Introduction
à la
Théorie de l'Information

➡ **Une information est un couple constitué:**

d'une représentation matérielle, qui en constitue le formant

et d'un ensemble d'interprétations, qui en constitue le formé

dont la nature, événementielle, consiste en un changement d'état qui, par l'occurence de cette représentation matérielle, provoque l'activation du champ interprétatif correspondant, selon les règles fixées par un code préétabli.

Georges Ifrah - Histoire universelle des chiffres

☞ **Seule la composante matérielle (formant) d'une information fait l'objet d'une communication: ce n'est pas le sens (formé) que l'on transmet**

☞ **L'information est la troisième dimension universelle après la matière et l'énergie. L'information n'est autre que la néguentropie (structure ordonnée)**

☞ **Etymologie: informare= donner une forme....**

➡ **Quantité d'information**

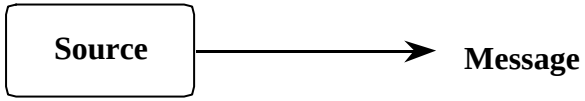
Mesure quantitative de l'incertitude d'un message en fonction du degré de probabilité de chaque signal composant ce message

- **Information (suite...)**
  - Séquence de signaux, correspondant à des règles de combinaisons précises, transmise entre une source et un collecteur par l'intermédiaire d'un canal
  - Ecrit, fait, notion ou instruction susceptible d'être traitée en tout ou partie par des moyens automatiques.
  - Renseignements obtenus de quelqu'un ou sur quelqu'un ou quelque chose, en particulier nouvelle communiquée par la presse, la radio,...
- **Message**
  - Lot d'information formant un tout intelligible ou exploitable et transmis en une seule fois
  - Séquence de signaux qui correspondent à des règles de combinaisons précises et qu'une source transmet à un collecteur par l'intermédiaire d'un canal
- **Signal**
  - Phénomène physique porteur d'une information et pouvant représenter des données
  - Variation d'une grandeur de nature quelconque grâce à laquelle, dans un équipement, un élément en influence un autre
  - Signe convenu pour avertir, annoncer, donner un ordre.

➡On ne s'intéresse ici qu'à l'aspect QUANTITATIF

➡L'aspect qualitatif de l'information, son "intérêt" est subjectif

➡Mesure basée sur la probabilité d'occurrence d'un événement, du message



➡L'attitude du collecteur de l'information est probabiliste

☞ La communication n'a d'objet que si le contenu du message est inconnu

➡Pour un événement X de probabilité P(X)

☞ I(X) = f (1 / P(X)) pour que I(X) croît quand P(X) décroît

☞ Il faut que I(X) soit toujours POSITIVE et ADDITIVE

➟ on choisit donc f= log

$$I(X) = \log \frac{1}{P(X)} = -\log(P(X))$$

➟ si $\log_e$ unité : Nat

➟ si $\log_{10}$ unité : Hartley

➟ si $\log_2$ unité: bit

- X — Symbole émis par une source
- Y — Symbole reçu par l'observateur au collecteur
- $P(x_k)$ — Probabilité que $X = x_k$
- $P(x_k / y_j)$ — Probabilité d'avoir émis $X = x_k$ si on a reçu $Y = y_j$
- $P(x_k ; y_j)$ — Probabilité émettre $X = x_k$ et de recevoir $Y = y_j$

$$I(x_k) = - \log_2 P(x_k) \text{ bits}$$

- **Alphabet:** Ensemble fini de symboles appelés lettres
  - ex : a b c d e ...
  - alphabet binaire : 0 1
- **Message:** Suite finie de symboles ou lettres
  - ex : Beuchot, 0110100110
- **Source de messages:** Ensemble de TOUS les messages susceptibles d'être formés à partir d'un alphabet
  - Source discrète ou continue
  - ex : dictionnaire
  - alphabet N° 5 (AI5) à 7 bits (0000000 à 1111111)
- **Extension d'une source**
  - Soit une source codée par un alphabet de taille k *par exemple k = 2 [0,1]*

  Une séquence de longueur l de lettres de cet alphabet constitue une nouvelle source appelée l ème extension de k
  - ex : Le code binaire correspondant à l'alphabet AI5 est une extension de taille 7 de l'alphabet binaire.

➡ **Source X de messages $x_1, x_2, \ldots x_i \ldots x_n$ de probabilité $p_1, p_2 \ldots p_i \ldots p_n$**

$$p_i > 0 \quad \forall i \quad \text{et} \quad \sum_{i=1}^{n} p_i = 1$$

$$\text{Information moyenne} \quad H = \sum_{i=1}^{n} p_i I(x_i)$$

➡ **Axiomes de Fadeev et Fenstein**

- ☞ si $p_1 = p_2 = .. p_i .. = p_n = 1/n$, H est fonction monotone croissante de n
- ☞ Soit 2 ensembles $X(x_i)$ et $Y(y_j)$ sources de messages indépendantes

  et $p(x_i) = 1/n \quad \forall i$ et $p(y_j) = 1/m \quad \forall j$

  $$H(\ldots, 1/mn, \ldots) = H(\ldots, 1/n, \ldots) + H(\ldots, 1/m, \ldots)$$
- ☞ (voir polycop) . On n'apporte pas plus d'information en fractionnant les expériences ....
- ☞ Pour un alphabet binaire, H(p, 1-p) est une fonction continue de p sur [0,1]

➡ **La seule fonction vérifiant ces 4 axiomes est :**

$$H(p_1,\dots p_i,\dots p_n) = - C\sum_{i=1}^{n} p_i \log_a p_i$$

$$\text{Information moyenne } H = \sum_{i=1}^{n} p_i I(x_i)$$

- **C constante arbitraire > 0 et a base du logarithme**

$$I(p_i) = - C \log_a p_i$$

- **I : information moyenne associée au résultat d'un épreuve**
- **H : information moyenne, espérance mathématique de I**
  - **ENTROPIE (Shannon, 1948)**

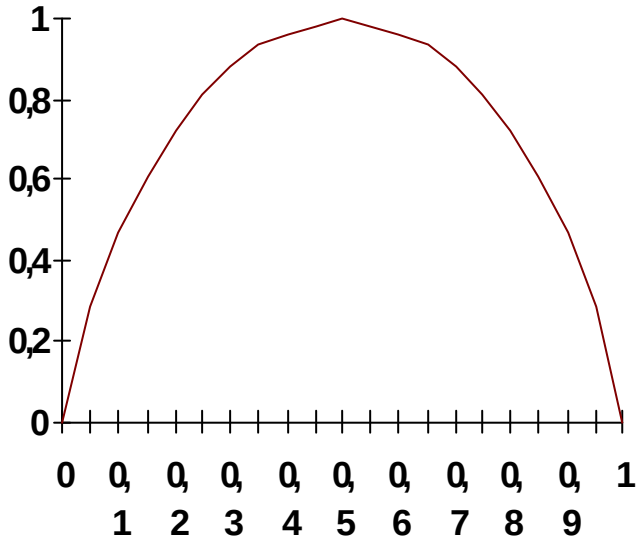
Alphabet + séparateur: 27 symboles

- si équiprobables $H = \frac{1}{27}\sum_{i=1}^{27} \log_e \frac{1}{1/27} = \log_2 27 = 4{,}7549$ bit par lettre
- en réalité, les lettres ne sont pas équiprobables et H = 3,98 bit par lettre
  - cette inégalité fait perdre 770 bits pour 1000 lettres

Source binaire

- quelconque
  - si p = 0,6 pour X_0 et p = 0,4 pour X_1
    - $I(X_0) = \log_2 0{,}6 = 0{,}73$
    - $I(X_1) = \log_2 0{,}4 = 1{,}32$ soit H = 0,97 bit
- symétrique
  - p = 0,5 pour X_0 et X_1
    - $I(X_0) = \log_2 0{,}5 = 1 = I(X_1)$ soit H = 1 bit

dé équiprobable H = 2,585 bits



➡ **l'entropie est maximale si les messages sont équiprobables**

➡ **Loi composée**

$$H(X,Y) = -\sum_{i=1}^{n}\sum_{j=1}^{m} p(x_i, y_j)\log\left(p(x_{i,}y_j)\right)$$

