Cours de

Transmission de l 'information

Jean-Yves Ramel

Département Informatique de l 'EPU de TOURS
Laboratoire d 'Informatique – RFAI

2007

Organisation du Cours

- 12 séances de 2 heures
- Pas d 'horaires de TD ou TP prévus !!!
- 1 contrôle final
- Plan :
  - Introduction
  - Théorie de l 'information
  - Codage de sources / compression
  - Codage canal / gestion des erreurs
  - Techniques de chiffrement, sécurisation
  - Supports et techniques de transmission

Introduction

Que va t 'on apprendre ? Pourquoi ?

- Comment définir l 'information ?
- Comment construire une information ?
  - Analogique / Numérique
  - Codage, détection des erreurs, ...
- Comment transmettre une information ?
  - Contraintes temporelles : débits, délais, synchro, ...
- Comment stocker / traiter une information ?
  - compression
  - chiffrement

Que va t 'on apprendre ? Pourquoi ?

- Aujourd'hui, les applications :
  - GSM, UMTS, GPRS -> codage de source & canal
  - TV numérique -> codage de source & canal
  - Réseaux -> codage de canal, détection des erreurs
  - E-commerce -> chiffrement
  - Multimédia, DVD -> Compression, transmission

L'information (une définition)

Définitions du (dictionnaire) Petit Robert :

- Élément ou système pouvant être transmis par un signal ou une combinaison de signaux (voir «message»)
- Ce qui est transmis: objet de connaissance, de mémoire
- Mesure de la densité de renseignements contenus dans un message (Petit Robert)
- **Contenu informatif d'un message**

D'autres définitions

- Signal : phénomène physique porteur d 'une information et pouvant représenter des données
- Alphabet : ensemble fini de signes appelés lettres
- Message : lot d 'informations formant un tout intelligible et exploitable transmis en 1 seule fois. Sequence finie de signaux (lettres) qu 'une source transmet à un collecteur
- Quantité d 'information : mesure quantitative de l 'incertitude d 'un message en fonction du degré de probabilité de chaque signal composant le message

Notion de communication

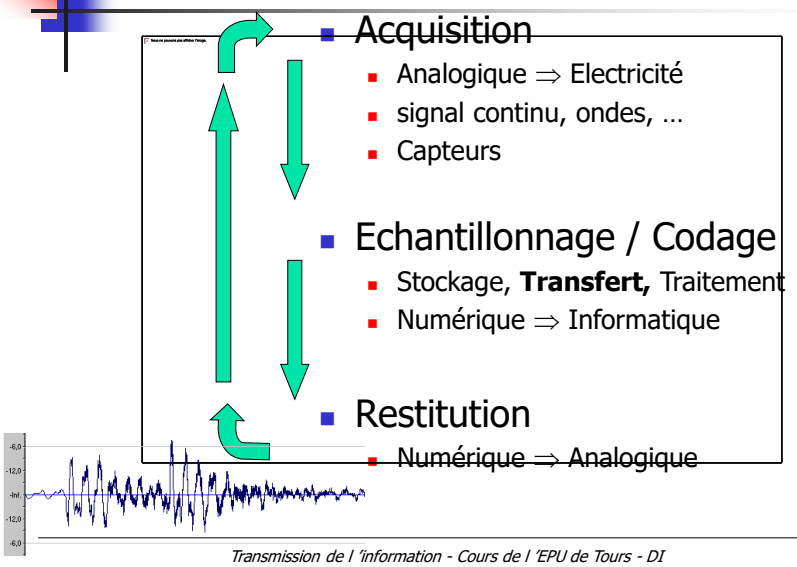
- La communication définit l'action d'échanger des informations. Cela induit un mécanisme de transmission (aspect physique) et la capacité du récepteur à recevoir et à réagir (aspect logique)

  → Il faut un langage commun

- **Quelles informations ?**

  - De plus en plus, le numérique

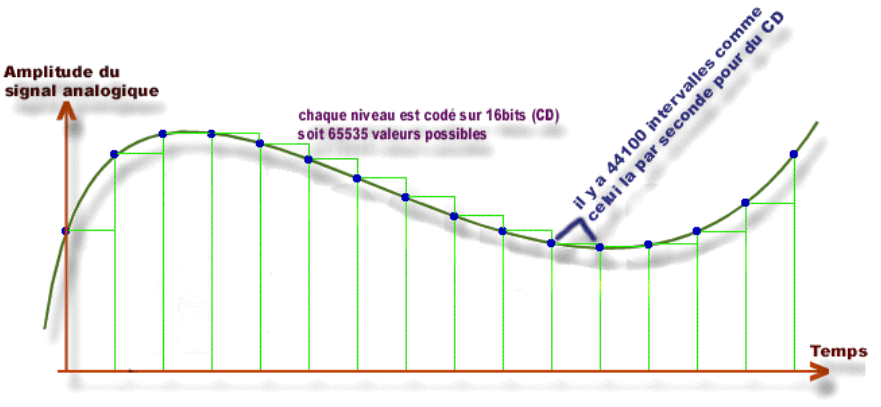
De l'analogique au Numérique



- **Acquisition**
  - Analogique ⇒ Electricité
  - signal continu, ondes, …
  - Capteurs
- **Echantillonnage / Codage**
  - Stockage, **Transfert,** Traitement
  - Numérique ⇒ Informatique
- **Restitution**
  - Numérique ⇒ Analogique

Conversion Analogique/Numérique

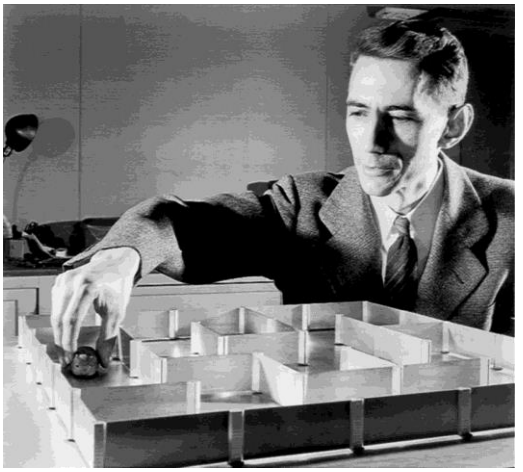
- Quantification / Résolution :



Chapitre 1 :

Théorie de l 'information

Claude Elwood Shannon (1916-2001)



Vue d 'ensemble de la théorie de l 'information

A côté de son usage devenu banal, le mot "information" a un contenu scientifique précis mais restrictif. La théorie de l'information élaborée et énoncée par l'ingénieur américain Claude Elwood Shannon en 1948, se présente comme un chapitre plutôt austère de la théorie des probabilités. Elle résume, en une magistrale synthèse, l'expérience théorique acquise avant et surtout pendant la Seconde Guerre mondiale sur les moyens de communication, en même temps qu'elle suggère des possibilités entièrement nouvelles. Elle affirme la possibilité paradoxale d'une communication sans erreur malgré des bruits perturbateurs affectant la transmission. pourvu qu'un codage approprié soit employé.

Vue d 'ensemble de la théorie de l 'information

Utile, indispensable même aux ingénieurs en tant que cadre conceptuel, elle n'a eu initialement qu'une faible influence directe sur les moyens de communication. Elle a pris de plus en plus d'importance à mesure qu'il devenait possible de réaliser des dispositifs complexes. Par une coïncidence qui fait rêver, 1948 est aussi l'année de l'invention du transistor. Le prodigieux développement de la technologie des semi-conducteurs a peu à peu fait entrer la théorie de l'information dans la pratique, et c'est peu dire qu'elle a fait désormais la preuve expérimentale de son utilité. La radiotéléphonie numérique et les CD seraient inconcevables sans les très efficaces procédés de codage qu'a directement suscités la théorie de l'information. Une immense expérience technique s'ajoute donc maintenant à la théorie proprement dite. Sa validité en est confirmée avec éclat et sa compréhension enrichie.

Vue d 'ensemble de la théorie de l 'information

Cette théorie est mal connue du public. L'une de ses caractéristiques fondamentale paraît si étrangère à la perception commune de l'information qu'elle étonne ou rebute et, en tout cas. fait obstacle à son assimilation : l'exclusion de la sémantique. La théorie de l'information est, en effet, indifférente à la signification des message. Au premier abord, la signification paraît l'essence même de l'information, au point que le refus de la sémantique semble la vider de tout contenu. Mais le point de vue de la théorie de l'information est modeste: celui d'un messager dont la fonction se limite au transfert d'un objet -une lettre par exemple- dont il n'a pas à connaître autre chose que le poids et les dimensions extérieures. L'information que peut porter cet objet n'a pas d'incidence sur les moyens de la transporter. Tel est aussi le point de vue de l'ingénieur en communications, seulement concerné par la quantité d'information qu'il doit transmettre, mesurable selon la théorie de Shannon.

G. Battail. Science et Avenir, hors série décembre 1999 janvier 2000, pp. 28-29

Quelle mesure quantitative pour l'information ?

Un constat :

la transmission d'un message certain est inutile

⌘ source d'information : siège d'événements *aléatoires* qui constituent le message

⌘ quantité d'information d'un *message* :

mesure de son *imprévisibilité*

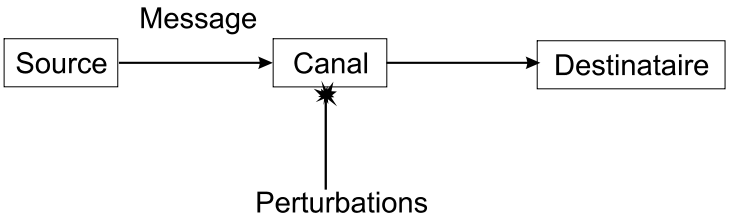
Voici la blague d'aujourd'hui

- **Claude Shannon, avait l'habitude de faire jouer à ce petit jeu de société quand il était invité quelque part. Il prenait un livre au hasard, l'ouvrait au hasard, commençait à lire un paragraphe et s'arrêtait. Il demandait ensuite à l'assistance de deviner une à une les lettres suivantes. L'assistance se débrouillait bien et trouvait la lettre dans environ 75 % des cas. Shannon en déduisait que la langue anglaise possède un taux de redondance de 75 %.**
- **Quand nous manipulons du texte, les caractères que nous utilisons n'ont pas la même probabilité d'apparition. De plus il a une structure interne forte (la grammaire). Quand le mot arbre est au pluriel on peut aisément prédire la lettre qui suit le « e » final.**
- **Quand nous travaillons avec de la musique, la distribution des probabilités d'apparition des sons n'est pas uniforme non plus.**
- **Quand nous manipulons des images, elles possèdent également des régularités, elles ne sont pas « aléatoires ».**
- **C'est cette caractéristiques qui incite à compresser les données et c'est elle qui permet, souvent, de réussir.**

Théorie de l'information (Shannon 1948)

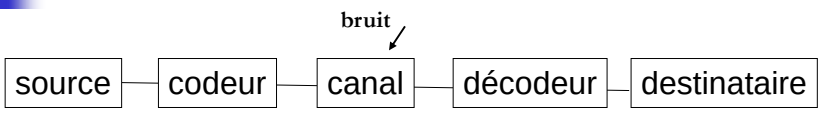
- Représentation efficace de l'information: le codage de source sans pertes (compaction de l'information)
- Théorie de la distorsion et codage de source avec pertes (compression de l'information)
- Capacité d'un canal de télécommunications et méthodes de codage de canal (transmission fiable de l'information à l'aide de codes correcteurs d'erreurs)
- Chiffrement et stratégies de cryptanalyse (confidentialité de l'information, authentification des utilisateurs, décryptement)

Modèle d 'un système de communication



Source = je parle
Canal = l'air ambiant
Perturbations = bruit sonore
Destinataire = tu écoutes

Modèle d 'un système de communication



- Th. Signaux → décrit messages et perturbations
- Modulation → modifie les signaux pour les propager
- Electronique → réalise les fonctions
- Th. Information → propose une mesure quantitative de l'**information** et étudie sa représentation, sa transmission, sa dégradation

Modèle d 'un système de communication

- **Source** : siège d'évènements aléatoires qui constituent le message émis → **Entropie**
- **Canal** : transmet et dégrade le message → **Capacité**

Des messages différents portent la même information, le **codage** cherche le message avec les meilleurs propriétés.

- Codage de source → supprime la redondance, reduit le coût
- Codage de canal → protège contre les perturbations
- Cryptage → protège contre les curieux

Information, grandeur mesurable ?

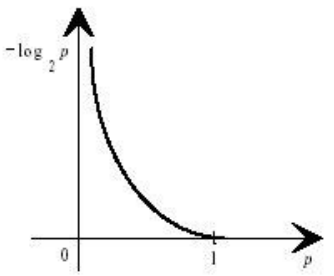
- **Aspects qualitatifs et quantitatifs de l 'information**
  -> apporter des valeurs chiffrées, mesurer
- Information liée à la nature aléatoire d 'un message
- Information + grandeur mesurable = probabilités
- communication = expérience aléatoire
- message = résultat de l'expérience qui apporte l 'information
- Exemple : montant sur bulletin de paye

Information, grandeur mesurable ?

- Soit I(x), la quantité d'information apportée par le message x
- I(x) est une fonction f de $\frac{1}{p_x}$ Avec f croissante & f(1)=0
- I(x) doit être positive
- I(x) doit être additive : I(x+y) = I(x) + I(y)

Information, grandeur mesurable ?

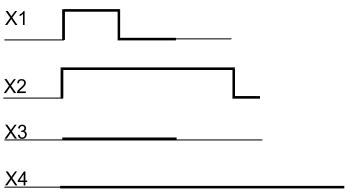
- Selon Shannon, $I(x) = \log(1/p(x)) = -\log(p(x))$
- Si log base 2, alors I(x) s 'exprime en bit
- $I(x_k)$ est aussi appelé Self-information de la source



Sources discrètes

- **Source discrète d'information** : suite de variables aléatoires discrètes X_1, X_2, … X_n
- **Symbole** ou **lettre** : élément fondamental irréductible contenant une information, cad réalisation particulière de la source d'information.
- **Mot** : succession finie de symboles
- **Alphabet** : totalité des D lettres
  [X] = [x_1,x_2, …., x_n]

Ex : Code morse, 4 symboles



Sources discrètes

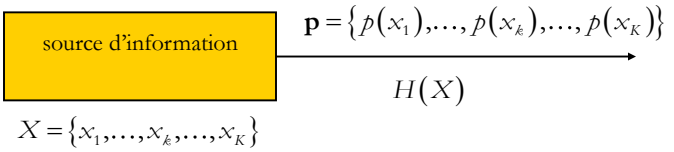
- **Source discrète sans mémoire** : source pour laquelle la probabilité d'apparition d'un symbole ne dépend pas des symboles précédents $p(x_{i_n} / x_{i_{n-1}}, x_{i_{n-2}}, ...) = p(x_{i_n})$
- **Source discrète à mémoire** : source pour laquelle la probabilité d'apparition d'un symbole dépend du ou des symboles précédents
- **Source stationnaire** : source pour laquelle les probabilités d'apparition des différents symboles ne dépendent pas de l'origine des temps $p(x_{i_n}) = p(x_{i_{n+k}}) \quad \forall k$
- **Source à débit contrôlable** : source pouvant générer des messages comme suite à une commande externe (Télégraphe, .)

Sources discrètes

- **Source à débit non contrôlable** : source générant des messages avec un débit fixé, propriété de la source (CD audio)
- **Source discrète à contraintes fixes** : source pour laquelle certains symboles ne peuvent être utilisés qu'en des conditions déterminées (Morse, …)
- **Source discrète à contraintes probabilistes** : source à mémoire. Dans un état, la source peut générer n'importe lequel des symboles avec une probabilité qui dépend des symboles précédents (texte …)
- **Source de Markov** : source pour laquelle la probabilité de générer un symbole ne dépend que du symbole à l'instant n-1

$$p(x_{i_n} / x_{i_{n-1}}, x_{i_{n-2}}, ...) = p(x_{i_n} / x_{i_{n-1}})$$

Entropie d'une source d'information



Hyp : source discrète finie stationnaire sans mémoire

Emission = variable aléatoire X

$$p_i = p(X = x_i) \quad \text{pour} \quad i = 1,2,...,k$$

$$\sum_{i=1}^{k} p_i = 1$$

Entropie d ’une source d ’information

Quantité d'information moyenne associée à chaque symbole de la source = **entropie**

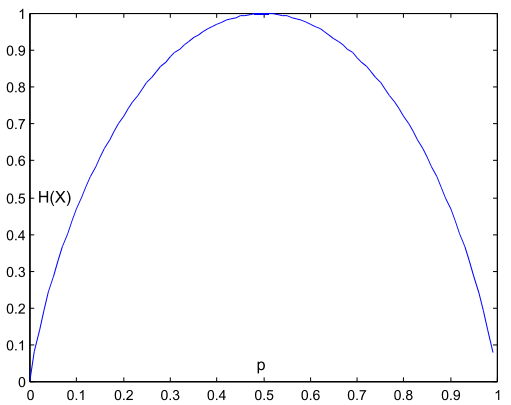
$$H(X) = E(I(X)) = \sum_{i=1}^{N} p_i.\log(1/\ p_i) = -\sum_{i=1}^{N} p_i.\log(\ p_i)$$

Entropie d ’une source binaire

$$H(X) = \begin{cases} -\ p.\log(\ p) - (1-p).\log(1-p) & \text{pour } 0 < p < 1 \\ 0 \quad \text{si} \quad p = 0 \text{ ou } 1 \end{cases}$$

$p(1) = p$

$p(0) = 1 - p$



Propriétés de l 'entropie

- **Additivité** : de part la définition de l'information propre.
- **Positive** : $H(X) = H(p_1, p_2, ..., p_n) \geq 0$
- **Bornée** : $H(X) \leq H(\frac{1}{n}, \frac{1}{n}, ..., \frac{1}{n}) = \log(n)$
- **Continuité** : l'entropie est une fonction continue de chaque variable p_i.

