x_1, x_2, \ldots, x_n\}$ avec les probabilités respectives $P_X = \{p_1, p_2, \ldots, p_n\}$:

$$P(X = x_i) = p_i \quad p_i \geq 0 \quad \text{et} \quad \sum_{x_i \in A_X} p_i = 1 \tag{1}$$

Probabilité conjointe

Soit deux variables aléatoires X et Y ayant pour espace de réalisations respectif $A_X = \{x_1, x_2, \ldots, x_n\}$ et $A_Y = \{y_1, y_2, \ldots, y_m\}$

On appelle $P(X = x_i, Y = y_j)$ la probabilité conjointe des évènements $X = x_i$ et $Y = y_j$. On a bien entendu :

$$\sum_{x_i \in A_x} \sum_{y_j \in A_y} P(X = x_i, Y = y_j) = 1 \tag{2}$$

Probabilité marginale

Il est possible d'obtenir la probabilité $P(X = x_i)$ à partir de la probabilité conjointe

$P(X = x_i, Y = y_j)$:

$$P(X = x_i) = \sum_{y_j \in A_y} P(X = x_i, Y = y_j) \tag{3}$$

Probabilité conditionnelle

$$P(X = x_i | Y = y_j) = \frac{P(X = x_i, Y = y_j)}{P(Y = y_j)} \tag{4}$$

De la même manière on a :

$$P(Y = y_j | X = x_i) = \frac{P(X = x_i, Y = y_j)}{P(X = x_i)} \tag{5}$$

Ainsi on a la relation

$$\begin{aligned} P(Y = y_j, X = x_i) &= P(X = x_i | Y = y_j) P(Y = y_j) \\ &= P(Y = y_j | X = x_i) P(X = x_i) \end{aligned} \tag{6}$$

Loi de Bayes

$$
\begin{aligned}
P(Y = y_j | X = x_i) &= \frac{P(X = x_i | Y = y_j) P(Y = y_j)}{P(X = x_i)} \\
&= \frac{P(X = x_i | Y = y_j) P(Y = y_j)}{\sum_{y_k \in A_y} P(X = x_i, Y = y_k)} \\
&= \frac{P(X = x_i | Y = y_j) P(Y = y_j)}{\sum_{y_k \in A_y} P(X = x_i | Y = y_k) P(Y = y_k)}
\end{aligned}
$$

$P(X = x_i | Y = y_j)$ est appelée la probabilité a posteriori
$P(Y = y_i)$ est appelée la probabilité a priori.

Indépendance

L'indépendance de deux variables aléatoires X et Y implique

$$P(X, Y) = P(X) P(Y) \tag{7}$$

et

$$P(X | Y) = P(X) \tag{8}$$

UNE MESURE LOGARITHMIQUE DE L'INFORMATION

$h(x_i)$ la mesure de l'information associée à l'événement $X = x_i$ doit satisfaire :

- $h(x_i)$ doit être continue pour $p(X = x_i)$ compris entre 0 et 1
- $h(x_i) = \infty$ si $P(X = x_i) = 0$
- $h(x_i) = 0$ si $P(X = x_i) = 1$ un événement certain n'apporte pas d'information
- $h(x_i) > h(y_j)$ si $P(Y = y_j) > P(X = x_i)$
- $h(x_i) + h(y_j) = h(x_i, y_j)$. La réalisation de 2 évènements indépendants $Y = y_j$ et $X = x_i$ apporte une quantité d'information égale à la somme des informations de ces 2 évènements $h(x_i)$ et $h(y_j)$.

La seule expression de la quantité d'information $h(x_i)$ associée à la réalisation de l'évènement $X = x_i$ satisfaisant les propriétés énumérées ci-dessus est la suivante:

$$h(x_i) = \log_2 \frac{1}{P(X = x_i)} = -\log_2 P(X = x_i) = -\log_2 p_i \tag{9}$$

Exemple : soit une source discrète produisant des bits (0 ou 1) avec la probabilité $\frac{1}{2}$. On a :

$$h(0) = h(1) = -\log_2 \frac{1}{2} = 1 \text{ Sh} \tag{10}$$

Considérons maintenant la réalisation de 2 évènements $X = x_i$ et $Y = x_j$. La quantité d'information associée est :

$$h(x_i, y_j) = \log_2 \frac{1}{P(X = x_i, Y = y_j)} = -\log_2 P(X = x_i, Y = y_j) \tag{11}$$

où $P(X = x_i, Y = y_j)$ est la probabilité conjointe des deux évènements.

La quantité d'information associée à la réalisation de l'événément $X = x_i$ conditionnellement à l'évènement $Y = y_j$ est la suivante :

$$h(x_i|y_j) = \log_2 \frac{1}{P(X = x_i|Y = y_j)} = -\log_2 P(X = x_i|Y = y_j) \tag{12}$$

On en déduit :

$$h(x_i, y_j) = h(x_i|y_j) + h(y_j) = h(y_j|x_i) + h(x_i) \tag{13}$$

INFORMATION MUTUELLE

On définit l'information mutuelle $i(x_i, y_j)$ comme la quantité d'information que la réalisation de l'évènement $Y = y_j$ apporte sur l'évènement $X = x_i$:

$$i(x_i, y_j) = \log \frac{P(X = x_i|Y = y_j)}{P(X = x_i)} \tag{14}$$

Si les deux événements sont indépendants, alors $P(X = x_i|Y = y_j) = P(X = x_i)$ et donc $i(x_i, y_j) = 0$.

Si l'événement $X = x_i$ est équivalent à l'événement $Y = y_j$, alors $P(X = x_i|Y = y_j) = 1$ et $i(x_i, y_j) = h(x_i)$.

L'information que la réalisation de l'évènement $Y = y_j$ apporte sur l'évènement $X = x_i$ est identique à l'information que la réalisation de l'évènement $X = x_i$ apporte sur l'évènement $Y = y_j$.

$$i(x_i, y_j) = i(y_j, x_i) \tag{15}$$

On a également les relations suivantes :

$$\begin{aligned} i(x_i, y_j) &= i(y_j, x_i) \\ &= h(x_i) - h(x_i|y_j) \\ &= h(x_i) + h(y_j) - h(x_i, y_j) \\ &= h(y_j) - h(y_j|x_i) \end{aligned}$$

Contrairement à $h(x_i)$, l'information mutuelle $i(x_i, y_j)$ peut être négative.

ENTROPIE ET INFORMATION MUTUELLE MOYENNE

Déterminons l'entropie d'une source décrite par la variable aléatoire X ayant pour espace de réalisation $A_X = \{x_1, x_2, \ldots, x_n\}$ avec les probabilités respectives $P_X = \{p_1, p_2, \ldots, p_n\}$. La quantité d'information moyenne ou entropie de la source est la moyenne des informations relatives à chaque réalisation de l'évènement $X = x_i$:

$$\begin{aligned} H(X) &= \sum_{i=1}^{n} p_i h(x_i) \\ &= \sum_{i=1}^{n} p_i \log_2 \frac{1}{p_i} \\ &= -\sum_{i=1}^{n} p_i \log_2 p_i \quad \text{en Sh/symbole} \end{aligned} \tag{16}$$

$H(X)$ mesure l'incertitude sur X.

Propriétés :

$$H(X) \geq 0 \quad \text{avec égalité si } p_i = 1 \text{ pour une valeur de } i \tag{17}$$

$$H(X) \leq \log_2 n \tag{18}$$

Si les probabilités $P(X = x_i) = p_i$ sont toutes égales à $\frac{1}{n}$, alors on a :

$$H(X) = \log_2 n \tag{19}$$

L'entropie est donc maximale lorsque toutes les probabilités p_i sont égales.

Exemple : soit une source à 2 états x_0 et x_1 avec $p_0 = p$ et $p_1 = 1 - p$
L'entropie de cette source est la suivante :

$$H(X) \equiv H(p, 1-p) = -p \log_2 p - (1-p) \log_2 (1-p) \tag{20}$$

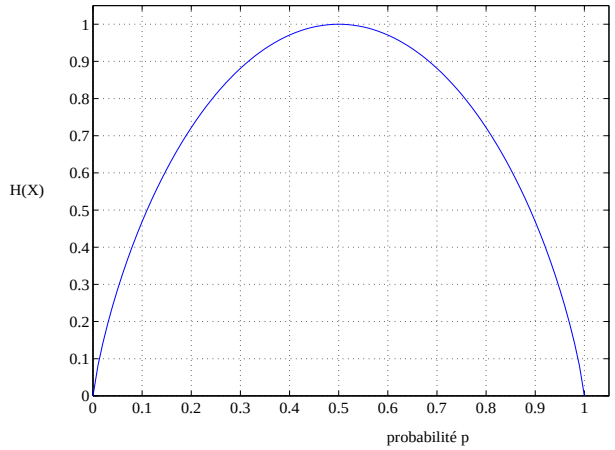


Figure 1: Entropie d'une source binaire

Ainsi, l'entropie de la source binaire est maximale (1 Sh/bit) lorsque $p = 0.5$

• Considérons deux variables aléatoires X et Y ayant respectivement pour espace de réalisations $A_X = \{x_1, x_2, \ldots, x_n\}$ et $A_Y = \{y_1, y_2, \ldots, y_m\}$.

• L'entropie conjointe $H(X, Y)$ est définie comme suit :

$$\begin{aligned} H(X,Y) &= \sum_{i=1}^{n}\sum_{j=1}^{m} P(X = x_i, Y = y_j) h(x_i, y_j) \\ &= -\sum_{i=1}^{n}\sum_{j=1}^{m} P(X = x_i, Y = y_j) \log_2 P(X = x_i, Y = y_j) \end{aligned} \tag{21}$$

• Si les variables aléatoires X et Y sont indépendantes alors $H(X, Y) = H(X)+H(Y)$.

• On peut aussi déterminer l'entropie conditionnelle $H(X/Y)$ qui détermine l'information sur X sachant l'observation Y à partir de $h(x_i|y_j)$:

$$\begin{aligned} H(X|Y) &= \sum_{i=1}^{n}\sum_{j=1}^{m} P(X = x_i, Y = y_j) h(x_i|y_j) \\ &= -\sum_{i=1}^{n}\sum_{j=1}^{m} P(X = x_i, Y = y_j) \log_2 P(X = x_i|Y = y_j) \end{aligned} \tag{22}$$

• Ces différentes relations permettent d'exprimer l'entropie conjointe en fonction de l'entropie et l'entropie conditionnelle :

$$H(X,Y) = H(X) + H(Y|X) = H(Y) + H(X|Y) \tag{23}$$

• L'incertitude sur X et Y est égale à l'incertitude sur X plus l'incertitude sur Y sachant X.

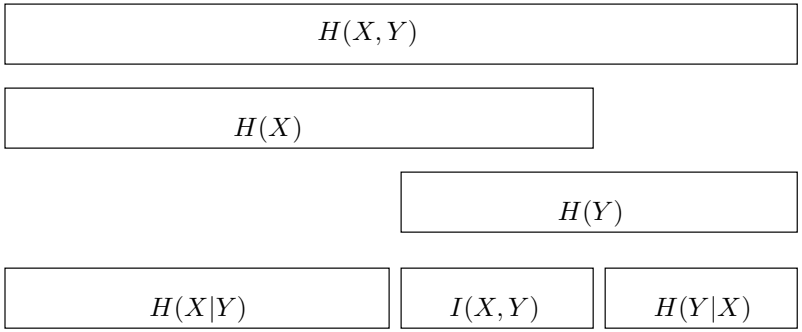
• L'information mutuelle associée à la réalisation d'un événement peut également être étendue aux variables aléatoires X et Y.

$$\begin{aligned} I(X,Y) &= \sum_{i=1}^{n}\sum_{j=1}^{m} P(X=x_i, Y=y_j) i(x_i,y_j) \\ &= \sum_{i=1}^{n}\sum_{j=1}^{m} P(X=x_i, Y=y_j) \log_2 \frac{P(X=x_i|Y=y_j)}{P(X=x_i)} \\ &= \sum_{i=1}^{n}\sum_{j=1}^{m} P(X=x_i, Y=y_j) \log_2 \frac{P(X=x_i, Y=y_j)}{P(X=x_i)P(Y=y_j)} \end{aligned} \tag{24}$$

Ainsi, on a les relations suivantes :

$$I(X,Y) = H(X) + H(Y) - H(X,Y) = H(X) - H(X|Y) = H(Y) - H(Y|X) \quad (25)$$

• L'information mutuelle $I(X,Y)$ mesure la réduction d'incertitude moyenne sur X qui résulte de la connaissance de Y.



COURS 2

théorème fondamental du codage de source, inégalité de Kraft, codage d'Huffman, débit d'entropie, codage LZ

CODAGE DE SOURCE

• l'opération de codage de source consiste à associer à chaque *message* issu de la source un *mot* composé d'un ou plusieurs symboles q-aire en cherchant à réduire au maximum le nombre moyen de ces symboles.

• Un *message* signifiera selon le contexte, soit un symbole Q-aire issu de la source, soit un ensemble de J symboles Q-aire.

• Nous nous restreindrons au cas où les symboles de sortie du codeur de source sont des bits ($q = 2$).

CODAGE DE SOURCE

- Le codage de source doit satisfaire les deux critères suivants :
- **décodage unique** : chaque message devra être codé par un mot différent
- **déchiffrabilité** : chaque mot devra pouvoir être dissocié sans ambiguité. Ce critère s'obtient par :
  - codage par mot de longueur fixe
  - codage avec un symbole de séparation distinguable (exemple : système morse)
  - codage par mot de longueur variable mais en évitant qu'un mot ne soit le début d'un autre.
- Un code est dit *instantané* si aucun mot code n'est le début d'un autre.

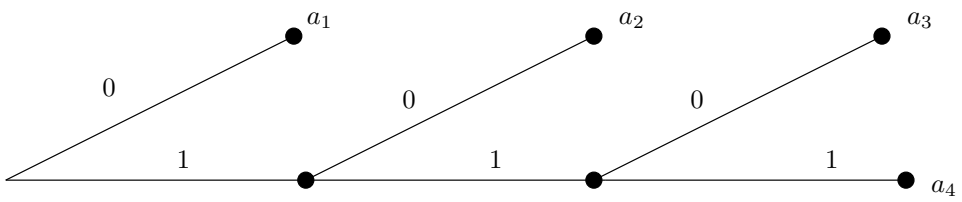
CODAGE PAR MOT DE LONGUEUR VARIABLE

exemple 1 :

message	**mot**
a_1	1
a_2	00
a_3	01
a_4	10

exemple 2 :

message	**mot**
a_1	0
a_2	10
a_3	110
a_4	111



INEGALITE DE KRAFT

théorème : un code instantané composé de M mots binaires de longueur respective $\{n_1, n_2, \ldots, n_M\}$ avec $n_1 \leq n_2 \leq \cdots \leq n_M$ doit satisfaire l'inégalité suivante :

$$\sum_{i=1}^{M} 2^{-n_i} \leq 1 \tag{26}$$

démonstration

Un code instantané peut être représenté graphiquement par un arbre binaire complet de profondeur M.

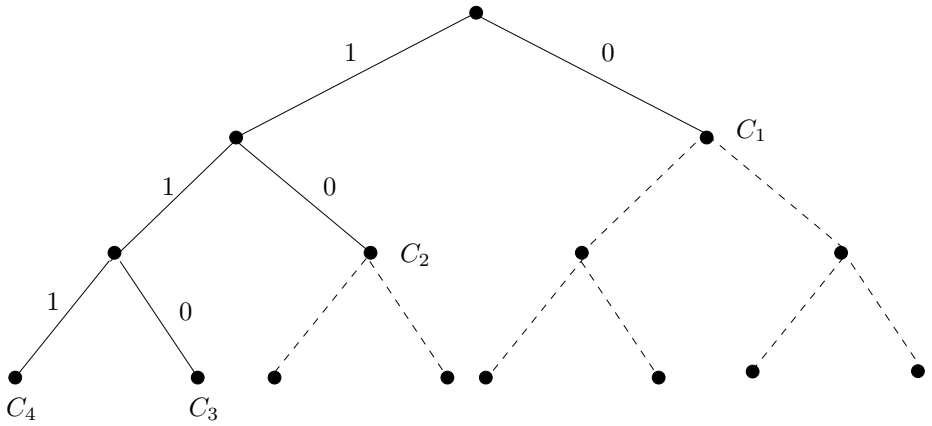


Figure 2: inégalité de Kraft

Le choix d'un noeud de degré n_1 comme premier mot C_1 élimine $2^{n_M - n_1}$ noeuds.

La condition pour obtenir un codage instantané devient donc :

$$\sum_{i=1}^{M} 2^{n_M - n_i} \leq 2^{n_M}$$

en divisant les deux termes par 2^{n_M}, on obtient bien l'inégalité de Kraft :
:

$$\sum_{i=1}^{M} 2^{-n_i} \leq 1 \tag{27}$$

THEOREME DU CODAGE DE SOURCE

• Soit une source sans mémoire d'entropie par symbole $H(X)$. Il est possible de construire un code instantané dont la longueur moyenne des mots R_{moy} satisfait l'inégalité suivante :

$$H(X) \leq R_{moy} < H(X) + 1 \tag{28}$$

démonstration

Soit $n_i = \lceil -\log_2 p_i \rceil$ longueur du mot associé au i-ième message.

$$\sum_{i=1}^{M} 2^{-n_i} = \sum_{i=1}^{M} 2^{-\lceil -\log_2 p_i \rceil} \leq \sum_{i=1}^{M} 2^{\log_2 p_i} = \sum_{i=1}^{M} p_i = 1 \tag{29}$$

Ainsi on a vérifié que ce code satisfait l'inégalité de Kraft.

On a alors :

$$R_{moy} = \sum_{i=1}^{M} p_i \lceil -\log_2 p_i \rceil < \sum_{i=1}^{M} p_i(-\log_2 p_i + 1) = H(X) + 1 \tag{30}$$

THEOREME DU CODAGE DE SOURCE

Théorème : pour toute source stationnaire d'entropie par symbole $H(X)$, il existe un procédé de codage de source binaire dont la longueur moyenne R_{moy} des mots est aussi voisine que l'on veut de $H(X)$.

$$H(X) \leq R_{moy} < H(X) + \epsilon \tag{31}$$

Groupons les symboles d'une source sans mémoire d'entropie par symbole $H(X)$ par paquet de J symboles : on obtient une nouvelle source. En utilisant le théorème précédent, on obtient:

$$JH(X) \leq R_{Jmoy} < JH(X) + 1 \tag{32}$$

En divisant les différents termes par J on obtient :

$$H(X) \leq R_{moy} < H(X) + \frac{1}{J} \tag{33}$$

avec R_{moy} nombre de bits moyens relatifs à un symbole de la source

Ce résultat se généralise immédiatement au cas des sources avec mémoire.

DEBIT D'ENTROPIE

Soit une source discrète et stationnaire X d'entropie $H(X)$ Sh/symbole délivrant D_S symboles par seconde.

débit d'entropie ou débit informationnel moyen D_I :

$$D_I = H(X)D_S \qquad \text{en Shannon/seconde} \tag{34}$$

l'entropie par bit en sortie du codeur de source binaire,

$$H'(X) = \frac{H(X)}{R_{moy}} \tag{35}$$

le débit binaire D'_B en sortie du codeur

$$D'_B = D_S.R_{moy} \tag{36}$$

le débit d'entropie D'_I en sortie du codeur

$$D'_I = H'(X).D'_B = D_I \quad \text{en Shannon/seconde} \tag{37}$$

En multipliant les 2 termes par D_S, on obtient :

$$D'_B \geq D_S.H(X) = D_I \tag{38}$$

ALGORITHME D’HUFFMAN

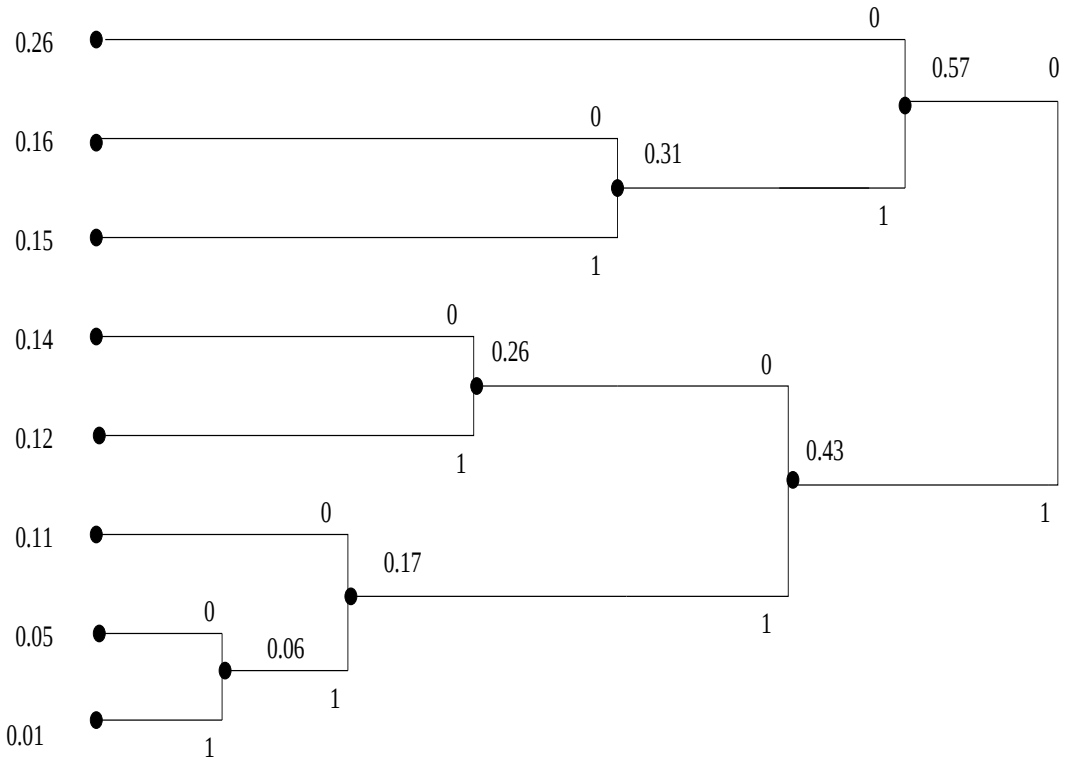
- On commence par classer la liste des messages de haut en bas par ordre de probabilité décroissante (chaque message correspond à un nœud).

- 1. Choix des deux messages de probabilités moindres.
- 2. Ces deux probabilités sont reliées avec la branche supérieure labelisée à 0 et la branche inférieure labelisée à 1.
- 3. La somme de ces deux probabilités est alors associée au nouveau nœud.
- 4. Suppression des deux messages précédemment choisis puis retour à la phase 1.

- On répète cette procédure jusqu’à ce qu’il ne reste plus aucun message. L’arbre ainsi obtenu décrit graphiquement l’ensemble du code. Les mots sont lus en parcourant chaque chemin de la droite vers la gauche.

exemple 3 : soit une source discrète à 8 messages $a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8$ avec les probabilités d'apparition respectives $\{0.16; 0.15; 0.01; 0.05; 0.26; 0.11; 0.14; 0.12\}$ d'entropie $H(X)$=2.7358.

L'arbre obtenu par codage de Huffman est le suivant :



La table de codage est la suivante :

message	mot	n_i
a_5	00	2
a_1	010	3
a_2	011	3
a_7	100	3
a_8	101	3
a_6	110	3
a_4	1110	4
a_3	1111	4

• Le nombre de bit moyen par mot est égal à 2.8.

• Pour les sources sans mémoire l'algorithme de Huffman fournit un codage de source optimal. Cependant, lorsque les symboles successifs sont corrélés, la complexité du codage de source utilisant l'algorithme de Huffman devient très grande (le nombre de code est égal à Q^J si les messages sont composés de J symboles Q-aire).

• autre codage de source : codage LZW, codage arithmétique.

ALGORITHME DE LEMPEL-ZIV

Cet algorithme, proposé en 1978 est indépendant des propriétés statistiques de la source. Le principe de l'algorithme LZ78 consiste à découper la séquence de sortie de la source en petites suites de longueurs variables.

exemple: considérons le début de séquence binaire issue d'une source :

0010000011000100100000101100010000001000

0110001010100001000001100000 1 01100000011

On commence par déterminer la suite la plus courte que nous n'avons pas encore rencontrée en commencant par la gauche

0,0100000110001001000001011000100000010000011

La seconde suite différente de 0 est 01

0,01,0000011000100100000101100010000001000011

La troisième suite est 00

0,01,00,000110001001000001011000100000010000011

Finalement, cette séquence est décomposée en suite comme suit :

0, 01, 00, 000, 1, 10, 001, 0010, 0000, 101, 100, 010, 00001, 000011

L’encodage se fait au fil de l’eau. Les suites sont décrites à partir des suites précédemment rencontrées

Nous obtenons la table suivante:

1	2	3	4	5	6	7	8	9	10	11	12	13	14
0	01	00	000	1	10	001	0010	0000	101	100	010	00001	000011
00	11	10	30	01	50	31	70	40	61	60	20	91	131

La première ligne de la table correspond à l’index des suites, la seconde aux suites et la troisième à l’encodage de ces suites.

Par exemple, la suite 0010 est encodée par 70 car elle est construite à partir de la séquence 001 (index 7) à laquelle on ajoute 0.

L’ensemble vide est encodé par 0.

Finalement la séquence encodée en binaire devient :

0 0 , 1 1 , 01 0 , 11 0 , 000 1 , 101 0 , 011 1 , 111 0 , 0100 0 , 0110 1 , 0110 0 , 0010 0 , 1001 1 , 1101 1

Afin de permettre le décodage, le nombre de bits utilisés pour encoder l’index est toujours croisant : 2 suites dont l’index est de longueur 1, puis 2 suites dont l’index est de longueur 2, puis 2^2 suites de longueur 3, 2^3 suites de longueur 4, ...

Le décodeur utilisera donc le même algorithme pour reconstruire la séquence initiale.

Le principal inconvénient de cet algorithme est que la taille du dictionnaire augmente sans limite. Une des solutions envisagées consiste à fixer une taille maximale comme dans l'algorithme de Lempel-Ziv-Welch.

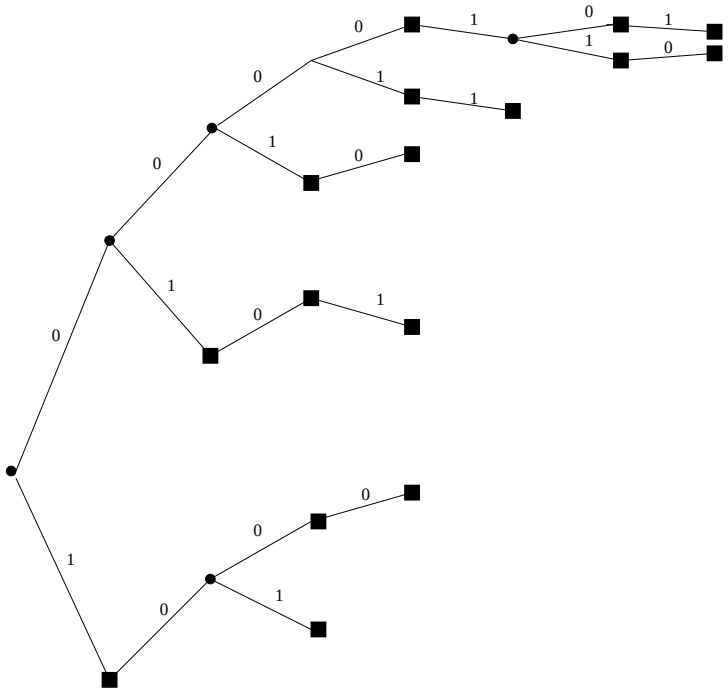
ALGORITHME DE LEMPEL-ZIV-WELCH

- Algorithme indépendant des propriétés statistiques de la source
- L'algorithme de Lempel-Ziv utilise pour coder une liste de suites stockées dans un dictionnaire.
- Principe de base : la séquence de sortie de la source est découpée en petites suites de longueurs variables. Les suites qui sont stockées dans un dictionnaire initialement vide sont appelées les suites prototypes.
- Une suite nouvelle est ajoutée dans le dictionnaire chaque fois qu'elle est différente des suites prototypes déjà stockées. De plus, cette suite à laquelle on ajoute un bit 0 ou 1 ne doit pas être déjà présente dans le dictionnaire.

EXEMPLE

00100000110001001000001011000100000100001100010101000010000011000001
01100000011

0, **01**, 00, 000, **1**, 10, **001**, **0010**, **0000**, **101**, **100**, **010**, 00001, **000011**, **0001**, **0101**, **000010**, **0000110**, **0000101**, **1000**, **00011**



EXEMPLE

- La table donne la liste des 16 suites prototypes dans cet exemple
- Chaque suite prototype est ici codée avec un mot de 4 bits.

position	suite prototype	mot de code
1	1	0000
2	01	0001
3	001	0010
4	010	0011
5	100	0100
6	101	0101
7	0000	0110
8	0001	0111
9	0010	1000
10	0101	1001
11	1000	1010
12	00011	1011
13	000010	1100
14	000011	1101
15	0000101	1110
16	0000110	1111

EXEMPLE

- Finalement, la séquence binaire issue d'une source est décomposée en utilisant les suites prototypes stockées dans le dictionnaire :

  0010, 0000110, 0010, 010, 0000101, 1000, 1000, 0010, 00011, 0001, 0101, 000010, 0000110, 0000101, 1000, 00011

- La sortie du codeur de source est la suivante :

  1000 1111 1000 0011 1110 1010 1010 1000 1011 0111 1001 1101 1111 1110 1010 1011

- Le décodeur de source utilisant le même algorithme pourra reconstruire le dictionnaire utilisé au codage et donc reconstituer instantanément la séquence émise par la source.
- Il est à noter que ici le codage Lempel-Ziv encode les 79 bits de la séquence de la source en 16 mots de 4 bits soit 64 bits au total.
- En pratique, le contenu du dictionnaire est adapté dynamiquement en fonction de l'évolution des caractéristiques de la source.
- L'algorithme Lempel-Ziv et ses variantes sont utilisés pour la compression des fichiers.

COURS 3

échantillonnage, quantification, loi A et mu, codage pour source analogique

ECHANTILLONNAGE

Soit le signal $x(t)$ à bande limitée B issu d'une source analogique.

En utilisant le théorème de l'échantillonnage, on montre que ce signal peut être représenté comme suit :

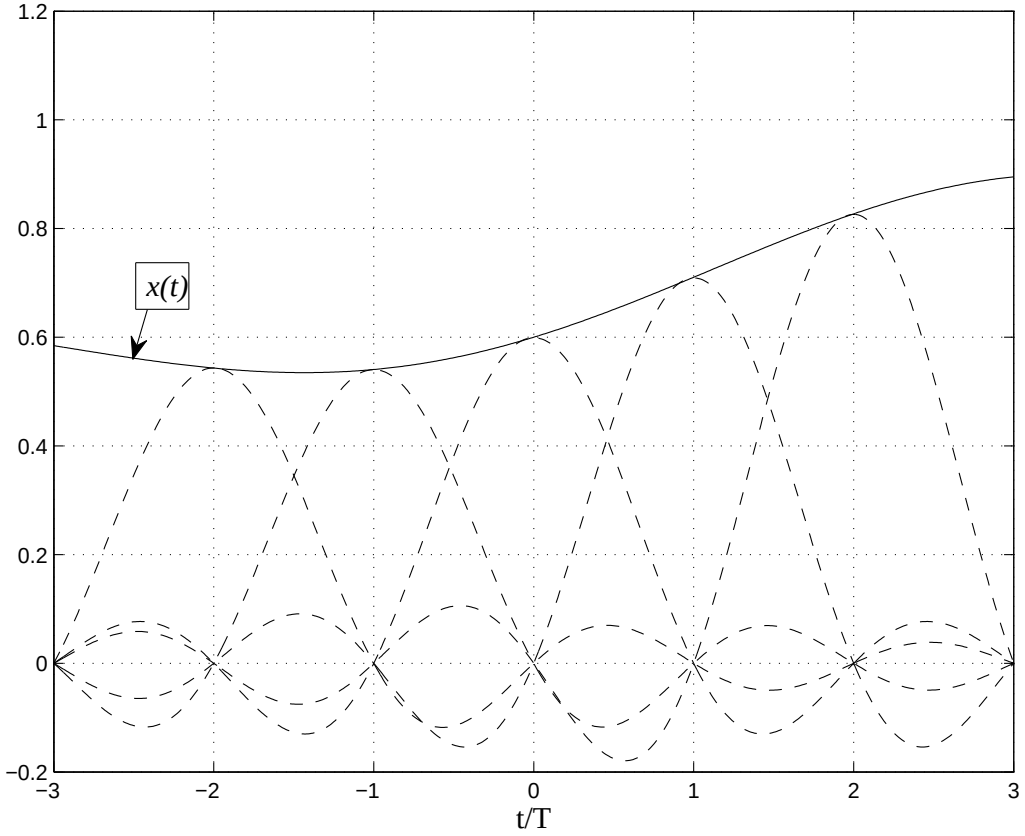
$$x(t) = \sum_{k=\infty}^{+\infty} x(kT)\text{sinc}\left(\frac{\pi}{T}(t - kT)\right) \tag{39}$$

avec $\text{sinc}(x) = \frac{sin(x)}{x}$ et $T = \frac{1}{2B}$.

La suite $x(kT)$ représente les échantillons du signal $x(t)$ aux instants $kT = \frac{k}{2B}$.

Cette suite est donc obtenue par échantillonnage de $x(t)$ à la fréquence de $2B$ échantillons par seconde.

ECHANTILLONNAGE



QUANTIFICATION

La quantification consiste à quantifier l'amplitude possible des échantillons sur L valeurs.

La valeur $\hat{x}$ choisie est la plus proche au sens de la distance euclidienne de l'amplitude x de l'échantillon.

Si L est une puissance de 2, ($L = 2^R$) alors chaque échantillon quantifié $\tilde{x}$ pourra être représenté par un mot binaire de R bits (opération de codage).

$$R = \log_2 L \qquad \text{bits/échantillon} \tag{40}$$

définition : soit un ensemble d'intervalles ou cellules $\mathcal{S} = [s_1, s_2, \ldots, s_L]$ et un ensemble de valeurs ou points $\mathcal{Y} = [y_1, y_2, \ldots, y_L]$. L'opération de quantification est définie mathématiquement par la relation suivante :

$$\tilde{x} = y_i \qquad \text{pour} \quad x \in S_i \tag{41}$$

Chaque intervalle ou cellule S_i est bornée par deux seuils notés a_{i-1} et a_i. Ainsi, la largeur de S_i est égale à $a_i - a_{i-1}$

L'opération de quantification introduit une erreur de quantification $q = \tilde{x} - x$

QUANTIFICATION UNIFORME

La quantification est dite uniforme si tous les intervalles ont la même largeur notée Δ.

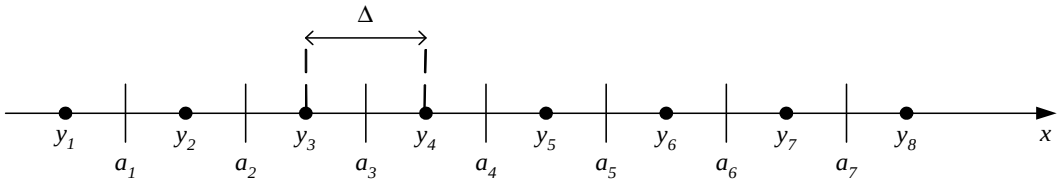


Figure 3: Quantification uniforme pour $L = 8$

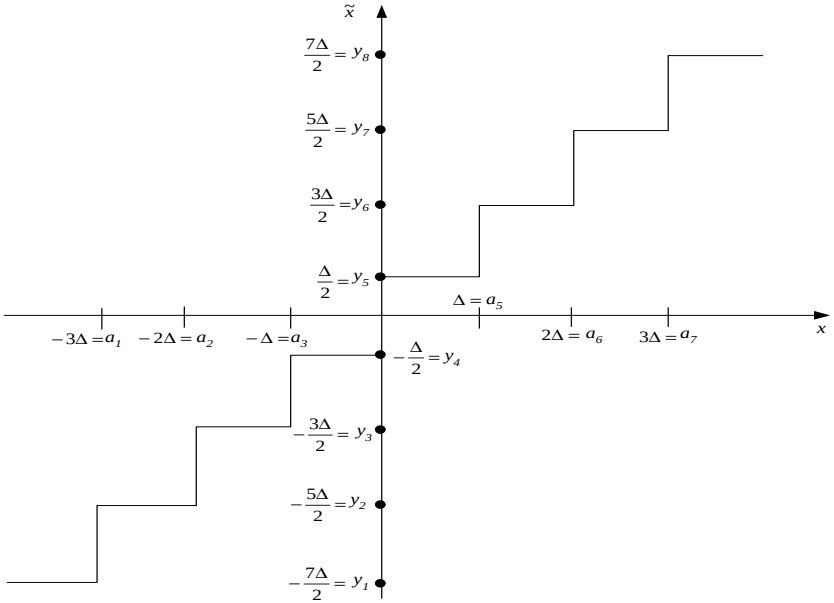


Figure 4: Quantification uniforme $L = 8$

QUANTIFICATION NON UNIFORME

La quantification est dite non uniforme si tous les intervalles n'ont la même largeur.