☞ **généralisable à N sources**

$$H(X,Y) \leq H(X) + H(Y)$$

➠ **avec égalité si indépendance des sources**

➠ **généralisé par**

$$H(X_1,..X_n) \leq \sum_{i=1}^{n} H(X_i)$$

Loi conditionnelle

$p\left(Y = y_j / X = x_i\right)$

$$H\left(Y/X = x_i\right) = -\sum_{j=1}^{k} p\left(y_j/x_i\right) \log\, p\left(y_j/x_i\right)$$

$$H\left(Y/X\right) = -\sum_{i=1}^{n} p(x_i) \sum_{j=1}^{m} p\left(y_j/x_i\right) \log\, p\left(y_j/x_i\right)$$

$$H\left(Y/X\right) = -\sum_{i=1}^{n} \sum_{j=1}^{m} p(x_i, y_j) \log\, p\left(y_j/x_i\right)$$

$$H(X, Y) = H(Y) + H\left(X/Y\right) = H(X) + H\left(Y/X\right)$$

$H\left(Y/X\right) \leq H(Y)$ égalité si indépendance

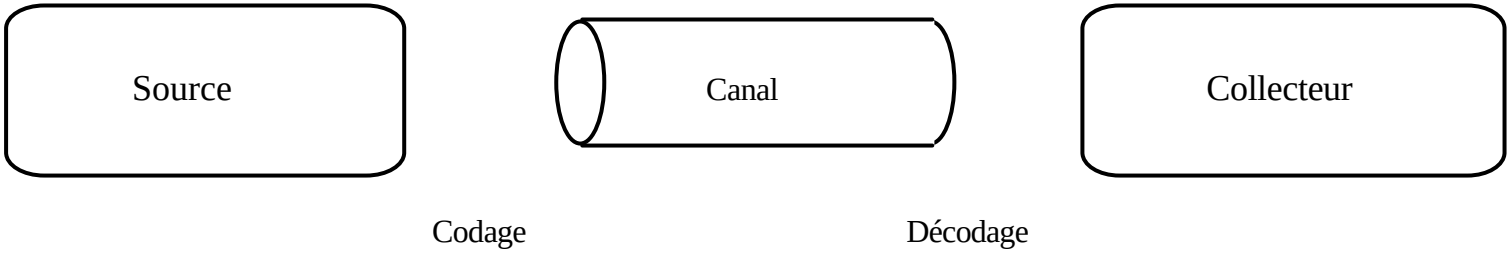
- L'entropie conditionnelle de Y relativement à $X = x_i$ est l'information moyenne apportée par Y si la source X a émis x_i
- Si X et Y sont indépendantes, la connaissance de X n'apporte aucune information au sujet de Y. L'entropie conjointe H(X,Y) est alors la somme des entropies des sources indépendantes.
- Si les sources sont liées, la connaissance de l'une apporte des information sur l'autre et l'entropie conjointe est inférieure à la somme des entropies de chaque source.

$$I(X,Y) = H(X) - H(X/Y)$$

$$= H(Y) - H(Y/X)$$

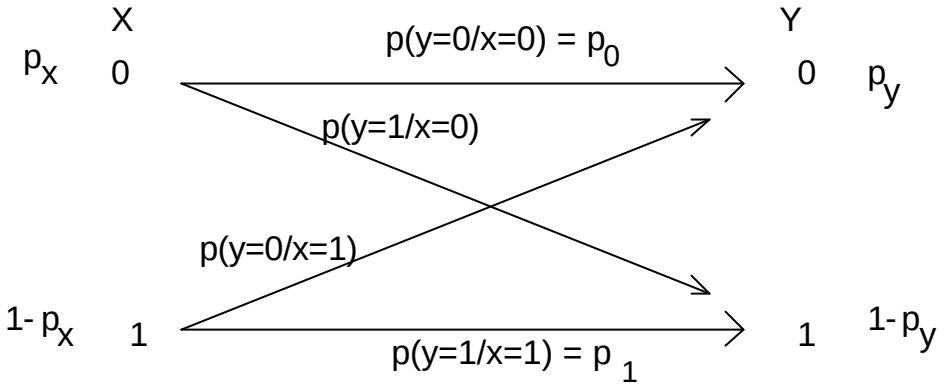
$$I(X,Y) = I(Y,X)$$

- **Notations**
  - **H(X) = entropie de la source**
  - **H(Y) = entropie du collecteur**
  - **H(X/Y) = partie de l'information source récupérée au vu de la sortie ou équivocation**
  - **H(Y/X) = bruit ou erreur sur le canal**
- **I(X,Y) est l'incertitude à priori sur X moins l'incertitude sur X lorsque Y a été réalisée**
  - **Pour un canal sans bruit I(X;Y) = H(X) = H(Y)**
  - **L'incertitude sur X décroît par la connaissance de Y**
  - **I(X,Y) est l'apport d'information de Y au sujet de X**
  - **0 £ I(X,Y) . I(X,Y) = 0 si les sources sont indépendantes; Y ne dit rien de X**



- **Au collecteur on observe Y qui est une conséquence de X**
  - **Stockage**
  - **Transmission**
- **Dans le canal l'information peut être perturbée, modifiée ....**

- $p_y = p_0 p_x + (1-p_1)(1-p_x)$
  - si $p_x = 0,5$ $p_y = 0,5\,(1 + p_0 - p_1)$
- **Le canal est symétrique si $p_0 = p_1$ alors $p_y = 0,5$**



$$H(Y/X) = -\sum_{i=1}^{2} p(x_i) \sum_{j=1}^{2} p(y_j/x_i) \log p(y_j/x_i)$$

$$H(Y/X) = -\left[\begin{array}{l} p(x=0)\left(p(0/0)\log p(0/0) + p(1/0)\log p(1/0)\right) + \\ p(x=1)\left(p(1/1)\log p(1/1) + p(0/1)\log p(0/1)\right) \end{array} \right]$$

si $p(x) = 0,5$ et $p(0/0) = p(1/1) = p$

$H(Y/X) = p \log (p) + (1-p) \log (1-p)$

p	H (Y / X)	I (X,Y)
1	0	1
0,9999	0,001473	0,9985
0,999	0,011408	0,988592
0,99	0,0808	0,9192
0,9	0,4690	0,5310
0,5	1	0

Le Canal de Transmission

- **Entre source et collecteur caractérisé par :**
  - alphabets d'entrée et de sortie
  - probabilités de transition
  - ensemble d'états

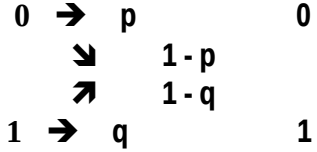
- **si le canal est SANS MEMOIRE les probabilités de transition sont indépendantes de l'état du canal**
  - Nous ne traiterons que ce cas
  - Une "mémoire" est un canal sans mémoire

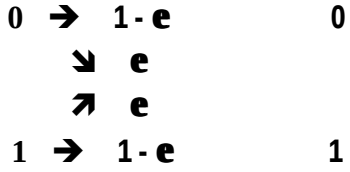
- sans perte (sortie implique l'entrée)
- déterministe (entrée détermine la sortie)
- sans erreur (déterministe et sans perte)

symétrique (matrice de transition symétrique)

➡ exemple : canal binaire symétrique

- (B.S.C. binary symetric channel)



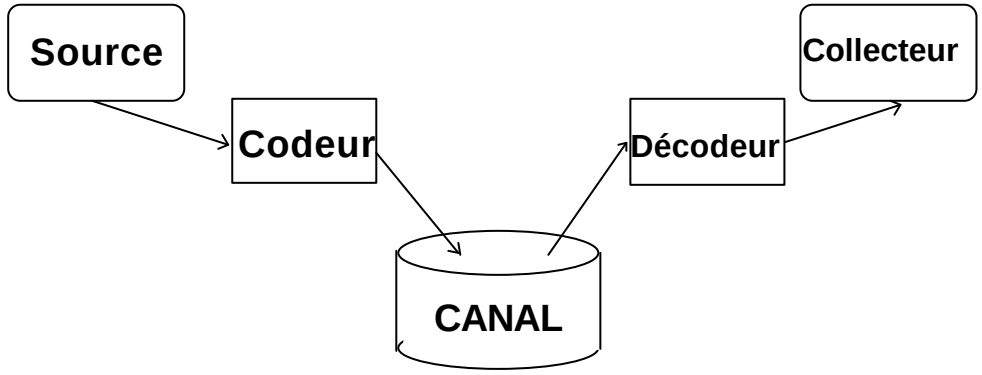


- **Entrée prise dans l'extension de l'alphabet**
  - ex : $p(00/00) = p^2$, $p(10/00) = qp$
  - Codage multiniveaux
- **exemple:**
  - matrice de probabilité de transition pour BSC