• **Redondance :**

$$R = H_{\max}(X) - H(X) \qquad \rho = 1 - \frac{H(X)}{H_{\max}(X)}$$

Entropie & Débit d 'information

- Le débit d'information d'**une source** est donné par le produit son entropie (valeur moyenne de l'info /symbole) par le nombre moyen de symboles par seconde, ce qui équivaut à :

$$D_X = \frac{H(X)}{\tau} \quad (bits.s^{-1}) \quad \text{avec } \tau \text{ durée moyenne d'un symbole}$$

• **Source Qaire :**

- **Source Q^{aire}** : source S dont l'alphabet possède Q éléments
- **k^{ième} extension** : source S^k dont l'alphabet est obtenu en groupant par bloc de k celui de la source S (ordre k)

Petit bilan

- 1 source est caractérisée par son entropie
- Le théorème du codage source affirme qu'il est toujours possible de trouver un code optimal pour transmettre l'information sans perte mais il ne dit pas comment trouver ce codage !
- Comment caractériser le canal ?

Information mutuelle

$$I(x_k ; y_k) = \log(1/\, p(x_k; y_k)) = \log(\, p(x_k \,/\, y_k)/\, p(x_k))$$

Exprime le lien qui existe entre un symbole émis et un symbole reçu

Propriétés :

$I(x;y) = I(y;x)$

$I(x/y) = I(x;y) + I(x)$

$I(x/y) = I(x)$ si x et y indépendants

Règle de Bayes : $p(x,y) = p(x/y).p(y) = p(y/x).p(x) = p(y,x)$

Transinformation

- Quantité moyenne d 'information transmise par le canal :

$$I(X;Y) = \sum_{i=1}^{n}\sum_{j=1}^{m} p(x_i, y_j).\log(\frac{p(x_i, y_j)}{p(x_i).p(y_j)})$$

$$I(X;Y) = \sum_{k=1}^{K}\sum_{j=1}^{J} p(x_k, y_j)\log\frac{p(y_j|x_k)}{p(y_j)} = \sum_{k=1}^{K}\sum_{j=1}^{J} p(x_k, y_j)\log\frac{p(x_k|y_j)}{p(x_k)}$$

- Entropie réunie ou conjointe (qtté d'info dans le syst. de comm.)

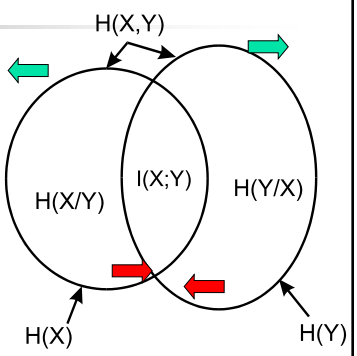
$$H(X,Y) = -\sum_{i=1}^{n}\sum_{j=1}^{m} p(x_i, y_j).\log(\ p(x_i, y_j))$$

- Entropie conditionnelle ou équivoque (= incertitude)

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

Transinformation

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



- Canaux non perturbés ⬅
  - $I(X;Y) = H(X) = H(Y)$
  - $H(X,Y) = H(X) = H(Y)$
  - $H(X/Y) = H(Y/X) = 0$
- Canaux très perturbés ➡
  - $I(X,Y) = 0$
  - $H(X,Y) = H(X) + H(Y)$
  - $H(X/Y) = H(X)$ et $H(Y/X) = H(Y)$

Notion de Capacité d'un canal

Nous avons vu que :

- H(X) caractérise la source
- I(X;Y) dépend de la source → p(x)
- I(X;Y) dépend du canal → p(x/y) = P

- Cas extrêmes :
  - I(X;Y) = H(X) → canal non bruité
  - I(X;Y) = 0 → canal bruité

- I(X;Y) varie entre $0 \leq I(X;Y) \leq H(X)$ → on définit C

Capacité d'un canal

Capacité: quantité maximum d'information que l'on peut transmettre dans un canal de télécommunications avec une probabilité d'erreur arbitrairement faible

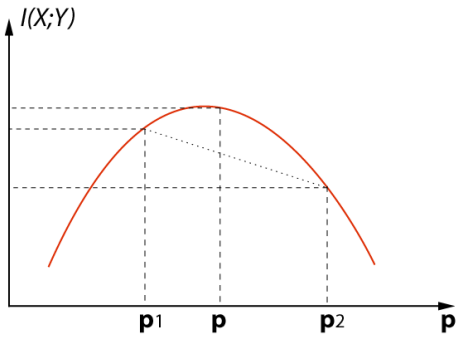
Autres définitions :

La capacité C d 'un canal est la plus grande quantité d'information moyenne qu 'il est capable de transmettre de son entrée à sa sortie.

La capacité C d 'un canal est le maximum de l'information mutuelle moyenne I(X;Y) avec X entrée, Y sortie.

Capacité d'un canal

$$C = \max_{\mathbf{p}} I(X;Y)$$



Capacité d'un canal

- Extensions d 'ordre n de la source :
  - On ajoute un buffer a la source qui attend d 'avoir reçu n symboles avant de transmettre *u*
  - En sortie : M^n messages *u* possibles
  - Récepteur recoit les messages *v*
- Information mutuelle :

$$I(X^n,Y^n) = \sum_{X^n}\sum_{Y^n} p(u,v).\log_2(p(v/u)/p(v))$$

Capacité d'un canal

- Si les symboles sont statistiquement indépendants
- Quantité moyenne fournie par un symbole

$$I(X;Y) = \frac{I(X^n,Y^n)}{n}$$

- $C = \max_{p(u),n} \frac{I(X^n,Y^n)}{n}$

Type de Canaux discrets

Canal : milieu de transmission de l'information situé entre la source et la destination. Le canal opère une transformation entre l'espace des symboles à l'entrée et celui de la sortie.

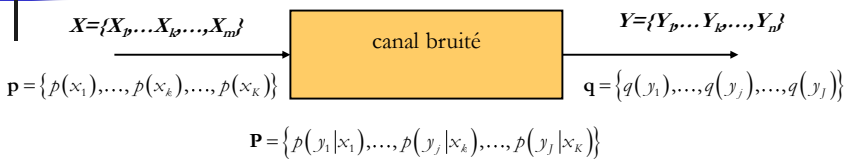
Canal discret : les espaces d'entrée et de sortie sont discrets

Canal continu : les espaces d'entrée et de sortie sont continus

Canal sans mémoire : si la transformation d'un symbole x à l'entrée en un symbole y en sortie ne dépend pas des transformations antérieures

Canal stationnaire : si les transformations ne dépendent pas de l'origine des temps

Canaux discrets



• **Matrice stochastique du canal :**

$$P(Y/X) = \begin{bmatrix} p(y_1/x_1) & p(y_1/x_2) & \dots & p(y_1/x_m) \\ p(y_2/x_1) & p(y_2/x_2) & & p(y_2/x_m) \\ \dots & & & \dots \\ p(y_n/x_1) & p(y_n/x_2) & \dots & p(y_n/x_m) \end{bmatrix}$$

Capacité d'un canal discret

- Rappel : Loi de Bayes

$$P(y_i) = \sum_{j=1}^{m} p(x_j, y_i) = \sum_{j=1}^{m} p(x_j).p(y_i / x_j)$$

- Sous forme matricielle :
  - P(Y)=P(X).P(Y/X)
  - P(X,Y)=P(X,X).P(Y/X) avec P(X,X)=

$$\begin{bmatrix} p(x_1) & 0 & \dots & 0 \\ 0 & p(x_2) & & 0 \\ \dots & & & \dots \\ 0 & 0 & \dots & p(x_m) \end{bmatrix}$$

Capacité d'un canal discret

- Canal uniforme en entrée / Canal uniforme en sortie (lignes)
- Canal uniforme en entrée et sortie :

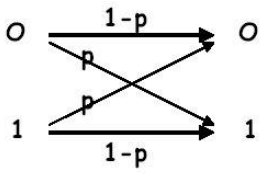
$$\begin{bmatrix} p_1 & p_2 & \cdots & p_m \\ p_2 & p_1 & & p_3 \\ \cdots & & & \cdots \\ p_m & p_2 & \cdots & p_1 \end{bmatrix}$$

$$C = \log_2 N + \sum_{j=1}^{N} p_j . \log_2 p_j$$

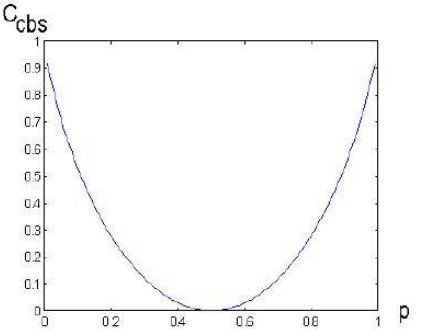
Capacité d'un canal

Exemple de Modélisation d'un canal :

Canal binaire symétrique (Canal stationnaire *sans* mémoire)



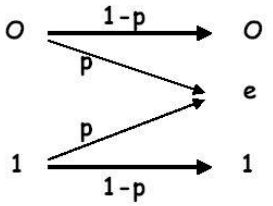
$C_{cbs} = 1 + (1-p) \log_2(1-p) + p \log_2(p)$



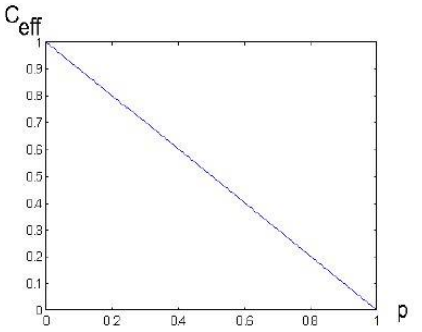
Capacité d'un canal

Exemple de Modélisation d'un canal :

Canal binaire à effacement (Canal stationnaire *sans* mémoire)



$C_{eff} = 1 - p$



D 'autres grandeurs

- **Efficacité d'un canal :** $\eta_c = \frac{I(X;Y)}{C}$
- **Taux d 'information (débit) :** $R_T = \frac{H(X)}{T_s}$
- **Capacité par unité de temps :** $C_T = \frac{C}{T_s}$
- **p_e : Probabilité moyenne d 'erreur :**

$$(R_T - C_T).T_s \leq H(p_e) + p_e.\log_2(M-1)$$

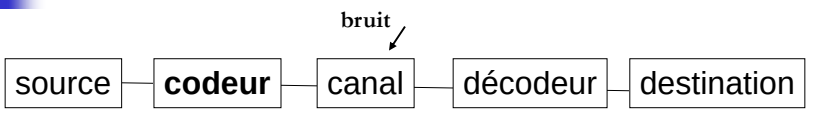
- **Entropie d 'erreur :**

$$H(p_e) = -p_e.\log p_e - (1-p_e).\log(1-p_e)$$

- **Théorème du codage source :**

Lorsque $R_t > C_t$* alors p_e tend vers 1 lorsque la lg des messages augmente. Inversement, *lorsque $R_t < C_t$ en utilisant une procédure adéquat de codage et décodage, on peut récupérer le message émis avec une probabilité d 'erreur relativement faible

Modèle d 'un système de communication



Alphabet du canal Z composé de D symboles

D < M (de la source) => codeur + décodeur

Message X_k devient Z_k de longueur n_k

$$n_m = \sum_{k=1}^{M} n_k . p(x_k)$$

On veut un codage pour lequel n_m est minimal

Modèle d 'un système de communication

Entropie max du codeur : $H(C)_{max} = \log D$

Entropie du codeur par symbole $H(C) = H(X) / n_m$

On peut définir l'efficacité du codeur : $e = H(C) / H(C)_{max}$

$e = H(X) / (n_m . \log D)$

e est maximum quand n_m est minimum

Modèle d 'un système de communication

Si n_k fixe alors $n_k = n_m$ et il faut $D^{n_m} \geq M$ d 'ou $n_m \geq \log M / \log D$

Si symboles équiprobables on a H(X)= log M d 'ou $n_m \geq H(X) / \log D$

Théorème de Shannon :

n_m est borné et on peut toujours trouvé un codage optimal en essayant d'avoir :

$$n_m = H(X) / \log D$$

Pour des codes bien choisi, on peut obtenir $\lim_{N \to \infty} n_m = H(X) / \log D$

Exemple

- Une source émet 8 lettres avec :
  p(a)=p(b)=1/4 p(c)=p(d)=1/8 p(e)=p(f)= p(e)=p(f)= 1/16
- Sur un canal binaire → nécessité d'un codeur
- 1ere solution de codage :
  - a → 000 b → 001 c → 010 d → 011
  - e → 100 f → 101 g → 110 h → 111
  - D'ou $n_{moy} = 3$ et $e_1 = H_X/3$
- 2e solution de codage :
  - a → 00 b → 01 c → 100 d → 101
  - e → 1100 f → 1101 g → 1110 h → 1111
  - D'ou $n_{moy} = 2{,}75$ et $e_2 = H_X/ 2{,}75$ → 2e solution mieux que 1ere
- Si destinataire reçoit 1100001001011111 → Pas d'ambiguïté car aucun code n'est le préfixe d'un autre → e a c d h

Et pour un signal continu ?

- On peut faire des raisonnements similaires mais c 'est beaucoup **moins simple !**

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

$$I(X;Y) = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} f(x,y).\log(f(x,y)/(f_1((x).f_2(y))dx$$

$$C = \max_{\mathbf{p}} I(X;Y)$$

  - Voir chapitre : Supports de transmission...

Et pour un signal continu ?

- **On se ramène la plupart du temps à :**
  - un signal limité dans le temps T, à une bande passante W
  - un canal soumis à un bruit blanc additif
  - Les répartitions en puissance du signal et du bruit suivent des distributions de probabilité gaussienne
  - Ce n 'est pas le cas dans la réalité mais cela fixe une référence dont on essaie de se rapprocher.

Chapitre 2 :

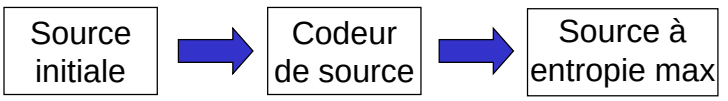
Codage de Source / Compression

Codage de source

Adapter la source au canal : - l'alphabet
- le débit

Utiliser la capacité du canal → maximiser I(X,Y)

- Hyp : Source stationnaire, canaux **sans perturbation**



Codeur de source → supprimer la **redondance**

Pourquoi ?