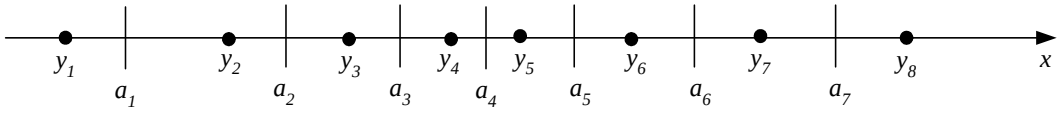


Figure 5: Quantification non uniforme pour $L = 8$

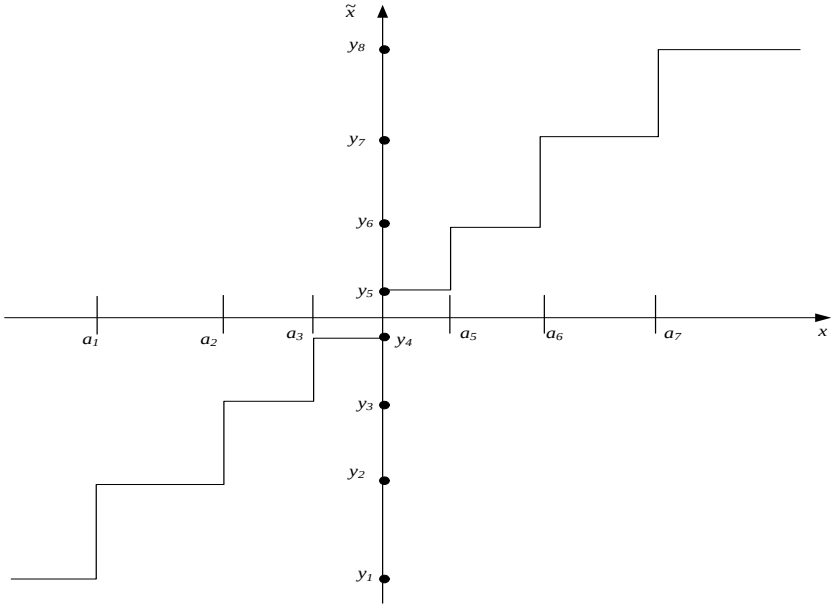


Figure 6: Quantification non uniforme $L = 8$

ERREUR QUADRATIQUE MOYENNE

La mesure la plus commune pour évaluer la différence entre le signal quantifié et le signal d'origine est l'erreur quadratique :

$$eq(x, \tilde{x}) = (x - \tilde{x})^2 \tag{42}$$

On définit l'erreur quadratique moyenne (EQM) ou distorsion moyenne D

$$\begin{aligned} D &= E(eq(x, \tilde{x})) \\ &= \int_{-\infty}^{+\infty} eq(x, \tilde{x}) f(x) dx \\ &= \sum_i \int_{S_i} eq(x, y_i) f(x) dx \end{aligned} \tag{43}$$

où $f(x)$ est la densité de probabilité de x.

Ainsi, lorsque la densité de probabilité $f(x)$ est connue, l'objectif de la quantification est de coder les échantillons de la source avec le minimum de bits en minimisant la distorsion moyenne D.

QUANTIFICATION VECTORIELLE

Si on associe à chaque échantillon un mot binaire, on réalise une quantification scalaire

Il est possible de grouper plusieurs échantillons ensemble avant de réaliser l'opération de quantification de ce groupe d'échantillons. On parlera de quantification vectorielle.

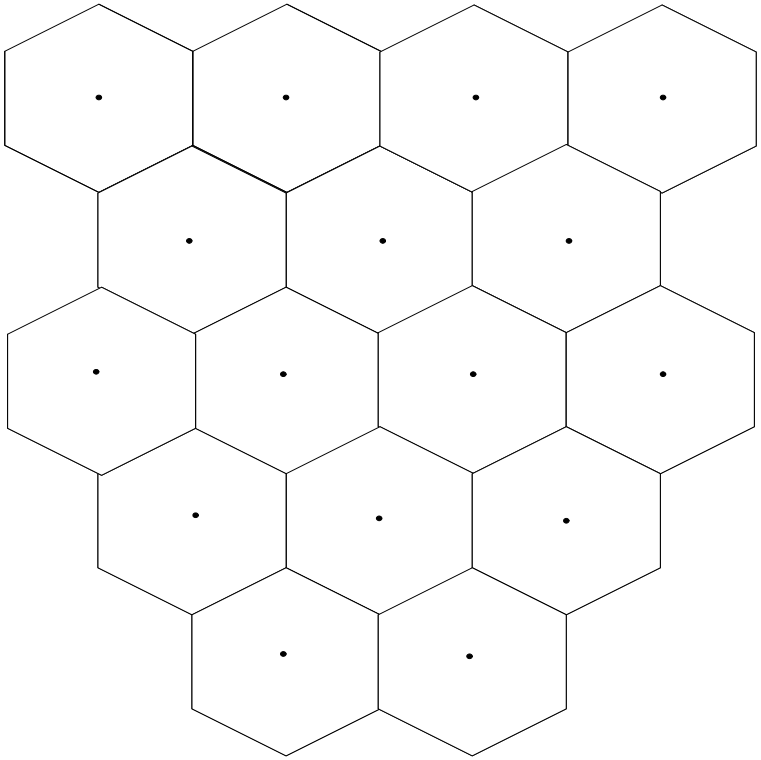


Figure 7: Region de Voronoi

QUANTIFICATION VECTORIELLE

Si on associe à chaque échantillon un mot binaire, on réalise une quantification scalaire

Il est possible de grouper plusieurs échantillons ensemble avant de réaliser l'opération de quantification de ce groupe d'échantillons. On parlera de quantification vectorielle.

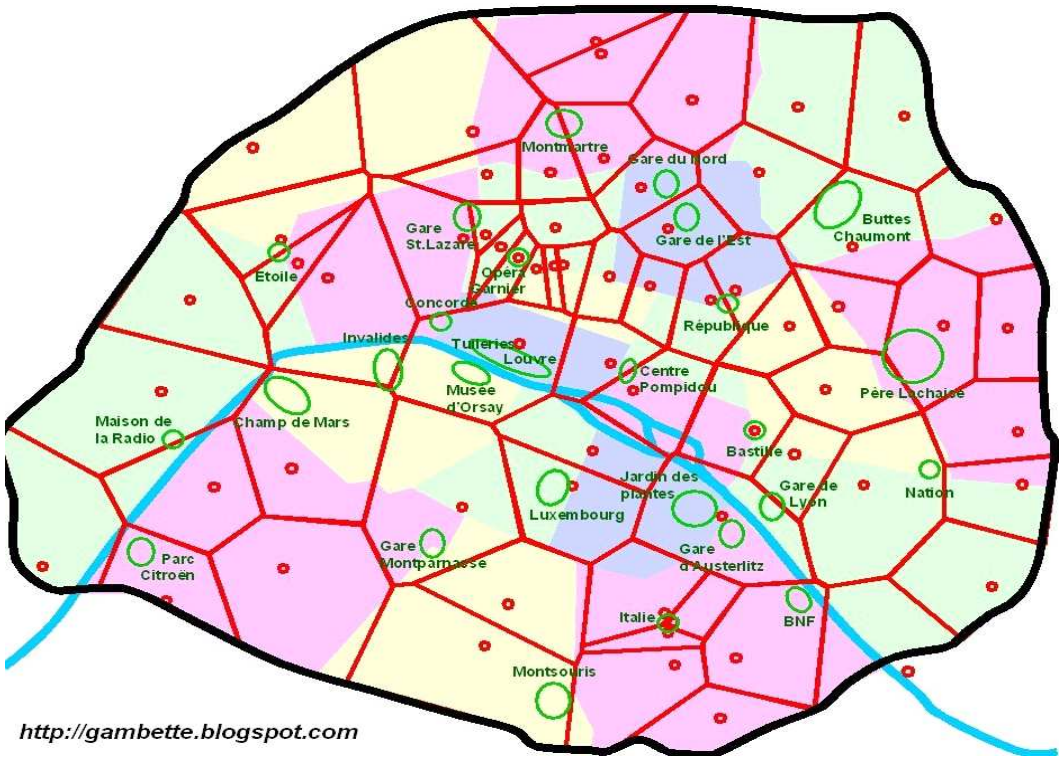


Figure 8: Quantification Vectorielle

THEOREME DU CODAGE DE SOURCE AVEC PERTES

Définition :

on définit pour une source sans mémoire la fonction taux-distorsion $R(D)$ (*rate distortion function* en anglais) comme suit :

$$R(D) = \min_{p(y|x)} \{I(X,Y) | E((X-Y)^2) \leq D\} \tag{44}$$

Théorème du codage de source avec pertes :

le nombre de bits minimum R permettant de représenter une suite d'échantillons avec une distorsion moyenne donnée D doit être supérieur ou égal à $R(D)$

$$R \geq R(D) \qquad \text{en bits} \tag{45}$$

THEOREME DU CODAGE DE SOURCE AVEC PERTES

On peut montrer que l'inégalité suivante est toujours vraie :

$$R(D) \leq \frac{1}{2} \log_2 \frac{\sigma_x^2}{D} \qquad 0 \leq D \leq \sigma_x^2 \tag{46}$$

où σ_x^2 est la variance de la source.

En introduisant la fonction distorsion-débit $D(R)$ la relation ci-dessus peut aussi s'écrire :

$$D(R) \leq \sigma_x^2 2^{-2R} \tag{47}$$

Les deux inégalités se transforment en égalités pour une distribution gaussienne.

EXEMPLE D'UNE SOURCE GAUSSIENNE

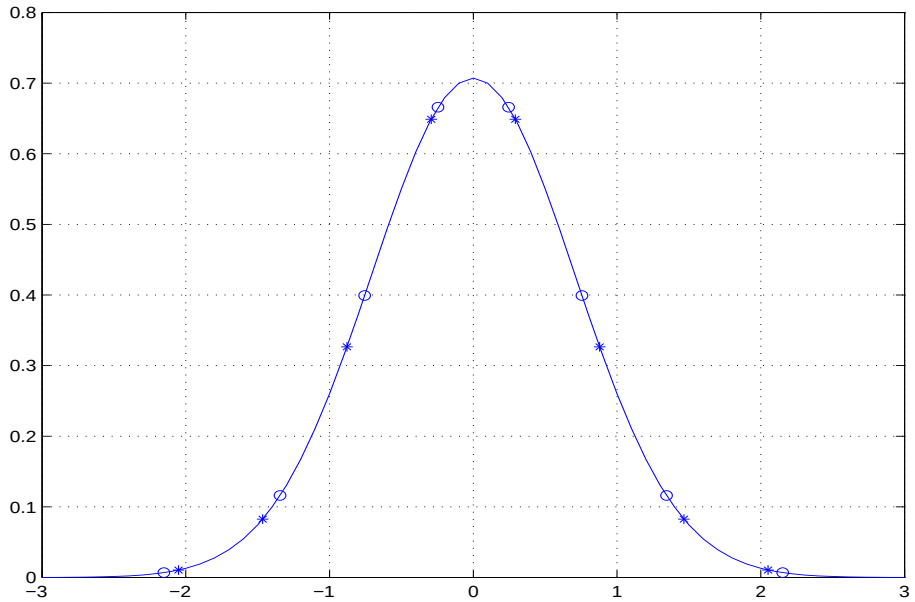


Figure 9: Quantification uniforme et non uniforme $L = 8$ pour une source gaussienne de variance $\sigma_x = 1$

Dans ce cas ($R = 3$ bits et $\sigma_x = 1$), les distorsions moyennes des quantifications uniforme et non uniforme sont respectivement égales à -14,27 dB et -14,62 dB.

La limite théorique est $10 \log_{10} 2^{-6} = -18.06$ dB

Comme les probabilités de chaque intervalle ne sont pas identiques, on peut appliquer un codage de Huffman après l'opération de quantification pour réduire encore le débit binaire. Ce codage permet de gagner ici environ 2 dB.

Pour atteindre la limite théorique, il faudra utiliser une quantification vectorielle

ERREUR DE QUANTIFICATION

L'opération de quantification introduit une erreur de quantification q entre l'amplitude x et l'amplitude quantifiée $\tilde{x}$. On a la relation suivante :

$$\tilde{x} = x + q \tag{48}$$

Pour la quantification uniforme, $q \in [-\frac{\Delta}{2}, +\frac{\Delta}{2}]$

En considérant que l'amplitude du signal est suffisamment grande devant Δ, la densité de probabilité de q est approximativement uniforme.

$$p(q) = \frac{1}{\Delta} \qquad -\frac{\Delta}{2} \leq q \leq \frac{\Delta}{2} \tag{49}$$

Calculons l'erreur quadratique moyenne $E(q^2)$:

$$\begin{aligned} E(q^2) &= E((eq(x, \tilde{x})) \\ &= \int_{-\infty}^{+\infty} q^2 p(q) dq \\ &= \frac{1}{\Delta} \int_{-\frac{\Delta}{2}}^{\frac{\Delta}{2}} q^2 dq = \frac{\Delta^2}{12} \end{aligned} \tag{50}$$

ERREUR DE QUANTIFICATION

Si la quantification uniforme est réalisée sur L niveaux avec $R = \log_2 L$ et que la dynamique du signal issu de la source est égal à A avec $A = \Delta L = \Delta 2^R$, alors le rapport signal à bruit en décibel s'exprime comme suit :

$$\begin{aligned} SNR &= 10 \log_{10} \frac{\sigma_x^2}{E(q^2)} \\ &= 10 \log_{10} \sigma_x^2 - 10 \log_{10} \frac{\Delta^2}{12} \\ &= 10 \log_{10} \sigma_x^2 - 10 \log_{10} \frac{A^2}{12} + 10 \log_{10} 2^{2R} \\ &= 10 \log_{10} \sigma_x^2 - 10 \log_{10} \frac{A^2}{12} + \frac{\ln 2}{\ln 10} 20R \\ &= 10 \log_{10} \sigma_x^2 - 10 \log_{10} \frac{A^2}{12} + 6R \end{aligned} \tag{51}$$

On peut noter qu'un bit supplémentaire améliore le rapport signal à bruit de 6 dB.

ERREUR DE QUANTIFICATION

Si on suppose que le signal est une sinusoide d'amplitude crête à crête A (soit une amplitude crête de $A/2$), on a :

$$\sigma_x^2 = \frac{A^2}{8} \tag{52}$$

On obtient alors :

$$\begin{aligned} SNR &= 10 \log_{10} \sigma_x^2 - 10 \log_{10} \frac{A^2}{12} + 6R \\ &= 10 \log_{10} \frac{3}{2} + 6R \\ &= 1,76 + 6R \quad dB \end{aligned} \tag{53}$$

MODULATION PAR IMPULSION ET CODAGE

Dans la modulation MIC ou PCM (*pulse coded modulation* en anglais), après échantillonnage, les échantillons sont simplement quantifiés puis codés.

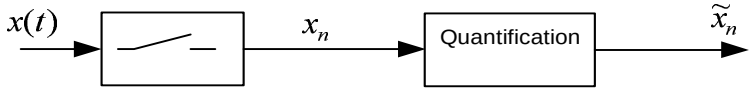


Figure 10: Codeur PCM

On a la relation suivante entre la sortie du codeur PCM et l'entrée:

$$\tilde{x_n} = x_n + q_n \tag{54}$$

où q_n est l'erreur de quantification

Cette technique est bien adaptée aux sources sans mémoire.

MODULATION PAR IMPULSION ET CODAGE

Dans de nombreuses applications comme le traitement de la parole par exemple, les signaux de petites amplitudes se produisent plus fréquemment que ceux de grandes amplitudes

Pour prendre en compte cette distribution non uniforme, plusieurs quantifications non uniformes ont été proposées pour améliorer les performances.

Une solution classique consiste à chercher à maintenir constant le rapport signal à bruit dans la dynamique du signal.

Pour le codage de la parole deux lois de compression (norme ITU-T G.711) sont principalement utilisées :

loi A (système européen)

$$y = \begin{cases} \frac{Ax}{1+\ln A} & \text{si} \quad |x| \leq \frac{1}{A} \quad A = 87,6 \\ (\text{signe de } x)\frac{1+\ln(A|x|)}{1+\ln A} & \text{si} \quad \frac{1}{A} \leq |x| \leq 1 \end{cases} \tag{55}$$

loi μ (système américain)

$$y = (\text{signe de } x)\frac{\ln(1+\mu(x))}{\ln(1+\mu)} \qquad \text{avec} \quad \mu = 255 \quad \text{et} \quad |x| \leq 1 \tag{56}$$

Pour un signal de parole standard, la loi A apporte une réduction de 24 dB du bruit de quantification par rapport à une quantification uniforme.

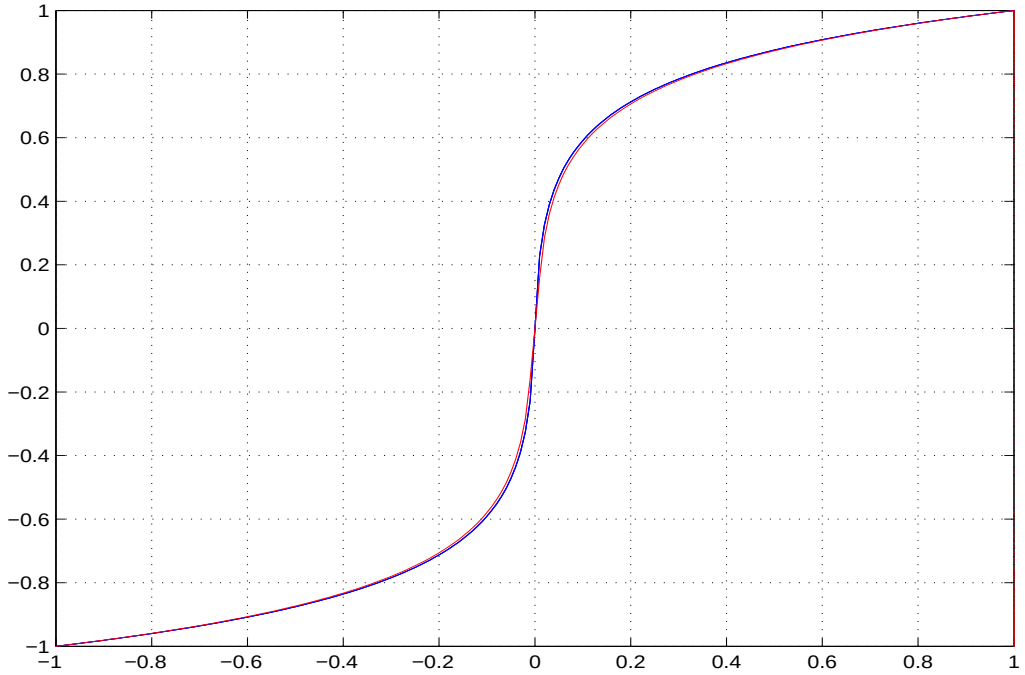


Figure 11: Caractéristiques de transfert d'un compresseur basé sur la loi A et la loi μ

MODULATION PAR IMPULSION ET CODAGE

En pratique, la quantification non uniforme peut être vue comme la concaténation d'une table de correspondance (*look up table* en anglais) appelée aussi compresseur et d'une quanfication uniforme. Le compresseur réalise l'opération non linéaire.

En pratique, la compression peut être réalisée à partir d'une quantification uniforme sur 12 bits. On modélise la loi par 13 segments.

La table d'encodage est présentée ci-dessous :

segment	mot d'entrée 12b	mot de sortie 8b
13	$0\ 1\ X_1\ X_2\ X_3\ X_4\ N\ N\ N\ N\ N\ N$	$0\ 1\ 1\ 1\ X_1\ X_2\ X_3\ X_4$
12	$0\ 0\ 1\ X_1\ X_2\ X_3\ X_4\ N\ N\ N\ N\ N$	$0\ 1\ 1\ 0\ X_1\ X_2\ X_3\ X_4$
11	$0\ 0\ 0\ 1\ X_1\ X_2\ X_3\ X_4\ N\ N\ N\ N$	$0\ 1\ 0\ 1\ X_1\ X_2\ X_3\ X_4$
10	$0\ 0\ 0\ 0\ 1\ X_1\ X_2\ X_3\ X_4\ N\ N\ N$	$0\ 1\ 0\ 0\ X_1\ X_2\ X_3\ X_4$
9	$0\ 0\ 0\ 0\ 0\ 1\ X_1\ X_2\ X_3\ X_4\ N\ N$	$0\ 0\ 1\ 1\ X_1\ X_2\ X_3\ X_4$
8	$0\ 0\ 0\ 0\ 0\ 0\ 1\ X_1\ X_2\ X_3\ X_4\ N$	$0\ 0\ 1\ 0\ X_1\ X_2\ X_3\ X_4$
7	$0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ X_1\ X_2\ X_3\ X_4$	$0\ 0\ 0\ 1\ X_1\ X_2\ X_3\ X_4$
7	$0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ X_1\ X_2\ X_3\ X_4$	$0\ 0\ 0\ 0\ X_1\ X_2\ X_3\ X_4$
7	$1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ X_1\ X_2\ X_3\ X_4$	$1\ 0\ 0\ 0\ X_1\ X_2\ X_3\ X_4$
7	$1\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ X_1\ X_2\ X_3\ X_4$	$1\ 0\ 0\ 1\ X_1\ X_2\ X_3\ X_4$
6	$1\ 0\ 0\ 0\ 0\ 0\ 1\ X_1\ X_2\ X_3\ X_4\ N$	$1\ 0\ 1\ 0\ X_1\ X_2\ X_3\ X_4$
5	$1\ 0\ 0\ 0\ 0\ 1\ X_1\ X_2\ X_3\ X_4\ N\ N$	$1\ 0\ 1\ 1\ X_1\ X_2\ X_3\ X_4$
4	$1\ 0\ 0\ 0\ 1\ X_1\ X_2\ X_3\ X_4\ N\ N\ N$	$1\ 1\ 0\ 0\ X_1\ X_2\ X_3\ X_4$
3	$1\ 0\ 0\ 1\ X_1\ X_2\ X_3\ X_4\ N\ N\ N\ N$	$1\ 1\ 0\ 1\ X_1\ X_2\ X_3\ X_4$
2	$1\ 0\ 1\ X_1\ X_2\ X_3\ X_4\ N\ N\ N\ N\ N$	$1\ 1\ 1\ 0\ X_1\ X_2\ X_3\ X_4$
1	$1\ 1\ X_1\ X_2\ X_3\ X_4\ N\ N\ N\ N\ N\ N$	$1\ 1\ 1\ 1\ X_1\ X_2\ X_3\ X_4$

CODAGE POUR SOURCES ANALOGIQUES CORRELES

Les sources analogiques (son et image) possède souvent une forte redondance qu'une simple modulation PCM ne peut exploiter. Cette redondance est directement liée à la corrélation entre échantillons : on parle aussi de source à mémoire.

Il existe 3 grandes familles de techniques pour exploiter cette propriété afin de réduire le nombre de bits nécessaire pour transmettre ces échantillons :

- les techniques basées sur une forme d'onde temporelle comme la modulation Delta, PCM, DPCM, ... souvent utilisé pour le codage de la parole.
- les techniques utilisant une décomposition spectrale comme le codage par sous-bande et le codage par transformée (cosinus discret ou ondelettes).
- les techniques basées sur des modèles de source comme le codage linéaire prédictif (LPC) utilisés pour le codage de la parole à très bas débit.

MODULATION DELTA

Lorsque la source est à mémoire, la variation de l'amplitude entre les échantillons successifs est relativement faible. En quantifiant les différences entre les amplitudes des échantillons successifs, il est possible de réduire le débit en sortie du codeur.

Le principe de base de la modulation Delta consiste à quantifier la différence d'amplitude e_n entre l'échantillon courant x_n et l'échantillon quantifié $\hat{x}_n$.

$$e_n = x_n - \hat{x}_n \tag{57}$$

La quantification est uniquement réalisée sur deux niveaux ($\tilde{e}_n = \pm\Delta$).

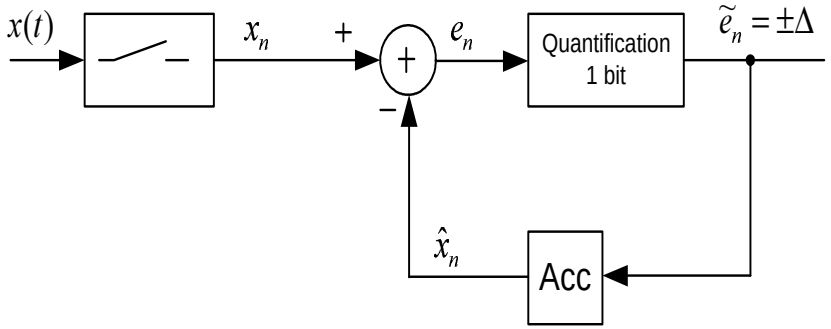


Figure 12: Modulateur Delta

MODULATION DELTA

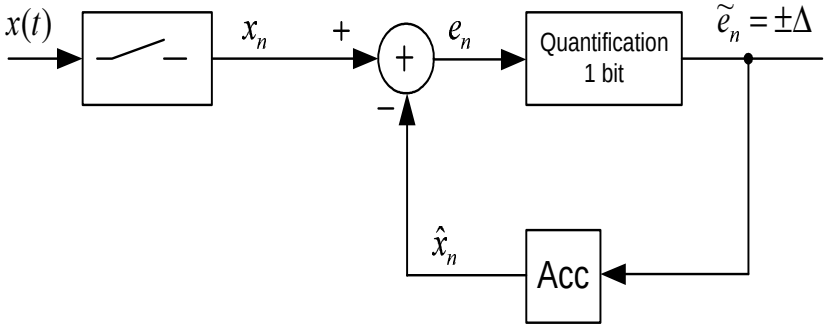


Figure 13: Modulateur Delta

L'accumulateur réalise l'opération suivante :

$$\hat{x}_n = \hat{x}_{n-1} + \tilde{e}_{n-1} \tag{58}$$

Si à l'instant n, on a $x_n > \hat{x}_n$, alors $\tilde{e}_n = +\Delta$ et la sortie de l'accumulateur à l'instant $n+1$ est incrémentée de Δ : $\hat{x}_{n+1} = \hat{x}_n + \Delta$

Si à l'instant n, on a $x_n \leq \hat{x}_n$, alors $\tilde{e}_n = -\Delta$ et la sortie de l'accumulateur à l'instant $n+1$ est décrémentée de Δ : $\hat{x}_{n+1} = \hat{x}_n - \Delta$

Le démodulateur Delta est un simple accumulateur. On a :

$$\hat{x}_n = \hat{x}_{n-1} + \tilde{e}_{n-1} \tag{59}$$

L'erreur de quantification q_n apportée par l'opération de quantification est donnée par :

$$q_n = \tilde{e}_n - e_n \tag{60}$$

Il est alors possible d'exprimer l'échantillon estimé $\hat{x}_n$ à partir de x_{n-1} et de l'erreur de quantification :

$$\begin{aligned} \hat{x}_n &= \hat{x}_{n-1} + \tilde{e}_{n-1} \\ &= (x_{n-1} + q_{n-1} - \tilde{e}_{n-1}) + \tilde{e}_{n-1} \\ &= x_{n-1} + q_{n-1} \end{aligned} \tag{61}$$

Ainsi l'échantillon estimé $\hat{x}_n$ est égal à l'échantillon précédent x_{n-1} entaché de l'erreur de quantification q_{n-1}

MODULATION DELTA

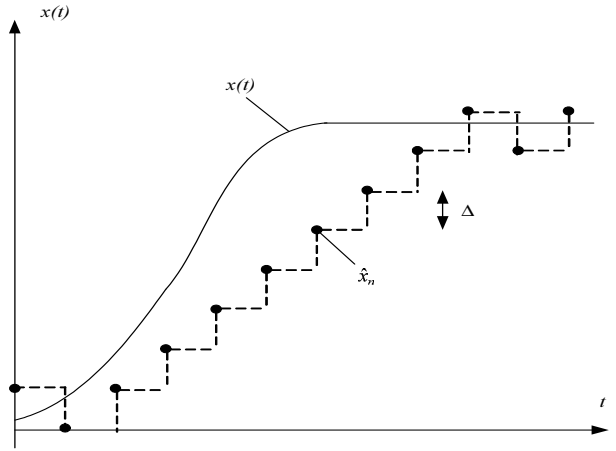


Figure 14: Exemple d'évolution de $\hat{x}_n$ dans un modulateur Delta

On peut observer deux types d'erreurs : l'erreur de poursuite et le bruit granulaire

L'erreur de poursuite est liée à la pente de $\hat{x}_n$ limitée à Δ/T_{ech}. Pour diminuer l'erreur de poursuite, la fréquence d'échantillonnage doit être égale à 4 à 5 fois la fréquence minimum d'échantillonnage. L'autre solution consiste à augmenter la valeur de Δ.

Le bruit granulaire se produit même si le signal $x(t)$ est constant. Les échantillons estimés $\hat{x}_n$ oscillent entre deux pas (bruit crête à crête de Δ. Une solution consiste alors à diminuer Δ.

Le choix de Δ est un compromis entre les deux types d'erreurs.

MODULATION PAR IMPULSION ET CODAGE DIFFERENTIEL

Le principe de base de la modulation DPCM (*differential pulse coded modulation* en anglais) aussi notée modulation par impulsion et codage différentiel (MICD) consiste à quantifier la différence d'amplitude e_n entre l'échantillon courant x_n et l'échantillon prédit $\hat{x}_n$.

e_n est aussi appelée l'erreur de prédiction.

$$e_n = x_n - \hat{x}_n \tag{62}$$

$\hat{x}_n$ est obtenu en utilisant un prédicteur d'ordre P :

$$\hat{x}_n = \sum_{i=1}^{P} a_i x_{n-i} \tag{63}$$

PREDICTION

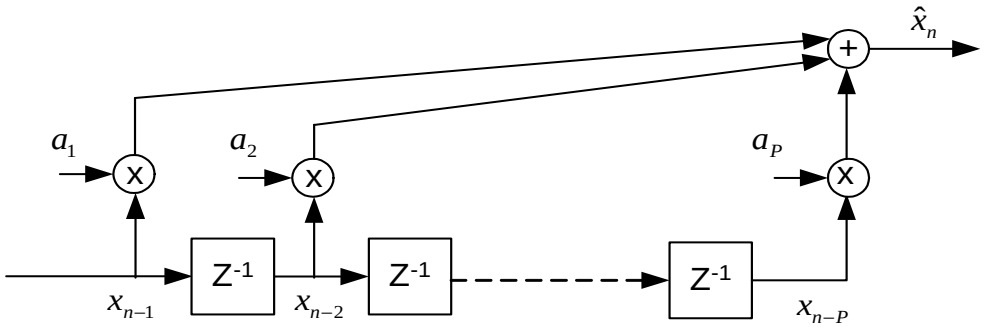


Figure 15: synoptique du prédicteur d'ordre P

On déterminera les P coefficients de prédiction a_i qui minimisent l'erreur quadratique moyenne $E(e_n^2) = E[(x_n - \hat{x}_n)^2]$. Ces coefficients sont les solutions du système linéaire suivant :

$$\sum_{i=1}^{P} a_i \phi(i-j) = \phi(j) \qquad \text{pour} \quad j = 1, 2, \ldots, P \tag{64}$$

où $\phi(m)$ est la fonction d'autocorrélation des échantillons x_n. Ce système se résout efficacement en utilisant l'algorithme de Levinson. Les coefficients a_i sont déterminés en début de transmission mais peuvent aussi être ajustés périodiquement

Avec cette structure, les échantillons à quantifier sont décorrélés et de très faible amplitude et nécessite donc un nombre de bits limités.

MODULATION PAR IMPULSION ET CODAGE DIFFERENTIEL

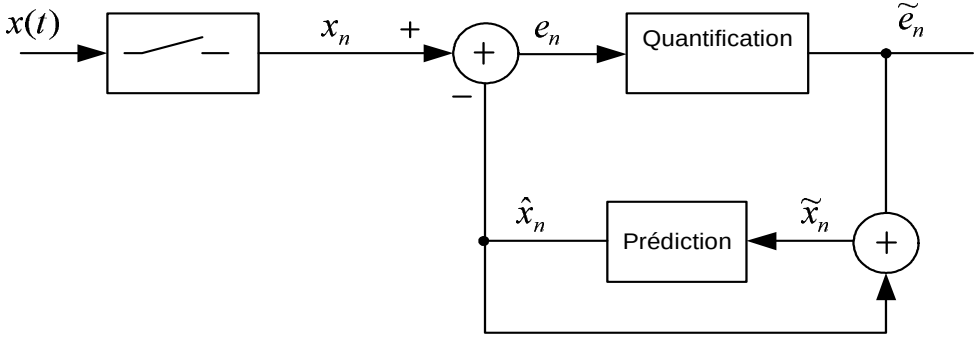


Figure 16: Codeur DPCM

En entrée du prédicteur, au lieu des échantillons de la source x_n , on utilise les échantillons modifiés par quantification $\tilde{x}_n = \hat{x}_n + \tilde{e}_n$:

$$\hat{x}_n = \sum_{i=1}^{P} a_i \tilde{x}_{n-i} \tag{65}$$

On peut vérifier que l'erreur de quantification $q_n = \tilde{e}_n - e_n$ est aussi égale à la différence $\tilde{x}_n - x_n$:

$$\begin{aligned} q_n &= \tilde{e}_n - e_n \\ &= \left((\tilde{x}_n - \hat{x}_n) - x_n - \hat{x}_n\right) \\ &= \tilde{x}_n - x_n \end{aligned} \tag{66}$$

MODULATION PAR IMPULSION ET CODAGE DIFFERENTIEL

Synoptique d'un décodeur DPCM :

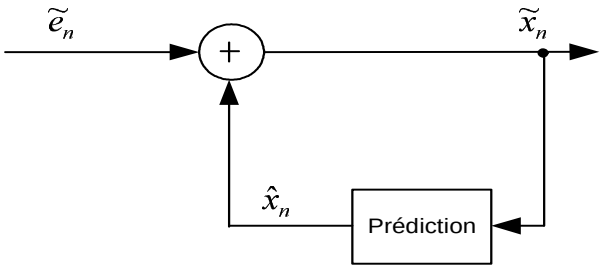


Figure 17: Décodeur DPCM

Pour le décodage, on utilise exactement le même prédicteur que pour le codage (en considérant l'absence d'erreurs de transmission). Ainsi, on peut reconstruire les échantillons $\tilde{x}_n = \tilde{e}_n + \hat{x}_n$.

La modulation DPCM est utilisée pour le codage de la parole dans les standards ITU G.721, G.722, G.723 et G.726.

COMPARAISON DES TECHNIQUES POUR LE CODAGE DE LA PAROLE

La table suivante présente une comparaison des différentes techniques de modulation pour le codage de la parole en considérant une fréquence d'échantillonnage de 8kHz.

Les paramètres choisis sont ceux qui sont les plus couramment utilisés.

Table 1: Comparaison des modulations pour le codage de la parole

technique	quantificateur	nbr de bits	débit
PCM	uniforme	12 bits	96 kb/s
log PCM	logarithmique	8 bits	64 kb/s
DPCM	logarithmique	4-6 bits	32-48 kb/s
ADPCM	adaptative	3-4 bits	24-32 kb/s
Delta	binaire	1 bit	32-64 kb/s
Delta adaptatif	binaire	1 bit	16-32 kb/s
LPC CELP			2,4-9,6 kb/s

COURS 4

Codes en ligne, NRZ

TRANSMISSION EN BANDE DE BASE

• Transmission en bande de base vs transmission par ondes modulées

• Une transmission en bande de base signifie que les symboles à émettre dans le canal de transmission ne subissent pas de translation de leur spectre autour d'une fréquence porteuse.