	00	01	10	11
00	$(1-e)^2$	$e(1-e)$	$(1-e)$	e^2
01	$e(1-e)$	$(1-e)^2$	e^2	$e(1-e)$
10	$e(1-e)$	e^2	$(1-e)^2$	$e(1-e)$
11	e^2	$e(1-e)$	$e(1-e)$	$(1-e)^2$

- **CANAL AVEC MEMOIRE**
  - La matrice de transition dépend de l'état du canal, donc des transitions antérieures
  - Application : paquets d'erreurs

- **I(X,Y) = H(Y) - H(Y/X) = H(X) - H(X/Y)**
  - Cette information dépend
    - de la source
    - de la matrice de transition fixée par le canal
  - cette matrice de transition est imposée.
- **On recherche le maximum d'information transmise. Pour cela il faut jouer sur la source par un codage approprié.**
  - analogie: adaptation d'impédance
  - On utilisera donc un codeur et un décodeur comme interfaces avec le canal



$$H(Y) = - \sum_j p_{y_j} \log p_{y_j}$$

$$H(Y) = - \sum_j \left[\sum_i p_{x_i} \log p_{y_j/x_i} \right] \log \left(\sum_i p_{x_i} \log p_{y_j/x_i} \right)$$

$$H(Y/X) = - \sum_i \sum_j p_{x_i} p_{y_j/x_i} \log p_{y_j/x_i}$$

$$\Rightarrow C = \underset{p(x_i)}{\text{Max}}[X,Y]$$

$$I(X,Y) = - \sum_i \sum_j p_{x_i} p_{y_j/x_i} \left[\log \sum_i p_{x_i} p_{y_j/x_i} - \log p_{y_j/x_i} \right]$$

$$I(X,Y) = \sum_i \sum_j p_{x_i} p_{y_j/x_i} \log \frac{p_{y_j/x_i}}{\sum_i p_{x_i} p_{y_j/x_i}}$$

➡ **Si L est la taille de l'alphabet B du collecteur**

$$C = \text{Max}\,(H(Y) - H(Y/X))$$

☞ **H(Y) est maximal pour une source équiprobable**

$p(x_i) = 1 / L$ et $\sum_{i=1}^{L} p_{x_i} = 1$

$$C = \log L + \sum_{j=1}^{L} p\left(y_j / x_i\right) \log p\left(y_j / x_i\right)$$

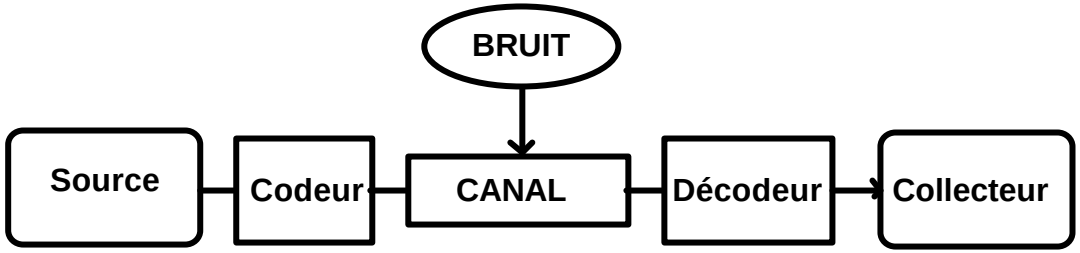
➡ **Pour un canal binaire symétrique (BSC)**

$$C = 1 + p \log p + (1-p) \log (1-p) \quad 1$$

☞ si p = 0,5 log p = log (1-p) = log 0,5 = - 1 **C = 0 bit**

☞ si p = 0,99 log p = -0,014495

1-p = 0,01 log(1-p) -6,643856 **C = 0,9192 bit**



- La matrice de transition du canal est liée à des perturbations induites par l'ajout d'une information parasite "le bruit" dans le canal.
- **Exemple : Canal binaire**

Source	1	1	0	1	1	0	0	1	0
bruit	0	0	0	1	0	1	0	0	0
Collecteur	1	1	0	0	1	1	0	1	0
erreur				+		+			

- Nous observons 2 erreurs de décodage. Nous allons rechercher la probabilité d'erreur p_{ei} sur un mot X_i et la probabilité moyenne d'erreurs p_e.

$$P_{e_i} = \sum_{Y_j} p(Y_j / X_i)$$

$$P_e = \sum_{i=1}^{M} p(X_i)\, p_{e_i}$$

- **A l'entrée du canal sont placés M mots X_i de longueur N. A chaque mot reçu Y_i on assigne un mot émis X_i. Il y a erreur si on reçoit Y_j .**
- **Le décodeur, placé à la sortie du canal, doit minimiser cette probabilité d'erreurs en recherchant :**
  - **le maximum de la loi de probabilité à postériori**
  - **ou le maximum de vraisemblance**
- **Pour une loi d'émission uniforme ces régles se confondent.**
- **Le calcul est en général très difficile. On se contente d'un majorant.**

- **Pour un canal B.S.C.**
  - avec des probabilités de transition **e** et 1-**e**
  - des mots de taille N

$$P_e \leq \left(2\sqrt{e(1-e)}\right)^N$$

en réalité

$$P_e \approx \sqrt{\frac{2}{pN}}\left(2\sqrt{e(1-e)}\right)^N$$

- Soit une source émettant des mots de taille N
  - Son taux est R = H / N bit par lettre
  - La probabilité d'erreur p_e ↘ si N ↗ mais R ↘ si N ↗
- En fait pour avoir une probabilité d'erreur $p_e = 0$ il faut N = ¥ soit R = 0 !
- On cherche à minimiser p_e
  - en fonction de R, N et de la capacité C du canal
  - Shannon a montré que si R £ C , il existe une suite de codes de longueur de mot N qui minimise la probabilité d'erreur.
  - La taille du code est *Partie entière* [exp (NR)]
  - Si R > C il n'existe aucune méthode de codage qui permette de transmettre de l'information avec un taux d'erreur négligeable.
  - Ce théorème fournit une condition limite qui permet d'orienter la recherche des codes les meilleurs ou de les comparer à cette valeur limite.

$$H(X) = -\int_{-\infty}^{+\infty} f_1(x) \log f_1(x)\, dx$$

$$H(Y) = -\int_{-\infty}^{+\infty} f_2(y) \log f_2(y)\, dy$$

$$H(X/Y) = -\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} f(x,y) \log \frac{f(x,y)}{f_2(y)} dxdy$$

$$I(X,Y) = -\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} f(x,y) \log \frac{f(x,y)}{f_{1(x).}\, f_2(y)} dxdy$$

Capacité du canal continu

$$C = \mathrm{Max}_{f_{1(x)}}\left[I(X,Y)\right]$$

- **Les définitions de l'information, de l'entropie, etc sont étendues à des sources ou des collecteurs ayant des fonctions densité de probabilité continues (et non plus discrètes) pour utiliser pleinement les canaux de communication analogiques**

- **Cette quantité d'information est liée à un symbole unique**

☞ Si T est l'intervalle de temps entre 2 symboles ("Moment")

➡ le DEBIT d'information vaut

$$C_T = C / T \quad \text{bit/s}$$

➡ Le taux de la source est $R_T = H(X) / T$

☞ D'après le second théorème de Shannon,

➡ si $R_T \leq C_T$ on peut avoir un taux d'erreurs aussi petit que l'on veut.