⇨ Photo scannée (diapo 36*24mm) 1200*600 dpi et 4 octets par pixels : 3,24 Mo

⇨ 5*36 pauses = 1 CDROM

⇨ Vidéo 25 images par secondes en 640*480 et 16 millions de couleurs : 176,4 Go pour 1h30 (17 disques durs de taille moyenne, 260 CD)

Propriétés d'un codeur de source

* **Régularité** : messages ≠ → codes ≠
* **Déchiffrabilité** : séparation des mots non ambiguë

• **Mot-code**

$[S]=[s_1,s_2, \ldots, s_N]$ $[X]=[x_1,x_2, \ldots, x_D]$

↳ $[C]=[c_1,c_2, \ldots, c_N]$

• **Exemple**

Symbole	Code A	Code B	Code C	Code D
S_1	00	0	0	0
S_2	01	10	01	10
S_3	10	110	011	110
S_4	11	1110	0111	111

Types de codage

- Code binaire = correspondance entre un ensemble d 'informations élémentaires (alphabet) et un ensemble de configurations binaires (mots codes)
  - souvent longueur fixe
  - pour un texte :
    - 10 chiffres + 26 lettres + symboles + caractères de contrôle
- CCITT (UIT-T) a normalisé plusieurs codes :
  - CCITT-2 : 5 bits
  - CCITT-5 : 7 bits ==> ASCII 7
- EBCDIC d 'IBM (8 bits)

Code ASCII 7

$b_6b_5b_4 \rightarrow$ $b_3b_2b_1b_0 \downarrow$	000	001	010	011	100	101	110	111
0000	NUL	DLE	SP	0	à (1)	P	\ (1)	p
0001	SOH	DC1	!	1	A	Q	a	q
0010	STX	DC2	"	2	B	R	b	r
0011	ETX	DC3	# (1)	3	C	S	c	s
0100	EOT	DC4	$ (1)	4	D	T	d	t
0101	ENQ	NAK	%	5	E	U	e	u
0110	ACK	SYN	&	6	F	V	f	v
0111	BEL	ETB	'	7	G	W	g	w
1000	BS	CAN	(	8	H	X	h	x
1001	HT	EM	)	9	I	Y	i	y
1010	LF	SUB	*	:	J	Z	j	z
1011	VT	ESC	+	;	K	° (1)	k	é (1)
1100	FF	FS	,	<	L	ç (1)	l	ù (1)
1101	CR	GS	–	=	M	§ (1)	m	è (1)
1110	SO	RS	.	>	N	^ (1)	n	~ (1)
1111	SI	US	/	?	O	_ (1)	o	DEL

Code ASCII 7

Fonctions de mise en page	
BS	*Backspace* (retour en arrière)
HT	*Horizontal Tabulation*
LF	*Line Feed* (nouvelle ligne)
VT	*Vertical Tabulation*
FF	*Form Feed* (nouvelle page)
CR	*Carriage Return* (retour chariot)
Fonctions de contrôle de la transmission	
SOH	*Start Of Heading* (début d'en-tête)
STX	*Start of Text* (début de texte)
ETX	*End of Text*
EOT	*End Of Transmission*
ENQ	*Enquiry* (demande)
ACK	*Acknowledge* (Acquittement)
DLE	*Data Link Escape* (echappement liaison de données)
NAK	*Negative Acknowledge* (acquittement négatif)
SYN	*Synchronous Idle* (caractère de synchronisation)
ETB	*End of Transmission Block* (fin de bloc)

Fonctions de commande des périphériques	
DC1	*X-on* (mise en route)
DC2	
DC3	*X-off* (arrêt)
DC4	
Fonctions de séparation dans les fichiers	
US	*Unit Separator* (séparateur de sous-articles)
RS	*Record Separator* (séparateur d'article)
GS	*Group Separator* (séparateur de groupes)
FS	*File Separator* (séparateur de fichiers)
Autres caractères	
NUL	caractère vide (sans aucun effet)
BEL	*Bell* (sonnerie)
SO	*Shift Out* (hors code)
SI	*Shift In* (retour en code)
CAN	*Cancel* (annulation)
EM	*End of Medium* (fin de support)
SUB	*Substitution*
ESC	*Escape* (echappement)
DEL	*Delete*

Code ASCII ANSI étendu

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	□	□	‚	ƒ	„	…	†	‡	ˆ	‰	Š	‹	Œ	□	□	□
9	□	‘	’	“	”	•	–	—	˜	™	š	›	œ	□	□	Ÿ
A		¡	¢	£	¤	¥	¦	§	¨	©	ª	«	¬	-	®	¯
B	°	±	²	³	´	µ	¶	·	¸	¹	º	»	¼	½	¾	¿
C	À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
D	Ð	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	Ý	Þ	ß
E	à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
F	ð	ñ	ò	ó	ô	õ	ö	÷	ø	ù	ú	û	ü	ý	þ	ÿ

Code à longueur variable / fixe

* **Code à décodage unique** : mot-code ⇔ symbole unique

* **Code séparable** : pas de signe de démarcation entre les mots

* **Code instantané** ou **irréductible** : on détermine les mots-codes à mesure que l'on reçoit les lettres de l'alphabet du code.
→ Aucun mot-code n'est le préfix d'un autre !

Code ASCII = 8 bits, pourtant dans un texte 2,5 bits / caractère suffiraient

Longueur moyenne d'un mot-code

D = nombre de symboles dans l 'alphabet du canal

• **Limite de la longueur moyenne**

$$\bar{l} = \sum_{i=1}^{N} p(x_i).l_i \qquad H(C) = H(X)/\bar{l} \qquad \bar{l} \geq \frac{H(X)}{\log D} = l_{\min}$$

• **Entropie - Efficacité - Redondance**

$$\eta = \frac{H(C)}{\log D} \qquad \eta = \frac{H(X)}{\bar{l}.\log D} \qquad \rho = 1 - \eta$$

Codes optimaux absolus

Codes dont l'efficacité est maximale : $\eta = 1$

$$\bar{l} = l_{\min} = \frac{H(X)}{\log D}$$

Théorème des canaux sans bruit (codage de source)

" Par un codage approprié (codage par groupe de n symboles de la source), l'information moyenne par lettre de l'alphabet du code peut être amenée aussi proche que l'on veut de la capacité du code, c'est-à-dire qu'**il existe toujours un codage optimal absolu** ."

Rq1 : à n fixé, le code qui donne $\eta_{max} < 1$ est dit 'optimal'

Méthodes de Compression

- Compression = réduction des informations
  - sans perte d 'information : réduction de la redondance
  - avec perte d 'information : images ou sons, restitution approximative de l 'original
- Méthodes :
  - analyse statistique générale : complexité élevée
  - heuristiques spécifiques : connaissances a priori du type d 'information
  - décomptaction par un programme externe ou auto-décompactable

Propriétés des compacteurs

- Compacteur => décompacteur
- Bijection ou non
- Pour tout compacteur , il existe un fichier non compactable
- plus il y a d 'heuristiques, plus il y a d 'efficacité

Quelques techniques primitives

- **Codage des répétitions :**
  - succession de symboles identiques (RLE)

00000111110000000000000000	→	5w5b17w
00000000000111110000000000	→	11w5b11w
A B C C C C C C A B C A B C	→	A B !6C A B C A B C

  - Variantes par choix de la taille des symboles(bit, octet,…) ou de la manière de balayer le fichier (ligne, colonne, fenêtre)
  - Utiliser sur les images : BMP, PCX

Quelques techniques primitives

- **Codage topologique :**
  - topologie d 'une suite de bits dominante
  - par exemple X
  - on utilise un octet topologique (bit à 1)pour décrire la position des X
    - ex : «aX19XXXXXXaXXXX» (16octets)
      «(01001111)a19(11101111)a»
  - Il faut que la fréquence de X > 1/8
  - Tc=T.(1+1/8-f)

Quelques techniques primitives

- **Codage relatif :**
  - succession d 'octets avec faible amplitude de variation
    - ex : 10101 101. 10101 000. 10101 110. 10101 001 (32b)
    - 5(3b) 10101 4(8b) 101 000 110 001 (28b)
  - Variante, on code les différences (c. à 2)
    - 10101101 1011 0001 1100 (poids faibles)

Méthodes statistiques

bananes et ananas

s	f(s)	p(s)	H(s)	H
A	5	0.294	1.765	8.825
N	4	0.235	2.09	8.36
E	2	0.118	3.1	6.2
S	2	0.118	3.1	6.2
<space>	2	0.118	3.1	6.2
B	1	0.059	4.08	4.08
T	1	0.059	4.08	4.08
Total	**17**		**21.315**	**43.945**

e	11	n	6,6	m	3	f	1,22	j	0,17	ç	0,04
s	9,27	o	6,04	p	2,79	v	1,01	z	0,12	w	0,03
i	8,61	é	4,87	d	2,23	q	0,88	k	0,12	ô	0,03
a	7,8	l	4,75	g	1,91	y	0,65	ï	0,08	î	0,03
r	7,42	u	4,49	h	1,67	x	0,41	â	0,08	û	0,03
t	6,8	c	3,87	b	1,64	è	0,3	ê	0,06	à	0,01

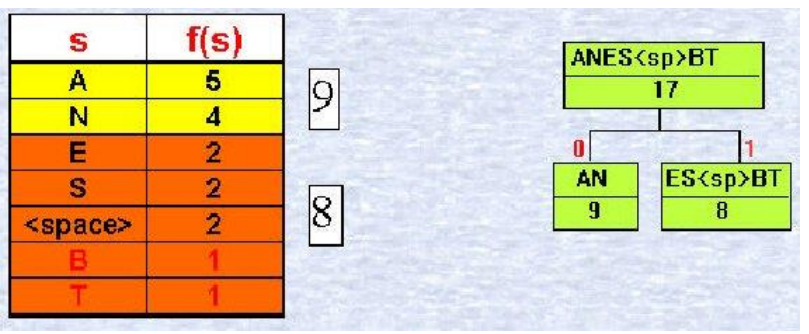
Codage de Shannon-Fano

Algorithme de génération d'un **codage optimal absolu**, pour des sources divisibles récursivement (jusqu'à un symbole par ensemble) en deux sous-ensembles équiprobables.

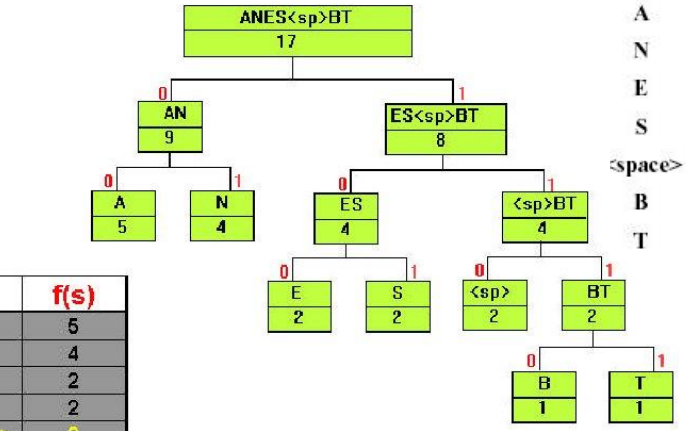
Symboles s_k	Proba $p(s_k)$					Mots-codes c_k	Longueur l_k
s_1	0.25	0	0			00	2
s_2	0.25		1			01	2
s_3	0.125	1	0	0		100	3
s_4	0.125			1		101	3
s_5	0.0625		1	0	0	1100	4
s_6	0.0625				1	1101	4
s_7	0.0625			1	0	1110	4
s_8	0.0625				1	1111	4

Exercice : Bananes et ananas

s	f(s)
A	5
N	4
E	2
S	2
<space>	2
B	1
T	1



Bananes et ananas



s	f(s)
A	5
N	4
E	2
S	2
<space>	2
B	1
T	1

A	00
N	01
E	100
S	101
<space>	110
B	1110
T	1111

Codage binaire de Huffman (1952)

- Algorithme de génération d'un **codage optimal** symbole par symbole.

- Longueur variable → codes longs pour probas faibles

Codage binaire de Huffman (1952)

L'algorithme de création de l ’arbre :

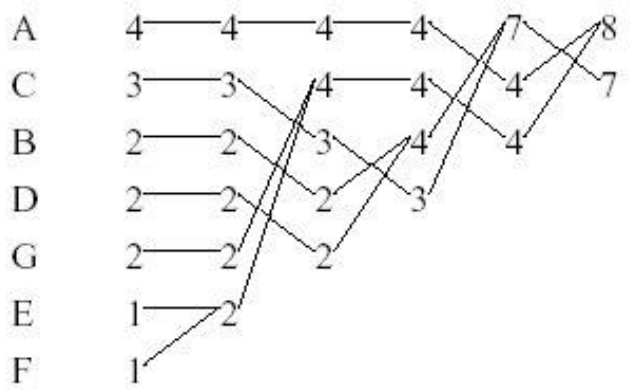
- 1° Evaluer les fréquences d'occurrence des symboles du fichier
- 2° Classer les symboles en ordre décroissant des fréquences d'apparition.
- 3° Regrouper de façon séquentielle les paires de symboles de plus faible probabilité, en reclassant
- symboles et groupes si nécessaire.
- 4° Calculer les codes avec retour en arrière en ajoutant, dans chaque point de regroupement, un 0 à une branche et un 1 à l'autre branche.

Codage binaire de Huffman (1952)

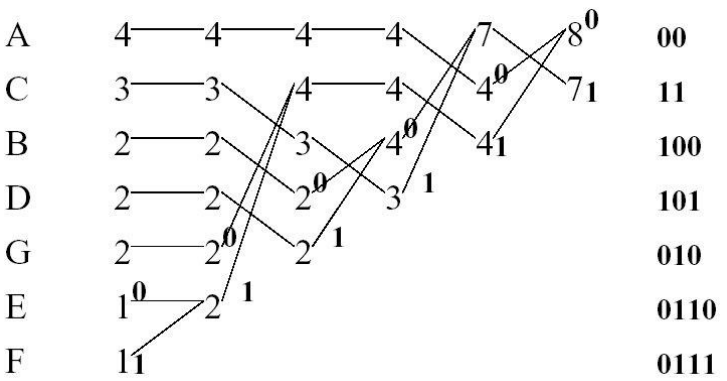
- **Exemple :**
  - Chaîne à comprimer : "ABCFGABDDACEACG" (45 bits), représentée avec 3 bits/lettre.
- **Classement des symboles :**

Symbole	Fréquence
A	4
C	3
B	2
D	2
G	2
E	1
F	1

Codage binaire de Huffman (1952)



Codage binaire de Huffman (1952)



Codage binaire de Huffman (1952)

Code séparable mais nécessité d 'avoir les codes employés => table

- Codage :
  - * Extraction des probabilités
  - * Création de l'arbre
  - * Création de la table d'Huffman
  - * Codage

On transmet la table + les codes en binaire :

- * Lecture de la table d'Huffman
- * Création de l'arbre de décodage
- * Lecture séquentielle et décodage

Codage binaire de Huffman (1952)

• La table de correspondance ou table des fréquences des codes employés est incluse dans l 'en tête du fichier :

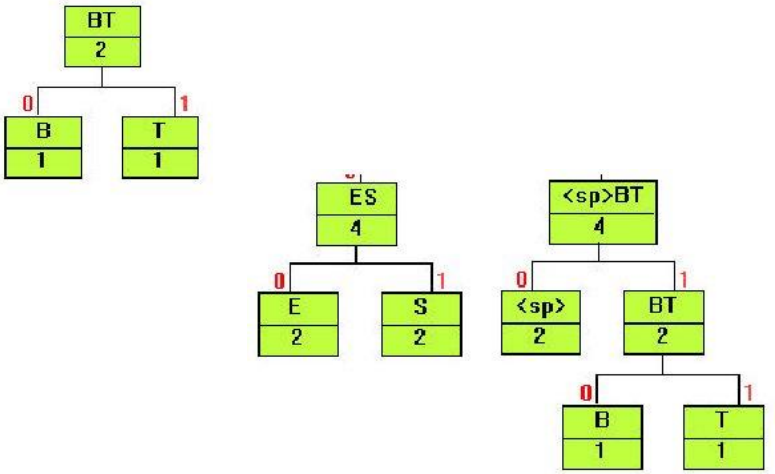
- • codage des fréquences d 'apparition de chaque code (256)
- • liste des codes avec leur taille

•Taux de compression et redondance :

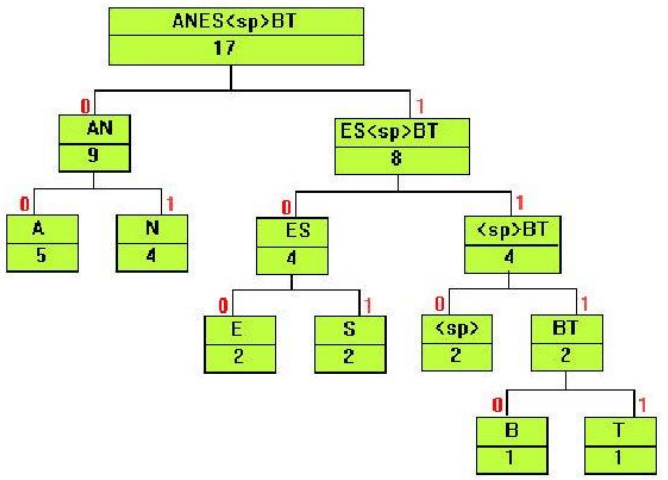
$$E_0 = -\sum_{i=1}^{256} p_i . \log p_i \qquad R_0 = \log N - E_0 \qquad R_0 \geq 8 + \sum_{i=1}^{64} 1/64 . \log 1/64 = 2$$

• Généralement, on travaille avec k=8, mais on peut choisir k > 8

Bananes et ananas



Bananes et ananas



Exercice : Comparaison Huffman / Fano

BANANES ET ANANAS

- bananes et ananas = 136 bits en ascii
- Avec Huffman = ?
- On ajoute en-tête => 120 bits !!!