• Dans le cas d'une transmission par ondes modulées, les symboles à émettre module une porteuse de fréquence f_c

• D'une manière générale, le signal émis s'écrit de la forme suivante :

$$x(t) = \sum_{k=-\infty}^{\infty} a_k g(t - kT)$$

T est la durée de l'impulsion élémentaire ou durée symbole

• Les critères principaux pour choisir un code en ligne sont les suivants :

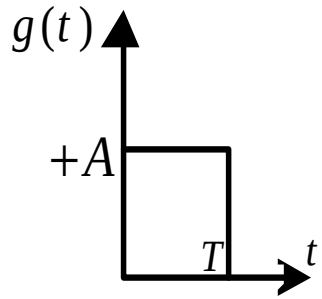
$\rightarrow$ la densité spectrale du code

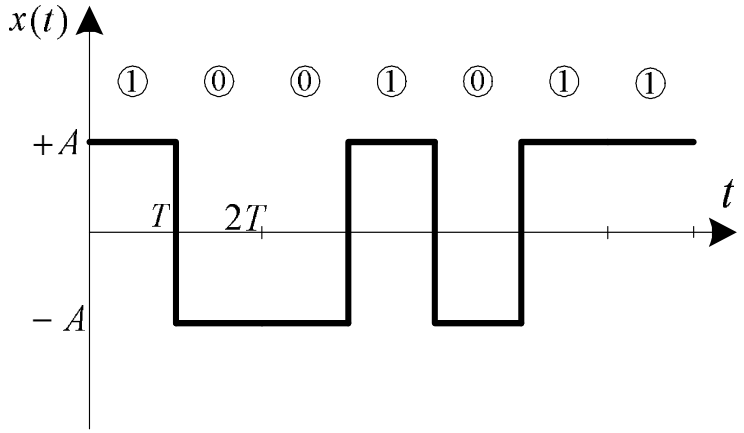
$\rightarrow$ la densité des transitions dans le signal émis

$\rightarrow$ la résistance au bruit

CODE NON RETOUR A ZERO

● Ce code associe un niveau +A à chaque bit égal à "1" et un niveau -A à chaque bit égal à "0".





● 1 transition dans le signal émis à chaque transition de la séquence binaire

● problème de la longue série de "0" ou de "1" → séquence pilote

CALCUL DE LA DENSITE SPECTRALE

Soit $\gamma_{XX}(f)$ la densité spectrale d'un signal $x(t)$ émis en bande de base.

Formule de Bennett :

$$\boxed{\gamma_{XX}(f) = \frac{1}{T}|G(f)|^2 \gamma_{AA}(f) \qquad (67)}$$

où $|G(f)|^2$ est la densité spectrale de puissance de l'impulsion $g(t)$ et $\gamma_{AA}(f)$ est la transformée de Fourier de la fonction d'autocorrélation $\gamma_{aa}(i) = E(a_k^* a_{k+i})$.

$$\gamma_{AA}(f) = \sum_{i=-\infty}^{+\infty} \gamma_{aa}(i) e^{-j2\pi f i T} \qquad (68)$$

DENSITE SPECTRALE DU CODE NRZ

Puisque $g(t)$ est une impulsion de largeur T et d'amplitude A, on a :

$$G(f) = \int_{-\infty}^{+\infty} e^{-j2\pi ft} g(t)dt = \left[\frac{Ae^{-j2\pi ft}}{-j2\pi ft}\right]_{-T/2}^{+T/2} = \frac{AT \sin \pi fT}{\pi fT} \tag{69}$$

$$|G(f)|^2 = A^2T^2 \left(\frac{\sin(\pi fT)}{\pi fT}\right)^2 \tag{70}$$

Dans le cas du code NRZ, les symboles a_k sont indépendants et peuvent prendre les valeurs +1 et -1. Leur moyenne $E(a_k)$ est nulle et leur variance $E((a_k)^2)$ est égale à 1.

la fonction d'autocorrélation $\gamma_{aa}(i)$ est égale à

$$\gamma_{aa}(i) = \begin{cases} 1 & \text{si} \quad i = 0 \\ 0 & \text{sinon} \end{cases} \tag{71}$$

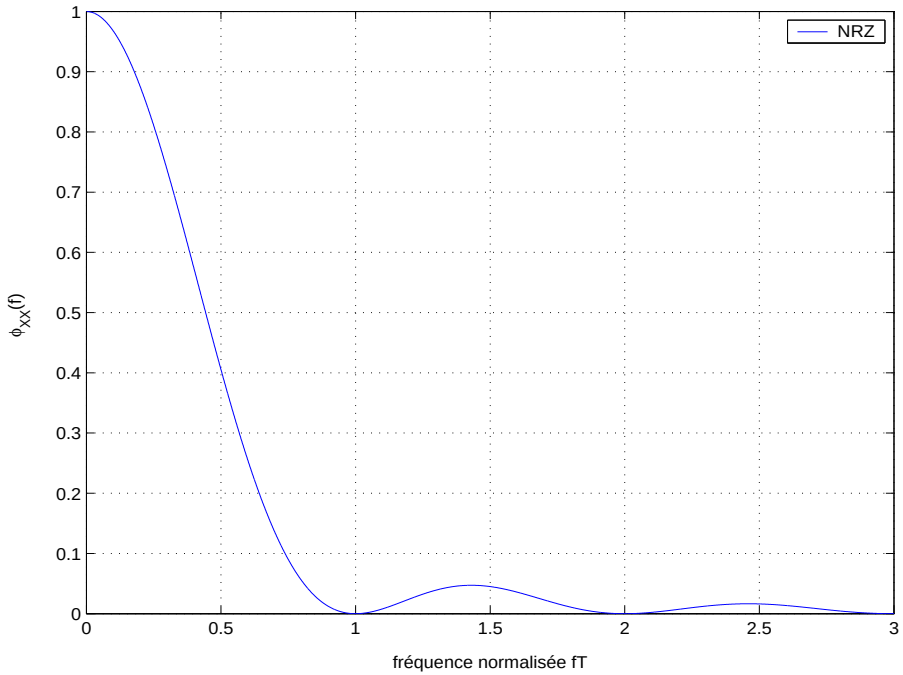
En conséquence, on a $\gamma_{AA}(f) = 1$

Finalement, on obtient donc la densité spectrale de puissance suivante :

$$\gamma_{XX}(f) = A^2T \left(\frac{\sin(\pi fT)}{\pi fT}\right)^2 \tag{72}$$

DENSITE SPECTRALE DU CODE NRZ

$$\gamma_{XX}(f) = A^2 T \left(\frac{\sin(\pi f T)}{\pi f T} \right)^2 \tag{73}$$

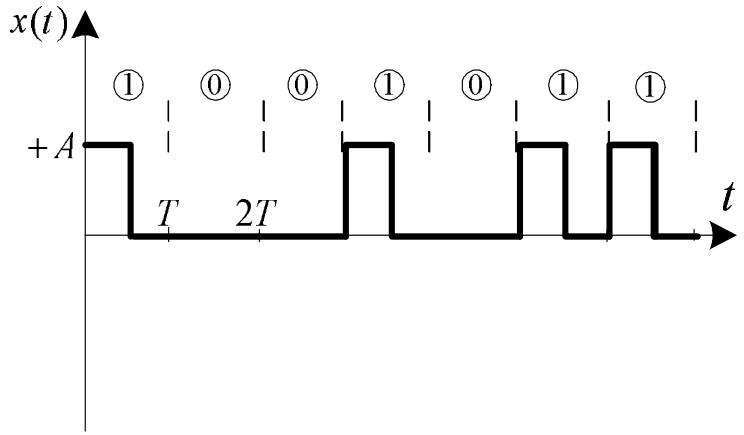


- Occupation spectrale théoriquement infinie
- 90 % de l'énergie est contenue dans le premier lobe ($f = 0$ à $1/T$)

CODE RETOUR A ZERO (RZ) UNIPOLAIRE

• Ce code associe à chaque bit égal à "1" un niveau +A pendant une durée $T/2$ puis un niveau O pendant $T/2$. A chaque bit égal à "0" est associé un niveau 0.

• Ce code est aussi appelé code RZ 1/2.

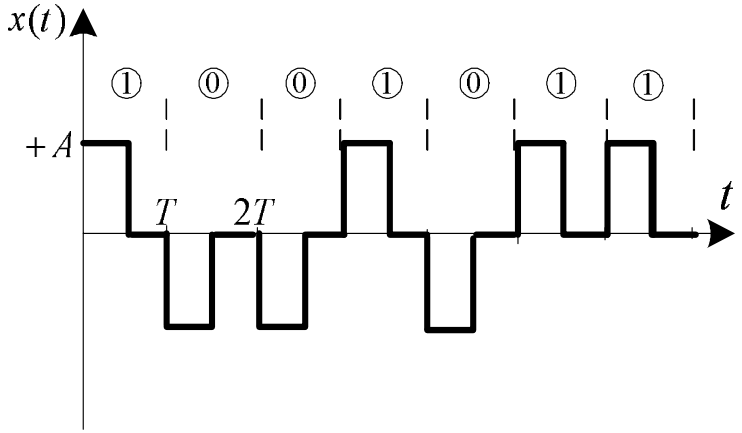


• moyenne du signal émis non nulle

• raie spectrale à $f = 1/T$

CODE RETOUR A ZERO (RZ) BIPOLAIRE SIMPLE

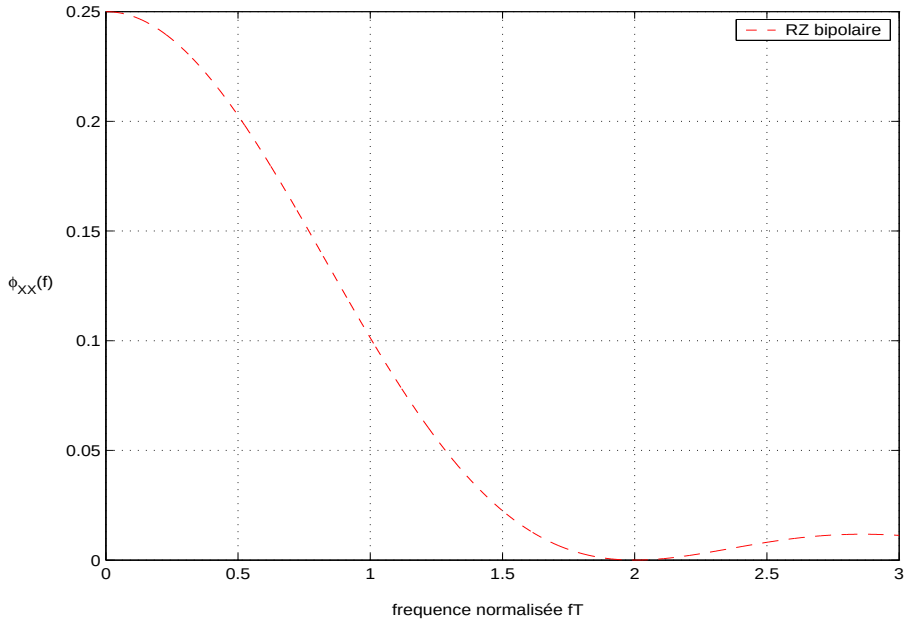
• Ce code associe à chaque bit égal à "1" un niveau +A pendant une durée $T/2$ puis un niveau O pendant $T/2$. A chaque bit égal à "0" est associé un niveau -A pendant une durée $T/2$ puis un niveau O pendant $T/2$.



• synchronisation facile

DSP DU CODE RZ BIPOLAIRE SIMPLE

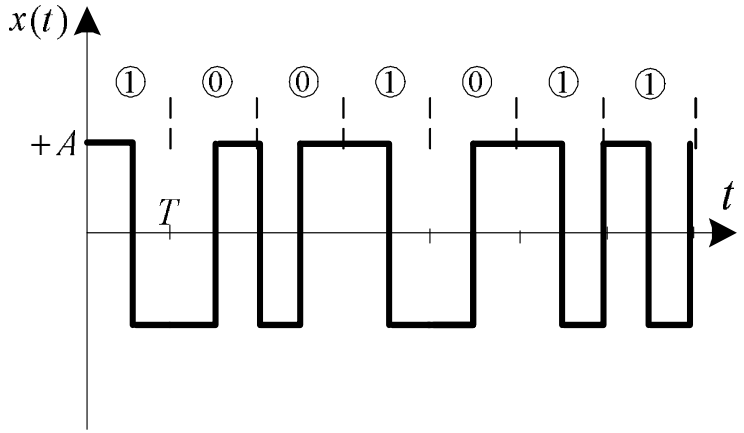
$$\gamma_{XX}(f) = \frac{A^2 T}{4} \left(\frac{\sin(\pi f T/2)}{\pi f T/2} \right)^2 \tag{74}$$



- mais occupation spectrale plus élevée que le code NRZ

CODE BIPHASE OU MANCHESTER

• Ce code associe à chaque bit égal à "1" un niveau +A pendant une durée $T/2$ puis un niveau -A pendant $T/2$. A chaque bit égal à "0" on associe un niveau -A pendant $T/2$ puis un niveau +A pendant $T/2$.

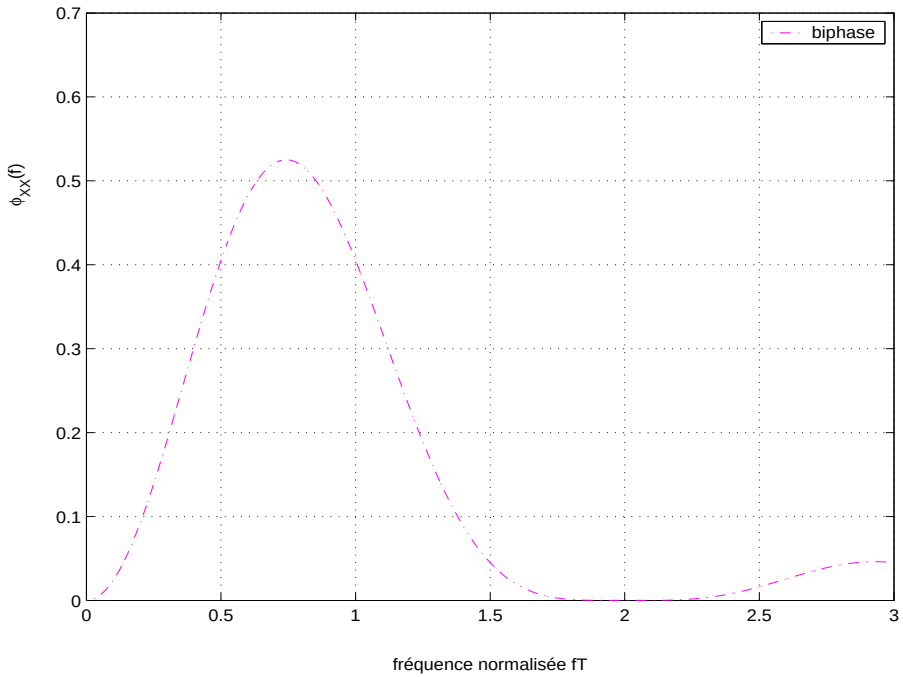


• synchronisation facilité par les transitions régulières

• utilisé pour Ethernet

DSP DU CODE BIPHASE

$$\gamma_{XX}(f) = A^2 T \frac{(\sin(\pi f T/2))^4}{(\pi f T/2)^2} \tag{75}$$

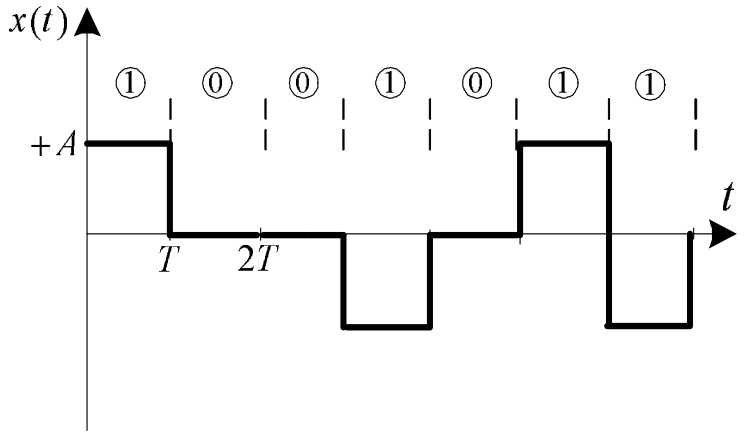


- occupation spectrale assez large
- intéressant lorsque le canal ne laisse pas passer les basses fréquences

CODE BIPOLAIRE OU AMI

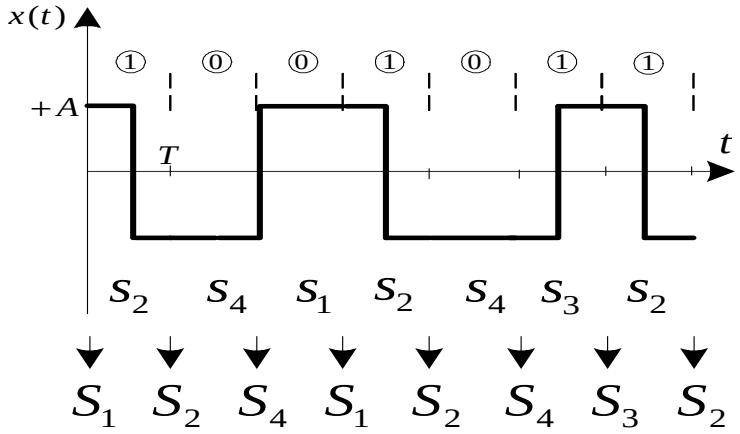
● Ce code associe à chaque bit égal à "1" successivement un niveau +A et un niveau -A. A chaque bit égal à "0" est associé un niveau 0.

Le code bipolaire est aussi appelé code Alternated Mark Inversion (AMI).

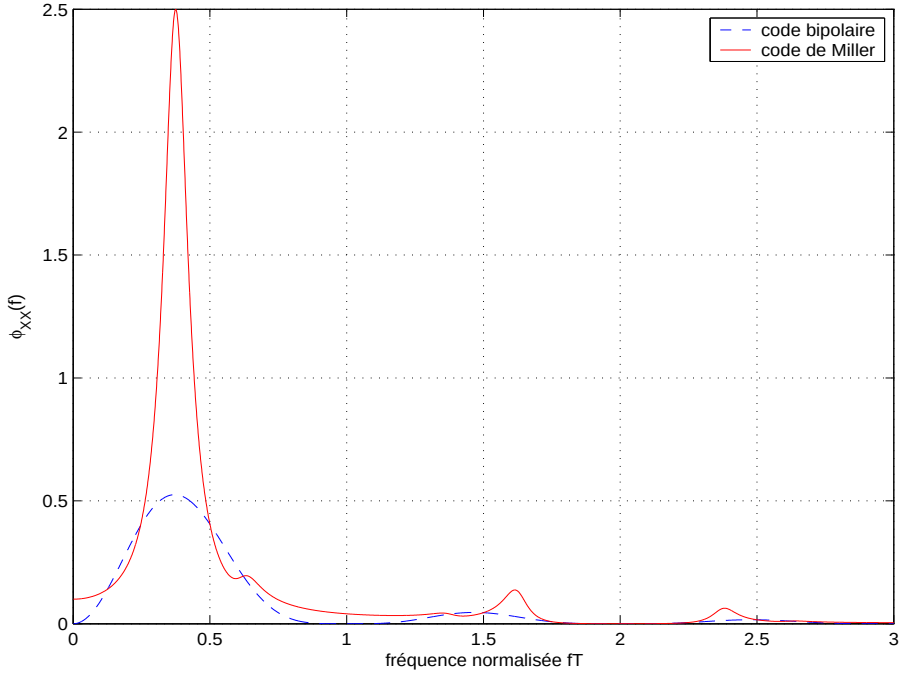


CODE DE MILLER

• Ce code associe à chaque bit égal à "1" soit un niveau +A pendant $T/2$ puis un niveau 0 soit un niveau 0 pendant $T/2$ puis un niveau 0. A chaque bit égal à "0" on associe soit un niveau -A et un niveau +A. La polarité du signal associé à un bit égal à "1" est choisie de façon à garantir une continuité avec l'impulsion précédente. La polarité du signal associé à un bit égal à "0" est choisie de façon à garantir une continuité avec l'impulsion précédente si celle ci portait un bit égal à "1".

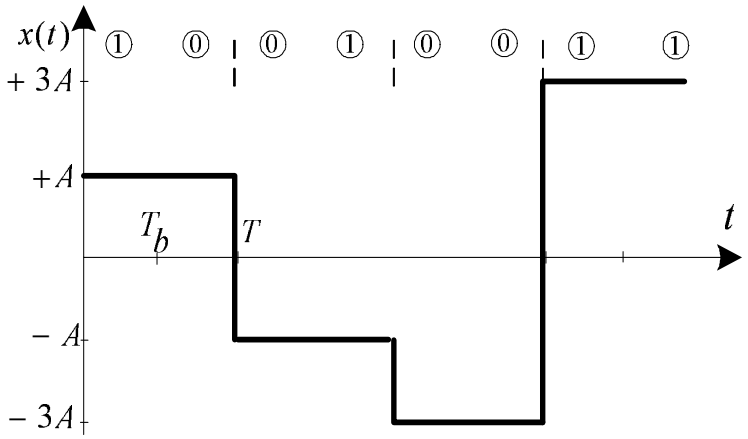


DENSITE SPECTRALE DU CODE BIPOLAIRE ET MILLER



CODE NRZ M-AIRE

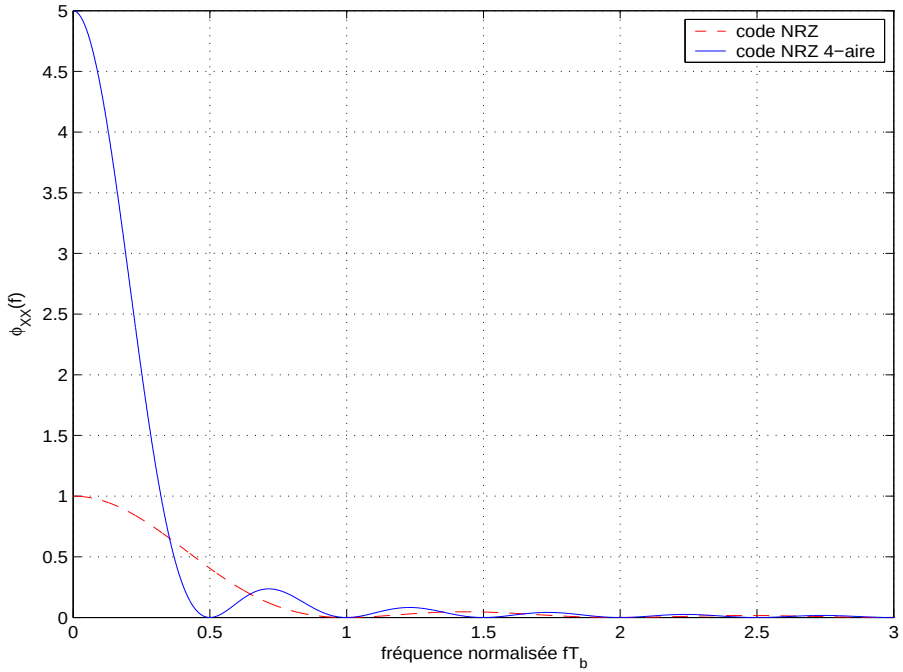
Exemple pour $M = 4$, $T = 2T_b$



DENSITE SPECTRALE DU CODE NRZ M-AIRE

$$\gamma_{XX}(f) = \frac{A^2 T}{3}(M^2 - 1)\left(\frac{\sin(\pi f T)}{\pi f T}\right)^2 \qquad (76)$$

Densité spectrale des codes NRZ et NRZ-4 aire :

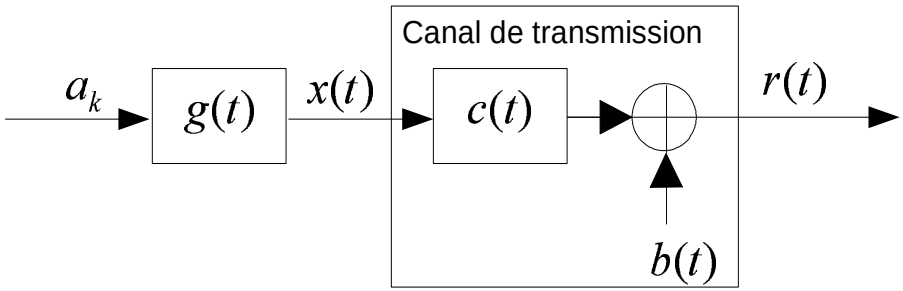


COURS 5

Filtrage adapté

CANAL DE TRANSMISSION

• Le signal émis $x(t)$ est modifié par le canal de transmission qui est en général modélisé par un filtre linéaire de réponse impulsionnelle $c(t)$. De plus, le canal ajoute un bruit blanc gaussien $b(t)$.

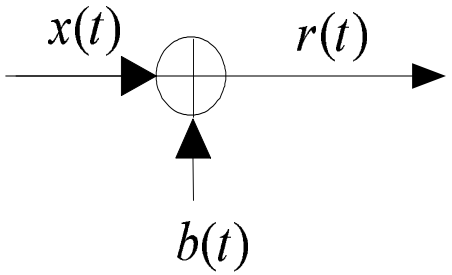


• Le signal reçu $r(t)$ est égal à :

$$\begin{aligned} r(t) &= x(t) * c(t) + b(t) \\ &= \int_{-\infty}^{+\infty} c(\tau)x(t-\tau)d\tau + b(t) \end{aligned}$$

CANAL A BRUIT BLANC ADDITIF GAUSSIEN

• Le canal à bruit blanc additif gaussien (BBAG) est un canal de transmission dont la réponse en fréquence $C(f)$ est égale à 1 sur toute la bande du signal émis :



• Il permet de modéliser les canaux dont le bruit prédominant est un bruit thermique
• On a la relation suivante entre l'entrée et la sortie du canal BBAG :

$$r(t) = x(t) + b(t) \tag{77}$$

• $b(t)$ est un bruit blanc centré gaussien

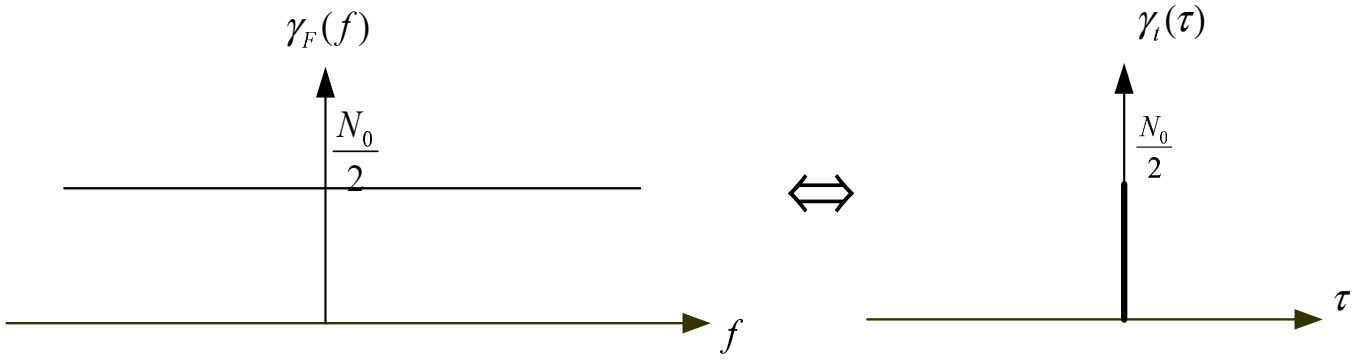
CANAL A BRUIT BLANC ADDITIF GAUSSIEN

- $b(t)$ est un bruit blanc centré gaussien de densité spectrale de puissance bilatérale :

$$\gamma_F(f) = \frac{N_0}{2}$$

- Sa fonction d'autocorrélation $\gamma_t(\tau)$ est égale à

$$\gamma_t(\tau) = \int_{-\infty}^{+\infty} \gamma_F(f) \exp(2j\pi f\tau) df = \frac{N_0}{2}\delta(\tau)$$

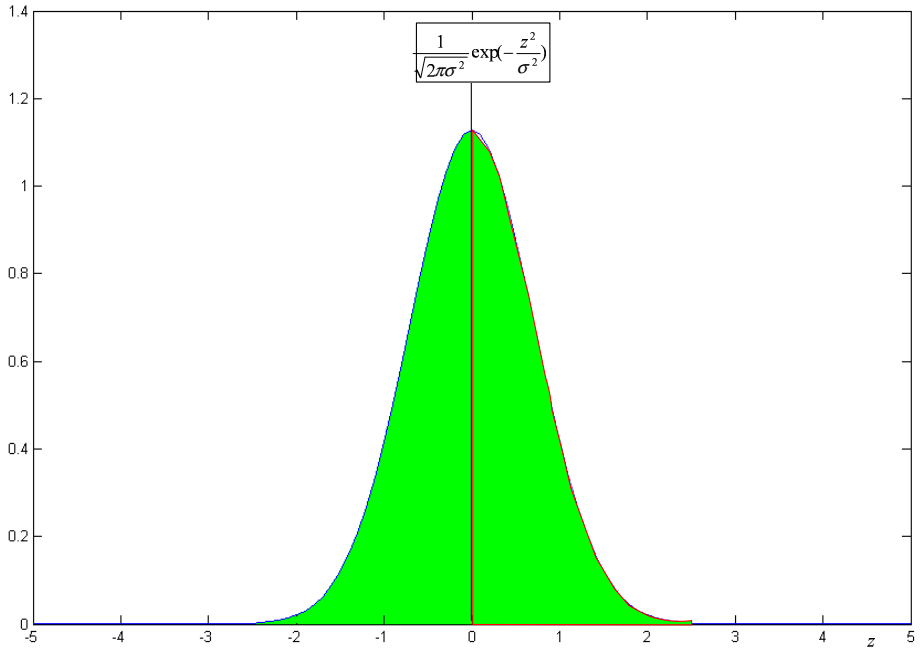


CANAL A BRUIT BLANC ADDITIF GAUSSIEN

• $b(t)$ est modélisé par un processus aléatoire gaussien centré dont la densité de probabilité est la suivante :

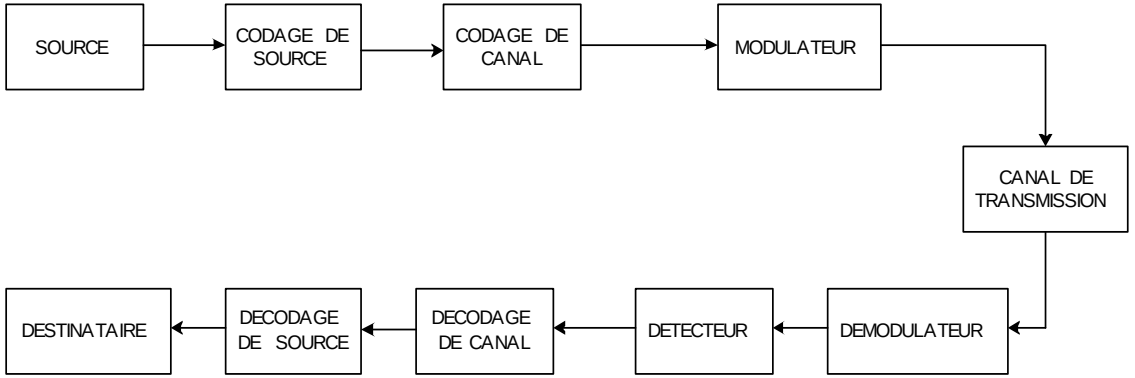
$$p(b) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{b^2}{2\sigma^2}\right) \tag{78}$$

avec $\sigma^2 = \frac{N_0}{2}$



CHAINE DE COMMUNICATION

● Rappelons la structure d'une chaîne de communication point à point:



● Le rôle du démodulateur est d'extraire les échantillons tout en maximisant le rapport signal à bruit.

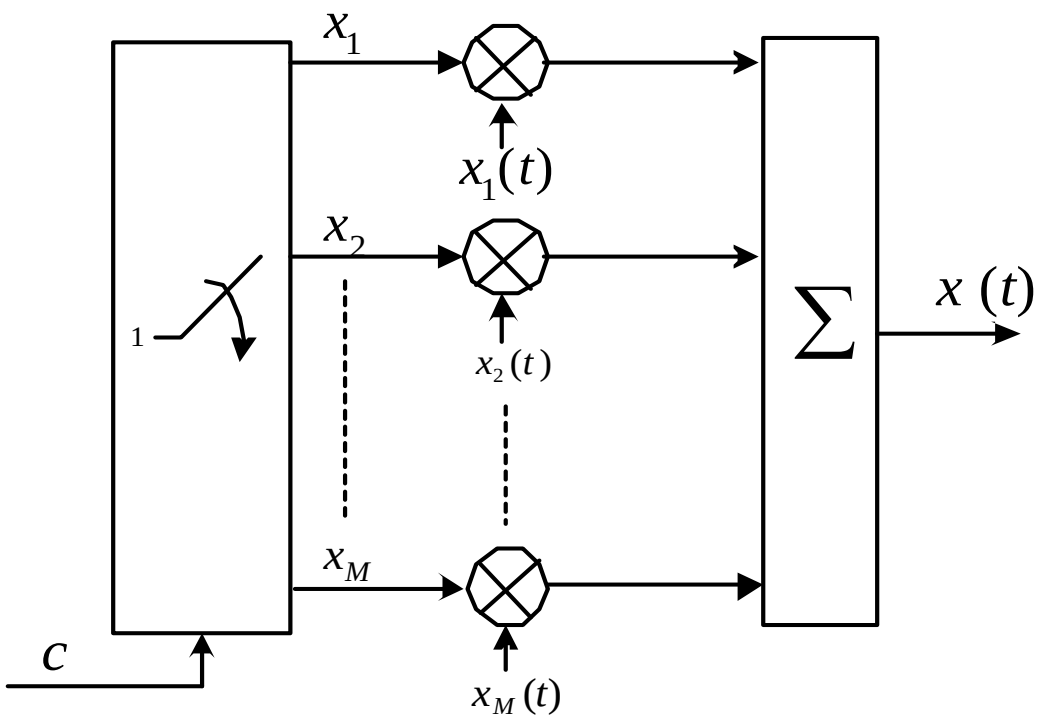
● Le role du détecteur est de décider en faveur des symboles les plus probablement émis.

● Récepteur cohérent = les paramètres comme la fréquence et la phase des signaux reçus sont connus ou ont été correctement estimés.

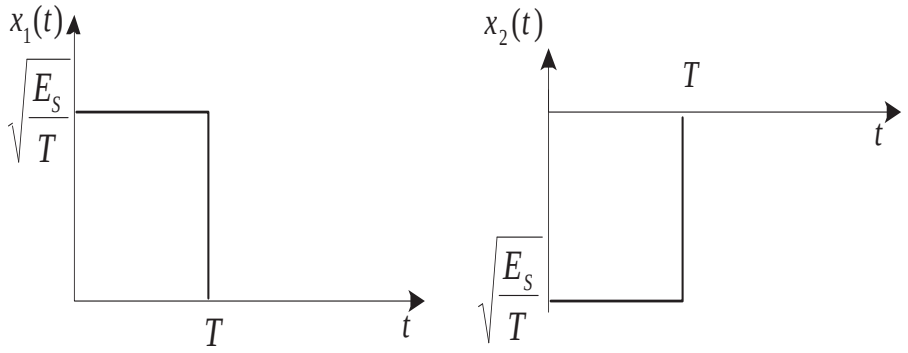
STRUCTURE DU MODULATEUR 1

• On suppose que le codeur de canal délivre des bits groupés par bloc de g bits. On a donc $M = 2^g$ messages différents possibles $c \in \{c_1, c_2, \ldots, c_M\}$.

• le modulateur associe à chaque message $c = c_i$ un signal $x_i(t)$, défini sur l'intervalle fini $0 \leq t \leq T$ et choisi parmi un jeu de $M = q^g$ signaux d'énergie égale à $\mathcal{E}_s$



exemple : cas du code en ligne non retour à zéro (NRZ) composé de $M = 2$ signaux élémentaires $x_1(t)$ et $x_2(t)$.



$$x(t) = \begin{cases} x_1(t) & \text{si} \quad c = 0 \quad (x_1 = 1, x_2 = 0) \\ x_2(t) & \text{si} \quad c = 1 \quad (x_1 = 0, x_2 = 1) \end{cases}$$

• L'énergie de chaque signal $x_i(t)$ entre 0 et T est égale à $\mathcal{E}_s$

$$\mathcal{E}_s = \int_0^T x_i(t)^2 dt \tag{79}$$

STRUCTURE DU MODULATEUR 2

• Il est également possible de représenter les M signaux possibles $x_i(t)$ par des combinaisons linéaires de $N \leq M$ fonctions de base orthonormées $f_i(t)$ et d'énergie unitaire.