$$H(Z) = -\int_{-\infty}^{+\infty} q(z)\log q(z)dz$$

avec, si B est la puissance du bruit

$$q(z) = \frac{1}{\sqrt{2pB}}\exp\left(-\frac{z^2}{2B}\right)$$

et si S est la puissance du signal

$$f_1(x) = \frac{1}{\sqrt{2pS}}\exp\left(-\frac{x^2}{2S}\right)$$

- **On suppose que le canal est additif et qu'il est perturbé par un "bruit" gaussien.**
- **Soit H(Z) l'entropie de cette" source de bruit" de densité de probabilité q(z)**
- **On suppose aussi que le signal utile issu de la source possède une densité de probabilité gaussienne $f_1(x)$**

- H (X) = log (2peS)
- H (Z) = log (2peB)

➡ **Y = X + Z donc** $H(Y) = \log(2\pi e(S+B))$

➡ $I(X,Y) = H(Y) - H(Z)$

$= \log(2\pi e(S+B)) - \log(2\pi eB)$

$$I(X,Y) = \frac{1}{2}\log_2\left(1 + \frac{S}{B}\right)$$

➡ **Si B est minimal et S maximal** $C = \frac{1}{2}\log_2\left(1 + \frac{S}{B}\right)$

- Nota : Pour avoir le débit maximal théorique d'information nous devrons multiplier cette capacité par le nombre de symboles transmis par unité de temps.

➡ **Bruit blanc maximum sur canal téléphonique :**

48 db au dessous du signal

☞ **nota : décibel =** $10 \log_{10} \frac{P_2}{P_1}$

➡ **soit 10 $\log_{10}$ (S/B) = 48 ou S/B ≈ 63000**

➡ **C = $\log_2$ (1 + S/B) =** $\frac{1}{2} \frac{\log_{10}(1+\frac{S}{B})}{\log_{10} 2}$

➡ **C = 3,32 $\log_{10}$ (1 + S/B)**

☞ **= 3,32 * 4,8 ≈ 7,970 bits par symbole**

- **Signal émis 1mW**
  - nota : sur une ligne téléphonique
    - puissance crête 0 dbm = 1 mw
    - puissance moyenne -10 dbm = 0,1 mw
  - Atténuation 30 db soit un facteur 1000
- **Signal reçu 1W**
- **Bruit à la réception 10 nW**
  - S/B = 100 (20 db)
  - $\log_2 (1 + S/B) = \log_2 (101) = 6,658$
- **C = 3,329 bit par symbole**
  - En pratique on pourra espérer transmettre 3 bit par symbole

Codage

SOURCES

- **Indépendantes**
  - x_i sont des var. aléatoires indépendantes de même loi , Si M = $x_1x_2x_3$ $p(M) = p_1p_2p_3$
- **Stationnaires**
  - x_i sont liées mais les probabilités pour une suite $x_1x_{2...}x_i$ ne dépendent pas de l'instant d'émission
- **de Markov**
  - Source à mémoire finie
  - loi à priori $p(x_i)$ + suite de lois conditionnelles $p(x_{1/}x_2)$, ...,$p(x_{1/}x_n)$

    $p(x_{1/}x_2x_3)$

    $p(x_{1/}x_1x_2x_{3...}x_n)$

    On utilise un diagramme d'état pour indiquer les probabilités à prendre en compte à un instant donné
  - exemple : langue naturelle (...ait, ...ment, ..)
  - Codage prédictif: images Vidéo

LONGUEUR MOYENNE

- n_i nombre de caractères du mot m_i codant un message x_i
- Longueur moyenne des messages :

$$\bar{n} = E(n_i) = \sum_{i=1}^{N} n_i p(x_i)$$

Cette grandeur est aussi appelé le coût moyen par message

TYPES DE CODES

- A longueur de mot fixe
  - tous les mots sont de même longueur
  - exemple : AI5 (7 bit/lettre) , AI2 (5 bit/lette)
- A longueur de mots variable
  - exemple : code Morse, Huffman, ...

DEBITS, ETC.

- débit littéral : nombre de caractères par unité de temps

  D lettres/ s

- Taux d'émission : $\frac{H(X)}{\bar{n}}$ bit/lettre

- Débit d'information : $D\frac{H(X)}{\bar{n}}$ bit/s

- Efficacité : $h = \frac{\bar{n}_{min}}{\bar{n}}$ $\bar{n}_{min}$ longueur moyenne pour le code le plus court

- Redondance : $\rho = 1 - \eta$

- Source codée par un alphabet de taille K : $a_1, a_2 \ldots a_k$
  - exemple [0,1] K = 2
- Soit une séquence de longueur l de lettres de cet alphabet
  - $x_1, x_2, \ldots\ldots x_l$ $\vec{X}_l$
  - On peut construire K^l exemple 2^l
  - Ces K^l séquences constituent une nouvelle source appelée l ème extension de K
  - exemples : AI2 l = 5 2^5 = 32 "lettres" de 00000 à 11111
    AI5 l = 7 2^7 = 128 "lettres de 0000000 à 1111111
- Pour coder ces suites par des mots de taille N à partir d'un alphabet de taille D,

$$\frac{N}{l} \geq \frac{\log K}{\log D}$$

on doit avoir

- Exemple : Décimal codé binaire
  - D = 2 K = 10 $N \geq 1 \frac{\log 10}{\log 2} = 3{,}32$ chiffre de l = 1 lettre (0 à 9)
  - if soit N = 4

- **Code caractérisé par :**
  - longueur moyenne des mots
  - non ambiguïté de la lecture
- **Code : partie d'un ensemble de suites finies de caractères issus d'un alphabet**
  - Un tel code doit être REGULIER et DECHIFFRABLE
  - exemples :

	$P(x_i)$	code 1	code 2	code 3	code 4
x_1	0,5	1	0	1	1
x_2	0,25	1	1	10	01
x_3	0,15	0	11	100	001
x_4	0,1	00	01	1000	000
		non régulier	non déchiffrable	non ambigu	irréductible

- Un code est déchiffrable (conditions suffisantes) si
  - il a une longueur fixe
  - ou il est préfixé
- Code
  - singulier
  - régulier non déchiffrable
    déchiffrable
    - réductible
    - irréductible (ou instantané)
- un code irréductible est construit à l'aide d'un arbre

1
⇒ 01
0 001
00
000 etc.

- **Soit un alphabet de taille D (nombre de mots code)**
- **Code irréductible**

$$\bar{n} \leq \frac{H(X)}{\log D} + 1$$

- **Code déchiffrable**

$$\bar{n} \geq \frac{H(X)}{\log D}$$

Si égalité : code optimal

- **Pour toute source indépendante, il existe un code irréductible de longueur moyenne aussi proche que l'on veut de** $\frac{H(X)}{\log D}$
  - **1er Théorème de Shannon**
- **Pour un code binaire** $\bar{n}_{min} = H(X)$ **(log D =1)**

➡ **Soit k(N) le nombre de textes codés avec N lettres**

$$k(N) \leq D^N$$

$$\lim_{N\to\infty} \frac{\log k(N)}{N} = C \leq \log D$$

$$\text{avec } C \text{ solution de } \sum_{i=1}^{N} 2^{-Cn_i} = 1$$

➡ **C est la capacité de codage du code**

➡ **Exemple : code binaire [0,1] D = 2**

☞ **longueur optimale** $n_i = - \frac{\log p_i}{\log 2}$

☞ **Si on classe $p_1 \geq p_2 \geq \ldots\ldots \geq p_{i\ldots\ldots} \geq p_n$ alors $n_1 \leq n_2 \leq \ldots\ldots \leq n_{i\ldots} \leq n_n$**

➠ **On cherche les plus petits entiers tels que** $p_i \geq 2^{-n_i}$

➡ **A chaque étape on regroupe les messages en 2 sous-ensembles de probabilité la plus voisine possible. On continue jusqu'à obtenir 2 messages que l'on code.**

➡ **exemple 1**

mes. code	$p(x_i)$						
x_1	0,51	0,51	0				0
x_2 10	0,29			0,29	10		
x_3 110	0,08	0,49	1	0,20	11	0,08	110
x_4 111	0,12					0,12	111