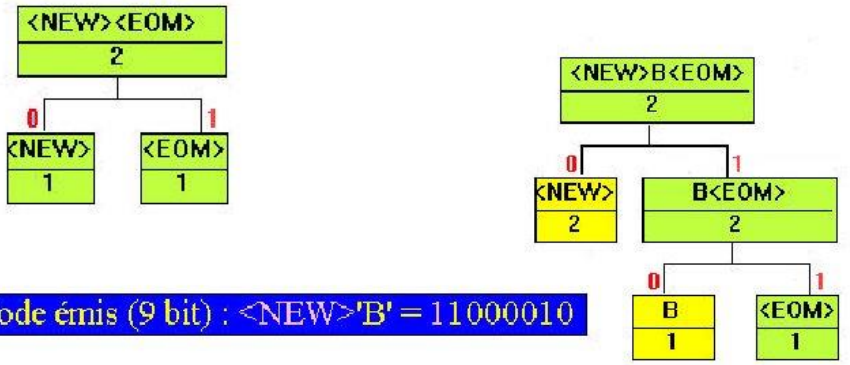
- Plus fichier grand => plus efficacité augmente

Algorithmes dynamiques

- Huffman :
  - en tête + 2 lectures + statistique sur fichier entier

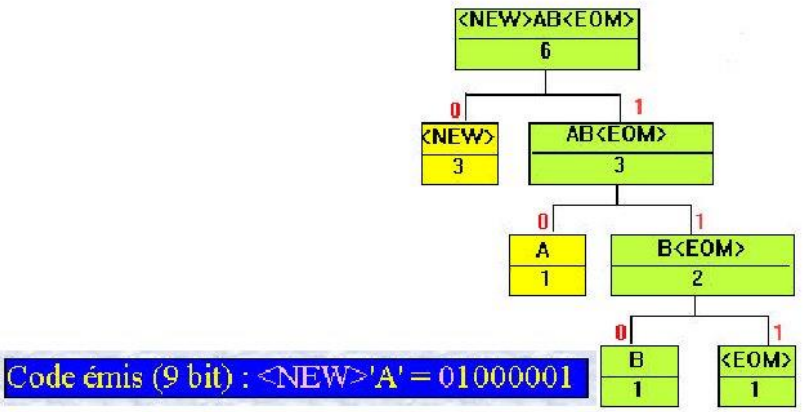
- Améliorations :
  - Algorithmes sans en tète :
    - **table des fréquences construire durant la lecture, code d 'un symboles varie au cours du temps. Au départ les codes proviennent d 'une table initiale connue par compacteur et décompacteur. Apres lecture de chaque symbole la table est reclassée en fonction de la fréquence : réattribution des codes aux symboles**
  - Algorithmes avec en tête plus court :
    - **idem mais les codes initiaux sont calculés au lieu d être pré-attribués => il faut envoyer la table finale = en-tète**

Bananes et ananas



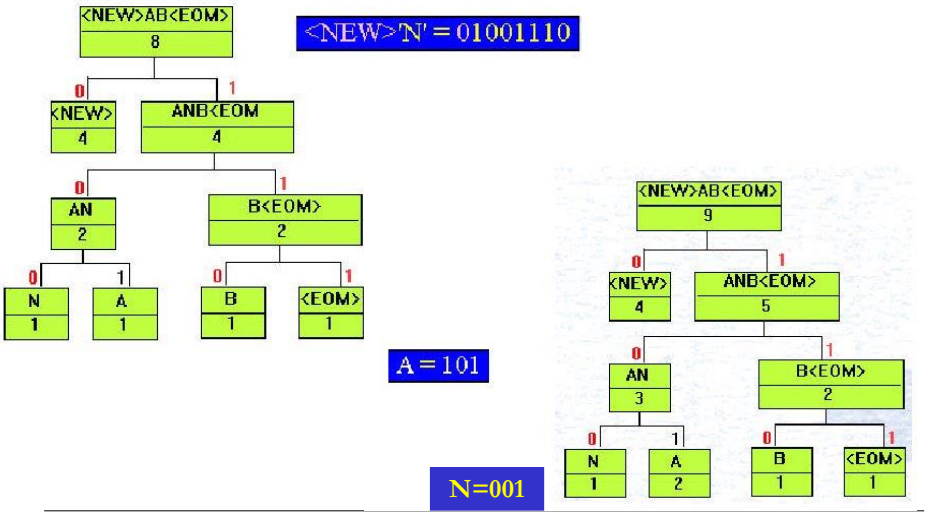
Code émis (9 bit) : <NEW>'B' = 11000010

Bananes et ananas



Code émis (9 bit) : <NEW>'A' = 01000001

Bananes et ananas



Algorithmes d 'ordre k > 0

- Ordre 0 : les symboles sont considérés indépendamment les uns des autres
- les redondances se manifestent aussi sur les sequences de symboles :
  - les méthodes d 'ordre k tiennent compte des k-ième antécédents de chaque symboles
  - Algo proche de celui d 'Huffman : on crée une table pour chaque antécédent
  - PB : plus k augmente => plus taille table (en-tète) augmente

Algorithmes d 'ordre k > 0

- Exemple : chaîne « ACABBDDBAAA »

Antécédent	$f_{i\|j}$	Code
sachant A	f(A) = 2 f(B) = 1 f(C) = 1	0 10 11
sachant B	f(A) = 1 f(B) = 1 f(D) = 1	1 00 01
sachant C	f(A) = 1	0
sachant D	f(B) = 1 f(D) = 1	0 1

Huffman : exemples d 'utilisation

- CCITT, Fax groupe III
  - ↳ Huffman sur les plages de 0 précédant les 1
- JPEG
  - ↳ Huffman sur les plages de 0 précédant les coeff. DCT

Table d'Huffman FAX III

0	00110101
1	000111
2	0111
3	1000
4	1011
5	1100
6	1110
7	1111
8	10011
9	10100
10	00111
11	01000
12	001000
13	000011
14	110100
15	110101
16	101010
17	101011
18	0100111
19	0001100
20	0001000
21	0010111
22	0000011
23	0000100
24	0101000
25	0101011
26	0010011
27	0100100
28	0011000
29	00000010
30	00000011
31	00011010

32	00011011
33	00010010
34	00010011
35	00010100
36	00010101
37	00010110
38	00010111
39	00101000
40	00101001
41	00101010
42	00101011
43	00101100
44	00101101
45	00000100
46	00000101
47	00001010
48	00001011
49	01010010
50	01010011
51	01010100
52	01010101
53	00100100
54	00100101
55	01011000
56	01011001
57	01011010
58	01011011
59	01001010
60	01001011
61	00110010
62	00110011
63	00110100

64	11011
128	10010
192	010111
256	0110111
320	00110110
384	00110111
448	01100100
512	01100101
576	01101000
640	01100111
704	011001100
768	011001101
832	011010010
896	011010011
960	011010100
1024	011010101
1088	011010110
1152	011010111
1216	011011000
1280	011011001
1344	011011010
1408	011011011
1472	010011000
1536	010011001
1600	010011010
1664	011000
1728	010011011
EOL	000000000001

Table 6.5 - Mots de code spécifiant la longueur des séquences correspondant au blanc lors de la transmission de documents par télécopie.

Codage de type dictionnaire (1977)

1977 : LZ (Lempel & Ziv) ⇨ 1984 : LZW (Welch)

• Dictionnaire = tableau dans lequel sont rangés des séquences de symboles de taille variables
• Remplacer les symboles/séquences par leur adresse dans le tableau

Dictionnaire de symboles incrémenté dynamiquement
↳ **apprentissage donc pas d'en-tete**

Fichier codé = suite des adresses des mots du dico

! Gérer l'incrément des bits d'adresse

Compression LZW

- Si le compresseur observe un séquence déjà rencontrée il utilise le code correspondant.
- Grandes étapes (S: séquence, P: Préfixe, C: symbole courant) :
  - P = LireFichier
  - C = LireFichier
  - On construit S = P + C
  - Si S est dans la Table alors P = S
  - Sinon seul P est dans la Table alors
    - on inscrit le code correspondant (à P) dans le fichier compressée
    - on ajoute S dans la table
    - on pose P = C
  - Répéter

Compression LZW

Pos	1	2	3	4	5	6	7	8	9
Char	A	B	B	A	B	A	B	A	C

Dictionnaire initial

(1) A
(2) B
(3) C

Code (index)

Étape/Pos 1. 2	S / P	Dictionnaire	Output
	AB / A	(4) A B	(1)

http://www.rasip.fer.hr/research/compress/algorithms/index.html

Compression LZW

Pos	1	2	3	4	5	6	7	8	9
Char	A	B	B	A	B	A	B	A	C

Dictionnaire initial

(1) A
(2) B
(3) C

Code (index)

Étape/Pos	S / P	Dictionnaire	Output
1. 2		(4) A B	
2. 3	BB / B	(5) B B	(1)(2)

http://www.rasip.fer.hr/research/compress/algorithms/index.html

Compression LZW

Pos	1	2	3	4	5	6	7	8	9
Char	A	B	B	A	B	A	B	A	C

Dictionnaire initial

(1) A
(2) B
(3) C

Code (index)

Étape/pos	S / P	Dictionnaire	Output
1. 2		(4) A B	
2. 3		(5) B B	
3. 4	BA / B	(6) B A	(1)(2)(2)

http://www.rasip.fer.hr/research/compress/algorithms/index.html

Compression LZW

Pos	1	2	3	4	5	6	7	8	9
Char	A	B	B	A	B	A	B	A	C

Dictionnaire initial

(1) A
(2) B
(3) C

Code (index)

Étape/pos	S / P	Dictionnaire	Output
1. 2		(4) A B	
2. 3		(5) B B	
3. 4		(6) B A	
4. 5,6	ABA / AB	(7) A B A	(1)(2)(2)(4)

http://www.rasip.fer.hr/research/compress/algorithms/index.html

Compression LZW

Pos	1	2	3	4	5	6	7	8	9
Char	A	B	B	A	B	A	B	A	C

Dictionnaire initial

(1) A
(2) B
(3) C

Code (index)

Étape/pos	S / P	Dictionnaire	Output
		(4) A B	
		(5) B B	
		(6) B A	
		(7) A B A	
5. 7,8,9	ABAC / ABA	(8) A B A C	(1)(2)(2)(4)(7)
6. 10	?C / C		(1)(2)(2)(4)(7)(3)

http://www.rasip.fer.hr/research/compress/algorithms/index.html

Décompression LZW

Code recu : (1)(2)(2)(4)(7)(3)

Dictionnaire initial

(1) A
(2) B
(3) C

Étape	Code lu	Output	Dictionnaire
1.	(1)	A	
2.	(2)	B	(4) A B
3.	(2)	B	(5) B B
4.	(4)	A B	(6) B A
5.	**(7)** ?	A B A	(7) A B A
6.	(3)	C	(8) A B A C

Algorithme LZSS

- SS = Storer et Szymanski
- Idem LZW mais avec un buffer circulaire
- Fichier à comprimer est mis par morceau dans buffer
- Grandes étapes :
  - lecture caractère
  - si répétition :
    - on stocke 0 + position relative + longueur seq
  - sinon
    - on stocke 1 + caractère lu
- Détection plus rapide des séquences répétitives maximales
- PB = taille tampon + taille du champs longueur
- + contenu initial du buffer

Algorithmes à dictionnaire

- Faible complexité des calculs
- Rapide
- Contenu initial du dictionnaire ou buffer ?
- Pkzip, GIF, TIFF utilisent LZW ou variantes
- Algorithmes mixtes :
  - Huffman + LZW
  - LZW + Huffman

Compression avec pertes

- Images, vidéos, sons → Pertes acceptées
- Systèmes sensoriels humains insensibles
- Applications : Visioconférences, mp3, ...

Compression d 'images

- Image = Matrice 2D de pixels
- Pixel = entier, plusieurs entiers, adresse dans table, …
- œil perçoit 24 millions de couleurs mais est moins sensible aux contours
- La compression se fait par filtrage des hautes fréquences (surtout celles de faibles amplitudes)
- Après des opérations de lissage

Compression et filtrage

- Techniques de Filtrage :
  - direct : à l'aide de masque de lissage :

$$M = \frac{1}{4}\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$$

Ce calcul est fait sur chaque pixel de l'image

  - par transformation :
    - Transformée de Fourrier
    - Transformée en cosinus discret

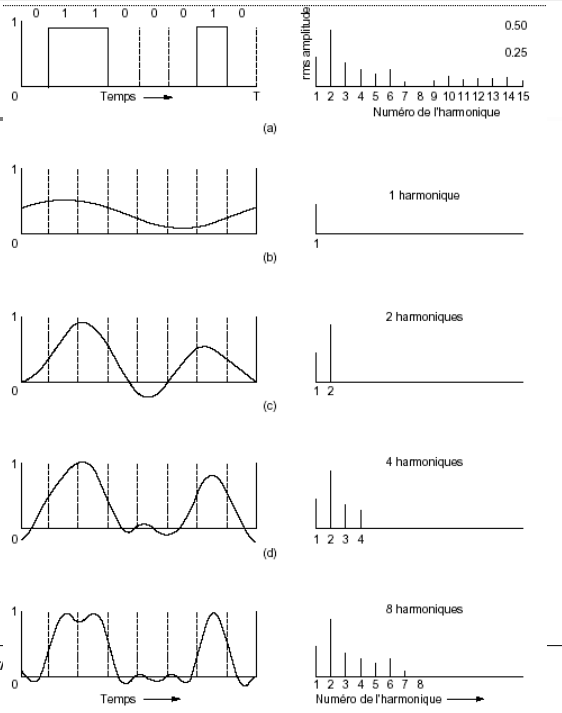
Transformée de Fourier

- un signal périodique quelconque peut être décomposé en une suite de fonctions périodiques sinusoïdales: les *harmoniques*.

$$s(t) = \frac{1}{2}a_0 + \sum_{n=1}^{\infty} a_n \sin(2\pi n f t) + \sum_{n=1}^{\infty} b_n \cos(2\pi n f t).$$

- f =1/T fréquence fondamentale
- an; bn amplitudes des composantes harmoniques
- Un signal rectangle est composé d'un nombre infini de composantes sinusoïdales avec une fréquence fondamentale f et les multiples 3f; 5f; . . .

Fourier



Transformée de Fourier 2D

Cas continu

$$\Im\{f(x,y)\} = F(u,v) = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} f(x,y)\, e^{-i2\pi(ux+vy)}\, dx\, dy$$

$$f(x,y) = \Im^{-1}\{F(u,v)\} = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} F(u,v)\, e^{+i2\pi(ux+vy)}\, du\, dv$$

Cas discret

$$\Im\{f(m,n)\} = F(u,v) = \frac{1}{MN}\sum_{m=0}^{M-1}\sum_{n=0}^{N-1} f(m,n)\, e^{-i2\pi\left(\frac{um}{M}+\frac{vn}{N}\right)}$$

$$f(m,n) = \Im^{-1}\{F(u,v)\} = \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} F(u,v)\, e^{+i2\pi\left(\frac{um}{M}+\frac{vn}{N}\right)}$$

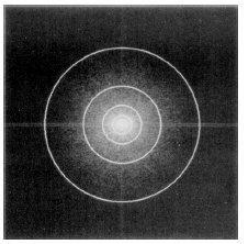
Cas M=N

$$\Im\{f(m,n)\} = F(u,v) = \frac{1}{N}\sum_{m=0}^{N-1}\sum_{n=0}^{N-1} f(m,n)\, e^{-\frac{i2\pi}{N}(um+vn)}$$

$$f(m,n) = \Im^{-1}\{F(u,v)\} = \frac{1}{N}\sum_{m=0}^{N-1}\sum_{n=0}^{N-1} F(u,v)\, e^{+\frac{i2\pi}{N}(um+vn)}$$

Transformée de Fourier 2D

Filtrage passe-bas



Spectre de l'image d'origine. Les cercles représentent respectivement 90, 93, 95, 99 et 99,5 % de la puissance totale de l'image

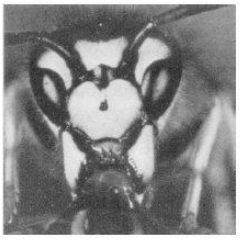
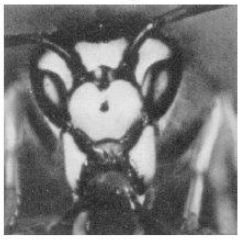