Les signaux $x_i(t)$ peuvent s'exprimer par :

$$x_i(t) = \sum_{n=1}^{N} x_n f_n(t) \tag{80}$$

où

$$x_n = \int_0^T x_i(t) f_n(t) dt \tag{81}$$

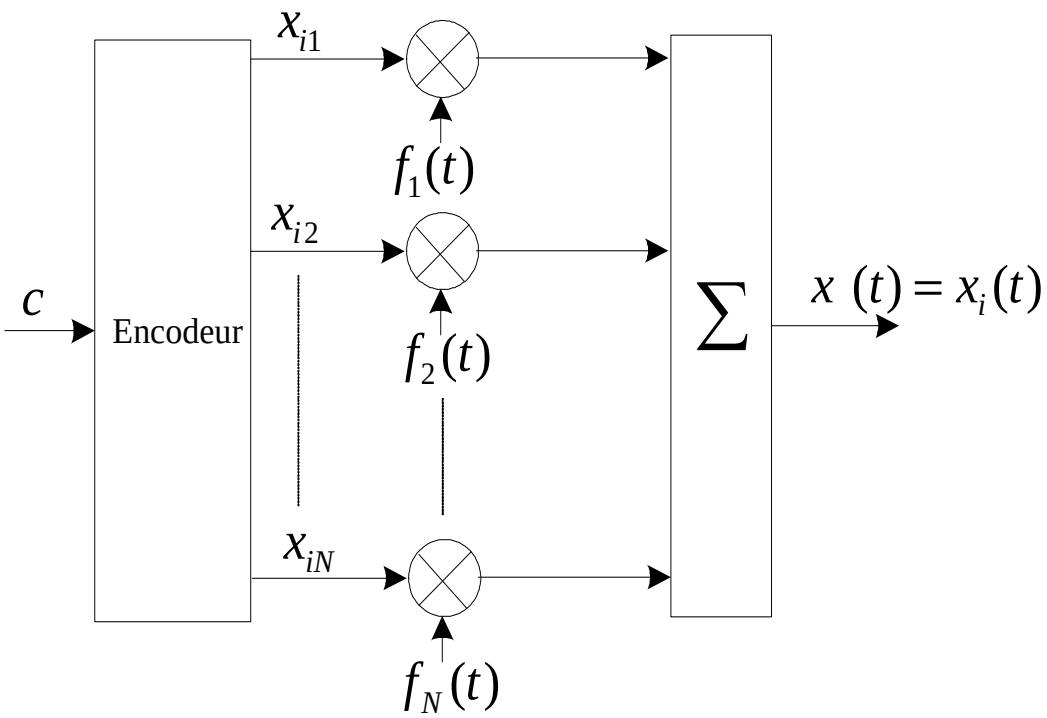
et où les fonctions de base $f_1(t), f_2(t), \ldots, f_N(t)$ sont orthonormées:

$$\int_0^T f_i(t) f_j(t) dt = \delta_{ij} = \begin{cases} 1 & \text{si} \quad i = j \\ 0 & \text{si} \quad i \neq j \end{cases} \tag{82}$$

Ainsi l'énergie des fonctions de base entre 0 et T est égale à 1.

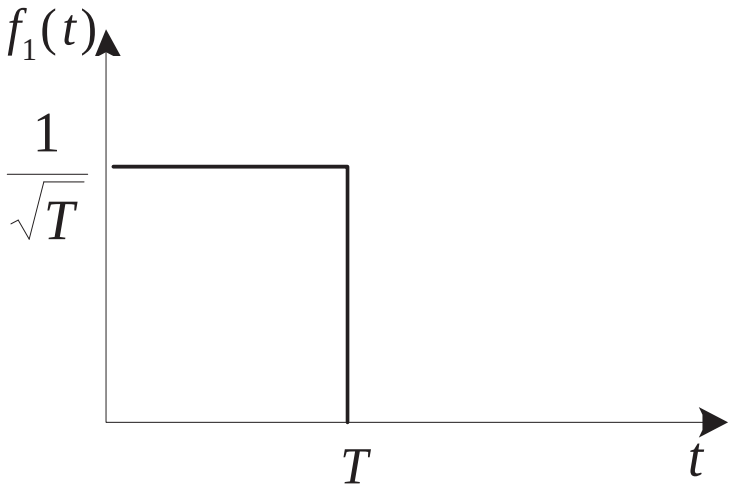
L'énergie de chaque signal $x_i(t)$ entre 0 et T est égale à $\mathcal{E}_s$

$$\mathcal{E}_s = \int_0^T x_i(t)^2 dt = \sum_{i=1}^{N} x_i^2 \tag{83}$$

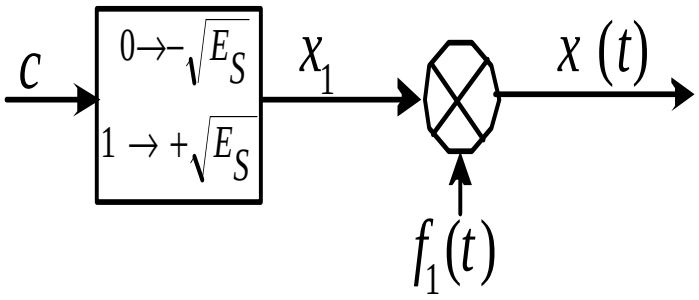


exemple 1(suite) : cas du code en ligne NRZ $M = 2$ signaux

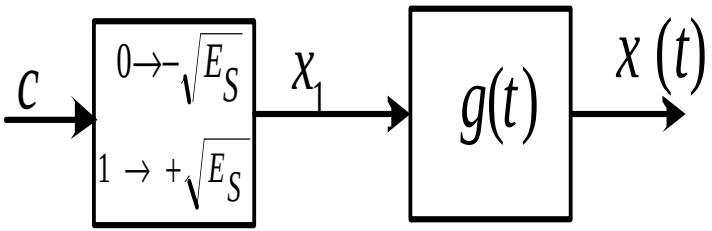
Une seule fonction $f_1(t)$ suffit pour générer les deux signaux $x_1(t)$ et $x_2(t)$.



$$x(t) = \begin{cases} -\sqrt{\mathcal{E}_s} f_1(t) & \text{si} \quad c = 0 \quad (x_1 = -\sqrt{\mathcal{E}_s}) \\ +\sqrt{\mathcal{E}_s} f_1(t) & \text{si} \quad c = 1 \quad (x_1 = +\sqrt{\mathcal{E}_s}) \end{cases}$$



- $f_1(t)$ correspond à la réponse impusionnelle $g(t)$ du filtre d'emission :



RECEPTEUR OPTIMAL POUR CANAL BBAG

• On considére qu'un des M signaux possibles $x_i(t)$ a été transmis sur un canal à bruit blanc additif gaussien (BBAG) pendant la durée T.

En entrée du récepteur, on a

$$r(t) = x_i(t) + b(t) \qquad 0 \leq t \leq T \tag{84}$$

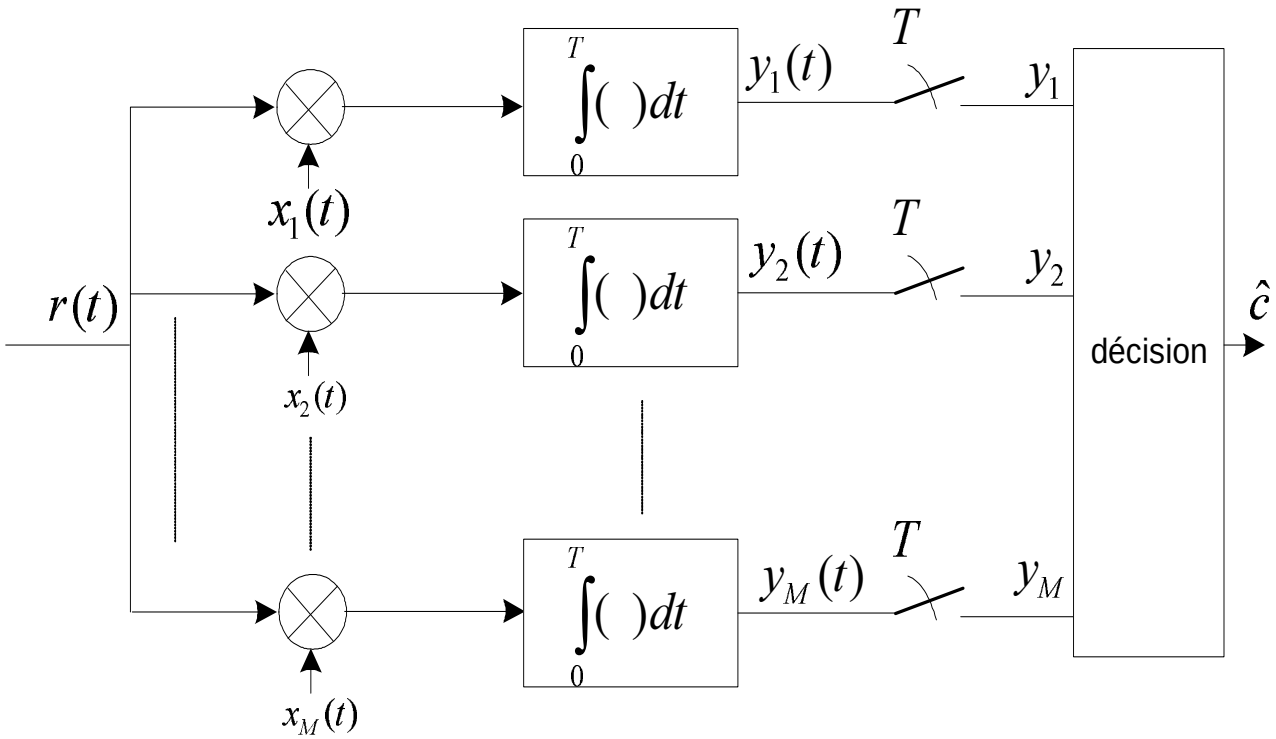
où $b(t)$ est un bruit blanc gaussien de densité spectrale de puissance unilatérale N_0

• L'objectif d'un récepteur optimal est de retrouver la séquence émise en minimisant le taux d'erreurs.

• Deux structures équivalentes de récepteur optimal : le corrélateur et le filtre adapté

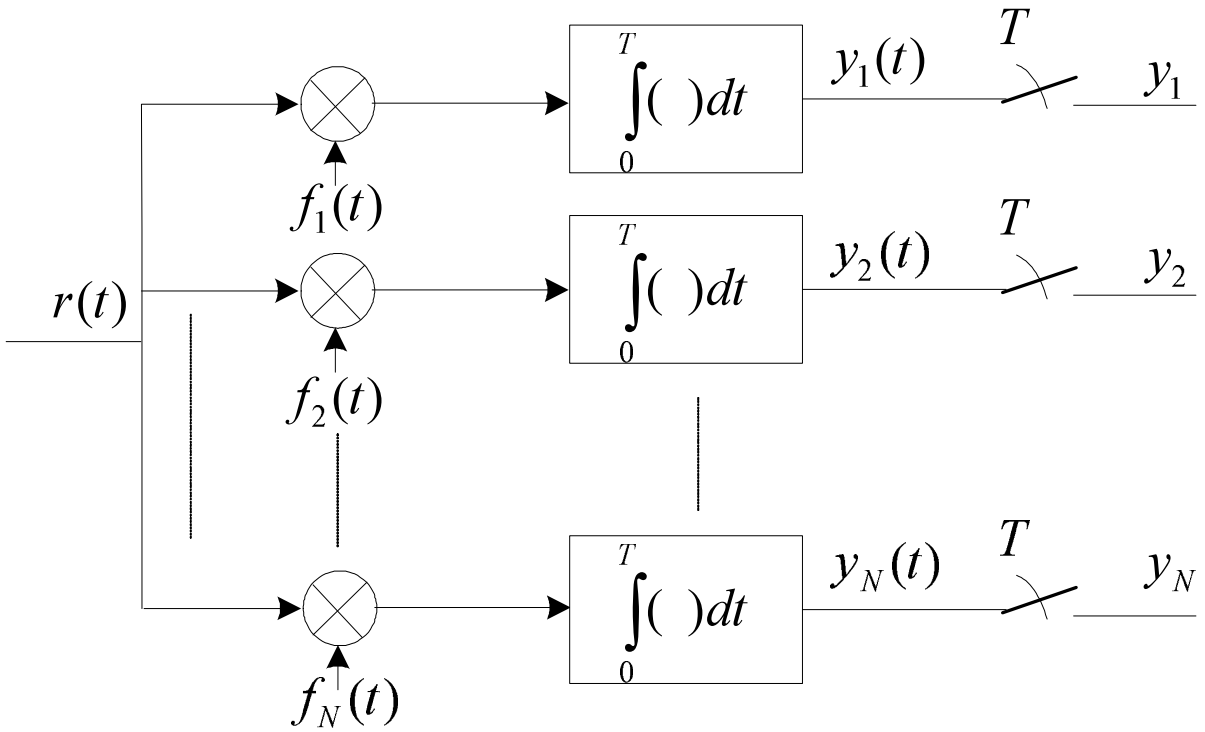
CORRELATEUR 1

• Une première structure de démodulateur consiste à projetter le signal reçu $r(t)$ sur chacun des M signaux $x_i(t)$ possibles.



CORRELATEUR 2

• Une autre solution consiste à projetter le signal reçu sur les N fonctions de base $f_i(t)$.



• Comme les fonctions de base $f_j(t)$ sont d'énergie unitaire entre 0 et T on a relation suivante :

$$
\begin{aligned}
y_j &= y_j(T) \\
&= \int_0^T r(t) f_j(t) dt \\
&= \int_0^T x_i(t) f_j(t) dt + \int_0^T b(t) f_j(t) dt \\
&= x_i + n_j
\end{aligned}
$$

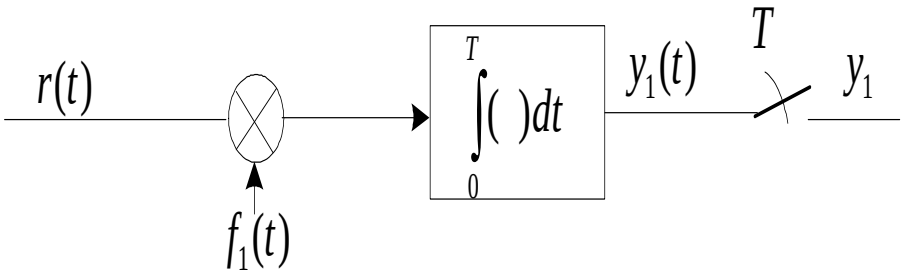
avec y_j la sortie du ième échantillonneur à l'instant T et $1 \leq j \leq N$.

• Le signal reçu $r(t)$ est maintenant représenté par un vecteur à N composantes y_j

• Les échantillons de bruit n_j sont des variables aléatoires relatives au bruit additif du canal de transmission. On peut montrer qu'ils sont centrés et de variance $\sigma^2 = \frac{N_0}{2}$.

• Il nous restera à décider en faveur du signal le plus probablement émis.

exemple 1(suite) :



$$\begin{aligned} y &= y_1(T) \\ &= \int_0^T y(t) f_1(t) dt \\ &= \int_0^T x_i(t) f_1(t) dt + \int_0^T n(t) f_1(t) dt \\ &= x_i + n \end{aligned}$$

• Le bruit n en sortie de l'échantillonneur est gaussien de moyenne nulle $E(n) = 0$ et de variance $\sigma^2 = \frac{N_0}{2}$.

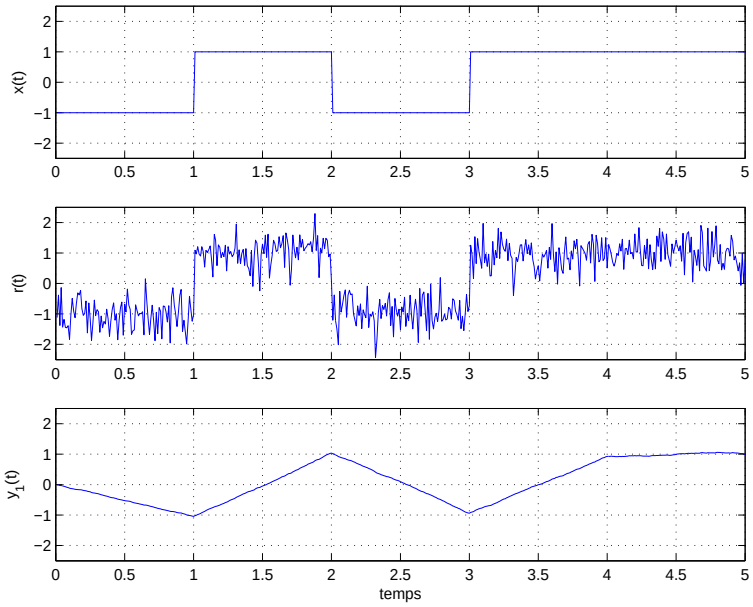
- Le rapport signal à bruit SNR après échantillonnage est donc

$$SNR = \frac{P_x}{P_n} = \frac{E\left[x_i^2\right]}{\sigma_n^2} = \frac{x_i^2}{\sigma_n^2}$$

- Comme $x_i^2 = \mathcal{E}_s$, et $\sigma_n^2 = \frac{N_0}{2}$, on a

$$SNR = \frac{2\mathcal{E}_s}{N_0}$$

- Signaux en sortie du modulateur, du canal et du corrélateur :



FILTRE ADAPTE

• On peut remplacer les corrélateurs par des filtres adaptés dont la réponse impulsionnelle est $h_j(t)$. La relation entre $h_j(t)$ et les fonctions de base $f_j(t)$ est la suivante :

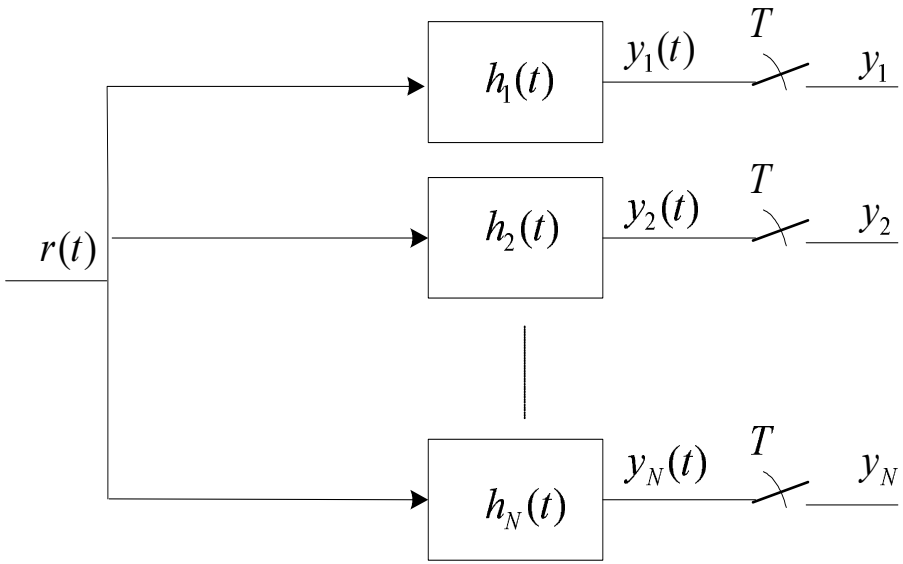
$$h_j(t) = f_j(T-t) \qquad \text{avec} \quad 0 \leq t \leq T \tag{85}$$

• La sortie de chaque filtre s'exprime comme suit :

$$\begin{aligned} y_j(t) &= \int_0^t y(t) h_j(t-\tau) d\tau \\ &= \int_0^t y(t) f_j(T-t+\tau) d\tau \end{aligned} \tag{86}$$

• Si on échantillonne la sortie des filtres à l'instant T, on retrouve la relation précédente $y_j(T) = y_j$

• la sortie des filtres adaptés $h_j(t)$ à l'instant $t = T$ est identique à celle de la structure avec corrélateur.



• On peut démontrer que l'utilisation de filtres adaptés permet de maximiser le rapport signal à bruit et par conséquence de minimiser le taux d'erreurs.

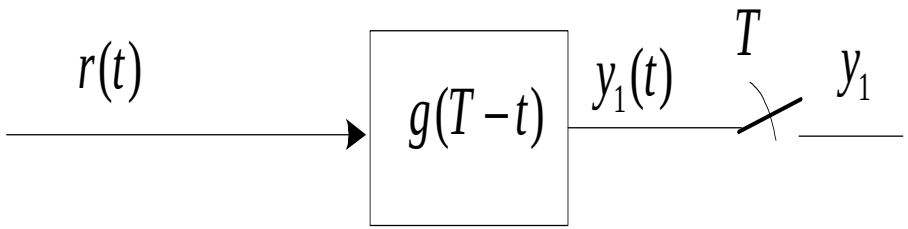
• Le rapport signal à bruit en sortie d'un filtre adapté ne dépend pas de la forme du signal mais de son énergie !

exemple 1(suite) :

• Pour le cas du code en ligne NRZ, la réponse impulsionnelle $h_1(t)$ du filtre adapté est la suivante :

$$h(t) = g(T - t)$$

où $g(t)$ est la réponse impulsionnelle du filtre d'émission.



COURS 6

Calcul du taux d’erreurs

DETECTION OPTIMAL

• L'objectif du détecteur optimal est de déterminer le message qui a été le plus vraisemblablement émis $\hat{\mathbf{x}}$.

• Soit le message $\mathbf{x}$ envoyé dans un canal discret stationnaire sans mémoire de densité de probabilité conditionnelle $p(y/x)$ et $\mathbf{y}$ le vecteur reçu après filtrage adapté.

• D'une manière générale, un détecteur *maximum a posteriori* (MAP) cherche parmi tous les messages possibles $\mathbf{x}$, le message estimé $\hat{\mathbf{x}}$ pour lequel la probabilité conditionnelle $Pr(\mathbf{x}|\mathbf{y})$ est la plus grande.

$$\hat{\mathbf{x}} = \arg\max_{\mathbf{x}} Pr(\mathbf{x}|\mathbf{y}) \tag{87}$$

• En utilisant la loi de Bayes, on peut écrire :

$$Pr(\mathbf{x}|\mathbf{y}) = \frac{Pr(\mathbf{y}|\mathbf{x})Pr(\mathbf{x})}{Pr(\mathbf{y})} \tag{88}$$

• Si tous les messages sont équiprobables, et comme le dénominateur $Pr(\mathbf{y})$ est commun à toutes les messages, le message estimé $\hat{\mathbf{x}}$ est le message pour lequel la probabilité conditionnelle $Pr(\mathbf{y}|\mathbf{x})$ est la plus grande.

$$\hat{\mathbf{x}} = \arg\max_{\mathbf{x}} Pr(\mathbf{y}|\mathbf{x}) \tag{89}$$

• Un détecteur utilisant ce critère est appelé un détecteur à *maximum de vraisemblance* (*maximum likelihood* en anglais ou ML).

• un détecteur ML calcule les distances euclidiennes entre l'échantillon reçu et les échantillons correspondant à toutes les séquences possibles.

• Messages équiprobables $\Rrightarrow$ détecteur MAP = détecteur ML.

CALCUL DU TAUX D'ERREURS BINAIRES POUR UN CODE NRZ SUR CANAL BBAG

• Modèle équivalent (après filtrage adapté) :

$$y = \sqrt{E_s} x_i + n$$

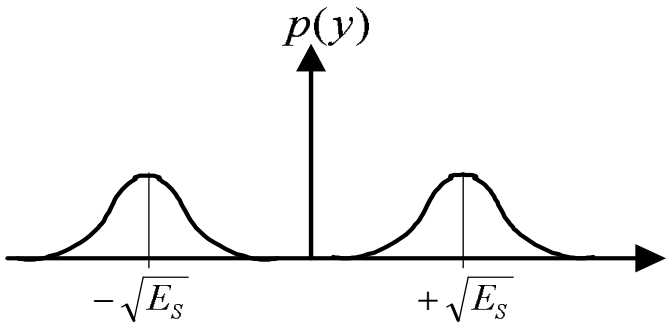
où $x_i = \pm 1$

n est une variable aléatoire gaussienne centrée de variance $\sigma^2 = \frac{N_0}{2}$

• Le détecteur ML réalise simplement l'opération de décision suivante :

$$\hat{c} = \begin{cases} 0 & \text{si} \quad y \leq s \\ 1 & \text{si} \quad y > s \end{cases} \tag{90}$$

• La densité de probabilité $p(y)$ a alors la forme suivante :



• Le seuil de décision est donc placé exactement entre $+\sqrt{E_S}$ et $-\sqrt{E_S}$: $s = 0$ (cas $Pr(x_i = +1) = Pr(x_i = -1) = 1/2$.

• La probabilité que l'amplitude de l'échantillon reçu y soit inférieure au seuil de décision $s = 0$ sachant que $c = 1$ (et donc $x_i = +1)$ est égale à :

$$p(y < 0|x_i = +1) = \frac{1}{\sqrt{2\pi\sigma^2}} \int_{-\infty}^{0} \exp\left\{ -\frac{(y - \sqrt{E_S})^2}{2\sigma^2} \right\} dy \tag{91}$$

• La probabilité que l'amplitude de l'échantillon y soit supérieure au seuil de décision sachant que $c = 0$ (et donc $x_i = -1)$ est égale à :

$$p(y > 0|x_i = -1) = \frac{1}{\sqrt{2\pi\sigma^2}} \int_{0}^{+\infty} \exp\left\{ -\frac{(y + \sqrt{E_S})^2}{2\sigma^2} \right\} dy \tag{92}$$

• Le taux d'erreurs bit (TEB) est alors :

$$\begin{aligned} TEB &= \frac{1}{2} p(y < 0|x_i = +1) + \frac{1}{2} p(y > 0|x_i = -1) \\ &= \frac{1}{\sqrt{2\pi\sigma^2}} \int_{0}^{+\infty} \exp\left\{ -\frac{(y + \sqrt{E_S})^2}{2\sigma^2} \right\} dy \end{aligned} \tag{93}$$

• Faisons un changement de variable $z = \frac{y+\sqrt{E_S}}{\sqrt{2}\sigma}$, $dz = \frac{dy}{\sqrt{2}\sigma}$. On obtient alors :

$$\begin{aligned} TEB &= \frac{1}{\sqrt{2\pi\sigma^2}} \int_0^{+\infty} \exp\left\{ -\frac{(y+\sqrt{E_S})^2}{2\sigma^2} \right\} dy \\ &= \frac{1}{\sqrt{\pi}} \int_{\sqrt{E_S}/\sqrt{2}\sigma}^{+\infty} \exp(-z^2) dz \\ &= \frac{1}{2}\text{erfc}\left\{ \sqrt{\frac{E_S}{N_0}} \right\} \end{aligned}$$

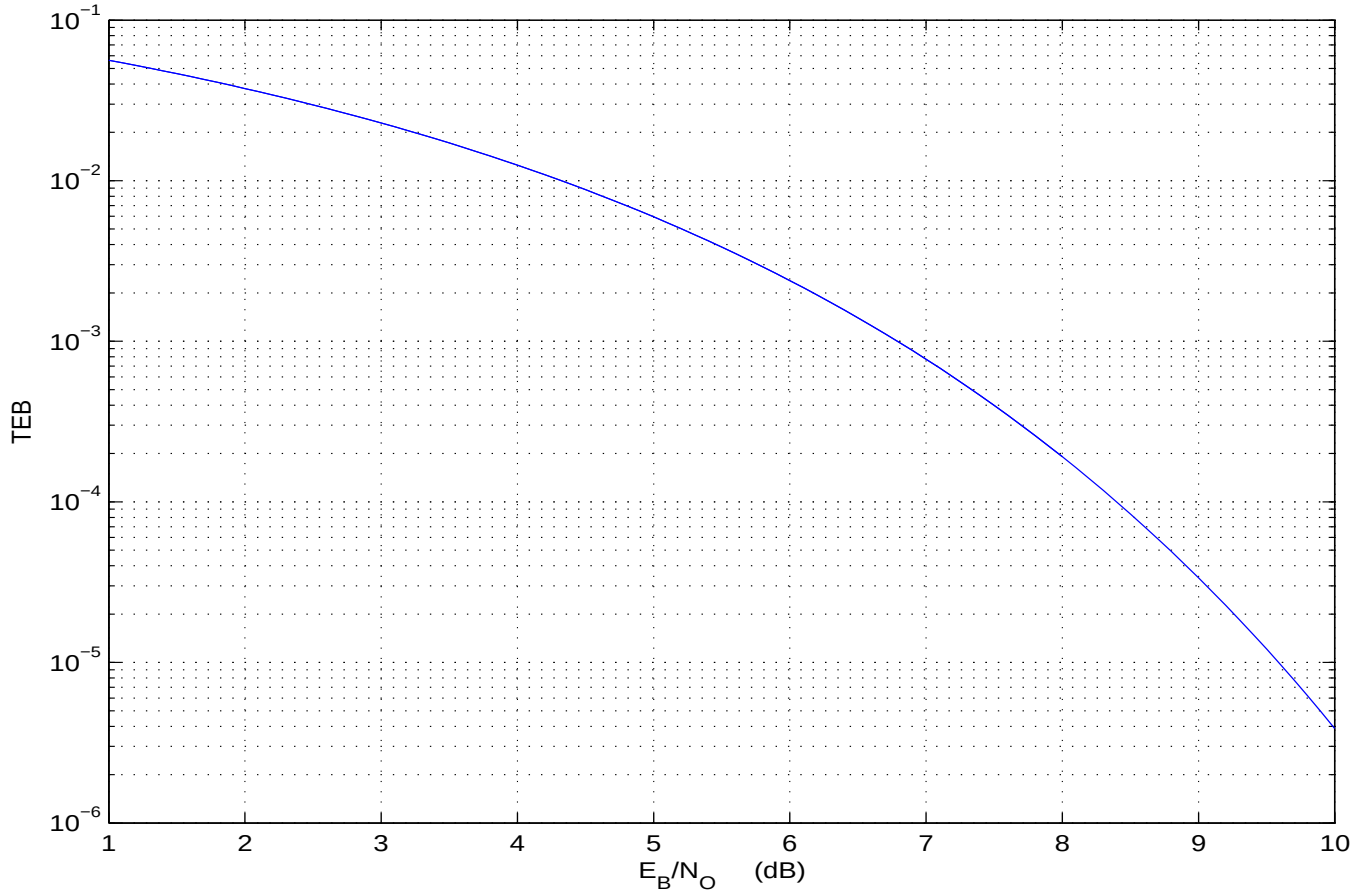
où la fonction erfc (erf complémentaire) est définie comme suit :

$$\text{erfc}(a) = 1 - \text{erf}(a) = \frac{2}{\sqrt{\pi}} \int_a^{+\infty} \exp(-z^2) dz \tag{94}$$

• Comme ici $E_S = E_B$, on a

$$TEB = \frac{1}{2}\text{erfc}\left\{ \sqrt{\frac{E_B}{N_0}} \right\}$$

- La courbe $TEB = f(E_b/N_0)$ pour un code NRZ sur canal BBAG est la suivante :



PROBABILITE D'ERREURS PAR PAIRE

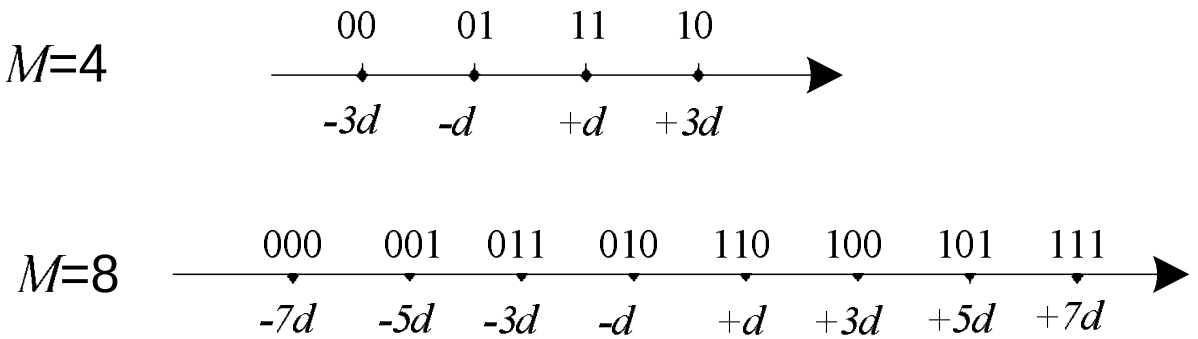
• Soient deux mots $\mathbf{x}_i$ et $\mathbf{x}_j$ dont la distance euclidienne est $d(\mathbf{x}_i, \mathbf{x}_j)$. Pour un canal BBAG, la probabilité $Pr(\mathbf{x}_j|\mathbf{x}_i)$ que **y** soit plus près de $\mathbf{x}_j$ que de $\mathbf{x}_i$ est donnée par :

$$Pr(\mathbf{x}_j|\mathbf{x}_i) = \frac{1}{2}\text{erfc}\left(\frac{d(\mathbf{x}_i, \mathbf{x}_j)}{2\sqrt{N_0}}\right) \tag{95}$$

• Dans le cas du code NRZ, le mot $\mathbf{x}_i$ est l'échantillon x_i. La distance euclidienne est égale à $2\sqrt{E_b}$ et on retrouve bien l'expression du TEB précédente.

CALCUL DU TES POUR UN CODE NRZ M-AIRE SUR CANAL BBAG

- Alphabet des symboles =$\{\pm d, \pm 3d, \ldots, \pm(M-1)d\}$.



L'amplitude des symboles est alors :

$$A_m = (2m - 1 - M)d \qquad \text{avec} \qquad m = 1, 2, \ldots, M$$

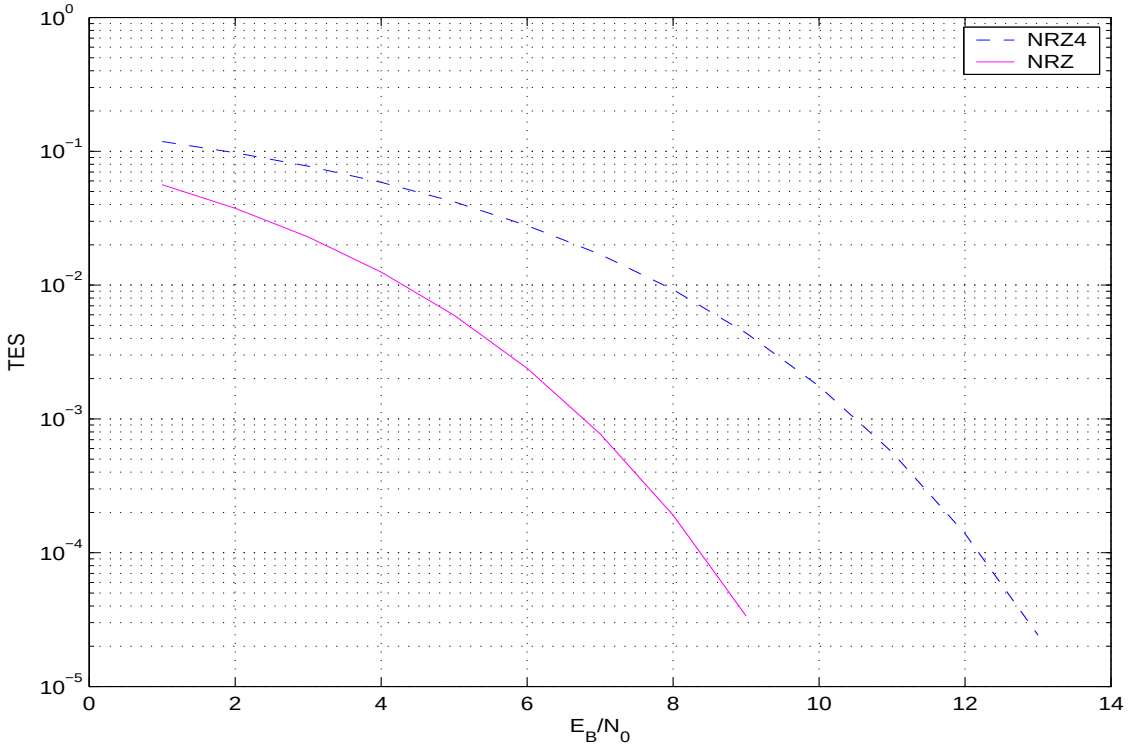
L'énergie moyenne par symbole est égale à :

$$E_s = \frac{1}{M}\sum_{m=1}^{M}(A_m)^2$$
$$= d^2\frac{M^2-1}{3}$$

• Calculons une approximation du TES en ne tenant compte que des cas où l'erreur symbole provient d'une décision en faveur du symbole adjacent.

• On a $2(M-1)$ paires de points adjacents et la distance euclidienne entre ces points est égale à $2d$.

$$TES = \frac{M-1}{M} erfc\left(\frac{2d}{2\sqrt{N_0}}\right) = \frac{M-1}{M} erfc\left(\sqrt{\frac{3log_2 M}{M^2-1}\frac{E_b}{N_0}}\right)$$



COURS 7

Critère de Nyquist

CRITERE DE NYQUIST

• Si la bande du signal émis est limitée, la forme de l'impulsion élémentaire $g(t)$ est de durée infinie

• Comment choisir $g(t)$ pour pouvoir reconstituer parfaitement à la réception les échantillons émis à la cadence symbole de $1/T$?