☞ longueur moyenne : 0,51 + 0,29 * 2 + (0.08 +0,12) * 3 = 1,69

Exemple 2

mes. code	$p(x_i)$								
x_1 10	0,4			0,4	10				
x_2 00	0,3	0,5	0	0,3	00				
x_3 01	0,2			0,2	01				
x_4 110	0,05	0,5	1			0,05	110		
x_5 1110	0,03			0,1	11	0,05	111	0,03	1110
x_6 1111	0,02							0,02	1111

$\overline{n}_{min} =$

- longueur moyenne : (0,4 + 0,3 +0,2) * 2 + 0,05 * 3 + (0,03 +0,02) * 4 = 2,15
- H(X) = 1,994987
- log D = 1 1,995

- if

Propriétés

- si $p_i > p_j$ alors n_i n_j
- les 2 mots les moins probables ont même longueur
- parmi les mots de longueur maximale n_m il y en a au moins 2 ayant les mêmes n_{m-1} premières lettres
- On classe les messages et on les regroupe 2 par 2

exemple 1:

mes.	$p(x_i)$						code
x_1	0,51	0,51	0,51	0			0
x_2	0,29	0,29	0,49	1	10		10
x_3	0,12	0,20			11	110	110
x_4	0,08					111	111

- Pour cet exemple même longueur moyenne que Shannon-Fano

Exemple 2

mes. $p(x_i)$										code
x_1	0,4	0,4	0,4	0,4	0,4	1				1
x_2	0,3	0,3	0,3	0,3	0,6	0	00			00
x_3	0,2	0,2	0,2	0,3		01	010			010
x_4	0,05	0,05	0,1			011	0110			0110
x_5	0,03	0,05					0111	01110		01110
x_6	0,02		$\overline{n}_{min}$ =					01111		01111

- longueur moyenne : 0,4 + 0,3 * 2 + 0,2 * 3 + 0,05 *4 + (0,03+0,02) * 5 = 2,05
- H(X) = 1,994987
- if log D = 1 1,995

COMPRESSION DE DONNEES

Si des données sont codées avec des code de longueur fixe on peut les comprimer

- en enlevant les caractères inutiles
- en évitant les répétitions : Code RLE (Run Lenght Encoding)
  - xxxxxxxxxxx Rx11
- en utilisant un code de taille variable
  - Huffman statique ou adaptatif
  - Huffman -Shannon- Fano autosynchronisant ...
  - Ziv et Lempel (codage par dictionnaire)
    - certaines séquences de caractères sont considérés comme des messages valides : extension de l'alphabet
    - On applique un code d'Huffman sur cette extension

- On peut aussi utiliser
  - des codes Arithmétiques
  - des transformations mathématiques etc.
- nota : code JPEG (Joint Photographic Expert Group)
  - CCITT T.83 ou ISO 10917-1
- code MPEG (Motion Picture Expert Group)
  - CCITT ?? , ISO ??
    - MPEG2 Codage vidéo : prédictif + Code d'Huffman
- CCITT H.261

QUESTIONNAIRES DICHOTOMIQUES

- Questionnaire à réponse par oui ou non à nombre de questions minimal

MPEG : Moving Picture Expert Group

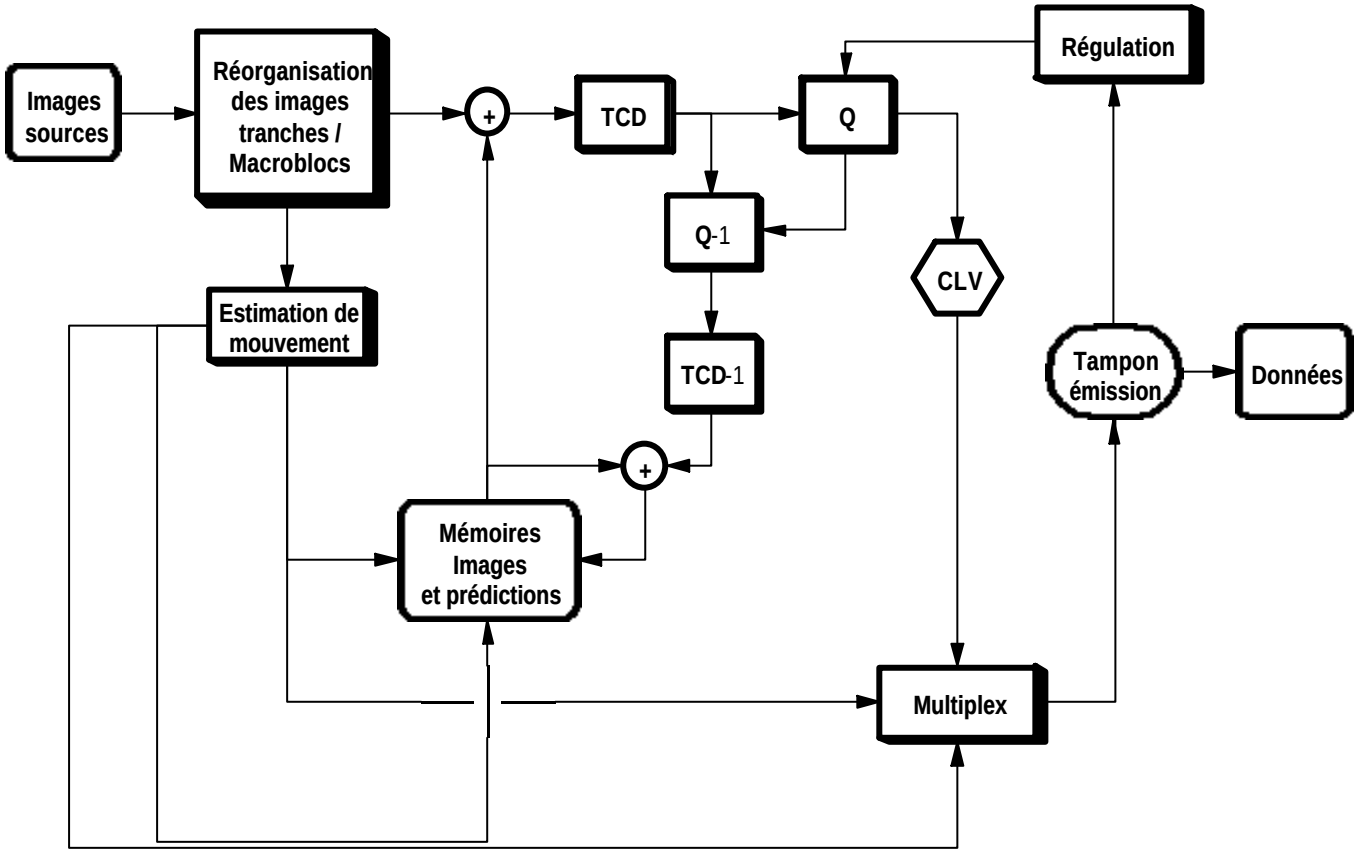
- MPEG-2 : ISO/IEC 13818
- IUT : H26x (H262) , H32x (visioconférence sur ATM)

Différents formats d'image

- Niveau (résolution) (pel = picture element)
  - Low (SIF) : 352*288 pels
  - Main (625l) : 720*576 pels (aspect 4/3)
  - High1440 : 1440*1152 pels (aspect 4/3)
  - High (TVHD): 1920*1152 pels (aspect 16/9)
- Profil (format) : décomposition des macroblocs
  - Simple : 4:2:0 (Main)
  - Main : 4:2:0 (tous niveaux)
  - SNR : 4:2:0 (Low, Main)
  - Spatial : 4:2:0 (High-1440)
  - High : 4:2:0, 4:2:2 (Main, High1440, TVHD)

- Séquence vidéo
- Images décomposées en trames (entrelacées ou non)
  - I : Intra Indépendante de toute autre (Non Prédite)
  - P : Prédites à partir des images I ou P Précédentes)
  - B : Bidirectionnelle (prédites par rapport à I ou P voisine)
- trame : ensembles de tranches (slice)
- tranche : un ou plusieurs macroblocs
- macrobloc :
  - 16*16pels (picture element)
  - luminance (jaune)+ Chrominance (2 couleurs : rouge +bleu)
  - 6, 8 ou 12 blocs ➔ 4:2:0 , 4:2:2 , 4:4:4
- bloc :
  - 8*8 pels luminance et 0 ou 8*8 chrominance