Image d'origine

Filtrage passe-bas avec 90 % de la puissance

Filtrage passe-bas avec 95 % de la puissance



Filtrage passe-bas avec 99 % de la puissance

10

Transformée Cosinus Discrète

$$C(u,v) = k_1(u)k_2(v)\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)\cdot cos\left(\pi u\frac{x+\frac{1}{2}}{N}\right)\cdot cos\left(\pi v\frac{y+\frac{1}{2}}{N}\right)$$

$$k_1(u) = \begin{cases}\sqrt{\frac{1}{N}} & \text{si } u = 0\\ \sqrt{\frac{2}{N}} & \text{sinon}\end{cases} \qquad k_2(v) = \begin{cases}\sqrt{\frac{1}{N}} & \text{si } v = 0\\ \sqrt{\frac{2}{N}} & \text{sinon}\end{cases}$$

$$f(x,y) = \sum_{u=0}^{N-1}\sum_{v=0}^{N-1} C(u,v)\cdot\cos\left(\pi x\frac{u+\frac{1}{2}}{N}\right)\cdot\cos\left(\pi y\frac{v+\frac{1}{2}}{N}\right)$$

JPEG (Joint Photographic Experts Group)

⇨ **4 méthodes de compression couplées :**

⇨ traduction de RVB à YUV, On conserve Y et on attribue la moyenne des chrominance à tout groupes de 4 pixels

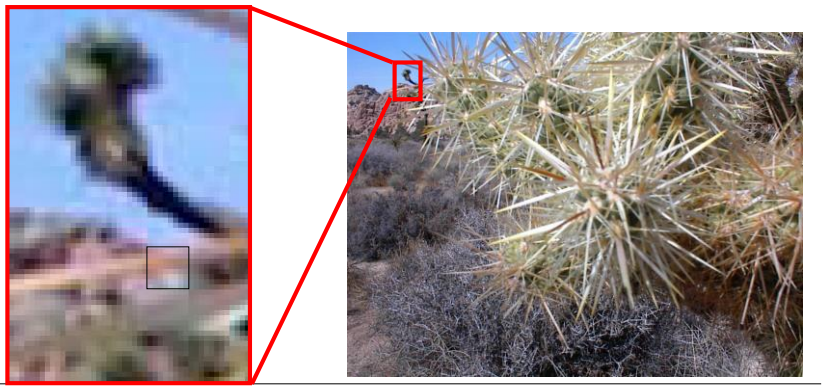
⇨ Découpage en 8*8, puis DCT (pour les 3 plans). Les 64 valeurs sont simplifiées en fonction de l 'indice de compression

⇨ RLE

⇨ Huffman

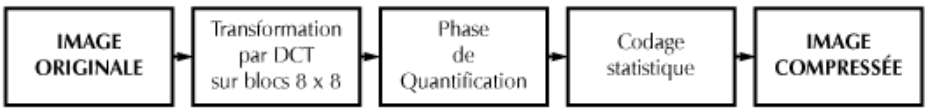
JPEG (Joint Photographic Experts Group)

- Compression forte MAIS avec perte (25:1)
- Transformée cosinus discrète rapide (FDCT) 2D **de sous-images 8 x 8**



JPEG (Joint Photographic Experts Group)

- Compression par DCT sur les blocs 8x8 :



JPEG (Joint Photographic Experts Group)

- Bloc de 64 pixels :

100	155	131	116	151	135	131	211
120	135	127	88	155	131	155	179
120	135	151	100	179	116	155	167
120	155	151	108	191	112	155	179
135	151	135	120	197	112	179	179
120	151	155	151	151	116	179	179
135	151	167	167	151	151	167	171
120	151	179	151	151	131	155	167

JPEG (Joint Photographic Experts Group)

- - 128 + Transformation DCT :

145	-84	34	-69	4	-66	-35	72
-45	-28	28	19	10	-54	5	15
0	-2	-8	-15	-9	0	30	-41
9	-14	15	-11	5	8	-12	-32
1	1	3	-11	7	-23	-4	0
18	4	-17	-10	4	-10	7	-10
-5	1	-7	-20	1	-1	-3	5
3	1	1	9	2	7	2	-2

JPEG (Joint Photographic Experts Group)

- Quel est l'intérêt de cette transformation ?
  - coefficients de forte valeur absolue sont situés en haut et à gauche
  - l'importance des coefficients pour la reconstitution de l'image diminue quand on se déplace en diagonale du haut à gauche vers le bas à droite.
  - Quantification

JPEG (Joint Photographic Experts Group)

- De quoi s'agit-il ?
- Pas de quantification = Perte de l'information « astucieuse »
  - le pas de quantification dont dépend la précision de l'image restituée va dépendre de la position de la valeur dans la matrice.
  - pas relativement petit pour les valeurs importantes (en haut à gauche)
  - pas de plus en plus grand quand on descend vers le bas et la droite.
  - L'ensemble des pas constituent une matrice de quantification.
    - construites en fonction de critères psycho-visuels, ...
      - nous allons en fabriquer une avec une petite formule :

$$Q(i,j) = 1 + (1 + i + j) \times Fq$$

JPEG (Joint Photographic Experts Group)

- Nous prendrons Fq = 5 . Il s'agit d'un facteur de qualité :

6	11	16	21	26	31	36	41
11	16	21	26	31	36	41	46
16	21	26	31	36	41	46	51
21	26	31	36	41	46	51	56
26	31	36	41	46	51	56	61
31	36	41	46	51	56	61	66
36	41	46	51	56	61	66	71
41	46	51	56	61	66	71	76

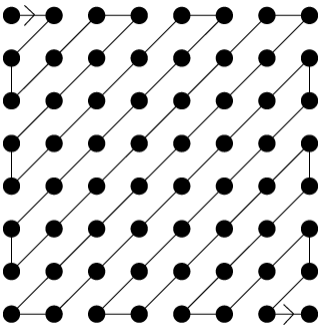
JPEG (Joint Photographic Experts Group)

- Division des valeurs de la matrice de données par les valeurs de la matrice de quantification :

24	-7	2	-3	0	-2	0	1
-4	-1	1	0	0	-1	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

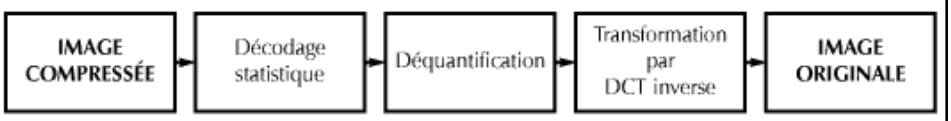
JPEG (Joint Photographic Experts Group)

- Huffman ou/et RLE sur les 0 :



JPEG (Joint Photographic Experts Group)

- Décompression :



JPEG (Joint Photographic Experts Group)

- décodage statistique + déquantification (xMq):

144	-77	32	-63	0	-62	0	41
-44	-16	21	0	0	-36	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

JPEG (Joint Photographic Experts Group)

- DCT inverse et ajouter 128 :

112	145	137	107	149	130	139	124
114	145	139	110	149	131	141	183
117	145	143	116	151	133	144	181
121	145	148	124	152	135	148	179
126	145	153	132	154	137	152	177
130	145	158	139	155	139	156	175
133	145	162	145	157	141	159	173
135	146	164	148	157	142	161	172

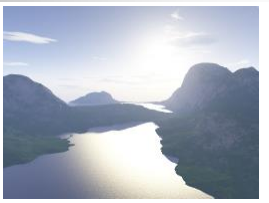
JPEG (Joint Photographic Experts Group)

100	155	131	116	151	135	131	211
120	135	127	88	155	131	155	179
120	135	151	100	179	116	155	167
120	155	151	108	191	112	155	179
135	151	135	120	197	112	179	179
120	151	155	151	151	116	179	179
135	151	167	167	151	151	167	171
120	151	179	151	151	131	155	167

112	145	137	107	149	130	139	124
114	145	139	110	149	131	141	183
117	145	143	116	151	133	144	181
121	145	148	124	152	135	148	179
126	145	153	132	154	137	152	177
130	145	158	139	155	139	156	175
133	145	162	145	157	141	159	173
135	146	164	148	157	142	161	172

JPEG : Illustrations

- Image BMP initiale : 144 Ko



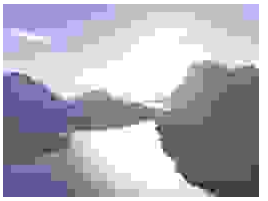
- JPEG indice 20 : 5,18 Ko
  - 28:1

- JPEG indice 50 : 2,97 Ko
  - 48,5:1

JPEG : Illustrations

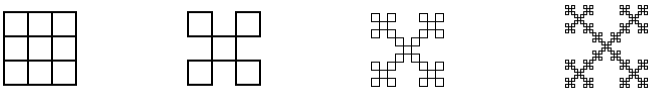
⇨ JPEG indice 92 : 1,05 Ko
⇨137:1

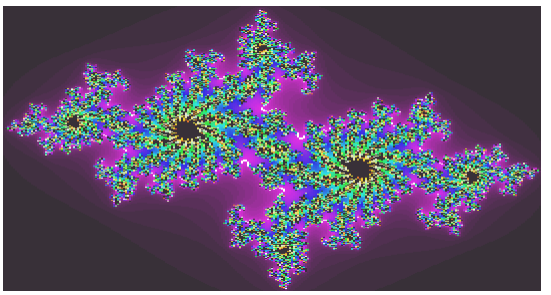
⇨ JPEG indice 99 : 0,694 Ko
⇨ 207:1



Compression Fractale

- Notion de fractalité





Compression Fractale

⇨**Concept de base :**

⇨ image décrite par un ensemble de motifs identiques en nombre limités, transformés par translations, rotations…

⇨ coder : décrire les motifs et les transformation

⇨ codage indépendant de la taille

⇨ codage long (plusieurs minutes)

⇨ Différent du JPEG : zoom = flou (non pas pixélisation)

⇨ Possibilité de zoomer

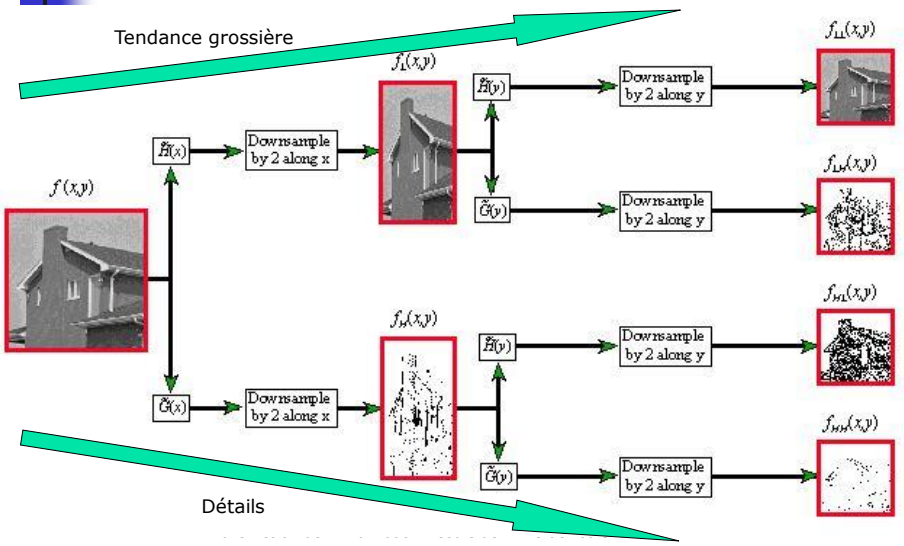
Compression par Ondelettes

• Une transformation par ondelettes consiste à décomposer un signal en une tendance grossière accompagnée de détails de plus en plus fins.

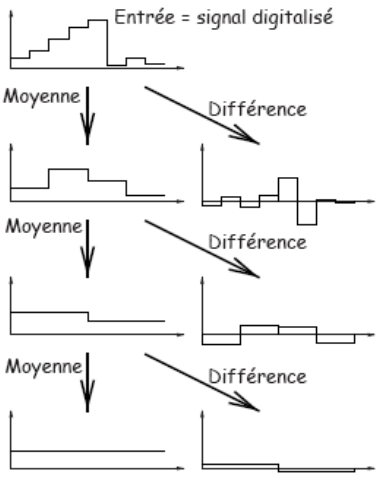
• Ainsi, pour reconstituer le signal avec une précision donnée, il suffira de connaître la tendance et les détails correspondant au niveau de précision recherché et de négliger les autres.

• La transformée par ondelettes est la décomposition d'un signal par une *ondelette mère* qui sera translatée et dilatée (idem fourier).

Compression par Ondelettes

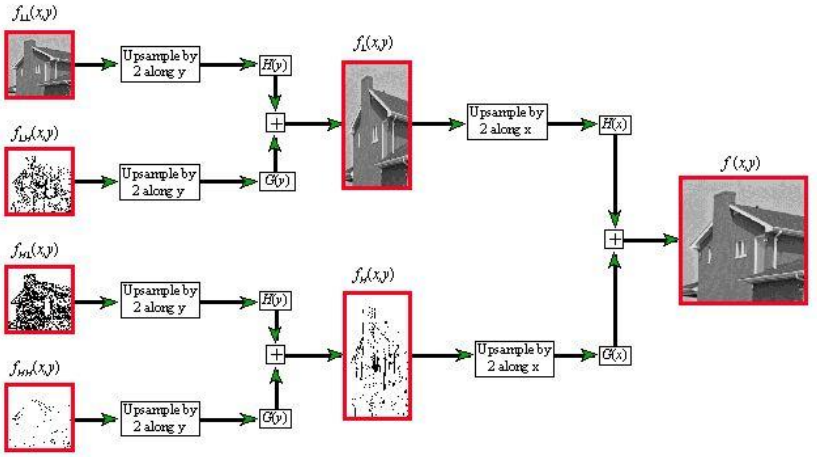


Compression par Ondelettes

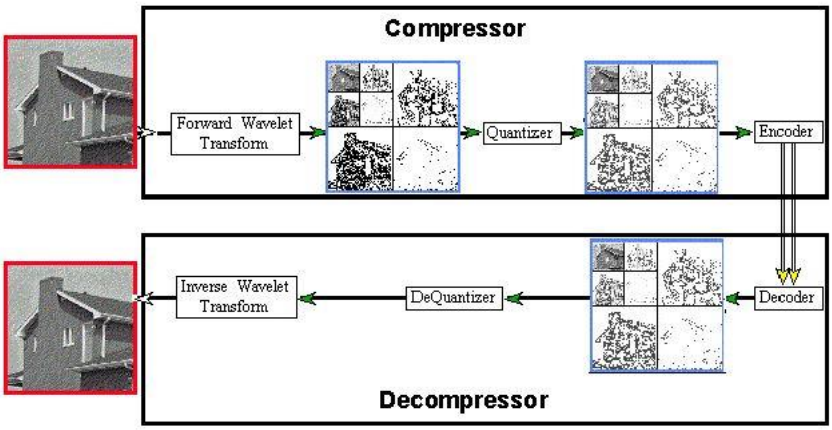


Résolution	Moyennes	Détails
8	[2 4 8 12 14 0 2 1]	
4	[3 10 7 1.5]	[-1 -2 7 0.5]
2	[6.5 4.25]	[-3.5 2.75]
1	[5.375]	[1.125]

Compression par Ondelettes



Compression par Ondelettes



Ondelettes et quantification

- ⇨ Les images d ’erreurs inférieures à un niveau donné sont éliminées
- ⇨ Compression destructive
- ⇨ Utilisation d ’une méthode non destructive pour le reste
- ⇨ En-tête plus compact
- ⇨ Avantage sur le JPEG flou pas pixelisation
- ⇨ Indice de 20 conseillé

Ondelettes et indice

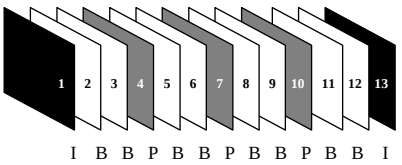
- ⇨ Original : 144 Ko
- ⇨ Indice 20 : 3,5 Ko = 41:1 : qualité excellente
- ⇨ Indice 30 : 1,7 Ko = 82:1 : flou artistique
- ⇨ Indice 50 : 805 o = 183:1 : Brume
- ⇨ Indice 92 : 171 o = 862:1

MPEG

- Pour la vidéo
- Compression dans les plans spatial et temporel
  - spatial : MJPEG (gros)
- MPEG 2
  - I,B,P
  - I toutes les 12 images en JPG
  - On compare les autres à I et on code les différences
- Découpage en blocs 16*16, on essai de trouver un bloc identique dans l 'image précédente.
- Compression dépend donc du mouvement

Compression MPEG

- La séquence vidéo est alors reconstruite avec
  - les images de référence,
  - des images prédites (P) qui sont construites par rapport à l'image I ou P précédente grâce aux vecteurs de déplacement et
  - des images interpolées (B) reconstituées à partir des images I et P suivantes et précédentes.