• Le signal après filtrage adapté s'écrit :

$$\begin{aligned} y(t) &= x(t) * c(t) * h(t) + n(t) \\ &= \sum_{k=-\infty}^{\infty} a_k p(t - kT) + n(t) \end{aligned} \tag{96}$$

avec $p(t) = g(t) * c(t) * h(t)$

• On échantillonne $y(t)$ aux instants $t = kT + \tau$ (τ est un délai permettant d'ajuster l'instant optimal d'échantillonnage)

CRITERE DE NYQUIST

$$y(kT) = a_k + \sum_{k \neq i}^{+\infty} a_i p((k-i)T) + n(kT) \tag{97}$$

• Cette expression est composée de 3 termes :

$\rightarrow$ Le premier terme est le k^{ieme} symbole transmis

$\rightarrow$ Le second terme est la contribution de tous les autres symboles transmis sur l'échantillon $y(kT)$. Ce terme est appelé l'interférence intersymbole.

$\rightarrow$ Le troisième terme est la contribution du bruit

• Comme le second et le troisième terme diminuent les performances du système de transmission, nous devrons choisir les filtres d'émission et de réception afin de les minimiser.

CRITERE DE NYQUIST

• Pour garantir l'absence d'interférence intersymbole, on doit avoir la condition suivante sur la forme d'onde $p(t)$:

$$p(kT) = \begin{cases} 1 & \text{pour} \quad k = 0 \\ 0 & \text{pour} \quad k \neq 0 \end{cases} \tag{98}$$

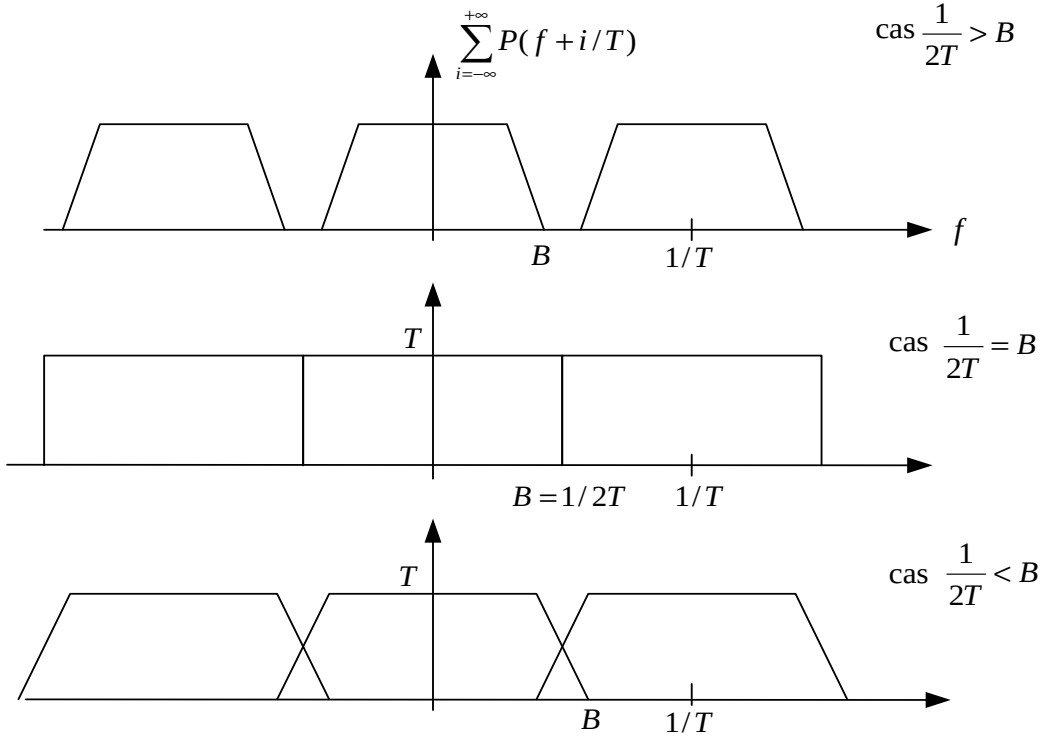
• Avec cette condition, on a bien $y(kT) = a_k + n(kT)$

• Dans le domaine fréquentiel, la condition devient :

$$\sum_{i=-\infty}^{+\infty} P\left(f + \frac{i}{T}\right) = T \tag{99}$$

• Quelle est la faisabilité de réaliser le critère selon la largeur de bande B du canal de transmission ($C(f) = 0$ pour f > B)?

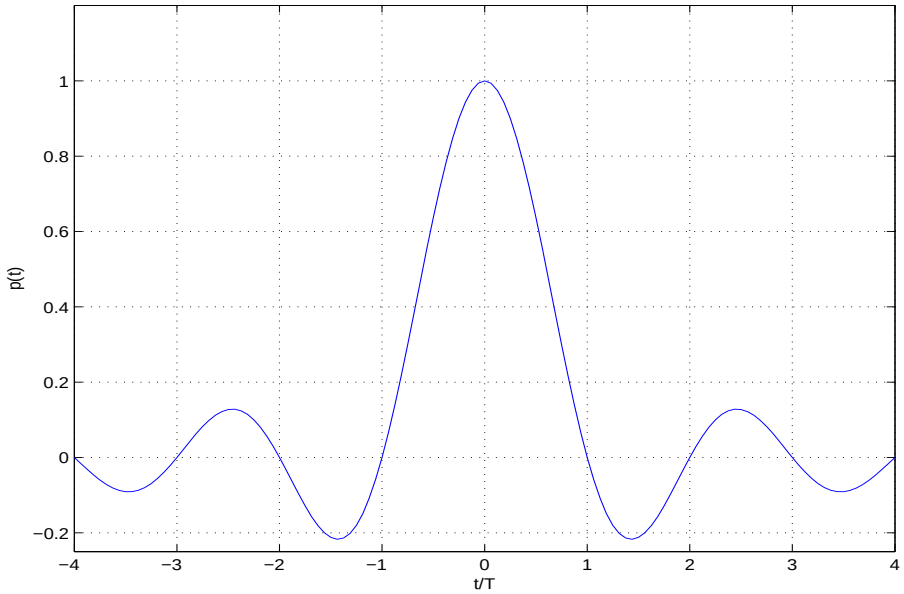
CRITERE DE NYQUIST



- Si $\frac{1}{2T} > B$, alors il n'existe pas de filtre $G(f)$ et $H(f)$ satisfaisant au critère de Nyquist

- Si $\frac{1}{2T} = B$, il existe une solution possible : le filtre passe-bas parfait de fréquence de coupure B

$$P(f) = \begin{cases} T & \text{si} \quad |f| < B = \frac{1}{2T} \\ 0 & \text{sinon} \end{cases} \qquad p(t) = \frac{\sin(\pi t/T)}{\pi t/T}$$



- Si $\frac{1}{2T} < B$, alors il existe une famille de filtre $G(f)$ et $H(f)$ tel que $P(f) = G(f)C(f)H(f)$ répond au critère de Nyquist. Les filtres appartenant à cette famille doivent satisfaire la condition suivante :

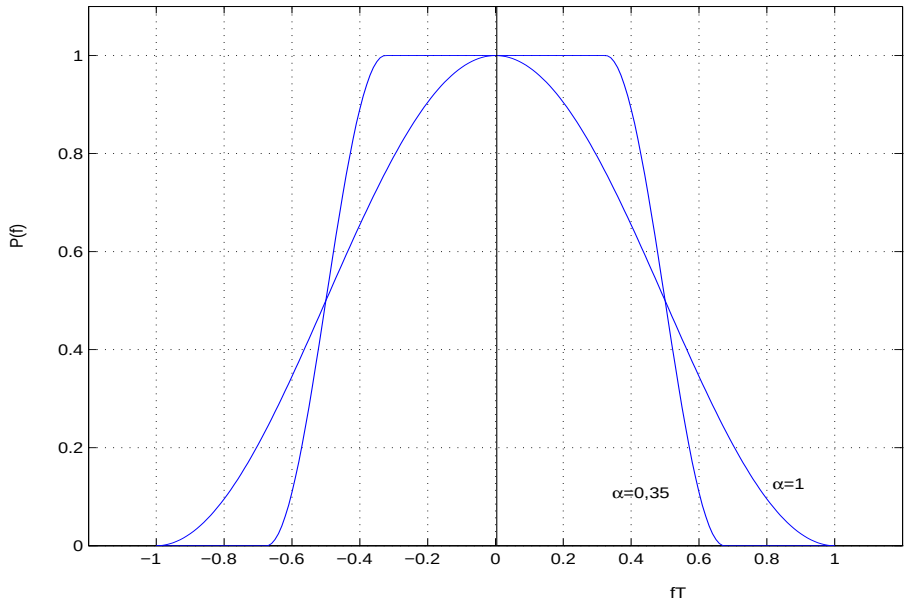
$$P(f) + P\left(f - \frac{1}{T}\right) = T \tag{100}$$

FILTRE EN COSINUS SURELEVE

- Fonction de transfert satisfait au critère

$$P(f) = \begin{cases} T & \text{si} \quad 0 \leq |f| \leq \frac{1-\alpha}{2T} \\ T\cos^2\left\{\frac{\pi}{4\alpha}(2fT-(1-\alpha))\right\} & \text{si} \quad \frac{1-\alpha}{2T} < |f| < \frac{1+\alpha}{2T} \\ 0 & \text{si} \quad |f| \geq \frac{1+\alpha}{2T} \end{cases} \tag{101}$$

α est le facteur d'arrondi (*roll-off* en anglais) et est compris entre 0 et 1 (en pratique cette valeur est choisie entre 0.22 et 0.35.



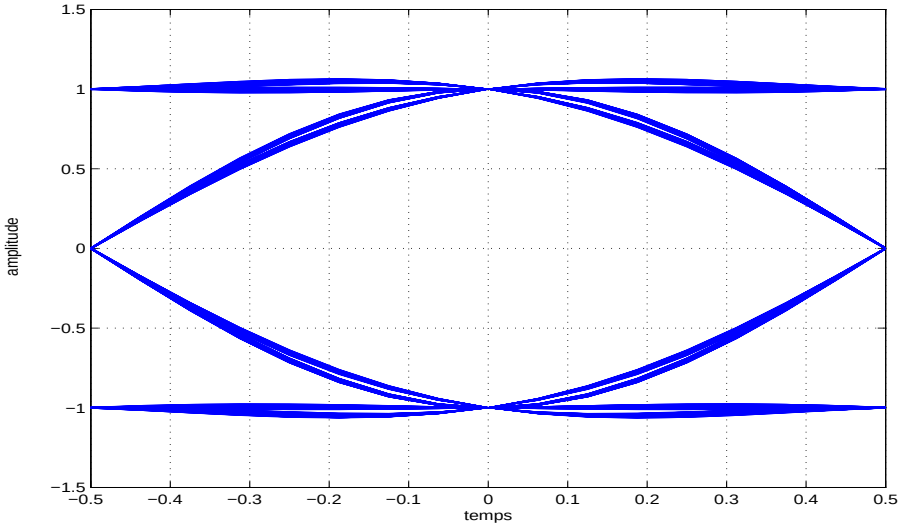
FILTRE EN RACINE DE COSINUS SURELEVE

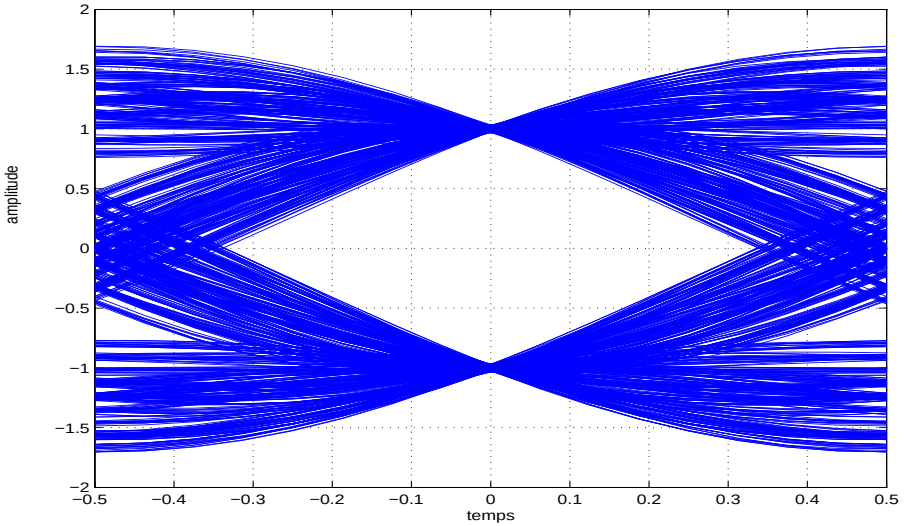
• Lorsque le canal de transmission est un canal BBAG, on a $P(f) = G(f)H(f)$ afin de maximiser le rapport signal à bruit à la réception on scinde le filtre en cosinus surélevé en deux filtres identiques :

$$G(f) = H(f) = \begin{cases} \sqrt{T} & \text{si} \quad 0 \leq |f| \leq \frac{1-\alpha}{2T} \\ \sqrt{T}\cos\left\{\frac{\pi}{4\alpha}(2fT-(1-\alpha))\right\} & \text{si} \quad \frac{1-\alpha}{2T} < |f| < \frac{1+\alpha}{2T} \\ 0 & \text{si} \quad |f| \geq \frac{1+\alpha}{2T} \end{cases} \tag{102}$$

DIAGRAMME DE L'OEIL

- Permet de visualiser les interférences intersymboles
- Il consiste à superposer toutes les sections de durée T du signal $y(t)$.
- Exemple pour un signal NRZ filtré par un filtre en cosinus surélevé $\alpha = 1$ et $\alpha = 0,35$.

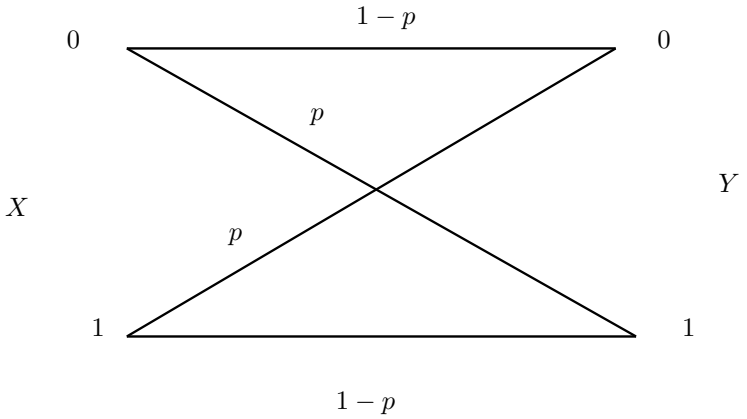




COURS 8

Capacité d’un canal de transmission

CANAL BINAIRE SYMETRIQUE



- **Ce canal est caractérisé par la probabilité de transition**:

$$P(Y=0|X=1) = P(Y=1|X=0) = p$$
$$P(Y=0|X=0) = P(Y=1|X=1) = 1-p \qquad (103)$$

- **Le canal binaire symétrique est sans mémoire** :

  Soit $\mathbf{x}$ et $\mathbf{y}$ respectivement les séquences d'entrée et de sortie composées de n bits du canal : $\mathbf{x} = [x_1, x_2, \ldots, x_n]$, et $\mathbf{y} = [y_1, y_2, \ldots, y_n]$.

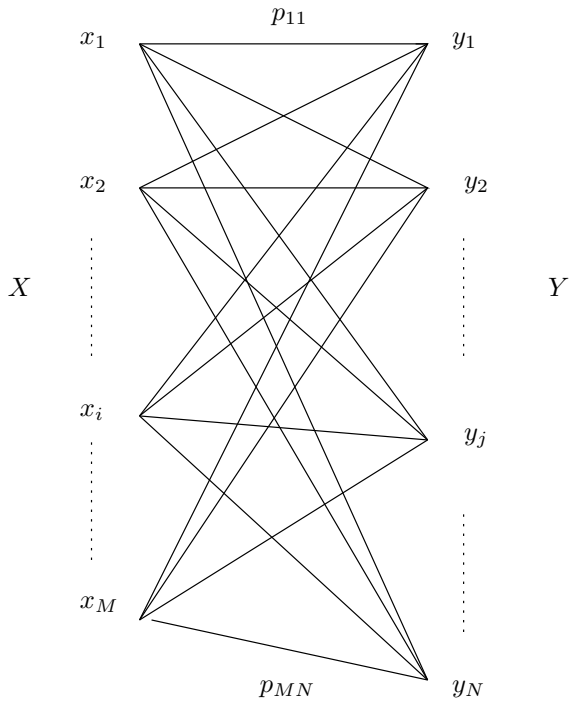
  La relation suivante justifie l'absence de mémoire dans le canal :

$$P(Y_1 = y_1, \ldots, Y_n = y_n | X_1 = x_1, \ldots, X_n = x_n) = \prod_{i=1}^{n} P(Y = y_i | X = x_i)$$

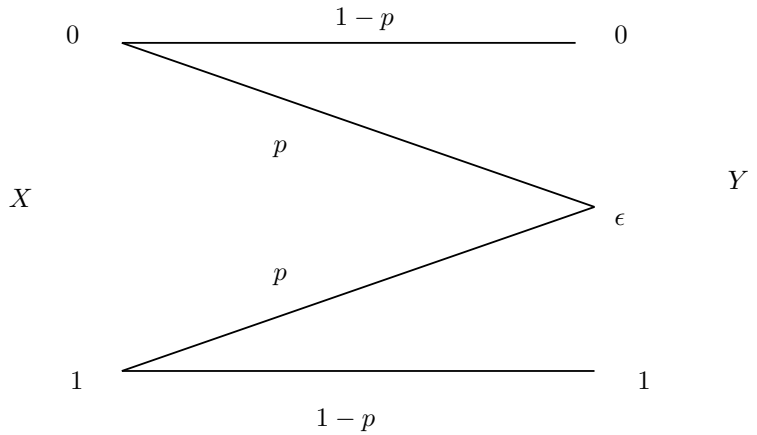
  La probilité conditionnelle jointe est le produit des n probabilités conditionnelles $P(Y = y_i | X = x_i)$.

CANAL DISCRET SANS MEMOIRE

- Les symboles en entrée de ce canal sont M-aire
- Les symboles en sortie sont N-aire.
- les probabilités conditionnelles sont de la forme $P(Y = y_j|X = x_i) \equiv p_{ij}$.



CANAL A EFFACEMENT



- **Ce canal sans mémoire est caractérisé par la probabilité d'effacement**:

$$P(Y=\epsilon|X=1)=P(Y=\epsilon|X=0)=p$$
$$P(Y=0|X=0)=P(Y=1|X=1)=1-p \qquad (104)$$

CANAL ADDITIF A BRUIT BLANC GAUSSIEN

• Modèle équivalent (après filtrage adapté et échantillonnage) :

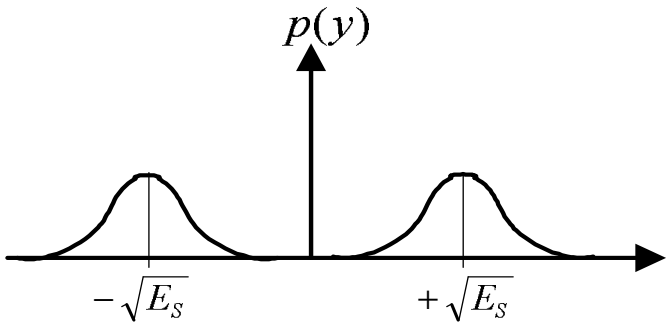
$$y_i = x_i + n_i$$

où $x_i = \pm\sqrt{E_s}$

n_i est une variable aléatoire gaussienne centrée de variance $\sigma^2 = \frac{N_0}{2}$

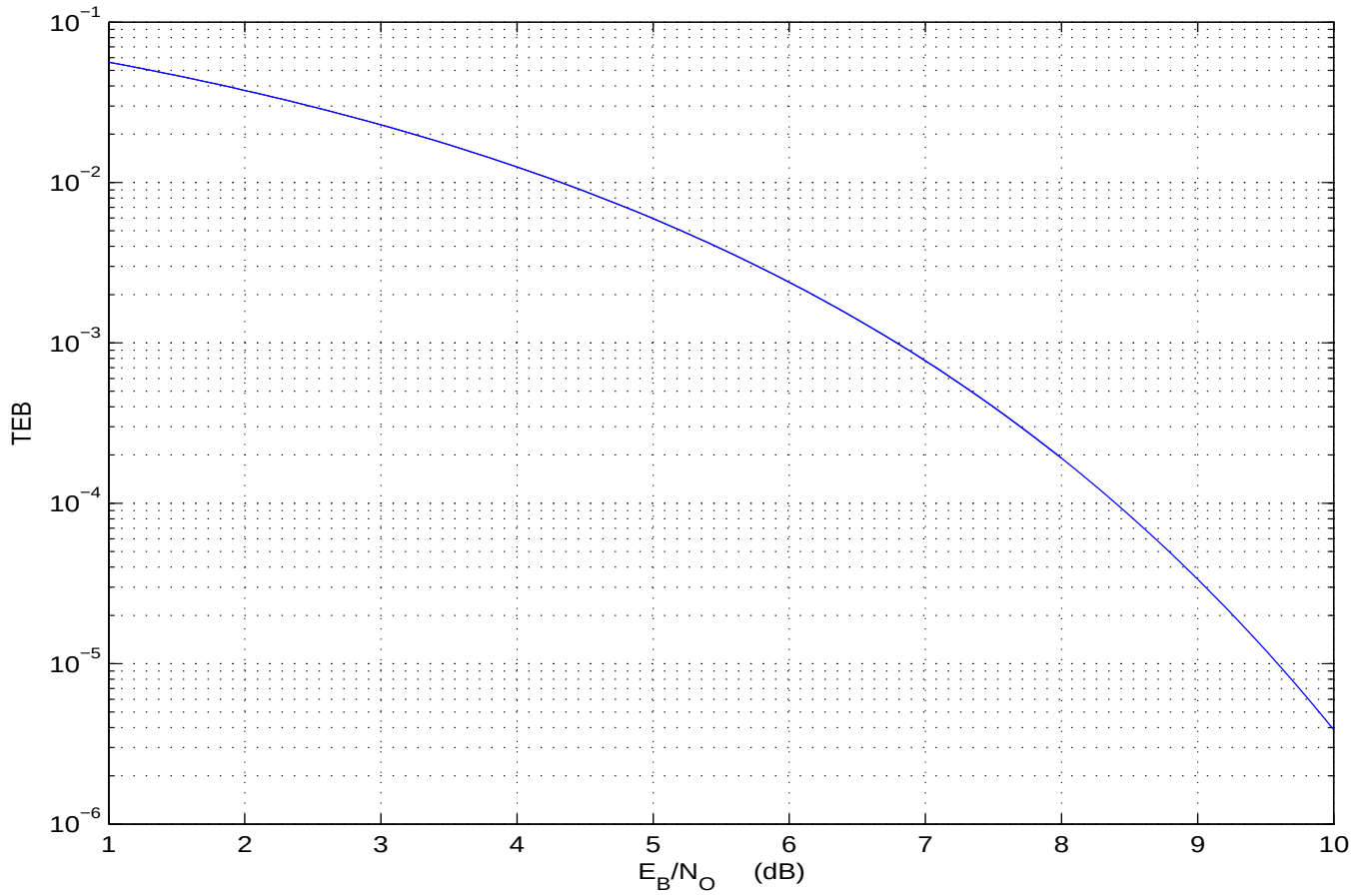
• Le détecteur ML réalise simplement l'opération de seuillage.

• La densité de probabilité $p(y)$ a la forme suivante :



$$TEB = \frac{1}{2}\text{erfc}\left\{\sqrt{\frac{E_B}{N_0}}\right\} \qquad \text{avec} \qquad \text{erfc}(a) = \frac{2}{\sqrt{\pi}}\int_a^{+\infty}\exp(-z^2)dz$$

- La courbe $TEB = f(E_b/N_0)$ associée :



CAPACITE D'UN CANAL DE TRANSMISSION

définition 1 : La capacité d'un canal de transmission est la quantité d'information moyenne maximale dont le canal peut assurer le transfert.

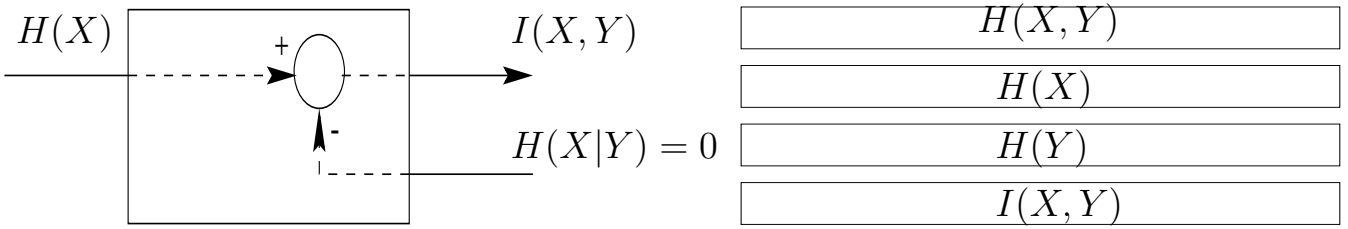
- C s'exprime en Shannon/symbole
- C' capacité par unité de temps $C' = C \times D_s$
- En absence de bruit $C = H_{MAX}(X) = \log_2 Q$
- En présence de bruit $C < H_{MAX}(X)$

définition 2 : La capacité d'un canal de transmission est le maximum de l'information mutuelle moyenne.

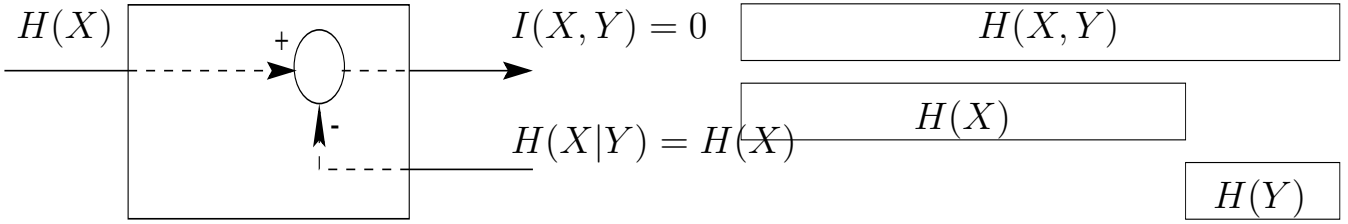
$$C = \max I(X,Y) \quad \text{avec} \quad I(X,Y) = H(X) - H(X|Y) \tag{105}$$

- La maximisation est réalisée sur l'ensemble de toutes les sources possibles.
- $H(X|Y)$ correspond à la quantité d'information moyenne perdue dans le canal.

Lorsque le canal est sans bruit, on a $H(X|Y) = 0$ et $C = H_{MAX}(X)$.

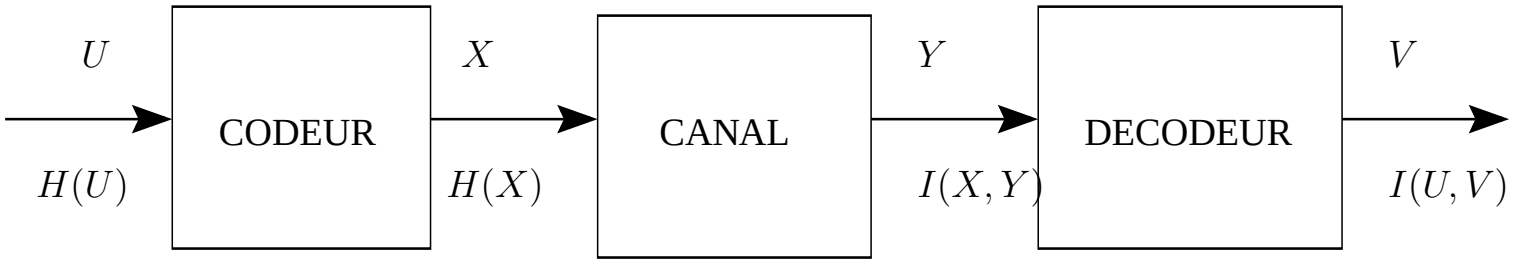


Lorsque X et Y sont indépendants, on a $H(X|Y) = H(X)$ et $C = 0$.



PRINCIPE DU CODAGE DE CANAL

- Si $H(X|Y)$ est non négligeable (cas du canal bruyant), il ne sera pas possible d'effectuer une communication sans erreur en reliant directement la source au canal de transmission.
- **Solution** : Il faut placer un élément appelé codeur de canal entre la source et le canal de transmission.



- On peut définir une nouvelle information mutuelle moyenne $I(U,V) = H(U) - H(U|V)$.
- On peut alors rendre la quantité d'information moyenne $H(U|V)$ aussi faible que souhaité. Il est alors possible de transmettre au travers de ce canal bruité une quantité d'information moyenne $H(U)$
- $H(U) < H(X)$ à cause de la redondance apportée par le codage de canal.

THEOREME FONDAMENTAL DU CODAGE DE CANAL

théorème : Il existe un codage de canal permettant de garantir une communication avec un taux d'erreurs aussi faible que souhaité à la condition que la quantité d'information moyenne entrant dans l'ensemble codeur-canal-décodeur soit inférieure à la capacité C du canal :

$$H(U) < C \quad \text{en Sh/symb} \tag{106}$$

Multiplions $H(U)$ et C par le débit symbole D_S de la source

$$H(U) \times D_S < C \times D_S \tag{107}$$

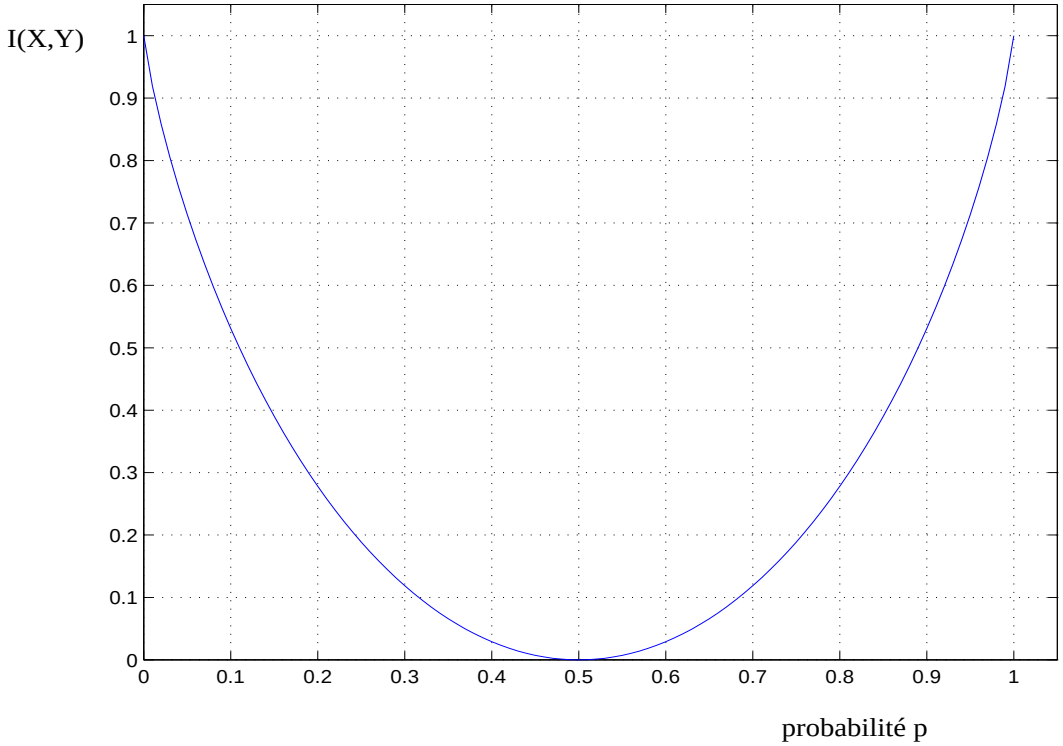
$$D_I < C' \quad \text{en Sh/sec} \tag{108}$$

La démonstration est basée sur les séquences typiques

CAPACITE D'UN CANAL BSC

Pour $q = 1/2$:

$$I(X,Y) = 1 + p\log_2(p) + (1-p)\log_2(1-p) \tag{109}$$



CAPACITE D'UN CANAL BBAG

On rappelle la relation entre le vecteur émis $\mathbf{x}$ et le vecteur reçu $\mathbf{y}$ de dimension D

$$\mathbf{y} = \mathbf{x} + \mathbf{n} \tag{110}$$

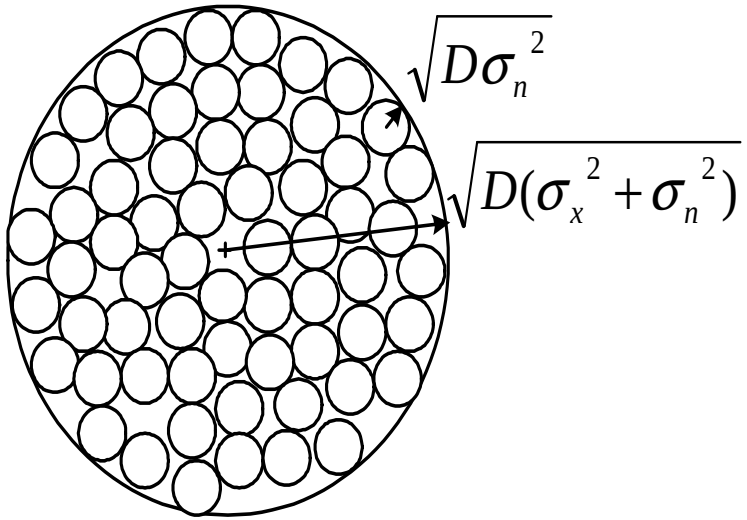
Soit $\mathbf{n} = (n_1, n_2, \ldots, n_D)$ le vecteur bruit composé de D composantes indépendantes gaussiennes de variance σ_n^2. La densité de probabilité du vecteur $\mathbf{n}$ s'exprime comme suit:

$$p(\mathbf{n}) = \frac{1}{(2\pi\sigma_n^2)^{D/2}} \exp\left(-\frac{\sum_{i=1}^{D} n_i^2}{2\sigma_n^2}\right) \tag{111}$$

Pour D tendant vers l'infini, on montre que la norme du vecteur de bruit $\mathbf{n}$ est concentrée à la surface de la sphère à D dimensions de rayon $\sqrt{D\sigma_n^2}$

Pour la même raison, la norme du vecteur $\mathbf{x}$ est concentrée à la surface de la sphère de rayon $\sqrt{D\sigma_x^2}$

Le vecteur $\mathbf{y}$ se trouve à la surface de la sphère à D dimension de rayon $\sqrt{D(\sigma_x^2 + \sigma_n^2)}$.



Soit M le nombre de vecteurs $\mathbf{x}$ distinguables.

Pour réaliser une transmission sans erreur, le volume des M sphères de bruit doit être inférieur au volume de la sphère de rayon $\sqrt{D(\sigma_x^2+\sigma_n^2)}$:

$$M \times V(\sqrt{D\sigma_n^2}, D) \leq V(\sqrt{D(\sigma_x^2+\sigma_n^2)}, D) \tag{112}$$

Le volume d'une sphère à D dimension et de rayon r est donné par

$$V(r, D) = \frac{\pi^{D/2}}{\Gamma(D/2+1)} r^D \tag{113}$$

On obtient l'inégalité suivante :

$$\begin{aligned} M &\leq \frac{V(\sqrt{D(\sigma_x^2+\sigma_n^2)}, D)}{V(\sqrt{D.\sigma_n^2}, D)} \\ &\leq \frac{(D(\sigma_x^2+\sigma_n^2))^{D/2}}{(D.\sigma_n^2)^{D/2}} \\ &\leq \left(\frac{\sigma_x^2+\sigma_n^2}{\sigma_n^2}\right)^{D/2} \end{aligned} \tag{114}$$

$$H(U) = \frac{1}{D}\log_2 M \leq C \tag{115}$$

Finalement on a :

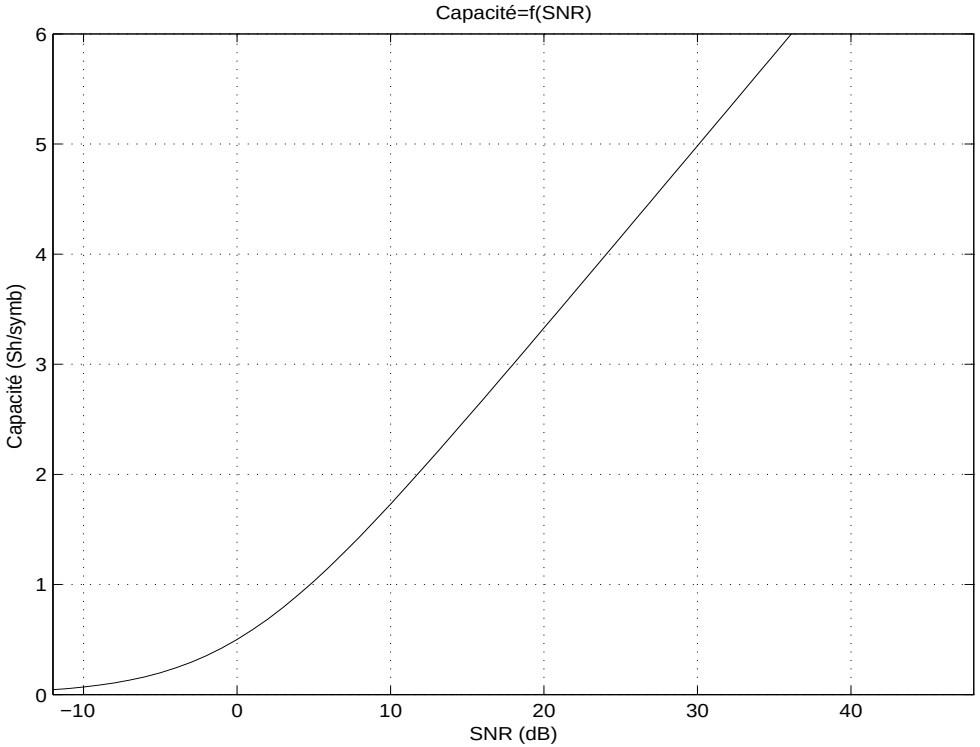
$$C = \frac{1}{2}\log_2\left(1+\frac{\sigma_x^2}{\sigma_n^2}\right) \tag{116}$$

Pour une bande passante B, la dimension D est égale à $D = 2BT$ (T durée de la transmission). La puissance du bruit est égale à $N = 2B\sigma_n^2$ et la puissance moyenne du signal X est égale à $P = 2B\sigma_x^2$.

$$C = \frac{1}{2}\log_2\left(1 + \frac{P}{N}\right) \qquad \text{Sh/dim} \tag{117}$$

$$C' = B\log_2\left(1 + \frac{P}{N}\right) \qquad \text{Sh/s} \tag{118}$$

CAPACITE D'UN CANAL BBAG



EFFICACITE SPECTRALE

Soit $E_b = PT_b$ l'énergie moyenne par bit d'information.