- **Seuls les blocs différents d'une trame à la suivante sont transmis**
  - L'adresse des blocs est codée par un code à longueur variable
- **Seules les différences entre les blocs sont transmises**
  - La repésentation des blocs est transformée par DCT : Transformation en Cosinus Discrète
  - Les coéfficients de la transformée sont codés sur 12 bits (Signe + 11 bits)
  - La différence entre les coéfficients, pour deux trames successives, est codée par un code à longueur variable (3 à 17 bits)
- **Les types de macroblocs et d'image sont codés par des codes à longueur variable**



TCD:

Transformée en Cosinus Discrète

Q: Quantification

CLV:

Codage à longueur variable (Huffmann)

Codes détecteurs et correcteurs d'erreurs

- **Pour éviter les erreurs de décodage (dues à des perturbations dans le canal) les messages doivent le plus "différent" possible les uns des autres.**
- **On appelle DISTANCE DE HAMMING de deux mots codes $\vec{x}_1$ et $\vec{x}_2$ le nombre de chiffres par lequel ils diffèrent.**
  - ☞ $\vec{x}_1$ = 00111010 $\vec{x}_2$ = 10110010 d($\vec{x}_1$, $\vec{x}_2$) = 2

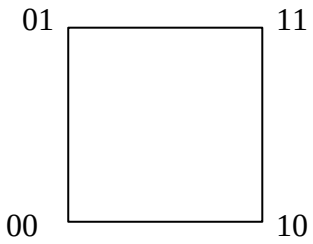
- **Pour des mots de longueur N**

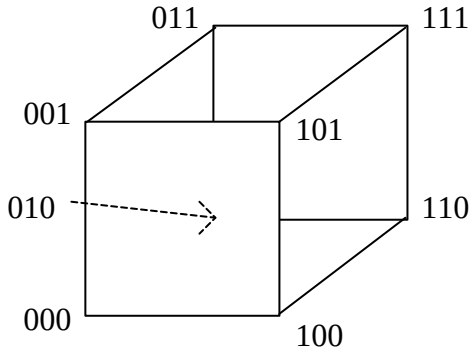
$$d(\vec{x}_1, \vec{x}_2) = \sum_{i=1}^{N} x_{1_i} \oplus x_{2_i}$$

- **La DISTANCE DE HAMMING d'un code est la plus petite distance observée entre les mots du code (pris 2 à 2).**

- $d(\vec{x}_1, \vec{x}_2) = 0$ si et seulement si $\vec{x}_1 = \vec{x}_2$
- $d(\vec{x}_1, \vec{x}_2) = d(\vec{x}_2, \vec{x}_1) > 0$ si $\vec{x}_2 \neq \vec{x}_1$
- $d(\vec{x}_1, \vec{x}_3) + d(\vec{x}_2, \vec{x}_3) \geq d(\vec{x}_1, \vec{x}_2)$

☞ Exemples "géométriques"





☞ Si pour l'espace à 3 dimensions (N=3) on ne garde que des mots codes sur des sommets opposés (000 et 111 ou 011 et 100 par exemple) leur distance est d = 3

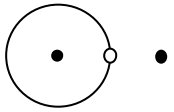
- Pour un canal B.S.C. décoder selon la règle du maximum de vraisemblance revient à prendre le mot le plus proche, au sens de la distance de Hamming, d'un mot code.
  - Par exemple si le code choisi est [001,110]
    et si on observe 000 à la sortie du canal,
    on décide que 001 a été émis à la source

- Ceci nous conduit à la notion de boule de décodage dans l'espace à N dimensions
  - Dans une boule de rayon r il y a $\sum_{i=0}^{N-1} C_N^i$ mots code
  - On pourra corriger une erreur simple ou multiple si le mot code érroné observé reste dans la boule de rayon r.
  - On pourra détecter une erreur simple ou multiple si le mot code observé n'est pas un mot du code.

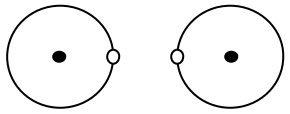
Avec un code de distance de Hamming d on peut

- detecter p erreurs si $d \geq p+1$
- corriger q erreurs si $d \geq 2q+1$
- corriger q et détecter p erreurs si $d \geq q+p+1$

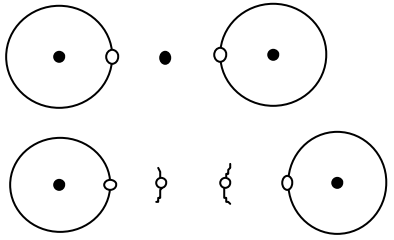
- d — Puissance
- 1 — erreur non détectable
- 2 — détection d'une erreur simple



- 3 — correction d'une erreur simple



- 4 — correction d'une erreur simple et detection d'une erreur double



- 5 — correction d'une erreur double

Codes à contrôle de parité

- Application d'un espace à 2^k élements dans un espace à 2^n éléments
- Dans ces codes un ou plusieurs bits de redondance sont ajoutés à une ensemble de k bits d'information
- notation : code (n,k)

Codes de blocs

- longueur de bloc (ou de code) n dont k bits d'information
- codes linéaires
  - Les bits de redondance sont placés sur des sites particuliers du bloc de taille n
  - ex : Code de Hamming
    - correction d'erreur simple
    - n-k bits de "parité"
    - $n \leq 2^{n-k} - 1$
    - code optimal si $n = 2^{n-k} - 1$

- codes cycliques
  - En général ("codage par division") les bits de redondance sont placés après les k bits d'information
  - ex : Code de Hamming

    Code B.C.H. (Bose-Chaudhuri-Hocquenghem)

Codes d'arbre ou codes continus

- codes convolutionnels
  - des bits de redondance sont placés au fil de l'eau, réguliérement, dans la séquence de bits d'information et portent sur les m bits précédents (qui peuvent être des bits de redondance ...)
  - Utilisés à l 'origine pour les bandes magnétiques, ils sonr=t redevenus très importants pour la téléphonie mobile (GSM)

Codes systématiques

$$G = \begin{bmatrix} 1 & 0 & . & .0 & a_{1,k+1} & . & a_{1,n} \\ 0 & 1 & 0 & .0 & a_{2,k+1} & . & a_{2,n} \\ 0 & . & . & .0 & a_{.,k+1} & . & a_{.,n} \\ 0 & 0 & 0 & .1 & a_{k,k+1} & . & a_{k,n} \end{bmatrix}$$

$$H = \begin{bmatrix} a_{1,k+1} & a_{1,k+2} & . & a_{1,n} \\ a_{2,k+1} & . & . & a_{2,n} \\ a_{.,k+1} & . & . & a_{.,n} \\ a_{k,k+1} & a_{k,k+2} & . & a_{k,n} \end{bmatrix}$$

- Matrice génératrice G k lignes , n colonnes
- Un mot code $\vec{x}$ est obtenu à partir d'une séquence d'information $\vec{u}$ par $\vec{x} = \vec{u}$ G
- si $\vec{u}$ = 00.. 010..0 (1 en ième position) $\vec{x}$ est la ième ligne de G
- Toute combinaison linéaire de lignes de cette matrice est un mot du code
- matrice de parité: k lignes n-k colonnes
- Permet de contrôler la réception par $\vec{x}$ H = 0 si aucune erreur

- Ensemble [0,1] muni
  - de l'addition (+ est le "ou exclusif")
  - de la multiplication

  forme le corps de Galois GF(2)

- si $a_i \in$ GF(2) (a_i = 0 ou1)

$$a_n x^n + a_{n-1} x^{n-1} + \quad +a_0 = P(x)$$

  - P(x) est un polynome de degré n sur GF(2)
  - L'ensemble de ces polynomes à une structure d'anneau
    - pas un groupe pour la multiplication

- Soit 2 polynomes $P_1(x)$ et $P_2(x)$ sur GF(2)

  $ deux polynomes $Q(x)$ et $R(x)$ tels que $P_1(x) = Q(x)\,P_2(x) + R(x)$

- 2 polynomes son équivalents modulo Q(x) si les restes de leur division par Q(x) sont identiques.