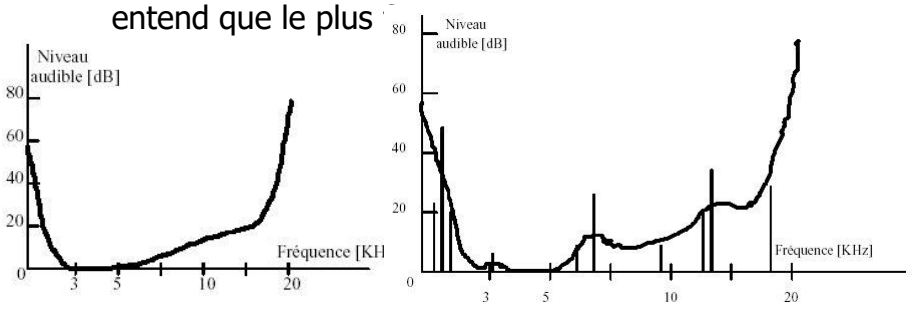
NB : MPEG4, 7 ...

Comparaison des compressions

- **Image originale 144 Ko**
- **JPEG indice 20 : 5,18 Ko : compression 28:1**
- **Ondelettes indice 20 : 3,50 Ko : compression 41:1**
- **LZW : 32 Ko : compression 4,5:1**

Compression des sons

- Echantillonnage CD 44KHz sur 16 bits
- Système auditif :
  - sensibilité => filtre passe bande
  - 2 sons simultanée avec fréquence proche => on entend que le plus



Compression des sons

- Méthode PASC (Precision Adaptative Sub band Coding):
  - Division spectrale en 32 bandes de 750 Hz
  - Calcul du niveau moyen sur chaque bande
  - Elimination de certaines bandes
  - Pour chaque bande retenue => division en sous bandes
  - codage sur nb de bits variable en fonction du nombre de sous bandes et du niveau moyen de la bande
  - Taux de compression autour de 3 ou 4

Conclusion sur le codage de source

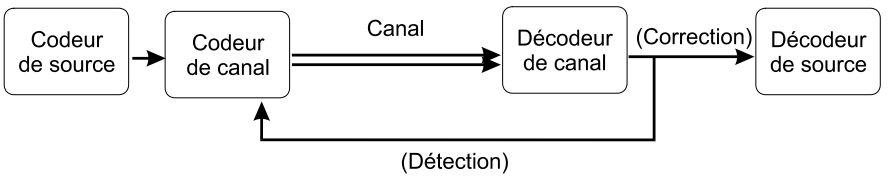
Supprime la redondance
→ Sensibilité au bruit
→ Codage de canal

Chapitre 3 :

Codage canal / Gestion des erreurs

Objectifs

- **Détecter** et/ou **corriger** les erreurs de transmission



Codeur de canal → **introduire** une **redondance** utilisable

En pratique

- **Sources d'erreurs :**
  - **le support ;**
  - **le débit ;**
  - **la modulation ;**
  - **le type de codage ;**
  - **le rapport S/N.**

- Taux d'erreur = 10^{-4} à 10^{-7} avec en plus des phénomènes de groupement d'erreurs par paquets.
- Erreur tolérée = 10^{-10} à 10^{-12} dans les réseaux locaux industriels

→ détection et correction

Théorème des canaux à perturbation

Rappel et remarques

" Pour une source à débit d'information de R bit/s et un canal de capacité C bit/s, si R<C, il existe un code ayant des mots de longueur n, de sorte que la probabilité d'erreur de décodage p_E soit minimale"

Rq2 : à p_E constant, n augmente si R tend vers C.

Rq3 : en pratique, si R<0.5 C, des codes existent avec p_E faible.

Quelques Définitions

- **Taux d'erreur** $T_e = \dfrac{Nombre\ de\ bits\ erronés}{Nombre\ de\ bits\ transmis}$

01100100100110010010100101O => 011001<u>1</u>01100101<u>1</u>0100<u>0</u>010

$T_e = 3/24 = 0.125$

- **Taux de codage / rendement**

$$R = \frac{k}{n}$$

- k taille du mot d 'information (avant codage)
- n taille du mot-code (après codage)

Quelques Définitions

- **Efficacité de la détection :**

$$E = \frac{Nombre\ messages\ reconnus\ erronés}{Nombre\ messages\ erronés}$$

- **Taux d 'erreurs brut :** $t = 1 - (1-p_e)^n$
- **Taux d 'erreurs résiduel :** $q = t.(1-E)$

Méthodes de détection d'erreurs

- Détection par écho / répétitions
- Détection par codes linéaires
  - Détection par bit de parité
  - Détection par Checksum
  - Détection par codage de Hamming
  - Détection par codes cycliques
- Détection par codes convolutifs

Méthodes de détection basiques

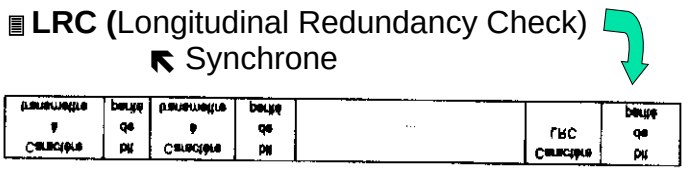
- **Echo :**
  - **Tout message émis est comparé à son écho et réémis si différent. Problème : il peut y avoir des erreurs dans l'écho, des compensations d'erreurs**, …
- **Codage par répétition**

Méthodes de détection basiques

- **Bit de parité (*checksum*)**
  - **Pour une trame donnée, t = a_1 … a_k ($a_i \in \{0,1\}$), on rajoute un bit a_{k+1} tel que le nombre de 1 dans la trame soit toujours pair.**
  - **On peut aussi effectuer des sommes sur certains des a_i. Cette technique du *checksum* est utilisée dans les entêtes des trames IP et TCP.**

Détection d'erreurs par bit de parité (caractère)

- **VRC (**Vertical Redundancy Check)
  - Asynchrone
- **LRC (**Longitudinal Redundancy Check)
  - Synchrone



Exercice →

	H	E	L	L	O	LRC →
bit 1	0	1	0	0	1	0
bit 2	0	0	0	0	1	1
bit 3	0	1	1	1	1	0
bit 4	1	0	1	1	1	0
bit 5	0	0	0	0	0	0
bit 6	0	0	0	0	0	0
bit 7	1	1	1	1	1	1
VRC ↓	0	1	1	1	1	0

0001001	0	1010001	1	0011001	1	0011001	1	1111100	1	0100001	0
H		E		L		L		O		LRC	

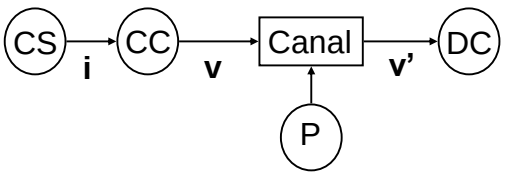
Caractéristiques des codages

- On note les codes avec C(n,k)
- k bits d'informations → n bits après codage
- k < n m = n - k

- Codes systématiques ou non :
  - Les mots codes contiennent les mots initiaux
  - Les mots initiaux ne sont pas visibles

- Autres caractéristiques d'un codage
  - Nombre d'erreurs détectables
  - Nombre d'erreurs corrigeables

Notations



• **Notations :**

• **Mot-code : v**

$$v = [a_1\ a_2\ \ldots . a_k\ a_{k+1}\ a_{k+2}\ \ldots . a_n] = [i\quad c]$$

[c] : *m* symboles de contrôle
[i] : *k =n-m* symboles d'information

• **Mot-erreur : ε**

$$\varepsilon = [\varepsilon_1\ \ \varepsilon_2\ \ldots\ldots \varepsilon_n] \qquad v_i = v'_i + \varepsilon \quad \Leftrightarrow \quad v'_i = v_i + \varepsilon$$

$$\varepsilon_i = \begin{bmatrix} 1 \text{ si erreur à la ième position} \\ 0 \text{ sinon} \end{bmatrix}$$

Codes linéaires

Définition :

Les symboles de contrôle sont obtenus par une combinaison linéaire des symboles d 'information

- **Code bloc linéaire**

  Symboles binaires et addition modulo 2 (XOR et AND)

Code et matrice génératrice

Code C(n,k) → G(k,n)

Soit $G_{(k,n)}$ la matrice génératrice, $[G] = \begin{bmatrix} g_{11} & g_{12} & \cdots & g_{1n} \\ g_{21} & g_{22} & & g_{2n} \\ \cdots & & & \cdots \\ g_{k1} & g_{k2} & \cdots & g_{kn} \end{bmatrix}$

$i = [i_1 \quad i_2 \quad \ldots \quad i_k]$

$$v = i.G$$

Codage et matrice génératrice

- Les lignes de G(k,n) sont les mots codes
  i peut, par exemple, être [1000]
- Les lignes sont linéairement indépendantes
  → base vectorielle

Décodage et matrice de contrôle

$$v = [a_1 \quad a_2 \quad \dots \quad a_n]$$

Soit $H_{(m,n)}$ la matrice de contrôle, $[H] = \begin{bmatrix} h_{11} & h_{12} & \dots & h_{1n} \\ h_{21} & h_{22} & & h_{2n} \\ \dots & & & \dots \\ h_{m1} & h_{m2} & \dots & h_{mn} \end{bmatrix}$

Soit z le syndrome (ou correcteur), $z = H.v'^{T} = \begin{bmatrix} z_1 \\ : \\ z_m \end{bmatrix}$

Si z=[0] pas d 'erreur, sinon erreur et + / - correction

Codage et matrice génératrice

Les matrices H et G sont liées par : $G.H^t = 0$

et peuvent se mettrent sous la forme systématique :

$$G = \left[\begin{array}{c:c} & \\ I_k & A_{k,m} \\ & \end{array}\right] \quad H = \left[\begin{array}{c:c} & \\ A^t{}_{k,m} & I_m \\ & \end{array}\right]$$

Tableau standard

Bloc données utiles	00	10	01	11
Mots de code	0000	1011	0101	1110
Classe 1	1000	0011	1101	0110
Classe 2	**0100**	1111	**0001**	1010
Classe 3	0010	1001	0111	1100

Classe : ensemble C(Z) = {Z + X, $\forall$ X$\in$Code}

Représentant de classe : mot de poids le plus faible

Codage et matrice

•*k* vecteurs constituant une base du code *C* sont utilisés pour former les lignes d'une matrice **G** de taille *k*×*n.*

•Tout mot de code est une combinaison linéaire des lignes de **G**.

•La correspondance entre G et H n'est pas unique.

•La correspondance entre mots d'information et mots de code n'est pas unique.

•**G** et **G'**, les matrices génératrices de deux codes équivalents sont reliées par :

- des permutation des colonnes
- des combinaisons linéaires sur les lignes.

Exemple k=2, m=1, n=3

$$[G_1]=\begin{bmatrix}0 & 1 & 1\\ 1 & 0 & 1\end{bmatrix} \quad [H]=\begin{bmatrix}1 & 1 & 1\end{bmatrix} \quad [G_2]=\begin{bmatrix}1 & 0 & 1\\ 0 & 1 & 1\end{bmatrix}$$

$$\begin{bmatrix}0 & 0\end{bmatrix}\times\begin{bmatrix}0 & 1 & 1\\ 1 & 0 & 1\end{bmatrix}=\begin{bmatrix}0 & 0 & 0\end{bmatrix} \qquad \begin{bmatrix}0 & 0\end{bmatrix}\times\begin{bmatrix}1 & 0 & 1\\ 0 & 1 & 1\end{bmatrix}=\begin{bmatrix}0 & 0 & 0\end{bmatrix}$$

$$\begin{bmatrix}0 & 1\end{bmatrix}\times\begin{bmatrix}0 & 1 & 1\\ 1 & 0 & 1\end{bmatrix}=\begin{bmatrix}1 & 0 & 1\end{bmatrix} \qquad \begin{bmatrix}0 & 1\end{bmatrix}\times\begin{bmatrix}1 & 0 & 1\\ 0 & 1 & 1\end{bmatrix}=\begin{bmatrix}0 & 1 & 1\end{bmatrix}$$

$$\begin{bmatrix}1 & 0\end{bmatrix}\times\begin{bmatrix}0 & 1 & 1\\ 1 & 0 & 1\end{bmatrix}=\begin{bmatrix}0 & 1 & 1\end{bmatrix} \qquad \begin{bmatrix}1 & 0\end{bmatrix}\times\begin{bmatrix}1 & 0 & 1\\ 0 & 1 & 1\end{bmatrix}=\begin{bmatrix}1 & 0 & 1\end{bmatrix}$$

$$\begin{bmatrix}1 & 1\end{bmatrix}\times\begin{bmatrix}0 & 1 & 1\\ 1 & 0 & 1\end{bmatrix}=\begin{bmatrix}1 & 1 & 0\end{bmatrix} \qquad \begin{bmatrix}1 & 1\end{bmatrix}\times\begin{bmatrix}1 & 0 & 1\\ 0 & 1 & 1\end{bmatrix}=\begin{bmatrix}1 & 1 & 0\end{bmatrix}$$

Codage de Hamming

- **Distance de Hamming**

$$D(v_i, v_j) = (a_{i1} \oplus a_{j1}) + (a_{i2} \oplus a_{j2}) + ... + (a_{in} \oplus a_{jn})$$

↳ Le nombre de coordonnées par lesquels les 2 mots diffèrent

Codage de Hamming → correction directe :

- C'est le récepteur qui corrige → Il faut donc un code très redondant.
- Ce mécanisme est très coûteux mais aussi très efficace. On parle alors de code auto-correcteur.
- Utilisation : Retrouver le code exact à partir d'un code erroné consiste à retrouver le plus proche voisin au sens de d_H.

Hamming et correction des erreurs

- Exemple : code d_H = 2, sur des mots de 4 bits
- Les codes autorisés sont :

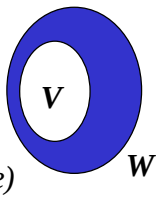
0 0 0 0	**(1)**
0 0 1 1	**(3)**
0 1 0 1	**(1)**
0 1 1 0	**(2)**
1 0 0 1	**(3)**
1 0 1 0	**(3)**
1 1 0 0	**(1)**
1 1 1 1	**(3)**

- Par exemple, soit le message suivant : 0 1 0 0. Ce n'est pas un code valide.
- Pour pouvoir corriger en plus de détecter, il faut un code de d_H > ou égale à 3 pour lequel les seuls mots autorisés sont :

0 0 0 0	**(1)**
1 1 1 1	**(3)**

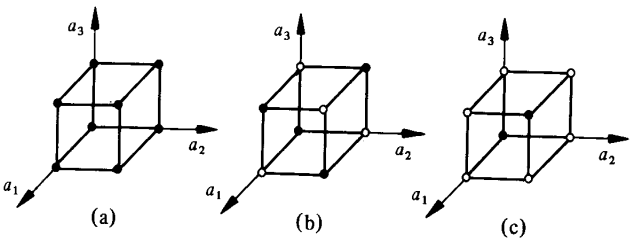
Illustration spatiale

• Un mot = un vecteur dans un espace à n dimensions !
w=[a_1 a_2 ... a_n]



• *W = ensemble des N = 2^n mots*

• *V = ensemble des S = 2^k mots ayant un sens (mot-code)*



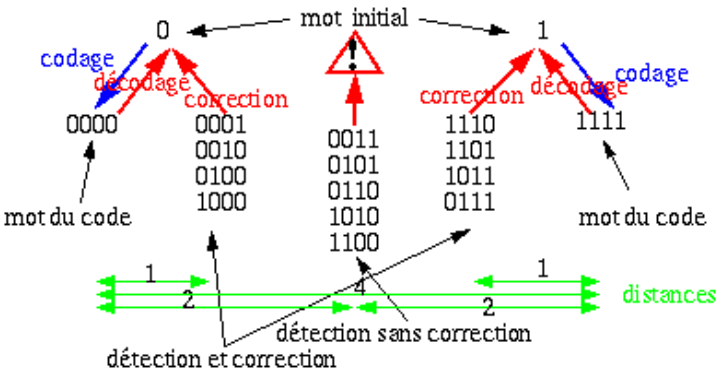
Hamming et correction

$v_i \rightarrow$ Région W_i dépendantes de d_H et disjointes

=> Détection et correction ↗ si W_i grand

- <u>Théorème de Hamming :</u>
- Si on veut :
  - **détecter p erreurs isolées alors il faut $d_H \geq p+1$.**
  - **corriger q erreurs isolées alors il faut $d_H \geq 2q+1$.**

Hamming et correction des erreurs



Poids de Hamming et distance min.

Le poids de Hamming $w(c)$ d'un mot de code c est égal au nombre de composantes non nulles de c.

Le poids minimal w^* d'un code C est le minimum des poids $w(c)$ des mots de code c non nuls.