On a la relation suivante entre le rapport signal à bruit P/N et le rapport E_b/N_0:

$$\frac{P}{N} = \frac{E_b}{N_0 B T_b} = \eta \frac{E_b}{N_0} \tag{119}$$

η est l'efficacité spectrale en bits/sec/Hz :

$$\eta = \frac{1}{BT_b} = \frac{D_b}{B} \quad \text{avec} \quad D_b = \frac{1}{T_b} \quad \text{débit binaire d'information} \tag{120}$$

η est maximale lorsque la bande passante est minimale soit $B_{min} = 1/T_s$, on a :

$$\eta_{max} = \frac{1}{T_b B_{min}} = \frac{T_s}{T_b} \tag{121}$$

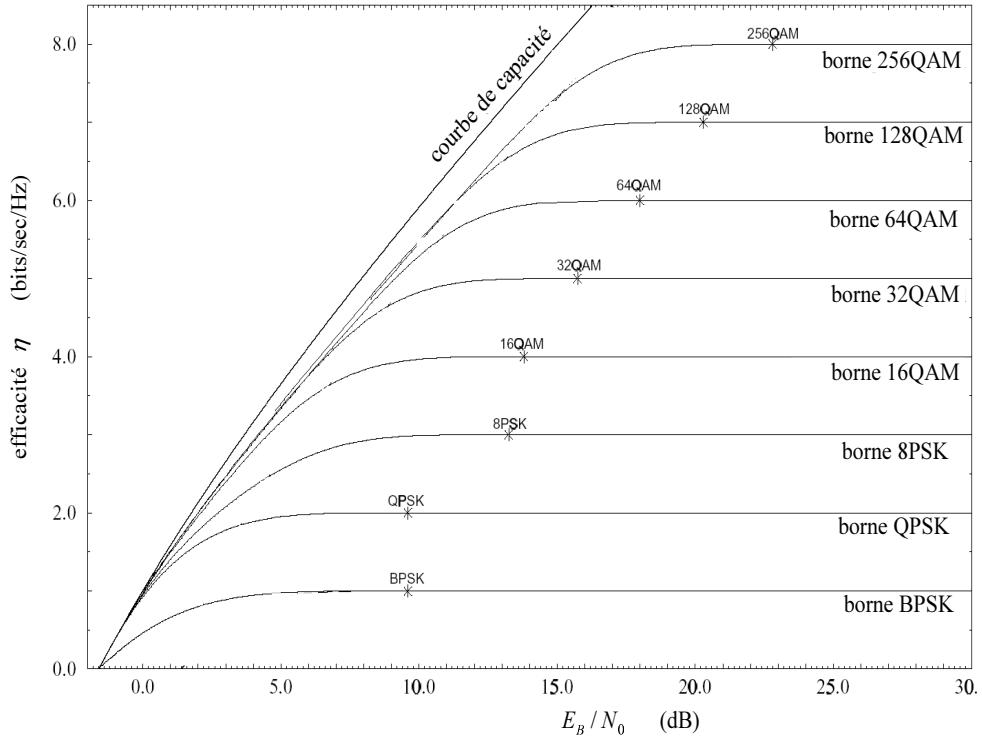
En considérant que $D_b = C'$, on obtient :

$$\eta_{max} = \frac{C'}{B_{min}} = \log_2\left(1 + \eta_{max}\frac{E_b}{N_0}\right) \qquad \text{en bits/sec/Hz} \tag{122}$$

EFFICACITE SPECTRALE (suite)

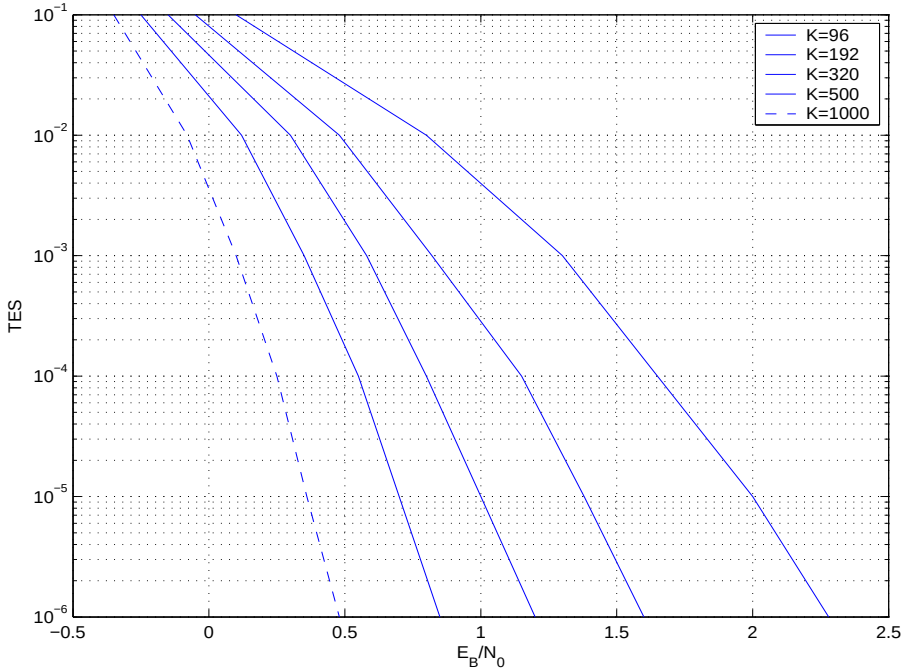
$$\frac{E_b}{N_0} = \frac{2^{\eta_{max}} - 1}{\eta_{max}} \tag{123}$$

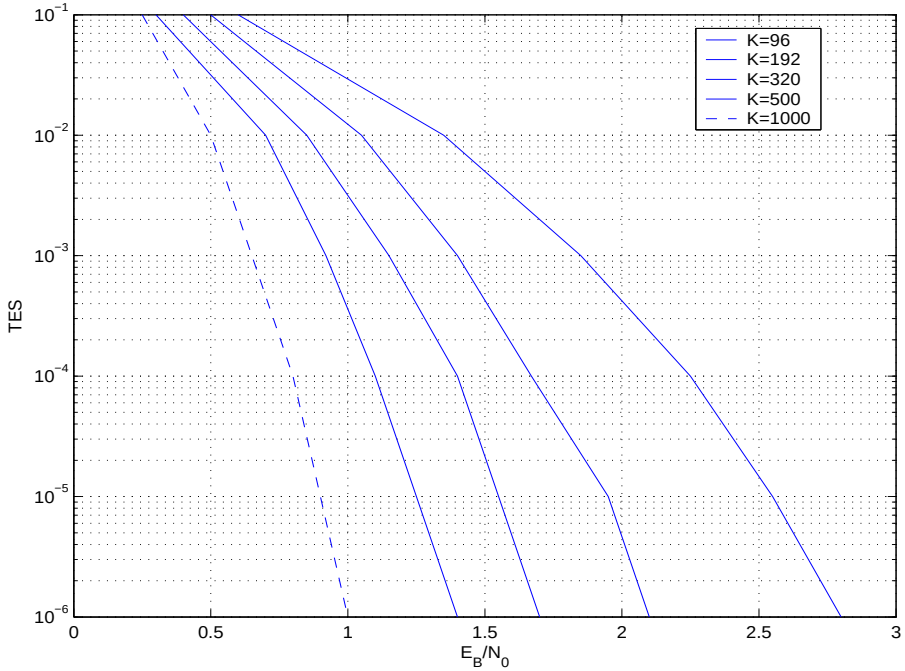
$$\lim_{\eta_{max} \to 0} \frac{E_b}{N_0} = \ln 2 \quad \text{soit} \quad \left.\frac{E_b}{N_0}\right|_{dB} = -1.59 \quad \text{dB} \tag{124}$$



BORNE PAR EMPILEMENT DE SPHERE

- borne par empilement de sphere pour $R = 1/2$ et $R = 1/3$.





COURS 9

Introduction codes en bloc

CODAGE DE CANAL

• L'objectif du codage de canal est de protéger les données issues du codage de source contre les erreurs de transmission.

• Ces erreurs peuvent être aléatoires ou se produire par paquet

• Au lieu d'utiliser un codage aléatoire, nous utiliserons des codes possédant une structure algébrique comme par exemple la linéarité et rendant ainsi les opérations de codage et de décodage plus simples à effectuer.

On distingue trois grandes familles de codes correcteurs d'erreurs

■ **les codes en bloc linéaires**

■ **les codes convolutifs**

■ **les codes concaténés**

• code en bloc + code convolutif

• plusieurs codes en bloc (code produit,LDPC,...)

• plusieurs codes convolutifs (turbo codes, ...)

RAPPEL SUR LES CORPS

Définition

Un corps F est un ensemble non vide muni de deux lois de composition internes, l'addition et la multiplication et satisfaisant :

- F est un groupe commutatif par rapport à l'addition (associativité, élément neutre noté 0, symétrique, commutativité)

- la multiplication est associative, commutative, distributive à droite et à gauche par rapport à l'addition
- le corps contient un élément neutre noté 1 pour la multiplication
- tout élément de F non nul est inversible

Les corps de Galois

Un corps de Galois $GF(q)$ est un corps fini possédant q éléments. q est soit un nombre premier ou soit de la forme $q = p^m$ avec p nombre premier.

exemple 1 : $GF(2)$

+	0	1
0	0	1
1	1	0

*	0	1
0	0	0
1	0	1

exemple 2 : $GF(5)$

+	0	1	2	3	4
0	0	1	2	3	4
1	1	2	3	4	0
2	2	3	4	0	1
3	3	4	0	1	2
4	4	0	1	2	3

*	0	1	2	3	4
0	0	0	0	0	0
1	0	1	2	3	4
2	0	2	4	1	3
3	0	3	1	4	2
4	0	4	3	2	1

CODES EN BLOC LINEAIRES

Soit $\mathbf{u} = [u_1, u_2, \ldots, u_K]$ un mot d'information composé de K bits d'information
Soit $\mathbf{x} = [x_1, x_2, \ldots, x_N]$ le mot de code associé composé de N bits.
On a la relation matricielle suivante entre **u** et **x**:

$$\mathbf{x} = \mathbf{u}\mathbf{G} \tag{125}$$

$\mathbf{G}$ est la matrice génératrice du codeur de dimension $K \times N$.

$$\mathbf{G} = \begin{pmatrix} \mathbf{g}_1 \\ \mathbf{g}_2 \\ \vdots \\ \mathbf{g}_K \end{pmatrix} = \begin{pmatrix} g_{11} & g_{12} & \cdots & g_{1N} \\ g_{21} & g_{22} & \cdots & g_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ g_{K1} & g_{K2} & \cdots & g_{KN} \end{pmatrix} \tag{126}$$

Chaque mot de code est une combinaison linéaire des vecteurs $\mathbf{g}_i$ de $\mathbf{G}$. Ainsi donc, un code en bloc linéaire peut être défini comme un sous espace vectoriel à $K < N$ dimensions construit suivant (126).

Il est toujours possible en combinant les lignes entre elles de mettre la matrice génératrice G sous la forme systématique suivante :

$$\mathbf{G} = [\mathbf{I_K} \quad \mathbf{P}] = \begin{pmatrix} 1 & 0 & 0 & \dots & 0 & p_{11} & p_{12} & \dots & p_{1N-K} \\ 0 & 1 & 0 & \dots & 0 & p_{21} & p_{22} & \dots & p_{2N-K} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 & p_{K1} & p_{K2} & \dots & p_{KN-K} \end{pmatrix} \tag{127}$$

PROPRIETES ET DEFINITIONS

- **rendement** : le rendement R d'un code en bloc (N, K) est égal à

$$R = \frac{K}{N} \tag{128}$$

- **linéarité** : soit $\mathbf{x}_1$ et $\mathbf{x}_2$ deux mots de code du code $\mathcal{C}$, et α_1 et α_2 deux éléments du corps fini. La linéarité implique que $\alpha_1\mathbf{x}_1 + \alpha_2\mathbf{x}_2$ est aussi un mot de code de $\mathcal{C}$. Par conséquence, le mot $\mathbf{x}_0 = [00\ldots0]$ est toujours un mot de code d'un code linéaire. On appellera ce mot de code le mot de code nul.

- **distance de Hamming** : soit $\mathbf{x}_1$ et $\mathbf{x}_2$ deux mots de code du code $\mathcal{C}$ de longueur N, la distance de Hamming $d_H(\mathbf{x}_1, \mathbf{x}_2)$ est égale aux nombres de bits qui différent.

  Exemple : $\mathbf{x}_1 = [001100]$ et $\mathbf{x}_2 = [001111]$, $d_H(\mathbf{x}_1, \mathbf{x}_2) = 2$

- **poids de Hamming** : le poids de Hamming $w(\mathbf{x})$ d'un mot de code binaire $\mathbf{x}$ est égal au nombre de bits non nuls de ce mot de code.

  Exemple : $\mathbf{x} = [001100]$, $w(\mathbf{x}) = 2$

- **distance minimale** : La distance minimale d_{min} du code $\mathcal{C}$ est le nombre de bits qui différent entre les deux mots de code les plus proches au sens de la distance de Hamming :

$$d_{min} = \min_{i,j,i \neq j} d_H(\mathbf{x}_i, \mathbf{x}_j) \tag{129}$$

Lorsque le code est linéaire, la distance minimale d_{min} est égale au poids de Hamming minimal du code $\mathcal{C}$ (en excluant le mot de code nul):

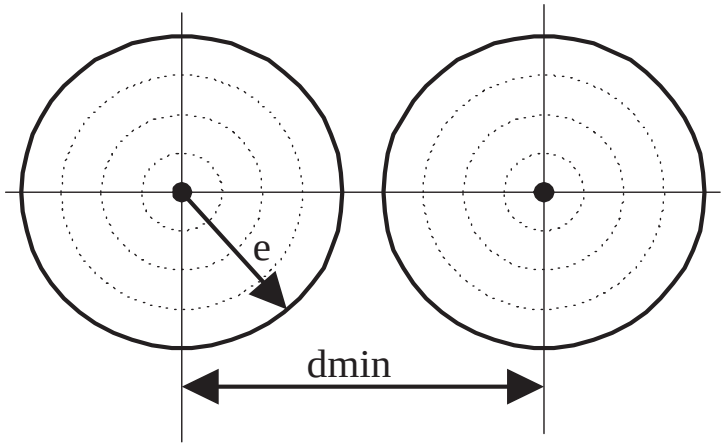
$$d_{min} = \min_{i,i \neq 0} w(\mathbf{x}_i) \tag{130}$$

PROPRIETES (SUITE)

• **Capacité de correction d'erreurs d'un code linéaire binaire en bloc**

Un décodeur à entrées dures peut corriger jusqu'à e erreurs avec :

$$e = \left\lfloor \frac{d_{min} - 1}{2} \right\rfloor \tag{131}$$



• **marge inférieure de Hamming** :

Pour un code (N, K) on a la relation suivante

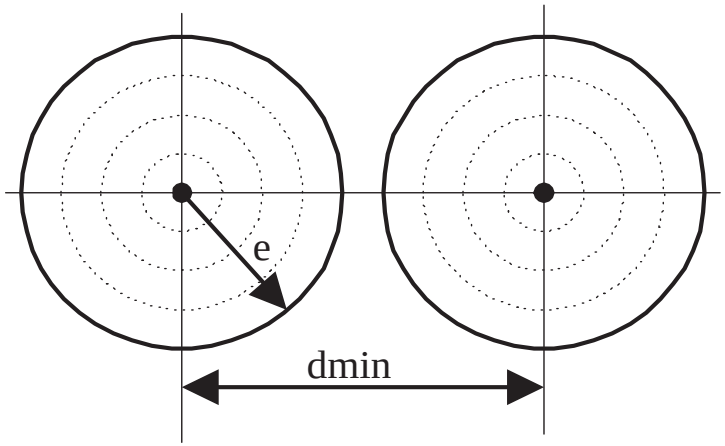
$$2^{N-K} \geq \sum_{i=0}^{e} C_i^N \tag{132}$$

PROPRIETES (SUITE)

• **Capacité de correction d'erreurs d'un code linéaire binaire en bloc**

Un décodeur à entrées dures peut corriger jusqu'à e erreurs avec :

$$e = \left\lfloor \frac{d_{min} - 1}{2} \right\rfloor \tag{133}$$



LES CODES LINEAIRES EN BLOC PARFAITS

• **codes parfaits** :

les codes parfaits possèdent la propriété que tous les 2^N mots possibles sont inclus dans les 2^K boules de Hamming de rayon e. L'inégalité ci-dessus se transforme en égalité.

• **Codes de Hamming**

Les codes de Hamming sont des codes en bloc linéaires parfaits binaires (N, K) avec $N = 2^J - 1$ et $K = 2^J - 1 - J$.

Par exemple pour $J = 3$, le code de Hamming est un code (7,4).

$$\mathbf{G} = \begin{pmatrix} 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix} \tag{134}$$

• **Code de Golay**

C'est un code (23,12) dont la distance minimale est égale à 7.

$$\mathbf{G} = \begin{pmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 1
\end{pmatrix} \tag{135}$$

MATRICE DE CONTROLE

Associé à chaque code $\mathcal{C}$ linéaire en bloc binaire (N, K) il existe un code linéaire en bloc binaire dual $(N, N-K)$

Soit **H** la matrice génératrice de ce code dual. Chacun des mots de code **x** du code $\mathcal{C}$ est orthogonal à tous les mots de code du code dual :

$$\mathbf{xH}^T = \mathbf{0}$$

Puisque cette relation est valide pour tous les mots de code du code $\mathcal{C}$, on a

$$\mathbf{GH}^T = \mathbf{0}$$

Si la matrice génératrice **G** est systématique **H** est de la forme :

$$\mathbf{H} = [\mathbf{P}^T \quad \mathbf{I}_{N-K}] = \begin{pmatrix} p_{11} & p_{21} & \dots & p_{K1} & 1 & 0 & 0 & \dots & 0 \\ p_{12} & p_{22} & \dots & p_{K2} & 0 & 1 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ p_{1N-K} & p_{2N-K} & \dots & p_{KN-K} & 0 & 0 & 0 & \dots & 1 \end{pmatrix}$$

exemple 3 (suite) : la matrice de contrôle du code $\mathcal{C}_3\ (7,4)$ est la suivante :

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

Chaque ligne de la matrice de parité correspond à une équation de parité

$$\begin{cases} u_1 + u_2 + u_3 + x_5 = 0 & \text{nœud} \quad T_1 \\ u_2 + u_3 + u_4 + x_6 = 0 & \text{nœud} \quad T_2 \\ u_1 + u_2 + u_4 + x_7 = 0 & \text{nœud} \quad T_3 \end{cases} \tag{136}$$

COURS 10

Décodage des codes en bloc (méthode du syndrome, Viterbi

DECODAGE A ENTREES DURES DES CODES EN BLOC LINEAIRES BINAIRES

- décodage à entrées dures versus décodage à entrées pondérées
- Le mot reçu **r** est la somme modulo 2 du mot de code émis **x** et du vecteur d'erreurs **e**

$$\mathbf{r} = \mathbf{x} + \mathbf{e}$$

3 méthodes de décodage

- recherche exhaustive
- méthode du tableau standard
- décodage par syndrome

Le syndrome d'erreurs **s** de dimension $1 \times (N - K)$ est défini par:

$$\begin{aligned} \mathbf{s} &= \mathbf{r}\mathbf{H}^T \\ &= \mathbf{x}\mathbf{H}^T + \mathbf{e}\mathbf{H}^T \\ &= \mathbf{e}\mathbf{H}^T \qquad \text{car} \quad \mathbf{x}\mathbf{H}^T = 0 \end{aligned} \tag{137}$$

En l'absence d'erreurs de transmission, le syndrome d'erreurs **s** est le vecteur nul.

METHODE DU TABLEAU STANDARD

- 2^{N-K} syndromes d'erreurs
- un même syndrome d'erreurs pour $2^N/2^{N-K} = 2^K$ vecteurs d'erreurs possibles

$\mathbf{x}_0$	$\mathbf{x}_1$	$\mathbf{x}_2$	$\ldots$	$\mathbf{x}_{2^K-1}$	$\mathbf{s}_0$
$\mathbf{x}_0 + \mathbf{e}_1$	$\mathbf{x}_1 + \mathbf{e}_1$	$\mathbf{x}_2 + \mathbf{e}_1$	$\ldots$	$\mathbf{x}_{2^K-1} + \mathbf{e}_1$	$\mathbf{s}_1$
$\mathbf{x}_0 + \mathbf{e}_2$	$\mathbf{x}_1 + \mathbf{e}_2$	$\mathbf{x}_2 + \mathbf{e}_2$	$\ldots$	$\mathbf{x}_{2^K-1} + \mathbf{e}_2$	$\mathbf{s}_2$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
$\mathbf{x}_0 + \mathbf{e}_{2^{N-K}-1}$	$\mathbf{x}_1 + \mathbf{e}_{2^{N-K}-1}$	$\mathbf{x}_2 + \mathbf{e}_{2^{N-K}-1}$	$\ldots$	$\mathbf{x}_{2^K-1} + \mathbf{e}_{2^{N-K}-1}$	$\mathbf{s}_{2^{N-K}-1}$

- chaque rangée est une classe correspondant à un syndrome
- le représentant est appelé chef de classe
- le chef de classe est le vecteur d'erreurs le plus vraisemblable parmis les mots de la classe.
- Le décodage consiste donc à rechercher dans le tableau la colonne dans laquelle se trouve le mot reçu. Le résultat du décodage sera le mot de code situé en première ligne de cette colonne.

DECODAGE PAR SYNDROME

- calcul du syndrome d'erreurs correspondant au mot reçu
- on associe à ce syndrome le vecteur d'erreurs estimé correspondant $\hat{\mathbf{e}}$.
- le mot de code estimé $\hat{\mathbf{x}}$ est alors :

$$\hat{\mathbf{x}} = \mathbf{r} + \hat{\mathbf{e}} \tag{138}$$

- Cette méthode est moins complexe que la méthode du tableau standard
- exemple :la table pour le décodage d'un code de Golay étendu (24,12) nécessite une mémoire de 4096 mots de 23 bits.

FONCTION D'ENUMERATION DE POIDS

Définition 1 : la fonction d'énumération de poids WEF (weight enumerator function *en anglais*) d'un codeur binaire en bloc systématique (N, K) est définie comme suit :

$$A(D) = \sum_{d=0}^{N} A_d D^d \tag{139}$$

A_d est le nombre de mots de code de longueur N de poids d.

Définition 2 : la fonction d'énumération de poids IRWEF (input redundancy weight enumerator function *en anglais*) d'un codeur binaire en bloc systématique (N, K) est définie comme suit :

$$A(W, Z) = \sum_{w=0}^{K} \sum_{z=0}^{N-K} A_{w,z} W^w Z^z \tag{140}$$

$A_{w,z}$ est le nombre de mots de code de longueur N dont le poids de la séquence des bits d'information est égal à w et dont le poids de la séquence des bits de redondance est égal à z.

BORNE PAR REUNION

• La borne par réunion est un outil très utile pour évaluer les performances des systèmes de transmission.

Considérons un décodeur à maximum de vraisemblance :

$$\hat{\mathbf{x}} = \arg\max_{\mathbf{x}} Pr(\mathbf{y}|\mathbf{x}) \tag{141}$$

Exemple : soit un ensemble de 4 mots de codes $\mathbf{x_1}, \mathbf{x_2}, \mathbf{x_3}$ et $\mathbf{x_4}$ et leurs zones de décision associées $\Lambda_1, \Lambda_2, \Lambda_3$ et Λ_4 donné sur la figure 18 (figure de gauche).

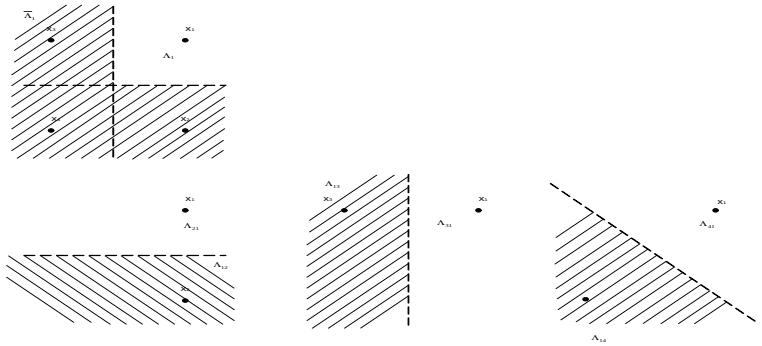


Figure 18: Zones de décision associées aux mots de code.

On peut par exemple borner la probabilité d'erreurs $Pr(\text{erreur mot}|\mathbf{x}_1)$:

$$
\begin{aligned}
Pr(\text{erreur mot}|\mathbf{x}_1) &= Pr(\mathbf{y} \in \bar{\Lambda_1}|\mathbf{x}_1) \\
&= Pr(\mathbf{y} \in (\Lambda_2 \cup \Lambda_3 \cup \Lambda_4)|\mathbf{x}_1) \\
&\leq Pr(\mathbf{y} \in \Lambda_{12}|\mathbf{x}_1) + Pr(\mathbf{y} \in \Lambda_{13}|\mathbf{x}_1) + Pr(\mathbf{y} \in \Lambda_{14}|\mathbf{x}_1) \\
&= Pr(\mathbf{x}_2|\mathbf{x}_1) + Pr(\mathbf{x}_3|\mathbf{x}_1) + Pr(\mathbf{x}_4|\mathbf{x}_1)
\end{aligned}
$$

La probabilité d'erreurs mot est égale à

$$
Pr(\text{erreur mot}) = \sum_i Pr(\mathbf{x}_i) Pr(\text{erreur mot}|\mathbf{x}_i)
$$

BORNE PAR REUNION

La probabilité de décoder un mauvais mot de code sachant un mot de code $\mathbf{x}_i$ transmis est bornée supérieurement comme suit :

$$\begin{aligned} Pr(\text{erreur mot}|\mathbf{x}_i) &= Pr(\mathbf{y} \in \bar{\Lambda}_i|\mathbf{x}_i) \\ &\leq \sum_{j:j\neq i} Pr(\mathbf{x}_j|\mathbf{x}_i) \end{aligned} \tag{142}$$

avec Λ_i zone de décision associée au mot de code $\mathbf{x}_i$.
$Pr(\mathbf{x}_j|\mathbf{x}_i)$ est la probabilité que $\mathbf{y}$ soit plus près de $\mathbf{x}_j$ que de $\mathbf{x}_i$.

• La borne par réunion ramène la comparaison d'un ensemble de mots de code à un certain nombre de comparaison de mots de code deux à deux.

• La borne par réunion n'est précise que pour les rapports signal à bruit élevés.

Si la distance de Hamming entre deux mots de code $\mathbf{x}_i$ et $\mathbf{x}_j$ est égale à d, leur distance euclidienne est égale à $2\sqrt{dRE_b}$ où $R = \frac{K}{N}$ est le rendement du code.

On a alors :

$$Pr(\mathbf{x}_j|\mathbf{x}_i) = \frac{1}{2}\text{erfc}\left(\sqrt{dR\frac{E_b}{N_0}}\right) \tag{143}$$

En utilisant la borne par réunion, on obtient la borne supérieure suivante sur le taux d'erreurs mot (TEM) et le taux d'erreurs bit (TEB) du décodeur à maximum de vraisemblance sur un canal BBAG associé à un code en bloc linéaire (N, K) :

$$TEM \leq \frac{1}{2}\sum_{d=d_{libre}}^{N} A_d\text{erfc}\left(\sqrt{dR\frac{E_b}{N_0}}\right)$$

où A_d est le nombre de mots de code de poids d.

$$TEB \approx \frac{1}{2}\sum_{d=d_{libre}}^{N} B_d\text{erfc}\left(\sqrt{dRE_b/N_0}\right)$$

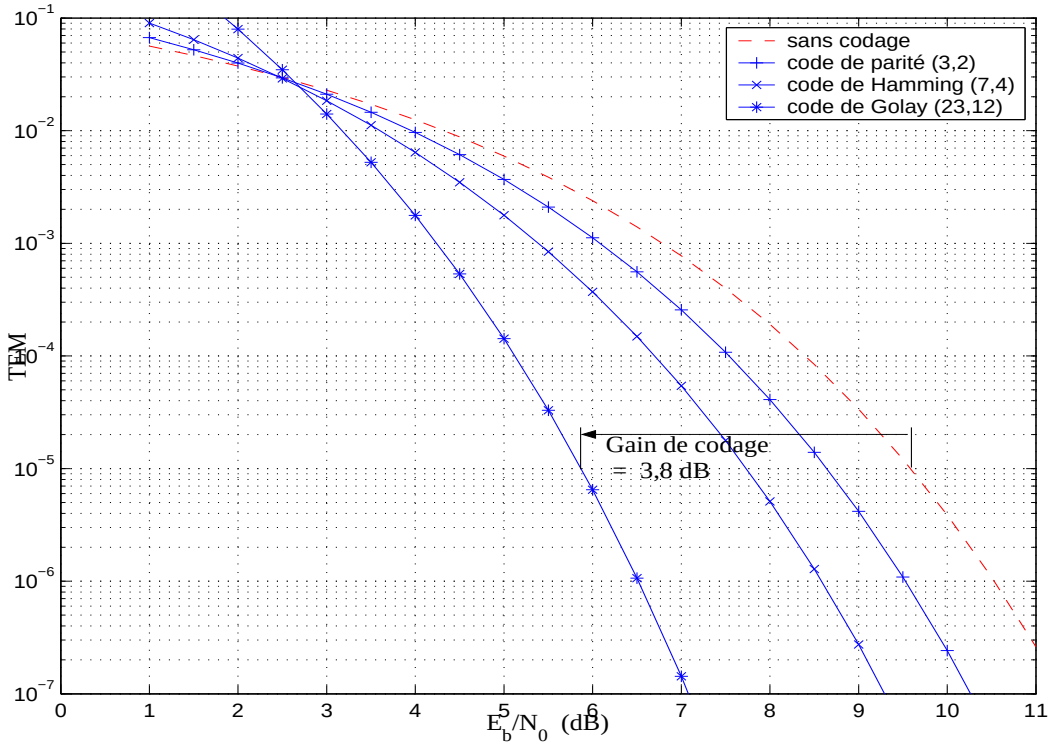
$$\text{avec} \qquad B_d = \sum_{d:d=w+z} \frac{w}{K} A_{w,z} \tag{144}$$

GAIN DE CODAGE

• Sur un canal BBAG, il est théoriquement possible de réaliser une transmission sans erreur à un rapport $E_b/N_0 = 0dB$ en utilisant un code de rendement 1/2.

• La différence entre une chaine de transmission utilisant une modulation bipodale sans codage et cette limite de capacité est de 9.6 dB pour un taux d'erreurs bit de 10^{-5})

• On définit le gain de codage d'un code correcteur d'erreurs comme étant la différence de rapport E_B/N_0 entre la chaine sans codage et la chaine utilisant ce code.



FONCTION D'ENUMERATION DE POIDS

Définition 1 : la fonction d'énumération de poids WEF (weight enumerator function *en anglais*) d'un codeur binaire en bloc systématique (N, K) est définie comme suit :

$$A(D) = \sum_{d=0}^{N} A_d D^d \tag{145}$$

A_d est le nombre de mots de code de longueur N de poids d.

Définition 2 : la fonction d'énumération de poids IRWEF (input redundancy weight enumerator function *en anglais*) d'un codeur binaire en bloc systématique (N, K) est définie comme suit :

$$A(W, Z) = \sum_{w=0}^{K} \sum_{z=0}^{N-K} A_{w,z} W^w Z^z \tag{146}$$

$A_{w,z}$ est le nombre de mots de code de longueur N dont le poids de la séquence des bits d'information est égal à w et dont le poids de la séquence des bits de redondance est égal à z.

COURS 11

Critère MAP/ML décodge à entrées pondérées

DETECTION OPTIMAL

• L'objectif du détecteur optimal est de déterminer le message qui a été le plus vraisemblablement émis $\hat{\mathbf{x}}$.

• Soit le message $\mathbf{x}$ envoyé dans un canal discret stationnaire sans mémoire de densité de probabilité conditionnelle $p(y/x)$ et $\mathbf{y}$ le vecteur reçu après filtrage adapté.

• D'une manière générale, un détecteur *maximum a posteriori* (MAP) cherche parmi tous les messages possibles $\mathbf{x}$, le message estimé $\hat{\mathbf{x}}$ pour lequel la probabilité conditionnelle $Pr(\mathbf{x}|\mathbf{y})$ est la plus grande.

$$\hat{\mathbf{x}} = \arg\max_{\mathbf{x}} Pr(\mathbf{x}|\mathbf{y}) \tag{147}$$

• En utilisant la loi de Bayes, on peut écrire :

$$Pr(\mathbf{x}|\mathbf{y}) = \frac{p(\mathbf{y}|\mathbf{x})Pr(\mathbf{x})}{p(\mathbf{y})} \tag{148}$$

• Si tous les messages sont équiprobables, et comme le dénominateur $p(\mathbf{y})$ est commun à toutes les messages, le message estimé $\hat{\mathbf{x}}$ est le message pour lequel la probabilité conditionnelle $p(\mathbf{y}|\mathbf{x})$ est la plus grande.

$$\hat{\mathbf{x}} = \arg\max_{\mathbf{x}} p(\mathbf{y}|\mathbf{x}) \tag{149}$$

• Un détecteur utilisant ce critère est appelé un détecteur à *maximum de vraisemblance* (*maximum likelihood* en anglais ou ML).

• un détecteur ML calcule les distances euclidiennes entre l'échantillon reçu et les échantillons correspondant à toutes les séquences possibles.

• Messages équiprobables $\Rrightarrow$ détecteur MAP = détecteur ML.

Il faut ici distinguer deux cas :
- si l'entrée du décodeur est binaire on parlera de décodage à entrées dures (*hard decoding* en anglais).
- si l'entrée du décodeur peut prendre une valeur continue entre $-V$ et $+V$, on dira que le décodage est à entrées souples ou pondérées (*soft decoding* en anglais).

cas du canal BSC

$$p(\mathbf{y}|\mathbf{x}) = p^{d_H(\mathbf{y},\mathbf{x})}(1-p)^{N-d_H(\mathbf{y},\mathbf{x})} = (1-p)^N \left(\frac{p}{1-p}\right)^{d_H(\mathbf{y},\mathbf{x})} \tag{150}$$

où $d_H(\mathbf{y},\mathbf{x})$ est la distance de Hamming entre la séquence reçue $\mathbf{y}$ et la séquence $\mathbf{x}$. Comme p est compris entre 0 et 0.5, on a $0 < \frac{p}{1-p} < 1$.