- Ceci permet de définir une classe d'équivalence. Ces polynômes peuvent être représenté par leur classe résiduelle de plus bas degré, soit ce reste R(x)

- exemple : $Q(x) = x^2 \oplus 1$

  classes résiduelles :

  0,1,x,x+1

- Pour un polynome de degré N, il y a 2^N classes résiduelles

- Un polynôme P(x) est réductible sur un corps s'il existe deux polynômes G(x) et H(x) tels que

$$P(x) = G(x)\,H(x)$$

- Si ce n'est pas le cas P(x) est irréductible

- Si P(x) est irréductible les racines de P(x) = 0 sont des éléments d'une extension de GF(2)

- soit $P(x) = \prod_i (x + a_i)$

- **Classe particulière des codes de groupe obtenu par permutation circulaire des chiffres**
  - si $a_1a_2 \ldots a_n$ C alors $a_2a_3 \ldots a_na_1$ C, $a_3a_4 \ldots a_2 \in$ C, etc
  - On utilise en général la notation polynomiale définie ci-dessus
  - si U(x) représente un mot du code, $x^iU(x)$ modulo $(x^n + 1)$ est aussi un mot du code
    - Ceci correspond à la ième permutation de U(x)
  - Toute combinaison linéaire de mots du code est encore un mot du code
  - Il existe un polynôme de plus bas degré qui divise $x^n + 1$
  - Ce polynôme $U_0(x)$, de degré m, est appelé POLYNOME GENERATEUR du code C
  - Il existe 2^{n-m} polynômes quotients de $x^n + 1$ par $U_0(x)$
    - Ce code contient donc k = n-m mots
    - code (n, n-m)

➡ MATRICE GENERATRICE

$$G = \begin{bmatrix} a_m & a_{m-1} & . & . & a_0 & 0 & . & 0 \\ 0 & a_m & a_{m-1} & . & . & a_0 & . & 0 \\ 0 & . & . & . & 0 & . & . & 0 \\ 0 & 0 & 0 & a_m & . & . & . & a_0 \end{bmatrix}$$

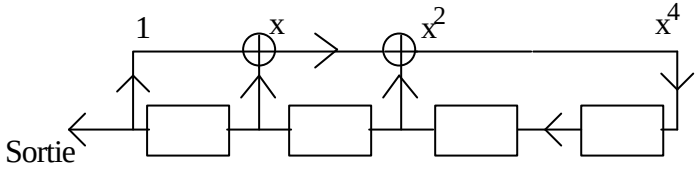
- ☞ U_0 et ses n-m-1 permutés forment une base du code, notée par la matrice G
- ☞ Exemple : Code (7,4) engendré par $X^3 + x + 1$

$$\begin{bmatrix} 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

➡ MATRICE de PARITE

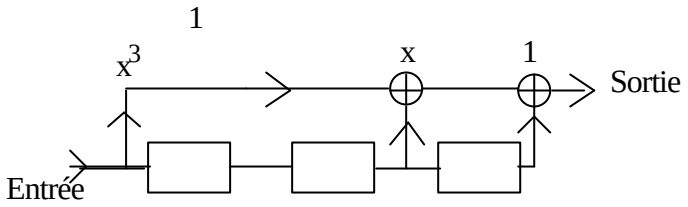
$$H = \begin{bmatrix} \bar{a}_0 & 0 & . & 0 \\ \bar{a}_1 & \bar{a}_0 & . & 0 \\ . & . & . & . \\ 0 & . & . & \bar{a}_{n-m} \end{bmatrix}$$

$$\begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

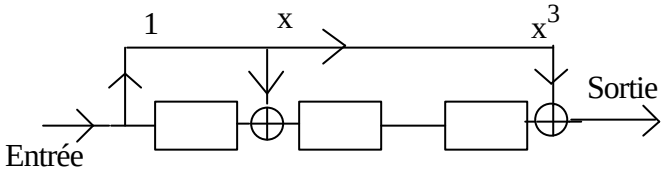


- On utilise des registres à décalage et des "ou exclusif" (addition)
- La multiplication
  - par 1 est notée par une connexion
  - par 0 par l'absence de connexion

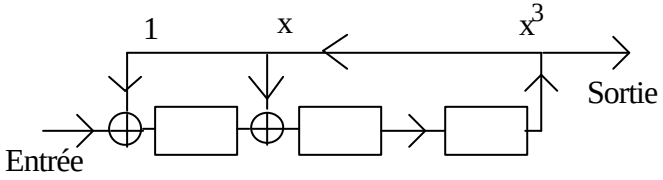
- CODEUR A k = n-m ETAGES (ENTREE PARALLELE)
  - Exemple : codeur (7,4)
  - Polynôme orthogonal à $x^3 + x + 1$ soit $x^4 + x^2 + x + 1$ (voir matrice de parité)
- CODEUR A m ETAGES (ENTREE SERIE)
  - Le registre est initialisé à 0. L'information est introduite poids forts en tête



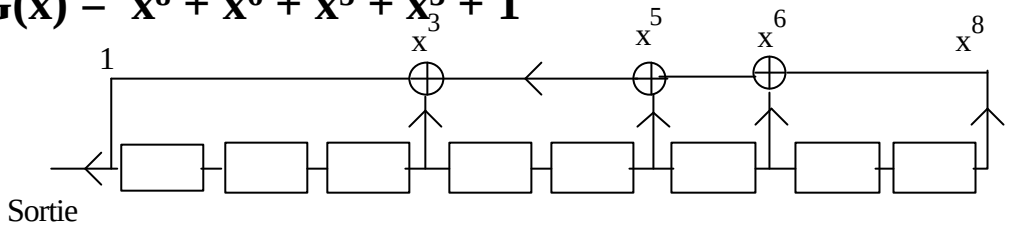
- **Codeurs par multiplication.**
  - **Exemples: Codeurs (7,4)**



- **Codeur par division**
  - **Les m premiers décalages donnent 0 en sortie et ne sont pas utilisés.**
  - **Exemple: Codeur (7,4)**



- Une séquence pseudo-aléatoire de taille $2^{m-1} - 1$ est générée par un polynôme de degré m : G(x) diviseur de $X^{n}+1$
  - Exemple : m = 8, n = 255
  - $G(x) = x^8 + x^6 + x^5 + x^3 + 1$



- La séquence de 255 bits comporte 128 bits à 1 et 127 bit à 0 répartis aléatoirement
- Si le circuit génère des bits en continu, la séquence est répétée périodiquement
- Technique utilisée pour
  - les générateurs de bruit
  - les séquences de test

- Soit un polynôme irréductible G(x) de degré m, diviseur ou produit de diviseurs de $X^{2^m - 1} + 1$
- et une séquence à protéger M(x) et E(x) un syndrome d'erreurs
  - sans erreur E(x) est une séquence nulle 000000....

CODAGE PAR MULTIPLICATION

- C(x) = M(x) • G(x) est le mot du code transmis
- On reçoit $C^*(x) = C(x) + E(x)$
- On décode $Q(x) = C^*(x) / G(x) + R(x)$
  - si R(x) = 0 M(x) = Q(x)
  - sinon erreur détectée
- Correction
  - On peut initialiser une table donnant E(x) en fonction de R(x)
  - On peut alors corriger les erreurs par $C(x) = C^*(x) + E(x)$

CODAGE PAR DIVISION

- On calcule
  - $I(x) = x^m \cdot M(x)$
    - ajout en fin de séquence de m bits à 0
  - $r(x) = I(x) - G(x)\ Q(x)$
    - reste de la division de I(x) par G(x)
  - $C(x) = I(x) + r(x)$
    - remplacer les m bits à 0 de fin par r(x)
- On décode
  - $Q(x) = C^*(x) / G(x) + R(x)$
  - Si $R(x) = 0$ $C(x) = C^*(x)$
    - $M(x) = C^*(x) / x^m$ troncature de C(x)

    Sinon erreur détectée
- ou correction par $C(x) = C^*(x) + E(x)$

  et utilisation d'une table de correction E (R(x))

- Considérons le polynôme $x^{n-1} + 1$
  - Ses racines $_i$ sont solutions de $x^{n-1} + 1 = 0$
  - Alors $x^{n-1} + 1 = \prod_{i=1}^{n-1} (x - a_i)$

- On peut montrer que $x^{n-1} + 1$ peut être décomposé en un produit de polynômes irréductibles $f_j(x)$
- $$x^{n-1} + 1 = \prod_{j=1}^{L} f_j(x)$$

  - $f_j(x)$ a pour racines des racines $_i$ de $x^{n-1} + 1$ soit $f_j(x) = \prod_i (x - a_i)$

- Soit P(x) un polynôme de degré n -1
- Si α est racine de p(x) alors $\alpha^2, \alpha^4, \alpha^8, \ldots,$ (2^k modulo n-1) sont racines de p(x)
- Les polynômes $f_j(x)$ sont difficiles à calculer . Plutôt que de donner des règles, fonctions du degré du polynôme permettant de les déterminer, Peterson en 1961 a fourni des tables qui les donnent tous jusqu'au degré 16 en partiellement jusqu'au degré 34.