La distance minimale d_{min} d'un code linéaire est égale à son poids minimal w^*

Correction des erreurs et mots codes

- On veut détecter et corriger toutes les erreurs de 1 bits
  - *Dans un mot code de n = m + k bits*
    - *Avec k bits de données*
    - *Avec m bits de validation*
  - *Chaque 2^k mot code est à distance d = 1 de n mots invalides*
  - *Pour chaque mot valide, on a donc besoin de n+1 mots (n inutilisés + 1 mot code)*
    - *On veut trouver m tel que $(n+1)2^k <= 2^n$, sachant que n = m + k*
    - *Limite théorique : le plus petit m tel que $(m+k+1) <= 2^m$*

Codage de Hamming : Principe

Codage de Hamming : Principe

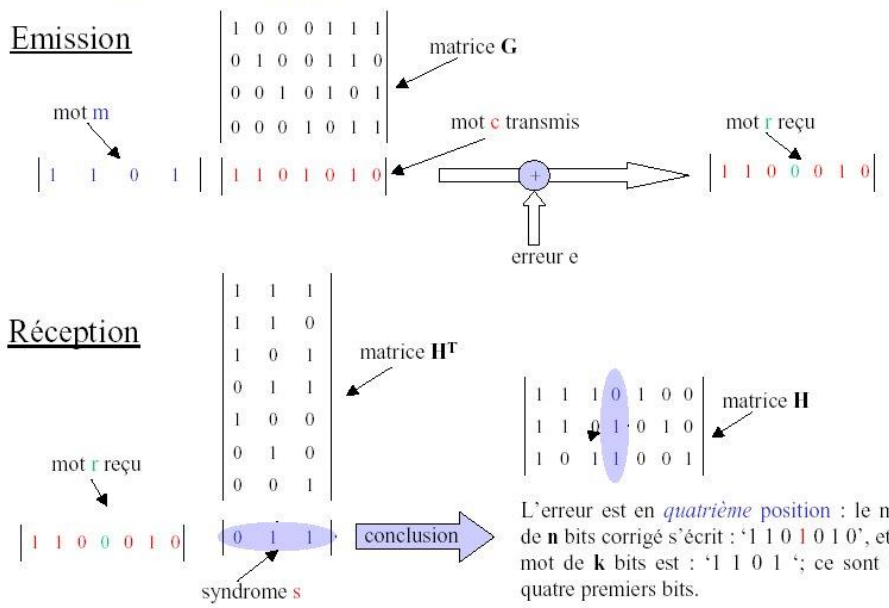
- Chaque bit est vérifié par un sous-ensemble distinct des bits de validation
- Une erreur sur un bit provoque une erreur de parité pour chaque bit de validation du sous-ensemble correspondant

- Conséquences → H doit vérifier :
  - Chaque colonne est la représentation binaire des nombres 1 à n.
  - Avec $n = 2^m - 1$ et $k = n - m = 2^m - 1 - m$

Code de Hamming : Principe

- Auto-correction :
  - le syndrome du mot reçu est identique à la colonne de la matrice de contrôle correspondant au bit à corriger.
  - si l'on trie les colonnes de H suivant leur poids binaire croissant et que les poids de ses colonnes couvrent l'intervalle [1, 2^m-1] alors la valeur binaire du syndrome est égale au numéro de bit erroné.

Exemple : mot m à transmettre '1 1 0 1'



Syndromes et tri des colonnes de H

Pour la correction d'une erreur

Si on a :

- $[H] = [h_1 \quad h_2 \quad \ldots \quad h_n] = \begin{bmatrix} 0 & 0 & \ldots & \\ : & : & : & \ldots \\ 0 & 1 & 1 & \ldots \\ 1 & 0 & 1 & \ldots \end{bmatrix}$ avec $h_i = bin(i)$
- Mot-erreur : $\varepsilon = [\ldots. \alpha_i \ldots.]$

$$v'_j = v_j + \varepsilon \Leftrightarrow z = H.v'_j = H.\varepsilon^T \Leftrightarrow z = h_i$$

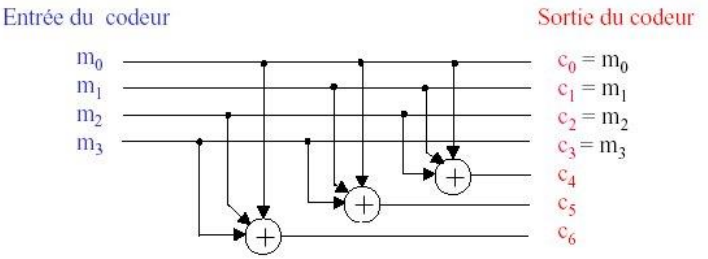
⇘ L'erreur est à la position $dec(h_i)$

Circuit de codage

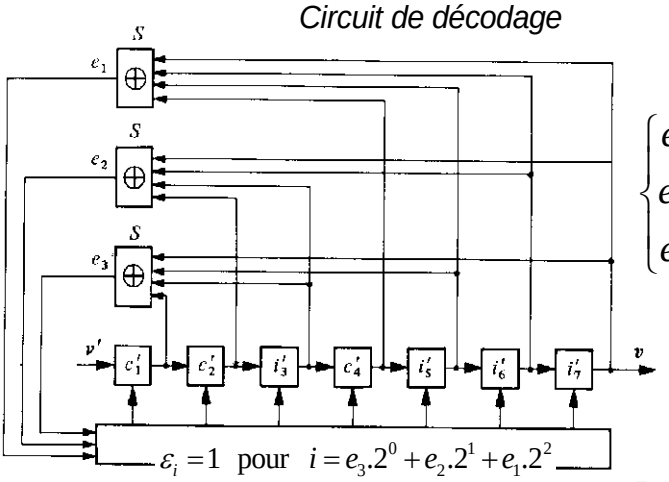
$$\mathbf{H} = \begin{vmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 \end{vmatrix}$$

Valeurs décimales : 7 6 5 3 4 2 1

La matrice génératrice **G** s'écrit ($\mathbf{GH^T = 0}$) : $\mathbf{G} = \begin{vmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{vmatrix}$



Circuit de décodage



$$\begin{cases} e_3 = c'_1 + i'_3 + i'_5 + i'_7 \\ e_2 = c'_2 + i'_3 + i'_6 + i'_7 \\ e_1 = c'_4 + i'_5 + i'_6 + i'_7 \end{cases}$$

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

Exercice sur le codage de Hamming

Codes polynomiaux

• Définition : Un code polynômial est un code linéaire systématique dont chacun des mots du code est un multiple du polynôme générateur (noté g(x)).

• Le degré du polynôme générateur définit la longueur du champ de contrôle d'erreur.

Codes polynomiaux

• Tout vecteur peut être présenté sous une forme polynômiale

• Les opérations (+, X, /) sont binaires : 1.x + 1.x = 0.x !

- Soit t= a_{n-1} … a_0. On peut lui associer un polynôme de degré n-1
- par : $P(x) = a_{n-1}x^{n-1} + \ldots + a_2 x^2 + a_1 x + a_0$
- Exemple : le mot 1001101 est associé à $P(x) = x^6 + x^3 + x^2 + 1$

Codage / Décodage

- **Division / Multiplication (en binaires)**

$a(x) = x^3 + x^2 + x$ et $b(x) = x^3 + x + 1 \rightarrow \quad c(x) = x^6 + x^5 + x$

$\boxed{c(x) = a(x) \times b(x)}$ $\quad [1110] \times [1011] = [1100010]$

- **Codage par division**

$v(x) = c(x) + x^m . i(x)$

Systématique !

$$\boxed{c(x) = Reste\left(\frac{x^m . i(x)}{g(x)}\right)}$$

- **Décodage par division**

Si *z(x)=0* => Transmission OK

Sinon => Détection ou correction

$$\boxed{z(x) = Reste\left(\frac{v'(x)}{g(x)}\right)}$$

Codage / décodage par division en détails

- Soit une clef G(x), polynôme de degré v. On pose :
- x^v P(x) = Q(x) G(x) + R(x)
- où Q(x) et R(x) sont deux polynômes de degré au plus égal à, respectivement, n-1 pour Q et v pour R.
- On travaille en binaire **donc** x^v P(x) + R(x) = Q(x) G(x) = Y(x)

Codage / décodage par division en détails

1. l'émetteur transmet les mots associés aux polynômes P(x) et R(x) (généralement par simple concaténation).
2. Le décodeur peut fabriquer Y(x) et la division puisque G est normalisé (donc v et G sont connus).
3. Si le reste est non nul (*i.e.* Y(x) ≠ G(x) Q(x)) alors il y a au moins une erreur.

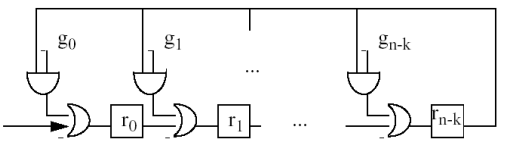
- On utilise pas n'importe quel polynôme générateur. Il est le plus souvent normalisé pour un protocole donné.
- Pour v = 16, le CCITT préconise dans l'avis V24, le polynôme :

$$G(x) = x^{16} + x^{12} + x^5 + 1$$

- L'implémentation Hardware de cet algorithme est facile.

Diviseur électronique

$$g(x) = g_0 + g_1.x + g_2.x^2 + \ldots + g_{n-k}.x^{n-k}$$



- **La multiplication est réalisée par un *ET logique*, l'addition par un *OU exclusif*, plus des registres à décalage.**
- **Procédé :**
  - **(i) les registres ri sont mis à zéros**
  - **(ii) les bits du mot à diviser sont insérés en entrée (k étapes), bits de poids fort en tête.**
  - **(iii) les registres ri contiennent alors le reste, qu'on extrait (n-k étapes).**
- **De nombreuses optimisations sont possibles :**
  - **- Lorsque gi=0 on supprime simplement la connexion et la porte ET !**
  - **- phase spécifique d'initialisation, etc**

Codes cycliques

Code cyclique = linéaire + propriété de permutation circulaire

Définition :

- toute permutation de tout mot code donne un autre mot code

- les polynômes associés aux mots codes sont tous multiples du polynome $(1+x^n)$

- Exemple :
  - Un code cyclique (1, 2) possède les mots de code suivants : {01, 10} ou {00, 11}, mais pas {01,11}.
  - Un code cyclique (1, 3) possède les mots de code suivants : {000, 111}.

Codes cycliques

- g(x) définit le codeur (n,k)
- g(x) est de degré m=n-k
- Il vérifie : $1+x^n = g(x)\times p(x)$

$$\mathbf{g(x) = 1 + g_1.x + g_2.x^2 + \ldots + g_{n-k}.x^{n-k}}$$

Exemple : code cyclique (n=7, k=4) :

$$1+x^7 = (1+x)\times(1+x^2+x^3)\times(1+x+x^3)$$

g(x) est de degré 3 soit :

$$g(x) = (1+x^2+x^3) \quad \text{ou} \quad g(x) = (1+x+x^3)$$

Bilan sur les codes cycliques

- La théorie des codes cycliques permet de montrer que ce type de code permet de détecter jusqu 'à k erreurs pour G(x) de degré k
- Pour CRC 16 :

  1) toutes les erreurs simples, doubles et triples

  2) toutes les salves d'erreurs de longueur impaire ou de moins de 17 bits

  3) 99,998% des salves d erreurs de plus de 16 bits.
- On peut ainsi obtenir un taux d'erreur résiduel de 10^{-10}.

Code cycliques BCH et RS

- Ce sont une extension des codes cycliques, ils sont non pas construit sur un alphabet binaire mais un alphabet composé d'un grand ensemble de symboles.
- Les codes BCH (Bose-Chaudhuri-Hocquenghem) sont ceux qui ont la plus grande capacité de correction d'erreurs indépendantes pour une redondance et une longueur données.
- Les codes RS (Reed-Solomon) sont des codes correcteurs très puissants. Ils peuvent être présentés comme des codes BCH dans lequel chaque bit des mots du code est remplacé par un entier.

Exemple de polynômes générateurs

ATM

$x^8 + x^2 + x + 1$ => Cellule ATM

$x^{10} + x^9 + x^5 + x^4 + x + 1$ => Couche AAL type 3/4

CCITT N°41 => X25 (HDLC)

$x^{16} + x^{12} + x^5 + 1$

IEEE 802 => Réseaux locaux

$x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + 1$

k peut varier de 200 à 3000 bits

Codes continus / convolutifs

- **Généralités**

=> Les symboles d'information sont traités en flux continu

- Rque :Blocs de n_0 symboles, mais dont les m_0 contrôleurs ne dépendent pas que des k_0 symboles d'information !

- Taux d'émission ou rendement : $R = \frac{k_0}{n_0}$

Codes convolutifs systématiques

- Mot-code : $V = \left[X_1 Y_1 X_2 Y_2 \ldots\ldots X_j Y_j \ldots\ldots_1\right]$

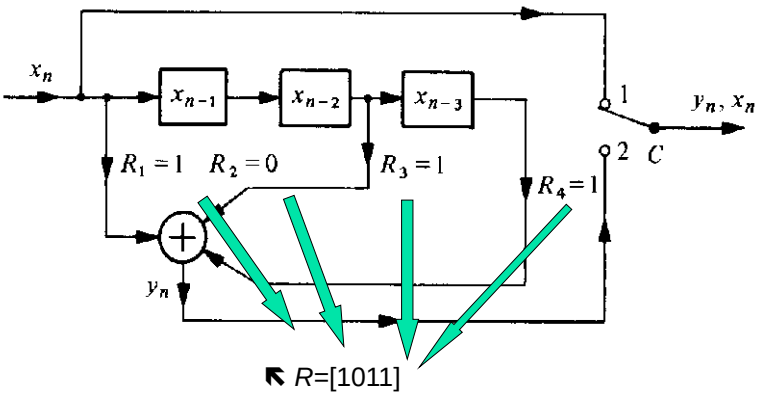
avec $X_j = \left[X_j^1 \ldots\ldots X_j^{k_0}\right]$ Information

$Y_j = \left[Y_j^1 \ldots\ldots Y_j^{m_0}\right]$ Contrôle

- **Codes convolutifs non systématiques**
  - Contrôle et information sont mélangés
    - Mot-code : $V = \left[U_1 U_2 \ldots\ldots U_j \ldots\ldots\right]$

Exemple : m=4, k_0=1, m_0=1, n_0=2

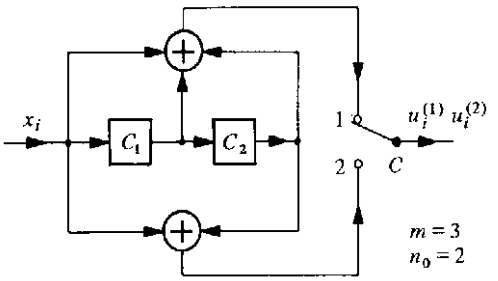
$$y_n = R_4.x_{n-3} + R_3.x_{n-2} + R_2.x_{n-1} + R_1.x_n$$



↖ *R*=[1011]

Exemple : n_0=2, R=0.5 , m=3

$$U_n^{(1)} = x_n + x_{n-1} + x_{n-2}$$
$$U_n^{(2)} = x_n + x_{n-2}$$

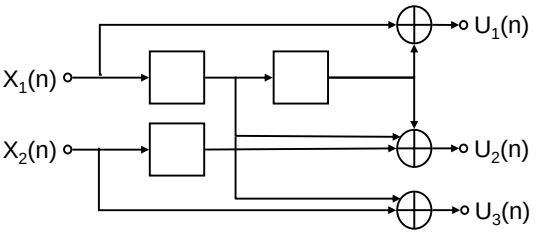


t_i	1	2	3	4	5	6	7	8
x_i	1	1	1	0	1	0	0	0
$C_1 C_2$	00	10	11	11	01	10	01	00
$u_i^{(1)} u_i^{(2)}$	11	01	10	01	00	10	11	00

191

Représentation des codes convolutifs

- Par le codeur

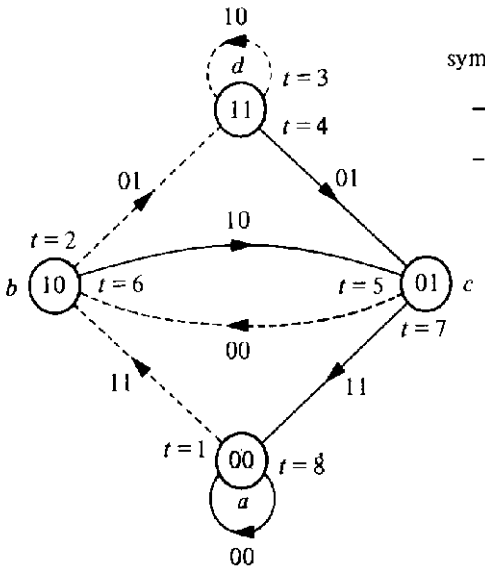


- Par une matrice de transfert

$$G_1 = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 5 \\ 0 \end{bmatrix} \qquad G_2 = \begin{bmatrix} 0 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 3 \\ 2 \end{bmatrix} \qquad G_3 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 2 \\ 4 \end{bmatrix}$$

$$G = \begin{bmatrix} 5 & 3 & 2 \\ 0 & 2 & 4 \end{bmatrix}$$

- Un diagramme d'état
- Un treillis → chemin → décodage par chemin le + probable

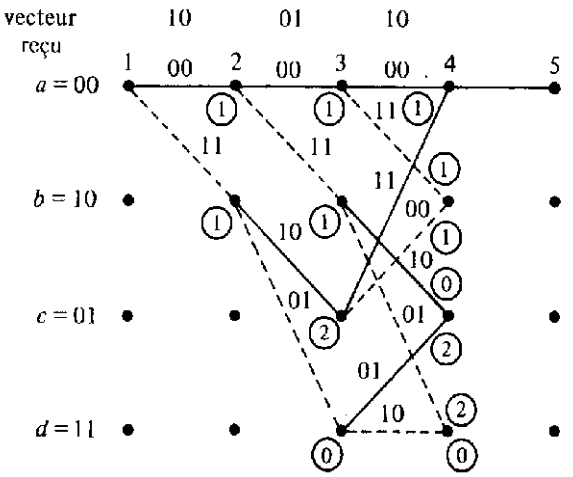


Décodage : algorithme de Viterbi

Stratégie de recherche de D_{min}

10 01 10 => $Min\left(\sum_{i=1}^{3} d_i\right) = ?$

=> 11 01 10



Conclusion sur le codage de canal

Indispensable

Théories mathématiques complexes → des solutions concrètes

- Reed-Salomon (1984) : BCH
- Turbo-Codes (1993) : Code convolutif

Recherche de codeurs conjoint source / canal

- complexité --
- robustesse ++
- flexibilité ++

Conclusion sur le codage de canal

- **Bilan sur la correction des Erreurs** :
- Pour les codes cycliques (de taille de clef = v) on peut détecter les paquets d'erreurs de taille ≤ v ou corriger les paquets de taille ≤ .
- La performance de ces codes est cependant contrebalancée par un coût important en longueur de code ce qui explique qu'ils sont très peu utilisés dans les réseaux locaux industriels.