Ainsi maximiser $p(\mathbf{y}|\mathbf{x})$ revient à minimiser la distance de Hamming entre $\mathbf{y}$ et $\mathbf{x}$.

cas du canal BBAG

Après filtrage adapté et échantillonnage, on a :

$$\mathbf{y} = \mathbf{x} + \mathbf{n} \tag{151}$$

avec $x_i = \pm\sqrt{RE_b}$ (modulation bipodale) et n_i échantillon gaussien centré de variance $\sigma^2 = \frac{N_0}{2}$.

$$p(y_i|x_i) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{ -\frac{(y_i - x_i)^2}{2\sigma^2} \right\} \tag{152}$$

$$
\begin{aligned}
\hat{\mathbf{x}} &= \arg\max_{\mathbf{x}} p(\mathbf{y}|\mathbf{x}) \\
&= \arg\max_{\mathbf{x}} \log p(\mathbf{y}|\mathbf{x}) \\
&= \arg\max_{\mathbf{x}} \log \prod p(y_i|x_i) \\
&= \arg\max_{\mathbf{x}} \sum_{i=0}^{N-1} \log p(y_i|x_i) \\
&= \arg\max_{\mathbf{x}} \sum_{i=0}^{N-1} \left\{ -\frac{(y_i - x_i)^2}{2\sigma^2} \right\} \\
&= \arg\min_{\mathbf{x}} \sum_{i=0}^{N-1} (y_i - x_i)^2
\end{aligned} \tag{153}
$$

• Une seconde version du décodeur à maximum de vraisemblance est obtenue en approximant les calculs :

$$
\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \sum_{i=0}^{N-1} |y_i - x_i| \tag{154}
$$

TAUX D'ERREURS POUR UN CODE NRZ SUR CANAL BBAG
PROBABILITE D'ERREURS PAR PAIRE

• Soient deux mots $\mathbf{x}_i$ et $\mathbf{x}_j$ dont la distance euclidienne est $d(\mathbf{x}_i, \mathbf{x}_j)$. Pour un canal BBAG, la probabilité $Pr(\mathbf{x}_i \rightarrow \mathbf{x}_j)$ que **y** soit plus près de $\mathbf{x}_j$ que de $\mathbf{x}_i$ est donnée par :

$$Pr(\mathbf{x}_i \rightarrow \mathbf{x}_j) = \frac{1}{2}\text{erfc}\left(\frac{d(\mathbf{x}_i, \mathbf{x}_j)}{2\sqrt{N_0}}\right) \tag{155}$$

• Dans le cas du code NRZ, le mot $\mathbf{x}_i$ est l'échantillon x_i. La distance euclidienne est égale à $2\sqrt{E_b}$.

La probabilité d'erreurs mot est égale à

$$Pr(\text{erreur mot}) = \sum_i Pr(\mathbf{x}_i) Pr(\text{erreur mot}|\mathbf{x}_i)$$

La probabilité de décoder un mauvais mot de code sachant un mot de code $\mathbf{x}_i$ transmis est bornée supérieurement comme suit :

$$Pr(\text{erreur mot}|\mathbf{x}_i) \leq \sum_{j:j\neq i} Pr(\mathbf{x}_i \rightarrow \mathbf{x}_j) \tag{156}$$

$Pr(\mathbf{x}_i \rightarrow \mathbf{x}_j)$ est la probabilité que $\mathbf{y}$ soit plus près de $\mathbf{x}_j$ que de $\mathbf{x}_i$ sachant $\mathbf{x}_i$ émis.

• La borne par réunion ramène la comparaison d'un ensemble de mots de code à un certain nombre de comparaison de mots de code deux à deux.

• La borne par réunion n'est précise que pour les rapports signal à bruit élevés.

Si la distance de Hamming entre deux mots de code $\mathbf{x}_i$ et $\mathbf{x}_j$ est égale à d, leur distance euclidienne est égale à $2\sqrt{dRE_b}$ où R est le rendement du code.

On a alors :

$$Pr(\mathbf{x}_i \rightarrow \mathbf{x}_j) = \frac{1}{2}\text{erfc}\left(\sqrt{dR\frac{E_b}{N_0}}\right) \tag{157}$$

En utilisant la borne par réunion, on obtient la borne supérieure suivante sur le taux d'erreurs mot (TEM) et le taux d'erreurs bit (TEB) du décodeur à maximum de vraisemblance sur un canal BBAG associé à un code en bloc linéaire (N, K) :

$$TEM \leq \frac{1}{2}\sum_{d=d_{libre}}^{N} A_d \text{erfc}\left(\sqrt{dR\frac{E_b}{N_0}}\right)$$

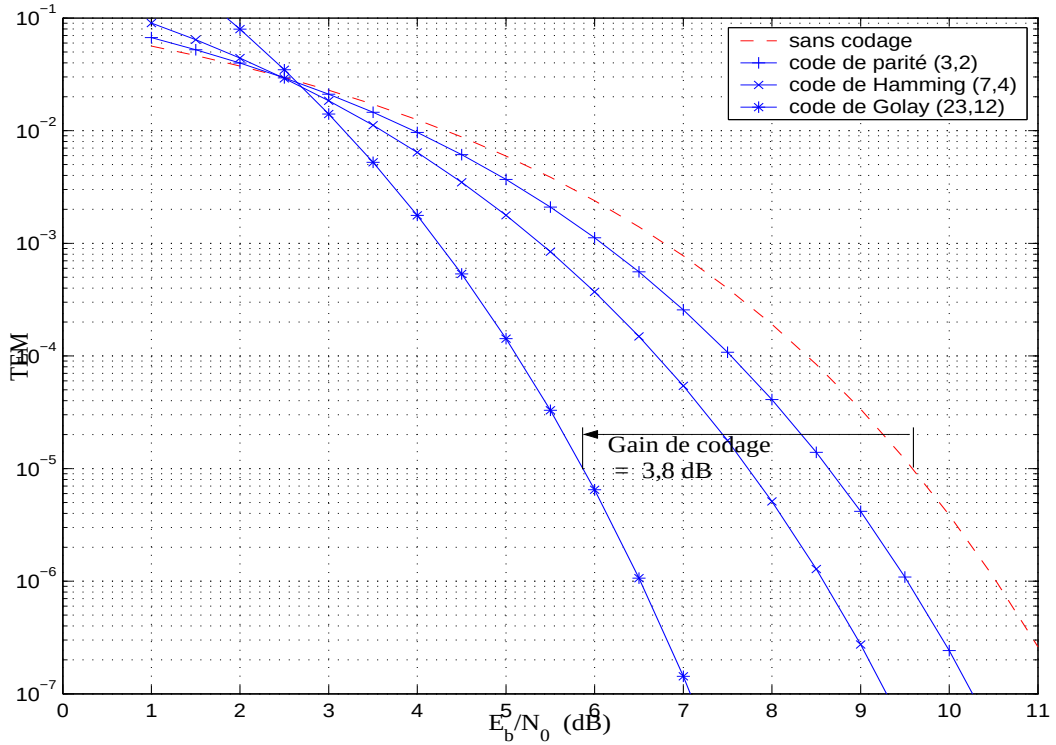
où A_d est le nombre de mots de code de poids d.

GAIN DE CODAGE

• Sur un canal BBAG, il est théoriquement possible de réaliser une transmission sans erreur à un rapport $E_b/N_0 = 0dB$ en utilisant un code de rendement 1/2.

• La différence entre une chaine de transmission utilisant une modulation bipodale sans codage et cette limite de capacité est de 9.6 dB pour un taux d'erreurs bit de 10^{-5})

• On définit le gain de codage d'un code correcteur d'erreurs comme étant la différence de rapport E_B/N_0 entre la chaine sans codage et la chaine utilisant ce code.



PERFORMANCES DES DECODEURS A ENTREES DURES ET PONDEREES

- entrées dures

$$TEM_{hard} \leq 1 - \sum_{i=0}^{e} C_N^i p^i (1-p)^{N-i} \tag{158}$$

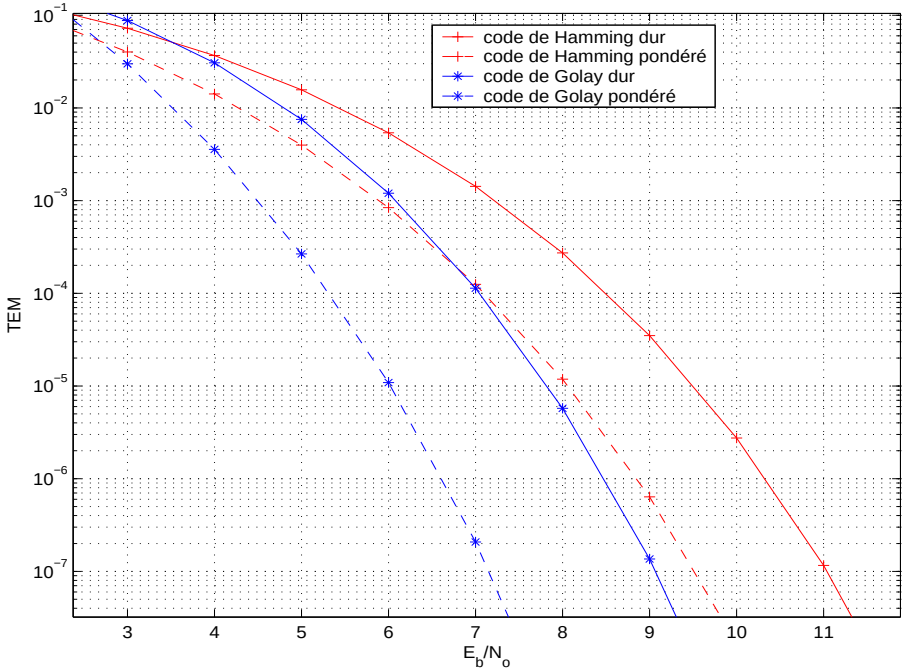
e capacité de correction du code correcteur d'erreurs

$$p = \frac{1}{2}\text{erfc}\left(\sqrt{\frac{REb}{N_0}}\right) \tag{159}$$

- entrées pondérées

$$TEM_{soft} \leq \frac{1}{2} \sum_{d=d_{libre}}^{N} A_d \text{erfc}\left(\sqrt{dR\frac{E_b}{N_0}}\right)$$

$TEM = f(E_b/N_0)$ d'une chaîne de transmission utilisant un code de Hamming (7,4) et un code de Golay (23,12).



On peut observer un gain d'environ 2 dB entre décodage à entrées dures et pondérées.

COURS 12

Codes cycliques, BCH, RS

CODES CYCLIQUES

• Les codes cycliques forment un sous ensemble des codes linéaires en bloc.

• Ces codes contiennent les familles de codes les plus importantes comme les codes de Hamming, de Golay, les codes BCH et Reed Solomon.

• Leurs propriétés permettent un codage et un décodage relativement aisé.

• Si **x** $= [x_0\ x_1\ \ldots\ x_{N-2}\ x_{N-1}]$ est un mot de code alors
x' $= [x_{N-1}\ x_0\ \ldots\ x_{N-3}\ x_{N-2}]$ est aussi un mot de code.

• On associe à chaque mot de code **x** $= [x_0\ x_1\ \ldots\ x_{N-2}\ x_{N-1}]$
un polynôme $x(p) = x_0 + x_1 p + \cdots + x_{N-2}p^{N-2} + x_{N-1}p^{N-1}$

• Calculons le polynôme $px(p)$:

$$px(p) = x_0 p + x_1 p^2 + \cdots + x_{N-2}p^{N-1} + x_{N-1}p^N$$

En ajoutant x_{N-1} puis retranchant x_{N-1} :

$$px(p) = x_{N-1} + x_0 p + x_1 p^2 + \cdots + x_{N-2}p^{N-1} + x_{N-1}(p^N - 1)$$

Effectuons le calcul de $px(p)$ modulo $(p^N - 1)$:

$$\begin{aligned} px(p) \mod (p^N - 1) &= x_{N-1} + x_0 p + x_1 p^2 + \cdots + x_{N-2} p^{N-1} \\ &= x'(p) \end{aligned}$$

• Un décalage circulaire à droite d'une position correspond à une multiplication par p modulo $p^N - 1$.

• Plus généralement, un décalage circulaire à droite de i positions correspond à la multiplication par p^i modulo $p^N - 1$.

PROPRIETES DES CODES CYCLIQUES

• Si $x(p)$ est un polynôme associé à un mot de code d'un code cyclique (N, K), alors

$$(a_0 + a_1 p + a_2 p^2 + \cdots + a_{K-1} p^{K-1}) x(p) \qquad \mod (p^N - 1)$$

est aussi un polynôme associé à un mot de code.

• Alors que pour un code en bloc linéaire K mots de code sont nécessaires pour déterminer l'ensemble des 2^K mots de code, pour les codes cycliques, un seul mot de code suffit.

Propriété 1 : Il est possible de construire un code cyclique (N, K) à partir d'un polynôme générateur noté $g(p)$ de degré $N - K$.

$$g(p) = g_0 + g_1 p + \cdots + g_{N-K-1} p^{N-K-1} + p^{N-K}$$

$g(p)$ est le polynôme de degré minimal appartenant au code cyclique.

Propriété 2 : Le polynôme $g(p)$ est un facteur de $p^N - 1$.

Propriété 3 : La liste de l'ensemble des 2^K polynômes s'obtient par multiplication de $g(x)$ par les 2^K polynômes de degré inférieur ou égal à $K - 1$.

Soit le mot d'information **u** $= [u_0\ u_1\ \ldots\ u_{K-2}\ u_{K-1}]$ et
$u(p) = u_0 + u_1 p + \cdots + u_{K-2}p^{K-2} + u_{K-1}p^{K-1}$ son polynôme associé.

On a la relation entre $u(p)$ et $x(p)$ suivante :

$$x(p) = u(p)g(p)$$

Propriété 4 : Tout polynôme facteur de $p^N - 1$ engendre un code cyclique.

• Décomposition en produit de polynômes irréductibles des polynômes de la forme $p^{2^j-1} - 1$:

$j = 2$	$p^3 - 1 = (1+p)(1+p+p^2)$
$j = 3$	$p^7 - 1 = (1+p)(1+p+p^3)(1+p^2+p^3)$
$j = 4$	$p^{15} - 1 = (1+p)(1+p+p^2)(1+p^3+p^4)(1+p+p^4)(1+p+p^2+p^3+p^4)$
$j = 5$	$p^{31} - 1 = (1+p)(1+p^3+p^5)(1+p^2+p^5)(1+p^2+p^3+p^4+p^5)$ $(1+p+p^3+p^4+p^5)(1+p+p^2+p^4+p^5)(1+p+p^2+p^3+p^5)$

exemple : on peut construire des codes de Hamming (7,4) avec $g(p) = 1 + p + p^3$ ou $g(p) = 1 + p^2 + p^3$.

Pour construire la matrice génératrice il suffit de choisir les polynômes suivants :

$$g(p) \quad g(p)p \quad g(p)p^2 \quad \ldots \quad g(p)p^{K-1}$$

exemple : pour le code de Hamming (7,4) avec $g(p) = 1+p+p^3$, la matrice génératrice est la suivante :

$$\mathbf{G} = \begin{pmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{pmatrix} \tag{160}$$

- La matrice génératrice ainsi obtenue n'est pas sous la forme systématique.

CODES CYCLIQUES SOUS FORME SYSTEMATIQUE

On cherche à obtenir un mot de code **x** sous la forme systématique $\mathbf{x} = [x_0\ x_1\ \ldots\ x_{N-1}] = [x_0\ x_1\ \ldots\ x_{N-K-1}\ u_0\ u_1\ \ldots\ u_{K-2}\ u_{K-1}]$

Le polynôme $x(p)$ associé $x(p) = q(p)g(p)$ est calculé comme suit:

-1 multiplier le polynôme $u(p)$ par p^{N-K}

-2 diviser $p^{N-K}u(p)$ par $g(p)$ pour obtenir le reste $r(p)$

-3 ajouter le reste $r(p)$ à $p^{N-K}u(p)$

$$x(p) = r(p) + p^{N-K}u(p)$$

COMPARAISON AVEC L'APPROCHE MATRICIELLE

Dans l'approche matricielle nous avons vu que la matrice génératrice avait la forme :

$$Gs = [I \quad P]$$

Avec l'approche codage cyclique, la matrice génératrice aura la forme suivante :

$$Gs = [P \quad I]$$

car le mot de code x se présente différemment :

$$\mathbf{x} = [x_0\ x_1\ \ldots\ x_{N-1}] = [x_0\ x_1\ \ldots\ x_{N-K-1}\ u_0\ u_1\ \ldots\ u_{K-2}\ u_{K-1}]$$

Exemple : pour le code de Hamming (7,4) avec $g(p) = 1+p+p^3$, la matrice génératrice avec l'approche matricielle est la suivante :

$$\mathbf{Gs} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{pmatrix} \tag{161}$$

avec l'approche cyclique on obtient :

$$\mathbf{Gs} = \begin{pmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{162}$$

DIVISION DE POLYNOMES DANS GF(2)

Soit la division d'un polynôme dividende $a(p)$ par un polynôme diviseur $g(p)$ de degré d dans le corps de Galois GF(2). On a la relation:

$$a(p) = g(p)q(p) + r(p)$$

Exemple :

$$
\begin{array}{l|l}
1p^6+0p^5+0p^4+0p^3+1p^2+0p+1 & 1p^2+1p+1 \\
\cline{2-2}
\underline{1p^6+1p^5+1p^4} & \\
0p^6+1p^5+1p^4+0p^3 & 1p^4+1p^3+0p^2+1p+0 \\
\quad \underline{1p^5+1p^4+1p^3} & \\
\quad 0p^5+0p^4+1p^3+1p^2 & \\
\quad\quad \underline{0p^4+0p^3+0p^2} & \\
\quad\quad 0p^4+1p^3+1p^2+0p & \\
\quad\quad\quad \underline{1p^3+1p^2+1p} & \\
\quad\quad\quad 0p^3+0p^2+1p+1 &
\end{array}
$$

-1 Au cours de la division, le degré du dividende décroît : les modifications du dividende sont réalisées des poids forts vers les poids faibles

-2 Pour un diviseur de degré d, à chaque itération, les modifications du dividende ne portent que sur les $(d+1)$ termes les plus à gauche.

-3 Lorsque la modification du dividende a lieu, elle consiste à ajouter terme à terme les $d+1$ coefficients du diviseur.

- Il est possible de déduire une structure matérielle pour réaliser cette division.

STRUCTURE MATERIELLE D'UN CODEUR CYCLIQUE

Cette structure comporte autant de registres à décalage que le degré du diviseur.

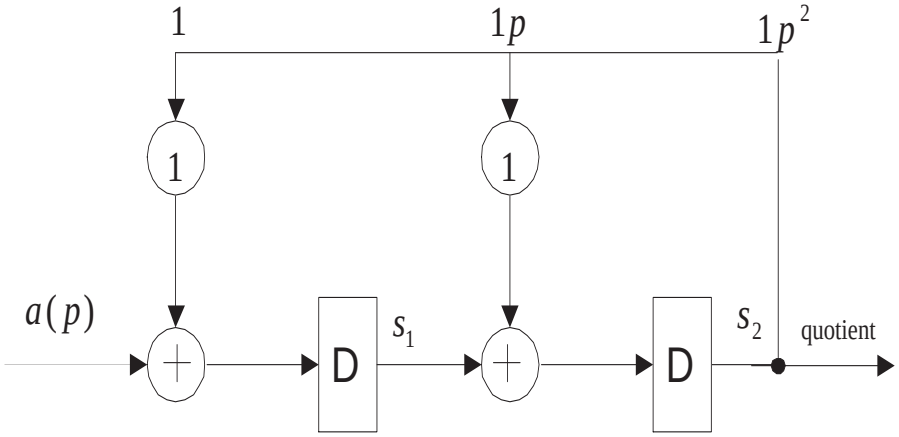


Figure 19: structure matérielle d'un diviseur par $1 + p + p^2$

• Les bits entrent dans le diviseur poids fort (MSB) en tête.

• A chaque coup d'horloge, un nouveau coefficient du polynôme $a(p)$ entre dans le circuit. Après d coups d'horloge, le premier coefficient non nul du quotient sort du dernier registre à décalage. Ce coefficient est multiplié par $g(p)$ puis soustrait comme dans la division.

STRUCTURE D'UN CODEUR CYCLIQUE

• Dans le cas des codes cycliques (N, K), nous avons vu qu'avant de calculer le reste $r(p)$ de la division il est nécessaire de prémultiplier le polynôme associé au mot d'information $u(p)$ par p^{N-K}.

• La multiplication par p^{N-K} est équivalente à ajouter p^{N-K} zéros à la suite de $u(p)$.

• Il est possible de réduire le nombre de coups d'horloge nécessaire pour calculer le reste en exploitant le fait que les $N - K$ derniers bits du polynôme $p^{N-K}u(p)$ sont nuls.

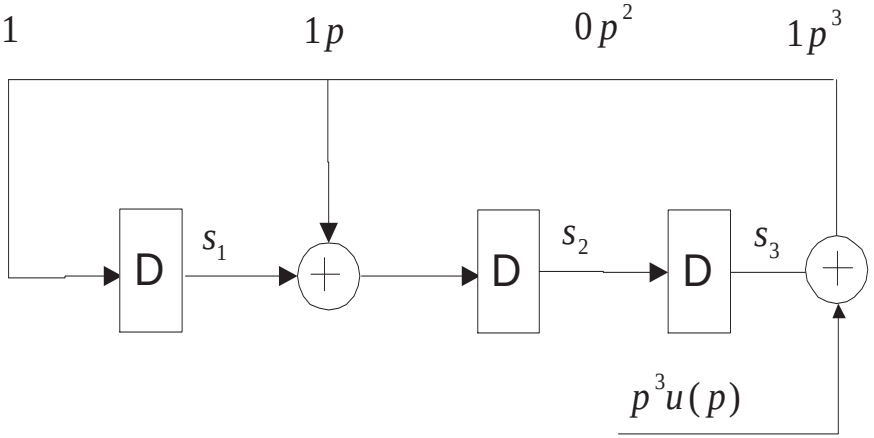


Figure 20: structure matérielle d'un codeur de Hamming $g(p) = 1 + p + p^3$

Cette structure permet de gagner $N - K$ coups d'horloge correspondant aux $N - K$ derniers bits nuls du polynôme $p^{N-K}u(p)$.

STRUCTURE MATERIELLE COMPLETE D'UN CODEUR CYCLIQUE

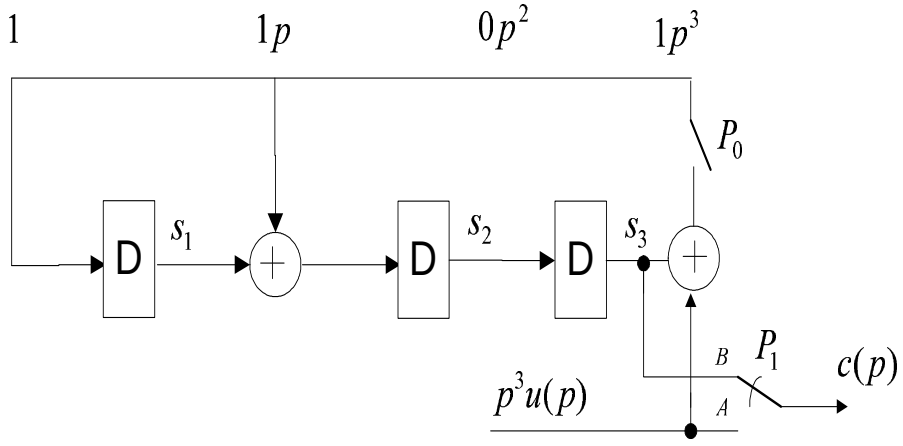


Figure 21: structure matérielle complète d'un codeur de Hamming $(7,4)$ $g(p) = 1 + p + p^3$

Le codage s'effectue en deux étapes :

1/ L'interrupteur P_0 est fermé et P_1 est en position A. On applique $K = 4$ coups d'horloge pour envoyer les K premiers bits du mot de code et calculer le reste $r(p)$.

2/ L'interrupteur P_0 est ouvert et P_1 est en position B. On applique $N - K = 3$ coups d'horloge pour envoyer les $N - K$ derniers bits du mot de code (correspondant au reste).

CORPS DE GALOIS $GF(2^4)$

• Le corps de Galois fini GF(2^4) est isomorphe au corps des polynômes à coefficients dans GF(2) modulo un polynôme irréductible dans GF(2) et de degré 4.

• Liste des polynomes irréductibles facteurs de $p^{15} - 1$

$1 + p + p^4$
$1 + p + p^2 + p^3 + p^4$
$1 + p + p^2$
$1 + p^3 + p^4$
$1 + p$

• Exemple : on choisit le polynôme irréductible $1 + p + p^4$

• Soit α une racine de ce polynôme.

• Les puissances successives de α engendrent les $2^m - 1$ éléments non nuls du corps GF(2^m).

• Liste des éléments du corps GF(2^4) construit avec $1 + p + p^4$

élément	polynôme	représentation binaire	élément	polynôme	représentation binaire
0	0	0000	α^7	$1 + p + p^3$	1011
1	1	0001	α^8	$1 + p^2$	0101
α	p	0010	α^9	$p + p^3$	1010
α^2	p^2	0100	α^{10}	$1 + p + p^2$	0111
α^3	p^3	1000	α^{11}	$p + p^2 + p^3$	1110
α^4	$1 + p$	0011	α^{12}	$1 + p + p^2 + p^3$	1111
α^5	$p + p^2$	0110	α^{13}	$1 + p^2 + p^3$	1101
α^6	$p^2 + p^3$	1100	α^{14}	$1 + p^3$	1001

POLYNOMES MINIMAUX

A chaque élément non nul du corps GF(2^m) on associe un polynôme minimal.

Définition : le polynôme minimal $m_i(p)$ est le polynôme irréductible de plus faible degré dont α^i est racine.

$$
\begin{aligned}
&m_1(p) = m_2(p) = m_4(p) = m_8(p) = 1 + p + p^4 \\
&m_3(p) = m_6(p) = m_9(p) = m_{12}(p) = 1 + p + p^2 + p^3 + p^4 \\
&m_5(p) = m_{10}(p) = 1 + p + p^2 \\
&m_7(p) = m_{11}(p) = m_{13}(p) = m_{14}(p) = 1 + p^3 + p^4 \\
&m_0(p) = 1 + p
\end{aligned}
\tag{163}
$$

LES CODES BCH

• Les codes BCH sont des codes cycliques binaires

• Le polynome générateur $g(p)$ possède parmi ses racines les $2e$ puissances consécutives de α.

• Ainsi, le polynôme générateur $g(p)$ est le produit des polynômes minimaux associés à $\alpha, \alpha^2, \alpha^3, \ldots, \alpha^{2e}$ sans qu'aucun des polynômes ne soit répété :

$$g(p) = \text{PPCM}\Big[m_1(p), m_2(p), m_3(p), \ldots, m_{2e}(p)\Big] \tag{164}$$

Propriétés

$$\begin{aligned} N &= 2^m - 1 \\ N - K &\leq me \\ d_{min} &\geq 2e + 1 \end{aligned} \tag{165}$$

• Liste des polynômes générateurs des codes BCH corrigeant jusqu'à 3 erreurs pour $N \leq 63$

N	K	e	$g(p)$
7	4	1	$p^3 + p + 1$
15	11	1	$p^4 + p + 1$
	7	2	$p^8 + p^7 + p^6 + p^4 + 1$
	5	3	$p^{10} + p^8 + p^5 + p^4 + p^2 + p + 1$
31	26	1	$p^5 + p^2 + 1$
	21	2	$p^{10} + p^9 + p^8 + p^6 + p^5 + p^3 + 1$
	16	3	$p^{15} + p^{11} + p^{10} + p^9 + p^8 + p^7 + p^5 + p^3 + p^2 + p + 1$
63	57	1	$p^6 + p + 1$
	51	2	$p^{12} + p^{10} + p^8 + p^5 + p^4 + p^3 + 1$
	45	3	$p^{18} + p^{17} + p^{16} + p^{15} + p^9 + p^7 + p^6 + p^3 + p^2 + p + 1$

DECODAGE DES CODES BCH

La matrice de parité des codes BCH s'écrit :

$$\mathbf{H} = \begin{pmatrix} 1 & \alpha & \dots & \alpha^{N-1} \\ 1 & \alpha^2 & \dots & \alpha^{2(N-1)} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \alpha^{2e} & \dots & \alpha^{2e(N-1)} \end{pmatrix} \tag{166}$$

On reçoit $r(p)$:

$$\begin{aligned} r(p) &= x(p) + e(p) \\ &= r_0 + r_1 p + \cdots + r_{N-2} p^{N-2} + r_{N-1} p^{N-1} \end{aligned} \tag{167}$$

La première étape consiste à calculer le syndrome **s** défini par un vecteur à $2e$ éléments s_i :

$$s_i = r(\alpha^i) = x(\alpha^i) + e(\alpha^i) = e(\alpha^i) \quad \forall i \quad 1 \leq i \leq 2e \tag{168}$$

DECODAGE DES CODES BCH

Propriétés

$$\begin{aligned} N &= 2^m - 1 \\ N - K &\leq me \\ d_{min} &\geq 2e + 1 \end{aligned} \tag{169}$$

Supposons que le mot d'erreur contiennent v erreurs (avec $v \leq e$):

$$e(p) = p^{j_1} + p^{j_2} + \cdots + p^{j_v} \tag{170}$$

avec $0 \leq j_1 \leq j_2 \cdots \leq j_v \leq N - 1$.

On obtient le système d'équation suivant :

$$
\begin{aligned}
s_1 &= \alpha^{j_1} + \alpha^{j_2} + \cdots + \alpha^{j_v} \\
s_2 &= (\alpha^{j_1})^2 + (\alpha^{j_2})^2 + \cdots + (\alpha^{j_v})^2 \\
s_3 &= (\alpha^{j_1})^3 + (\alpha^{j_2})^3 + \cdots + (\alpha^{j_v})^3 \\
&\vdots \\
s_{2e} &= (\alpha^{j_1})^{2e} + (\alpha^{j_2})^{2e} + \cdots + (\alpha^{j_v})^{2e}
\end{aligned} \tag{171}
$$

Posons $\beta_k = \alpha^{j_k} \quad 1 \leq k \leq v$

$$
\begin{aligned}
s_1 &= \beta_1 + \beta_2 + \cdots + \beta_v \\
s_2 &= (\beta_1)^2 + (\beta_2)^2 + \cdots + (\beta_v)^2 \\
s_3 &= (\beta_1)^3 + (\beta_2)^3 + \cdots + (\beta_v)^3 \\
&\vdots \\
s_{2e} &= (\beta_1)^{2e} + (\beta_2)^{2e} + \cdots + (\beta_v)^{2e}
\end{aligned} \tag{172}
$$

• Définissons le polynome suivant :

$$\sigma(p) = \prod_{k=1}^{v}(1 - \beta_k p)$$
$$= \sigma_0 + \sigma_1 p + \sigma_2 p^2 + \ldots \sigma_v p^v \quad (173)$$

• Ce polynôme est appelé polynôme localisateur d'erreur car les inverses des racines de $\sigma(p)$ sont égales à $\beta_k = \alpha^{j_k}$ et permettent de localiser la position des erreurs j_k.

DECODAGE DES CODES BCH

- On peut alors mettre le système la forme matricielle suivante :

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & \dots & 0 \\ s_2 & s_1 & 1 & 0 & 0 & \dots & 0 \\ s_4 & s_3 & s_2 & s_1 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ s_{2v-4} & s_{2v-5} & \dots & \dots & \dots & \dots & s_{v-3} \\ s_{2v-2} & s_{2v-3} & \dots & \dots & \dots & \dots & s_{v-1} \end{pmatrix} \begin{pmatrix} \sigma_1 \\ \sigma_2 \\ \sigma_3 \\ \vdots \\ \sigma_{v-1} \\ \sigma_v \end{pmatrix} = \begin{pmatrix} s_1 \\ s_3 \\ s_5 \\ \vdots \\ s_{2v-3} \\ s_{2v-1} \end{pmatrix} \quad (174)$$

CODES REED-SOLOMON

• Codes en bloc linéaire non binaire (N, K)

Propriétés

$$N = q - 1 = 2^k - 1$$
$$K = 1, 2, \ldots, N - 1$$
$$d_{min} = N - K + 1$$

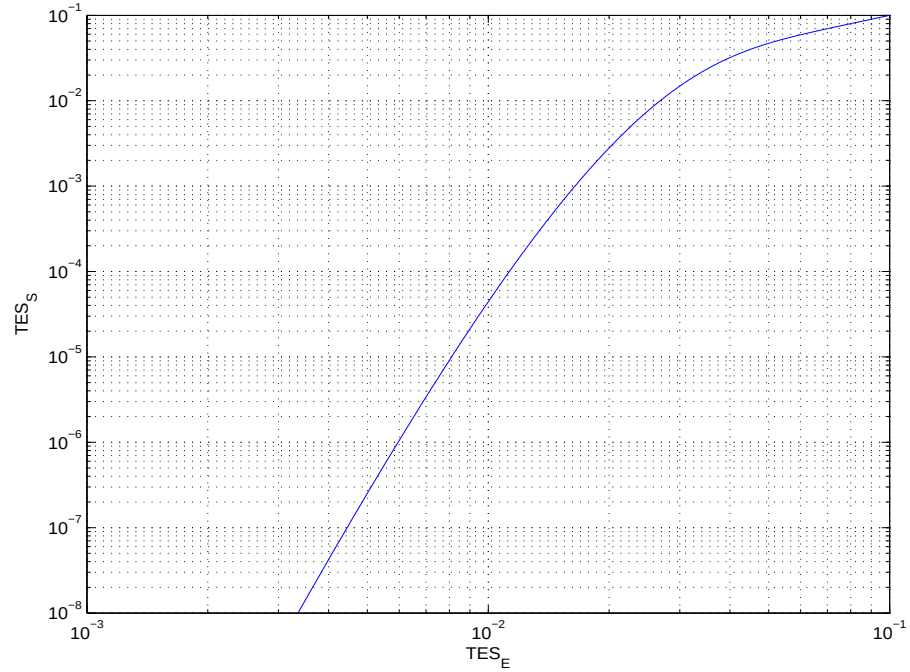
• Exemple $(255, 239, 17)$: permet de corriger jusqu'à 8 octets par bloc.

• Le taux d'erreurs symbole TES_S en sortie d'un décodeur de Reed-Solomon à entrées dures est le suivant :

$$TES_S = \frac{1}{N} \sum_{i=e+1}^{N} iC_i^N (TES_E)^i (1 - TES_E)^{N-i}$$

où TES_E est le taux d'erreurs symbole en entrée.

• $TES_S = f(TES_E)$ pour le code Reed Solomon $(255, 239, 17)$:



COURS 13

Codes convolutifs

LES CODES CONVOLUTIFS

Un code convolutif est un code qui transforme une séquence semi-infinie de mots d'informations en une séquence semi-infinie de mots de code

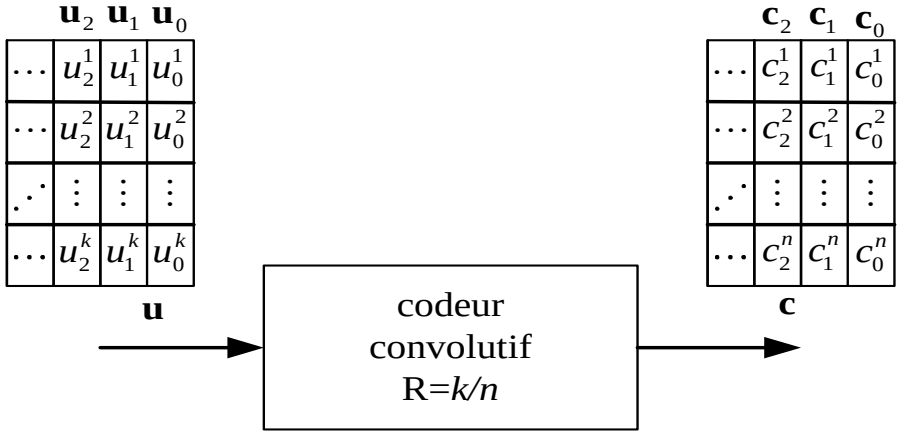
u : séquence de mots d'information de dimension k

$$\mathbf{u} = \mathbf{u}_0, \mathbf{u}_1, \mathbf{u}_2, \ldots \qquad \text{avec} \qquad \mathbf{u}_i = [u_i^1, u_i^2, ..., u_i^k]$$

x : séquence de mots de code de dimension n

$$\mathbf{c} = \mathbf{c}_0, \mathbf{c}_1, \mathbf{c}_2, \ldots \qquad \text{avec} \qquad \mathbf{c}_i = [c_i^1, c_i^2, ..., c_i^n]$$

Le rendement du code convolutif est $\frac{k}{n}$.



REPRESENTATION POLYNOMIALE DES SEQUENCES

$$\mathbf{x}(D) = [x^1(D), x^2(D), ..., x^n(D)]$$

$$\mathbf{u}(D) = [u^1(D), u^2(D), ..., u^k(D)]$$

avec

$$u^j(D) = \sum_{i=0}^{\infty} u_i^j D^i \qquad \text{et} \qquad x^j(D) = \sum_{i=0}^{\infty} x_i^j D^i \qquad \text{avec} \qquad u_i^j, x_i^j \in \mathbb{F}_2$$

Un codeur convolutif binaire de rendement k/n est la réalisation par un circuit linéaire de l'association d'une séquence de mots d'information de k bits $\mathbf{u}(D)$ avec une séquence de mots de code de n bits $\mathbf{x}(D)$.