➡ On lit dans la table

☞ degré 5 1 45E 3 75G 5 67H

☞ Ceci indique 3 polynômes $f_j(x)$ correspondant aux racines de base α^1, α^3 et α^5

☞ La lettre donne les propriétés du polynôme $f_j(x)$

❶ A,B,C,D non primitif

❶ E,F,G,H primitif

❷ A,B,E,F racines dépendantes

❷ C,D,G,H racines indépendantes

❸ A,C,E,G racines du réciproque dépendantes

❸ B,D,F,H racines du réciproque indépendantes

☞ La valeur donne le polynôme codé en octal (base 8)

- On décompose $x^{31} + 1$
  - 31 est de la forme $2^5 - 1$
  - Il sera décomposé en 6 polynômes de degré 5 et le polynôme (x+1)
- On recherche ces polynômes dans la table de Peterson
  - Les degrés des racines sont modulo 31
    - par exemple $\alpha^{48} = \alpha^{17}$
  - 45 100101 X^5+x^2+1 racines $\mathbf{a}$, $\mathbf{a}^2$, $\mathbf{a}^4$, $\mathbf{a}^8$, $\mathbf{a}^{16}$
  - 75 111101 $X^5+x^4+x^3+x^2+1$ racines $\mathbf{a}^3$, $\mathbf{a}^6$, $\mathbf{a}^{12}$, $\mathbf{a}^{24}$, $\mathbf{a}^{17}$
  - 67 110111 $X^5+x^4+x^2+x+1$ racines $\mathbf{a}^5$, $\mathbf{a}^{10}$, $\mathbf{a}^{20}$, $\mathbf{a}^9$, $\mathbf{a}^{18}$
- Les polynômes réciproques sont obtenus en prenant les bits dans l'ordre inverse
  - soit 100101 101001 = 51 racines α^{30}, α^{29}, α^{27}, α^{23}, α^{15}
  - 111101 101111 = 57 racines α^{28}, α^{25}, α^{19}, α^{7}, α^{14}
  - 110111 111011 = 73 racines α^{26}, α^{21}, α^{11}, α^{22}, α^{13}
- Ils correspondent à la racine α^{30}, α^{28} et α^{26}. Les puissances sont complémentaires de 31
- On peut vérifier que $x^{31} + 1 = (x+1)(X^5 + x^2 + 1)(X^5 + x^4 + x^3 + x^2 + 1)$
  $(X^5 + x^4 + x^2 + x + 1)(X^5 + x^3 + 1)(X^5 + x^3 + x^2 + x + 1)(X^5 + x^4 + x^3 + x + 1)$

➡ CODES DE HAMMING

- ☞ Codes de distance 3, correcteurs d'une erreur
- ☞ Ces codes sont optimaux
- ☞ Construits à partir d'un seul polynôme irréductible G(x) de degré m
  - ➠ **génèrent des mots de longueur $n = 2^m - 1$**
  - ➠ **exemple pour m = 5 , n = 31 code (31,26)**
- ☞ G(x) est un polynôme irréductible (normal ou réciproque) quelconque pris dans la table correspondante de Peterson
  - ➠ **par exemple $X^5 + x^2 + 1$ ou $X^5 + x^3 + x^2 + x + 1$**

➡ CODES B.C.H.

- ☞ Hocquenghem et Bose et Chaudhuri ont donné une condition suffisante pour qu'un code ai une distance de Hamming donnée :
- ☞ Leur polynôme générateur doit être un produit de polynômes irréductibles ayant au moins s = d - 1 racines α^i consécutives

- Code BCH: Exemple
  - $(X^5 + x^2 + 1)(x^5 + x^4 + x^3 + x^2 + 1)$
  - a 4 racines consécutives α , α^2, α^3, α^4
- s = 4 , d = 5 . Il est correcteur de 2 erreurs (d = 2q +1)
- taille n = 31, m =10 code (31,21)

CODES TRONQUES

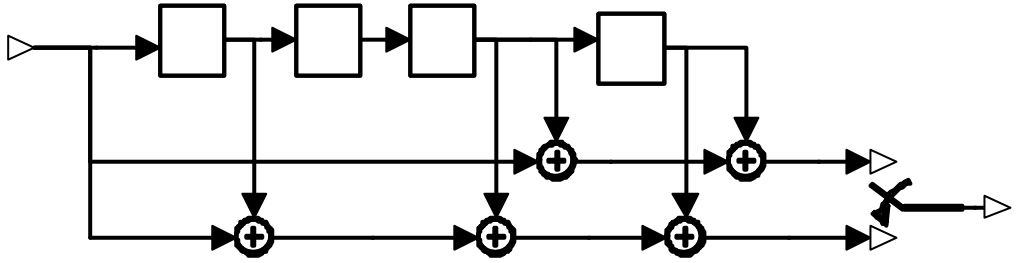
- On peut n'avoir que l < n bits à protéger (par exemple l = 16)
- On allonge fictivement le mot par n - l zéros et on utilise de code de Hamming ou B.C.H. correspondant
  - (31,26) $\Rightarrow$ (21,16) Hamming
- Mémoires protégées (22,16) avec $(x+1)(X^5 + x^2 + 1)$
  - (31,21) $\Rightarrow$ (26,16) B.C.H.
    $\Rightarrow$ (27,16) en ajoutant (x+1) à G(x)

- **Codes continus, au fil de l'eau…**
  - Ajout de bits de parité en fonctions des X bits précédents
  - taux de code: k/n
    - k bits d 'information
    - n bits au total
    - en pratique k/n = 1/2 ou 4/5 (Hagelbarger)
- **Codes correcteurs d'erreurs**
  - Peuvent être très performants en théorie mais pas de codes connus avec un très bon rendement
  - nécessité d'une séquence assez longue au décodage pour retrouver l'information
    - si blocs, trainée nécessaire pour que tous les bits utiles soient protégés
  - Distance libre= distance de Hamming minimale
    - d_{free}= capacité de correction
    - si taux = 1/2 , d_{free} = 7 correction de 3 erreurs

- **Pour un code 1/2:**
  - **Entrée U_i, sortie** $C_i = (C_i^{'}, C_i^{''})$
  - **Polynômes** $G'(X)$ et $G''(X)$

$$C'(X) = U(X)G'(X)$$
$$C''(X) = U(X)G''(X)$$

  - **Décodage** $S_i = C_i^{'} + C_i^{''}$



Codes cycliques

nature	Canal logique	Taille k	Taille CRC reXonXance	polynômes
Parole Classe I.a	TCH/FS	50	3	X^3+X+1
Signalisation Contrôle	SACCH, FACCH, BCCH, PCH, ...	184	40	$(X^{23}+1)(X^{17}+X^3+1)$
Accès	RACH	8	6	$X^6+ X^5+ X^3+ X^2 +X+1$
Synchronis.	SCH	25	10	$(X^5+1)(X^7+1)/(X+1)^2$

Codes convolutionnels

- code principal utilisé pour les données: taux 1/2
- polynômes : $X^4+ X^3+ 1$ et $X^4+ X^3+X+1$