Nous avons :

$$\mathbf{x}(D) = \mathbf{u}(D)\mathbf{G}(D)$$

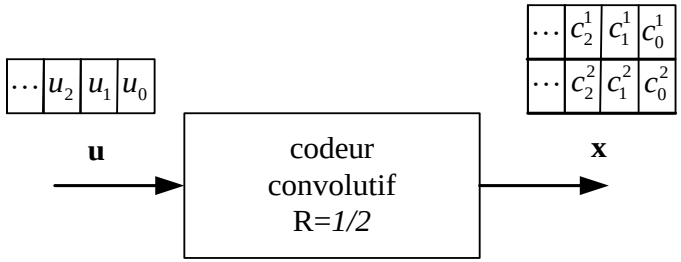
$\mathbf{G}(D)$ est la matrice génératrice utilisée par le codeur. $\mathbf{G}(D)$ est de dimension $k \times n$ et de rang k.

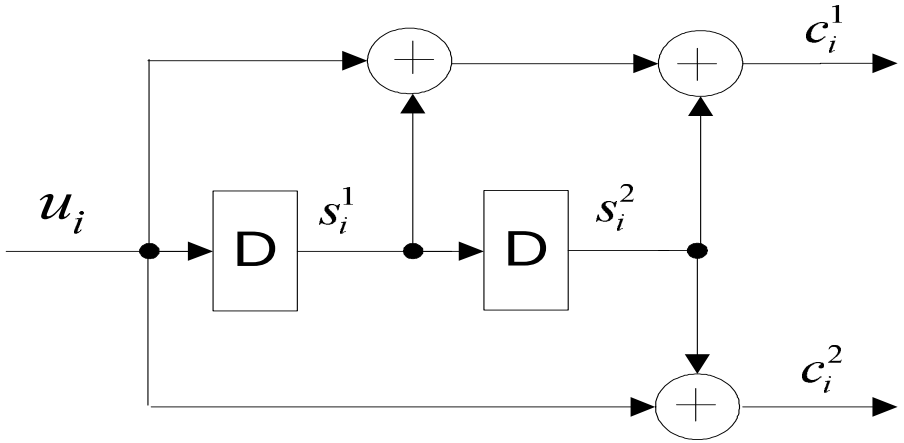
$$\mathbf{G}(D) = \begin{pmatrix} g_{1,1}(D) & g_{1,2}(D) & \cdots & g_{1,n}(D) \\ g_{2,1}(D) & g_{2,2}(D) & \cdots & g_{2,n}(D) \\ \cdots & \cdots & \cdots & \cdots \\ g_{k,1}(D) & g_{k,2}(D) & \cdots & g_{k,n}(D) \end{pmatrix}$$

Les éléments $g_{i,j}(D)$ de la matrice génératrice $\mathbf{G}(D)$ sont des polynômes de D ou des fonctions rationnelles de polynômes de D.

un code convolutif est l'ensemble de toutes les séquences de sortie possibles $\mathbf{x}(D)$ du codeur convolutif

Exemple : codeur convolutif non récursif $k = 1$, $n = 2$ $M = 2$





$$\mathbf{G}(D) = (g_1(D), g_2(D))$$

$$g_1(D) = 1 + D + D^2$$

$$g_1 = [111]_{\text{bin}} = 7_{\text{oct}}$$

$$g_2(D) = 1 + D^2$$
$$g_2 = [101]_{\text{bin}} = 5_{\text{oct}}$$

On a:

$$u(D) = u_0 + u_1 D + u_2 D^2 + \dots$$
$$x^1(D) = x_0^1 + x_1^1 D + x_2^1 D^2 + \cdots = u(D)(1 + D + D^2)$$
$$x^2(D) = x_0^2 + x_1^2 D + x_2^2 D^2 + \cdots = u(D)(1 + D^2)$$

On peut aussi écrire :

$$c_i^1 = u_i + u_{i-1} + u_{i-2}$$
$$c_i^2 = u_i + \qquad + u_{i-2}$$

TABLE DES MEILLEURS CODES CONVOLUTIFS R=1/2

nbr mémoire	code	d_{libre}
1	$(2, 3)$	3
2	$(5, 7)$	5
3	$(15, 17)$	6
4	$(23, 35)$	7
5	$(53, 75)$	8
6	$(133, 171)$	10

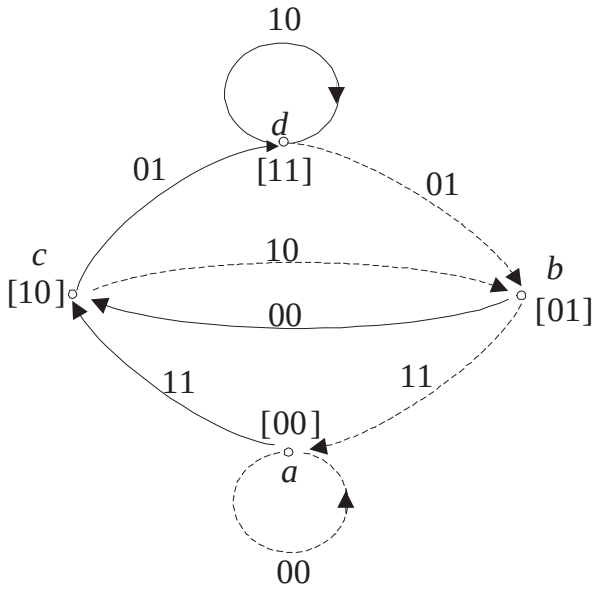
DIAGRAMME DE TRANSITIONS D'ETAT

On définit l'état interne du codeur à l'instant i par un vecteur $\mathbf{s}_i$ de dimension M :

$$\mathbf{s}_i = \left[s_{1i}, s_{2i}, ...s_{Mi}\right] \ .$$

s_{ji} est l'état à l'instant i de la j-ième cellule mémoire.

Diagramme de transition pour le codeur convolutif non récursif (7,5) de rendement 1/2 de l'exemple précédent



Chaque branche est renseignée avec les bits de sortie (ici c_{1i} et c_{2i}). Les branches en traits pointillés et continus correspondent respectivement à un bit d'entrée égal à 0 et à 1.

état interne	s_{1i}	s_{2i}
a	0	0
b	0	1
c	1	0
d	1	1

TREILLIS ELEMENTAIRE

A partir du diagramme de transitions d'état, il est possible de construire le treillis élémentaire du code convolutif. Chaque branche b relie un état de départ $s^{-}(b)$ et un état d'arrivée $s^{+}(b)$.

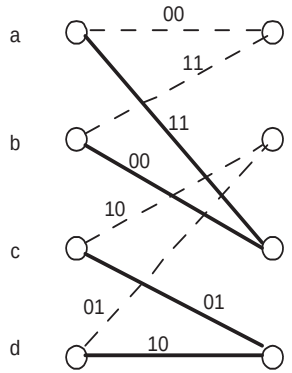
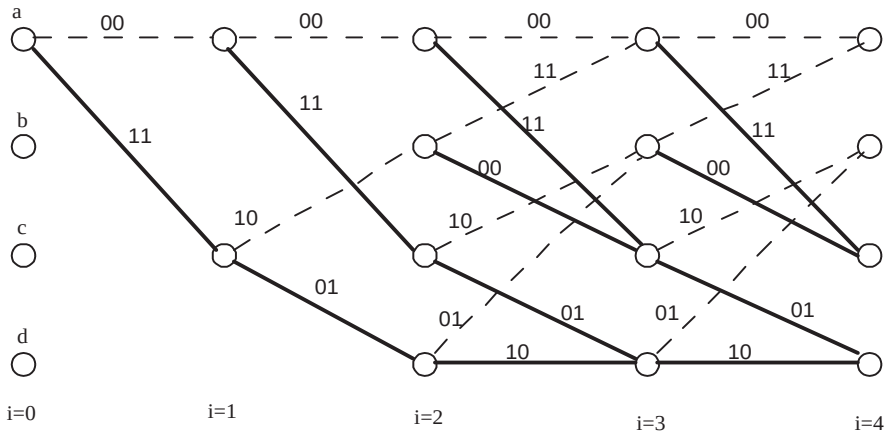


DIAGRAMME EN TREILLIS

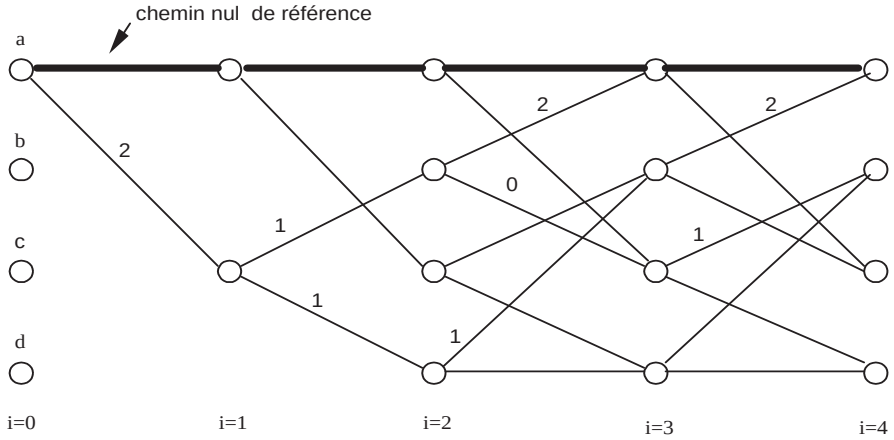
Le diagramme en treillis est un diagramme de transitions d'état où l'abscisse correspond au temps. Il est obtenu simplement en assemblant un nombre infini de treillis élémentaires.

Diagramme en treillis du codeur précédent :



Sur chaque branche on renseigne la valeur des deux bits c_{1i} et c_{2i}. Les traits en gras et en pointillés correspondent respectivement à $u_i = 1$ et $u_i = 0$.

DISTANCE MINIMALE DES CODES CONVOLUTIFS



A partir de ce diagramme on constate qu'un seul chemin quittant le chemin de référence à l'instant $i = 0$ est à la distance 5 du chemin de référence. La distance minimale de ce code convolutif est égale à 5.

La fonction de transfert d'un code convolutif permet de déterminer les propriétés de distance de celui-ci.

ALGORITHME DE VITERBI

L'algorithme de Viterbi est un décodeur ML. Il recherche le chemin ou la séquence d'état le plus probable. La séquence $\hat{\mathbf{x}}$ se déduit alors immédiatement.

cas du canal binaire symétrique

$$Pr(\mathbf{y}|\mathbf{x}) = p^{d_H(\mathbf{y},\mathbf{x})}(1-p)^{N-d_H(\mathbf{y},\mathbf{x})} = (1-p)^N \left(\frac{p}{1-p}\right)^{d_H(\mathbf{y},\mathbf{x})} \tag{175}$$

où $d_H(\mathbf{y},\mathbf{x})$ est la distance de Hamming entre la séquence reçue $\mathbf{y}$ et la séquence $\mathbf{x}$. Comme p est compris entre 0 et 0.5, on a $0 < \frac{p}{1-p} < 1$.

Ainsi maximiser $Pr(\mathbf{y}|\mathbf{x})$ revient à minimiser la distance de Hamming entre $\mathbf{y}$ et $\mathbf{x}$.

cas du décodage à entrées souples d'une séquence reçue pour un canal BBAG

$$y_i = \sqrt{E_s}x_i + n_i \tag{176}$$

avec E_s énergie moyenne par symbole et n_i échantillon bruit blanc gaussien centré de variance $\sigma^2 = \frac{N_0}{2}$.

A la sortie du démodulateur on a :

$$p(y_i|x_i) = \frac{1}{\sqrt{\pi N_0}} \exp\left\{ - \frac{[y_i - \sqrt{E_s}x_i]^2}{N_0} \right\} \tag{177}$$

Comme la fonction logarithme est croissante, au lieu d'utiliser la relation (149) pour déterminer $\hat{\mathbf{x}}$, on peut utiliser :

$$\hat{\mathbf{x}} = \arg\max_{\mathbf{x}} \log Pr(\mathbf{y}|\mathbf{x}) \tag{178}$$

On a alors :

$$\log Pr(\mathbf{y}|\mathbf{x}) = \sum_{i=0}^{N-1} \log Pr(y_i|x_i) \tag{179}$$

• On obtient une première version du décodeur à maximum de vraisemblance :

$$\begin{aligned} \hat{\mathbf{x}} &= \arg\max_{\mathbf{x}} \sum_{i=0}^{N-1} \left\{ - \frac{[y_i - \sqrt{E_s}x_i]^2}{N_0} \right\} \\ &= \arg\min_{\mathbf{x}} \sum_{i=0}^{N-1} (y_i - \sqrt{E_s}x_i)^2 \end{aligned} \tag{180}$$

• Une seconde version du décodeur à maximum de vraisemblance est obtenue en approximant les calculs :

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \sum_{i=0}^{N-1} |y_i - \sqrt{E_s} x_i| \tag{181}$$

Cette version sous optimale donne cependant de bonnes performances en pratique.

• Dans le cas où $x_i = \pm 1)$ il est possible d'obtenir une troisième version exacte :

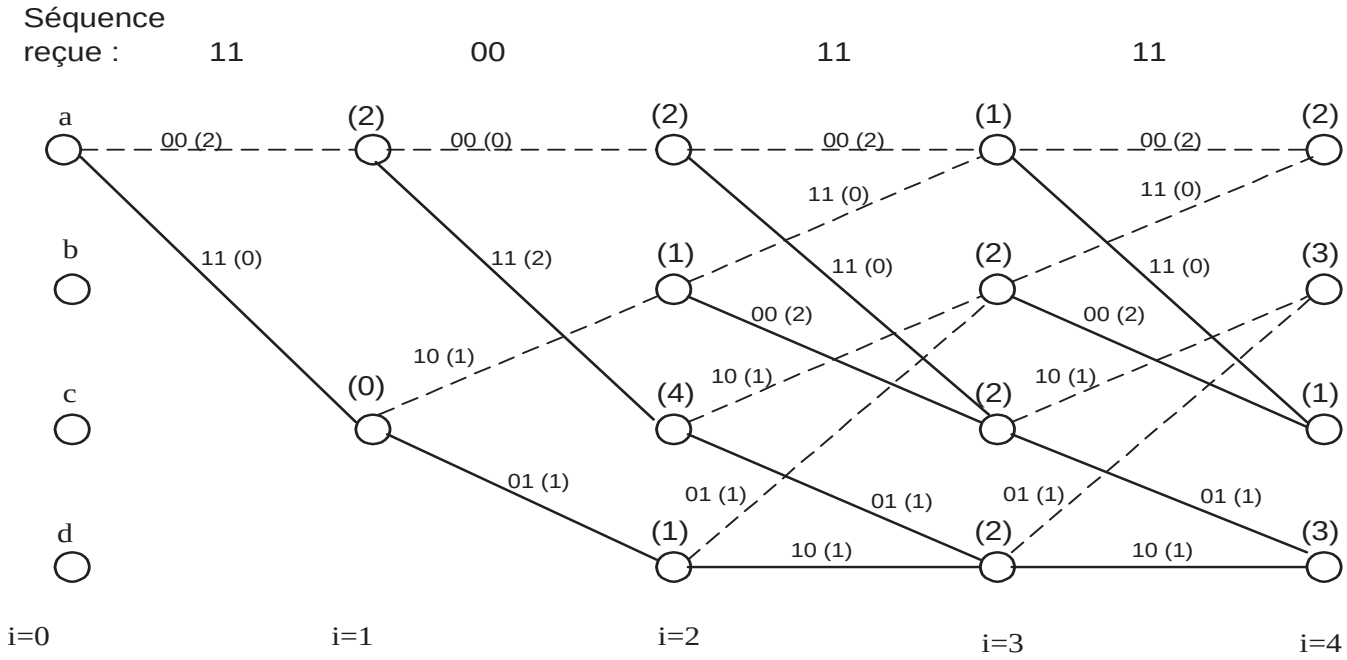
$$\begin{aligned}
\hat{\mathbf{x}} &= \arg\max_{\mathbf{x}} \log Pr(\mathbf{y}|\mathbf{x}) \\
&= \arg\max_{\mathbf{x}} \sum_{i=0}^{N-1} -(y_i - \sqrt{E_s} x_i)^2 \\
&= \arg\max_{\mathbf{x}} \sum_{i=0}^{N-1} -y_i^2 + 2\sqrt{E_s} y_i x_i - x_i^2 E_s \\
&= \arg\max_{\mathbf{x}} \sum_{i=0}^{N-1} y_i x_i
\end{aligned} \tag{182}$$

• L'algorithme de Viterbi évite de calculer les métriques associées à chacune des séquences possibles.

• A chaque section du diagramme en treillis, on élimine les chemins qui ne peuvent pas être le chemin le plus vraisemblable.

• Supposons la séquence en entrée du codeur soit 1001. La séquence en sortie est 11 10 11 11. On considère que la séquence reçue est 11 00 11 11

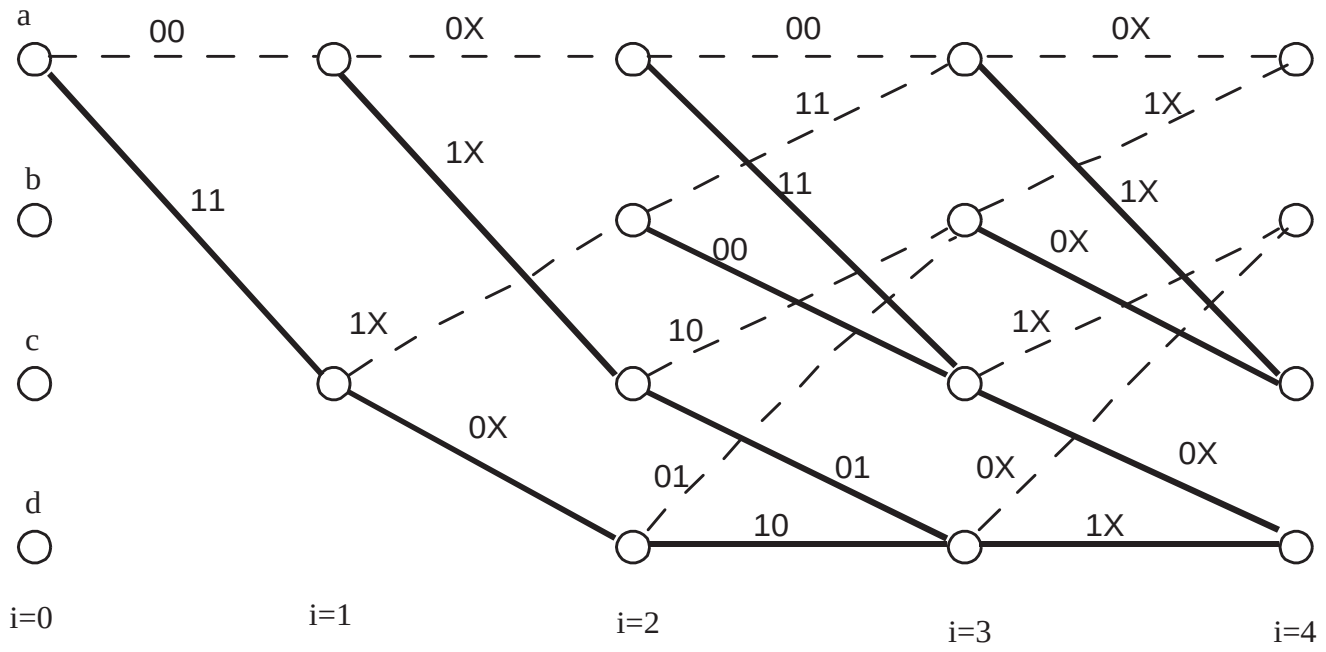
• A chaque étape l'algorithme de Viterbi calcule pour chaque nouvelle branche la distance de Hamming entre le couple de bits reçus et le couple de bits associés à la branche considérée. Puis il calcule les $2^k M$ métriques cumulées et ne conserve qu'un seul chemin par noeud baptisé chemin survivant.


Séquence
reçue : 11 00 11 11
a
b
c
d
(2)
(2)
(1)
(2)
(1)
(2)
(3)
(0)
(4)
(2)
(1)
(1)
(2)
(3)
00 (2)
00 (0)
00 (2)
00 (2)
11 (0)
11 (0)
11 (0)
11 (0)
11 (2)
00 (2)
00 (2)
10 (1)
10 (1)
10 (1)
01 (1)
01 (1)
01 (1)
01 (1)
10 (1)
10 (1)
i=0
i=1
i=2
i=3
i=4

POINCONNAGE DES CODES CONVOLUTIFS

• A partir d'un code convolutif on peut réaliser des codes convolutifs de rendement supérieur.

• Exemple : codeur R=2/3 obtenu à partir du codeur précédent.



• Le poinçonnage s'obtient ici en supprimant un symbole sur 4 en sortie du codeur de rendement 1/2.

COURS 14

Introduction aux modulations numériques

INTRODUCTION AUX MODULATIONS NUMERIQUES

- L’opération de modulation consiste à adapter le signal à émettre au canal de transmission.
- Translate le spectre du signal autour d’une fréquence porteuse.
- L’information est portée par l’amplitude, la phase, la fréquence

MODULATION A DEPLACEMENT D'AMPLITUDE

• La modulation à déplacement d'amplitude (MDA) ou *pulse amplitude modulation* (PAM) en anglais établit une correspondance entre les symboles a_k et le signal modulé comme suit :

$$\begin{aligned} x(t) &= \sum_{k=-\infty}^{\infty} a_k p(t-kT) \\ &= \sum_{k=-\infty}^{\infty} a_k g(t-kT) cos(\omega_0 t) \end{aligned} \tag{183}$$

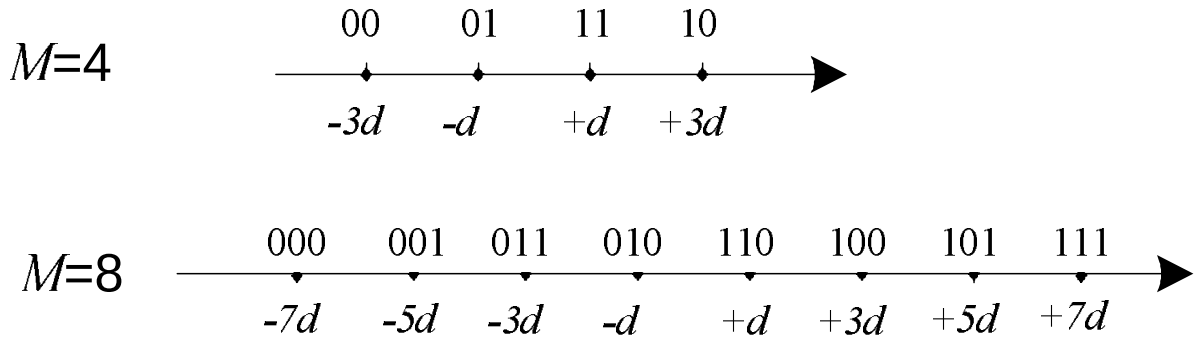
$g(t)$ et $p(t) = g(t)cos(\omega_0 t)$ sont les formes d'onde du signal respectivement avant et après la modulation.

• les symboles a_k prennent leurs valeurs dans l'alphabet $\{\pm d, \pm 3d, \ldots, \pm(M-1)d\}$ (NRZ multiniveaux).

• Le signal $x(t)$ peut également s'écrire sous la forme suivante :

$$x(t) = \sum_{k=-\infty}^{\infty} \Re\left\{ a_k g(t-kT) e^{j\omega_0 t} \right\} \tag{184}$$

• Représentation géométrique des signaux PAM pour $M = 4$ et $M = 8$:



Sur le canal gaussien, le taux d'erreurs symbole est donné par :

$$TES = \frac{M-1}{M} erfc\left(\sqrt{\frac{3log_2 M}{M^2-1}\frac{E_b}{N_0}}\right) \tag{185}$$

• Le codage de Gray consiste à choisir les mots binaires associés à deux points adjacents afin qu'ils ne diffèrent que d'un seul bit. Ainsi, une erreur symbole entre deux points adjacents n'engendre qu'une erreur bit sur les $log_2 M$ bits du mot binaire.

• Avec un codage de Gray, on a donc la relation suivante entre le TEB et le TES :

$$TEB = \frac{TES}{log_2 M} \tag{186}$$

MODULATION A DEPLACEMENT DE PHASE

• La modulation à déplacement de phase (MDP) ou *phase shift keying* (PSK) en anglais consiste à moduler la phase de la porteuse ϕ_k par le symbole à transmettre b_k:

$$\begin{aligned} x(t) &= \sum_{k=-\infty}^{\infty} Ag(t-kT)cos\left(\omega_0 t + \frac{2\pi b_k}{M} + \phi_0\right) \\ &= \sum_{k=-\infty}^{\infty} Ag(t-kT)cos\left(\omega_0 t + \phi_k + \phi_0\right) \end{aligned} \tag{187}$$

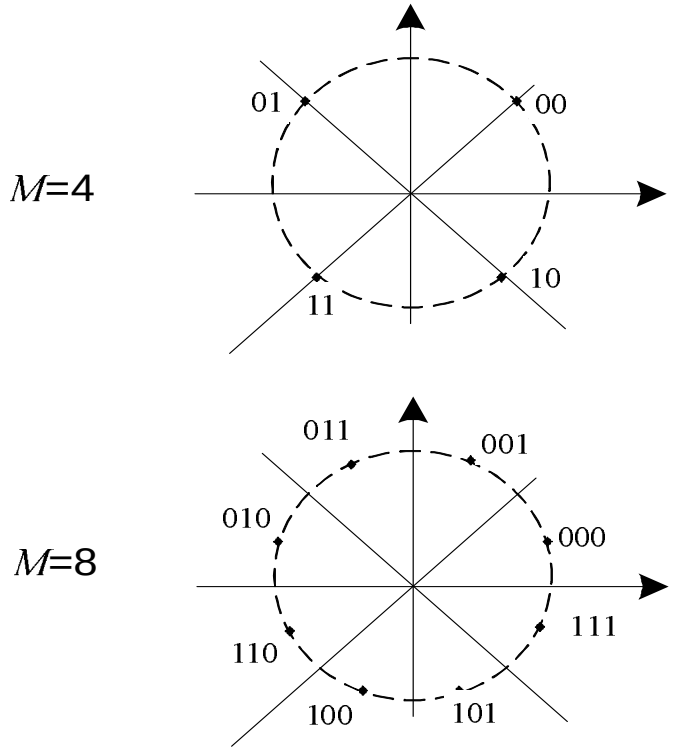
$b_k \in \{0, 1, \ldots, M-1\}$ et $g(t)$ est la forme d'onde du signal en bande de base. ϕ_0 est une phase de référence. On choisit en général $\phi_0 = \pi/M$.

exemple : $M = 4$: la phase ϕ_k peut prendre les valeurs suivantes : $0, \frac{\pi}{2}, \pi$ et $\frac{3\pi}{2}$.

Le signal $x(t)$ peut également s'écrire sous la forme suivante :

$$x(t) = \sum_{k=-\infty}^{\infty} \Re\left\{a_k g(t-kT)e^{j\omega_0 t}\right\} \qquad \text{avec} \qquad a_k = A\exp\left\{j\left(\frac{2\pi b_k}{M} + \phi_0\right)\right\}$$

• Représentation géométrique des signaux PSK pour $M = 4$ (QPSK) et $M = 8$ (8PSK)



MODULATION D'AMPLITUDE DE DEUX PORTEUSES EN QUADRATURE(QAM)

• La modulation d'amplitude de deux porteuses en quadrature(MAQ) ou *quadrature amplitude modulation* (QAM) en anglais consiste à moduler simultanément l'amplitude et la phase de la porteuse par le symbole à transmettre a_k:

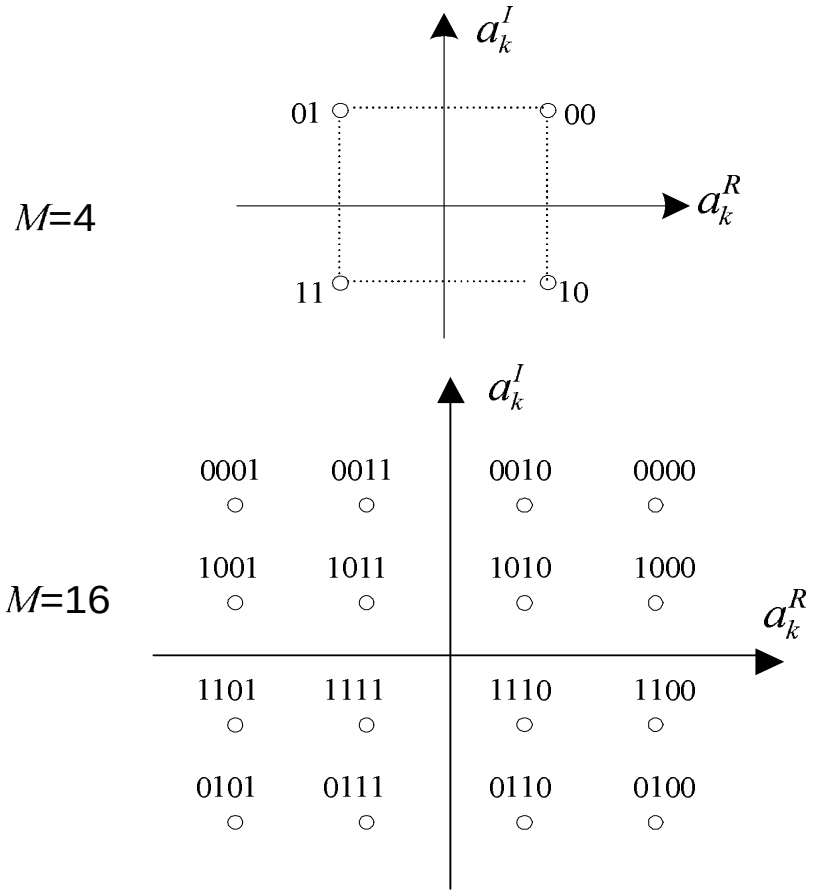
$$\begin{aligned} x(t) &= \sum_{k=-\infty}^{\infty} \Re\left\{ a_k g(t-kT) e^{j\omega_0 t} \right\} \\ &= \sum_{k=-\infty}^{\infty} \Re\left\{ (a_k^R + j a_k^I) g(t-kT) e^{j\omega_0 t} \right\} \\ &= \sum_{k=-\infty}^{\infty} a_k^R g(t-kT) cos\omega_0 t - a_k^I g(t-kT) sin\omega_0 t \\ &= \sum_{k=-\infty}^{\infty} v_k g(t-kT) cos(\omega_0 t + \phi_k) \end{aligned} \tag{188}$$

où

$$v_k = \sqrt{(a_k^R)^2 + (a_k^I)^2} \qquad \text{et} \qquad \phi_k = \arctan \frac{a_k^I}{a_k^R}$$

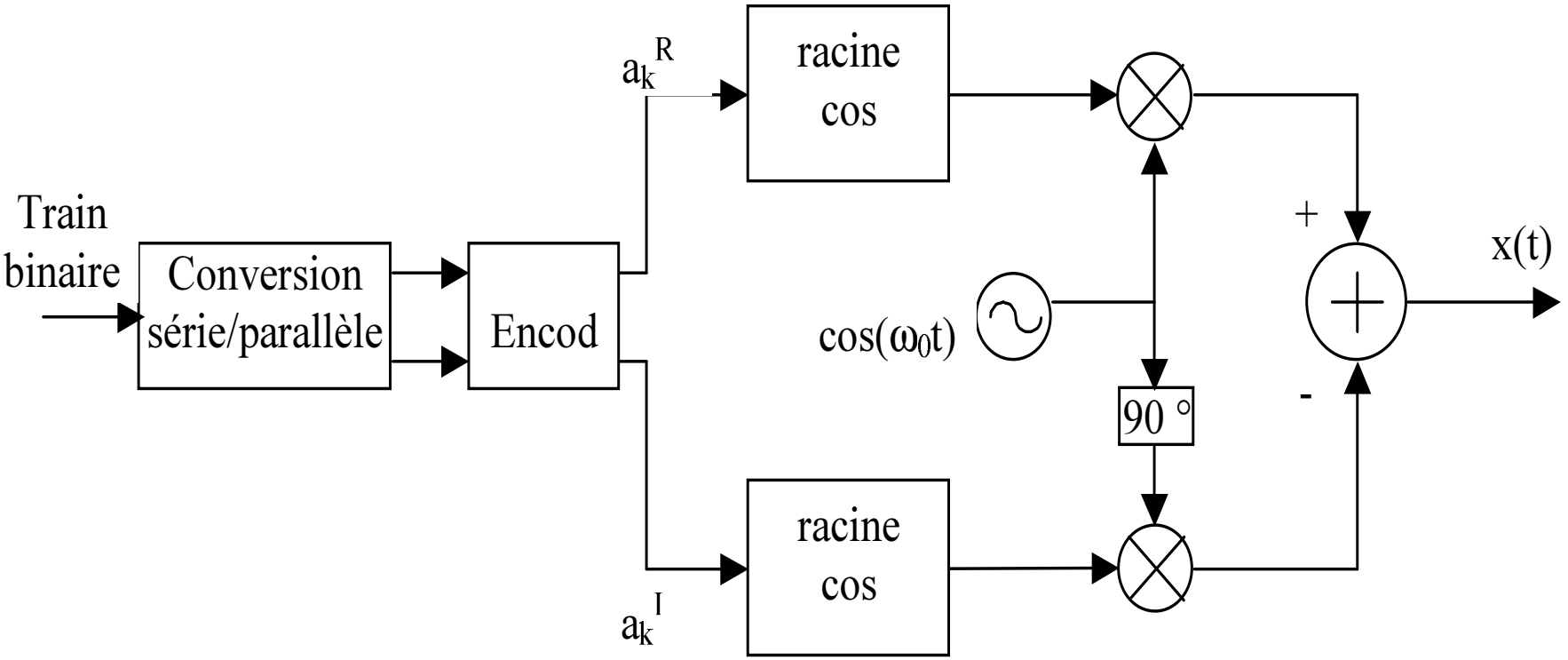
• Cette modulation permet de réaliser des transmissions numériques d'efficacité spectrale élevée.

• Représentation géométrique des signaux QAM pour $M = 4$ $M = 16$



• Nous pouvons observer que la modulation $QAM4$ est identique à la modulation à déplacement de phase à 4 états de phase QPSK.

SYNOPTIQUE MODULATEUR QAM



TAUX D'ERREURS SYMBOLE D'UNE MODULATION QAM

• Pour les constellations rectangulaires ($M = 2^k$), avec $k = log_2 M$ nombre pair de bits par symbole, la modulation QAM est équivalente à deux modulations PAM en quadrature ayant chacune $\sqrt{M} = 2^{k/2}$ points.

• Comme les signaux en phase et en quadrature peuvent être parfaitement séparés à la démodulation, le TES d'une modulation QAM peut s'obtenir aisément à partir du TES de la modulation PAM composée de $\sqrt{M}$ points :

$$TES = 1 - (1 - TES_{PAM\sqrt{M}})^2 \tag{189}$$

où $TES_{PAM\sqrt{M}}$ est le taux d'erreurs symbole de la modulation PAM avec la moitié de la puissance moyenne d'une modulation QAM équivalente :

$$TES_{PAM\sqrt{M}} = \left(1 - \frac{1}{\sqrt{M}}\right) erfc\left(\sqrt{\frac{3log_2 M}{2(M-1)}\frac{E_b}{N_0}}\right) \tag{190}$$

TAUX D'ERREURS SYMBOLE D'UNE MODULATION QAM

• Le TES d'une modulation QAM s'écrit suit :

$$TES = 2\left(1-\frac{1}{\sqrt{M}}\right) erfc\left(\sqrt{\frac{3log_2 M}{2(M-1)}\frac{E_b}{N_0}}\right)\left[1-\frac{1}{2}\left(1-\frac{1}{\sqrt{M}}\right) erfc\left(\sqrt{\frac{3log_2 M}{2(M-1)}\frac{E_b}{N_0}}\right)\right]$$

• Pour la modulation QAM 4, on obtient par exemple :

$$TES = 1-\left(1-\frac{1}{2}erfc\left(\sqrt{\frac{E_b}{N_0}}\right)\right)^2 = erfc\left(\sqrt{\frac{E_b}{N_0}}\right) - \frac{1}{4}erfc\left(\sqrt{\frac{E_b}{N_0}}\right)^2 \qquad (191)$$

• Par ailleurs avec un codage de Gray, le taux d'erreurs bit pour la modulation QAM 4 est égal à :

$$TEB = \frac{1}{2}erfc\left(\sqrt{\frac{E_b}{N_0}}\right) \qquad (192)$$

• En négligeant le second terme de (191), on retrouve la relation $TEB = TES/2$

TES ET TEB D'UNE MODULATION QAM