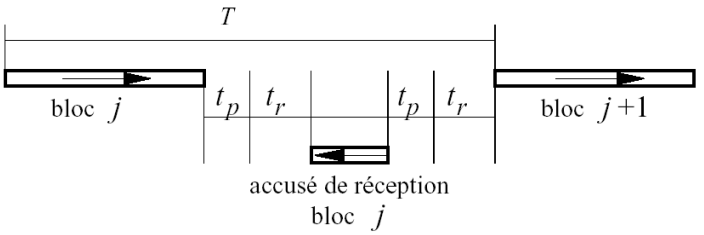
==> Correction par retransmission

Correction par retransmission

- Quand ?
  - Correction de l 'erreurs impossible
- Types de retransmission :
  - send and wait
  - envoi avec arrêt et attente d 'ACK ou NACK ou echo ou time-out
  - Nécessité d 'identification des messages

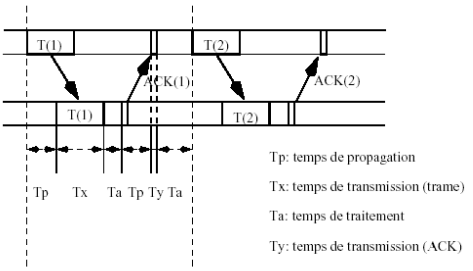
Correction par retransmission

$T_{trans} = (N_k + N_m + N_{ack})/D + 2.(t_p + t_r)$



- Rendement très faible

Correction par retransmission



Utilisation: $U = \frac{Tx}{Tt}$ $Tt = Tx + Ty + 2Tp + 2Ta$

Exemple: Ligne satellite à 56kbps et trames de 1000 bits. Le satellite est stationné à 33.000km sur la terre et la vitesse de propagation $3 \cdot 10^8$ m/sec. L'utilisation $U = \frac{T_x}{T_t}$ est de 3.4% seulement en utilisant un protocole 'envoyer et attendre' pour envoyer la trame et renvoyer l'acquittement.

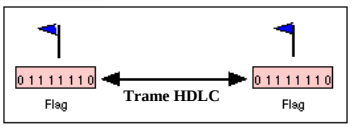
Correction par retransmission

- Transmission continue : pas d 'attente de l 'ACK
  - renvoi total a partir de l 'erreur
  - fenêtre d 'anticipation
  - Rendement moyen
- Retransmission sélective :
  - Mise en œuvre complexe
  - bon rendement
  - Satellites, TCP

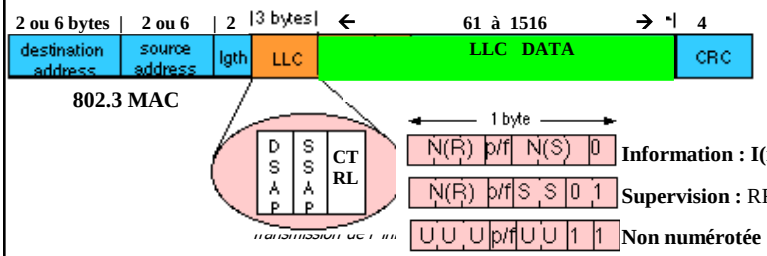
==> Voir cours de Réseaux

Gestion de la communication : Norme HDLC

- La norme HDLC (High level Data Link Control) fournit un service de transmission synchrone transparent de niveau 2.
- Service Sans ou Avec connexion :
- **Délimiteurs de trames**



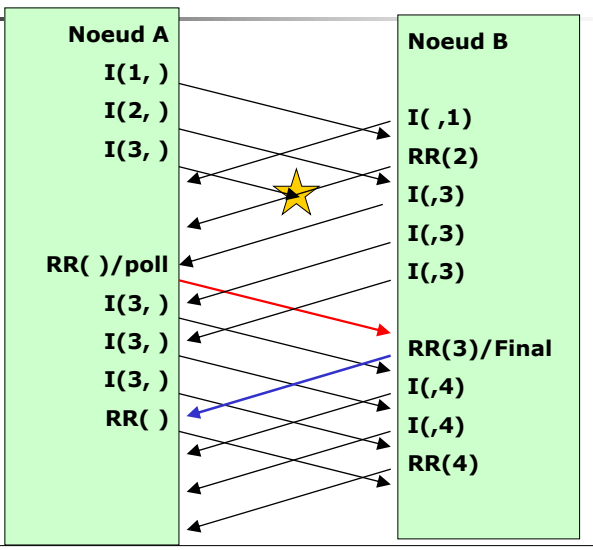
- **La trame HDLC :**



N(S) : Numéro de trame émise

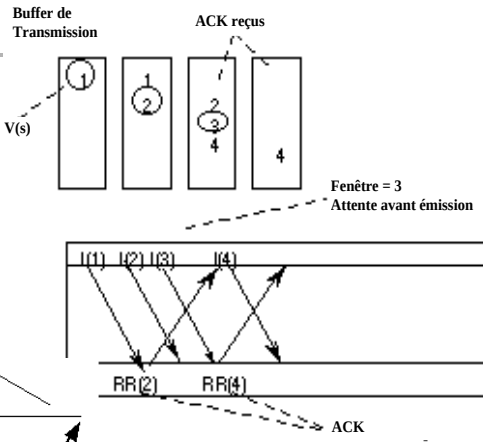
N(R) : Numéro de trame attendue

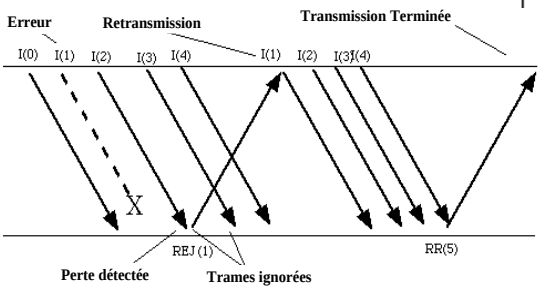
Norme HDLC

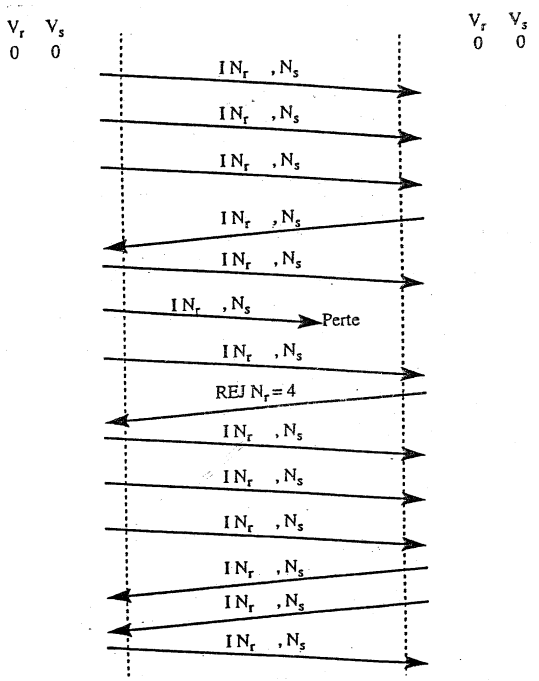


Norme HDLC

- **Exemples de fctnt (rejet, …)**
- **Fenêtre d'anticipation:** Nombre maximum de trames pouvant être envoyées sans attendre d'ACK.
- **Notion de Timer :** délais préétabli au bout duquel on réémet la trame







- Fin
- Suite = annexes = diapos supprimées

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

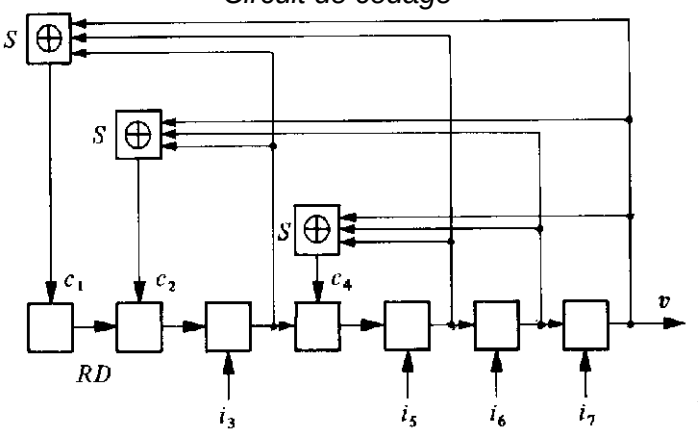
$$v = \begin{bmatrix} c_1 & c_2 & i_3 & c_4 & i_5 & i_6 & i_7 \end{bmatrix}$$

$$H.v^T = 0$$

$$\Downarrow$$

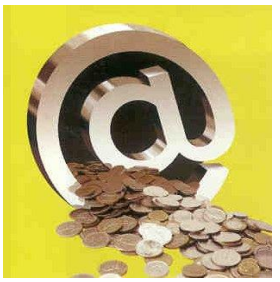
$$\begin{cases} c_1 = i_3 + i_5 + i_7 \\ c_2 = i_3 + i_6 + i_7 \\ c_4 = i_5 + i_6 + i_7 \end{cases}$$

Circuit de codage



Chapitre 4 :

Cryptographie



Introduction : Sécurité dans les SI & Réseaux

- Risques et Menaces :
  - **vulnérabilité : degré d'exposition à des dangers**
  - **sensibilité : caractère stratégique d'un élément**
  - **menaces passives : écoute des informations**
  - **menaces actives : modification de l'intégrité des données**

Introduction : Sécurité dans les SI & Réseaux

- Des garanties doivent être fournies :
  - authentification
  - contrôle d'accès
  - confidentialité des données
  - intégrité des données
  - non répudiation

→ login + mot de passe
→ droits d'accés par utilisateur
→ signature de messages
→ chiffrement de message
→ biométrie

Cryptographie

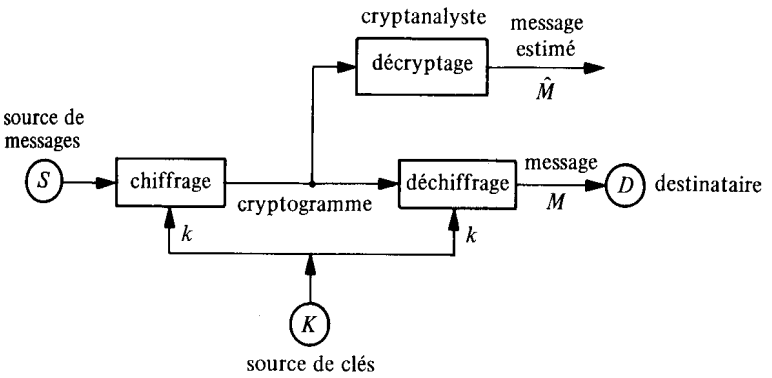
- **Objectifs**

Garantir la **confidentialité** des données
Garantir l'**intégrité** des données
Garantir l'**identité** des correspondants
→ Non répudiation des transactions

- **Applications**

Militaires
Biométrie
Sécurité réseaux et données
Téléphonie
Commerce électronique
…

Vocabulaire



Cryptographie : techniques de chiffrage

Cryptologie : cryptographie & cryptanalyse

Vocabulaire

Cryptanalyse

⟹ méthodes de pénétration d'un cryptosystème

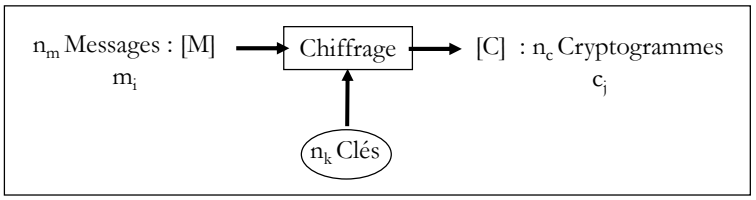
Cryptosystème vulnérable

⟹ à partir du texte crypté, il est possible de dériver

- le texte initial
- la clé

Vue de la théorie de l'information

Chiffrage = Canal très perturbé



Secret parfait ssi : $I(M;C) = 0$

Méthodes d'attaques d'un cryptosystème

Texte crypté
- détermination de la clé à partir du texte crypté
- Analyse statistique, connaissance de la langue.

Texte initial (imposé)
- le cryptanalyste connaît quelques paires (texte initial, texte crypté).

Texte initial choisi
- le cryptanalyste est capable d'acquérir le texte crypté d'un texte initial qu'il choisit.

Remarque

Un cryptosystème est inconditionnellement sûr si quelle que soit la façon dont un message chiffré est intercepté, il n'y a pas suffisamment d'informations dans le texte crypté pour déterminer de façon non ambiguë le texte initial.

Chiffrage efficace

ssi

(Coût + temps) de décryptage >> Valeur de l'info

Ordre de grandeur :

Tester 2^{128} possibilités

= 10^9 fois l 'âge de l 'univers sur la base

d ' 1 million de tests par seconde.

Taille actuelle des clés = 1024 bits voire plus…

Les grandes approches

- **Approches classiques** (concepts de base)
  - **Chiffrage par substitution**
    Jules César, l'Abbé Trithème
  - **Chiffrage par transposition**

- **Approches actuelles**
  - **Chiffrage à clé privée (symétrique)**
    DES, IDEA, AES
  - **Chiffrage à clé publique (asymétrique)**
    RSA, PGP

Chiffrage par substitution → S_Box

↖ Chaque lettre (ou groupe de lettres) est remplacée par une lettre (ou un groupe de lettres)

- Abbé Trithème (1499)

Dans son royaume à perpétuité,
En Paradis à perpétuité,
Ainsi qu'en toute éternité;
Dans la gloire à perpétuité,
Mais dans son règne;
Sempiternel, toujours dans la félicité,
Tant dans la lumière que dans la béatitude,
Et toujours dans la gloire à perpétuité,
Mais dans son règne;
En une infinité encore à perpétuité,
Comme dans la gloire autant que dans les Cieux,
A tout jamais, oui ! à tout jamais à perpétuité;
Dans son royaume et dans la félicité,
Irrévocablement, dans son royaume,
Et sans cesse qu'il soit à perpétuité dans la lumière,
Et encore à perpétuité !

A = dans les cieux
B = à tout jamais
C = un monde sans fin
D = en une infinité
E = à perpétuité
F = sempiternel
G = durable
H = sans cesse
I-J = irrévocablement
K = éternellement
L = dans la gloire
M = dans la lumière
N = en paradis
O = toujours
P = dans la divinité
Q = dans la déité
R = dans la félicité
S = dans son règne
T = dans son royaume
U-V-W = dans la béatitude
X = dans la magnificience
Y = au trône
Z = en toute éternité

219

Chiffrage par permutation / transposition (→ P_Box)

- on découpe le texte en blocs de m lettres

- sur chacun des blocs, on applique une permutation Π de m éléments

Exemple : m = 6

$$K \rightarrow \Pi \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 3 & 5 & 1 & 6 & 4 & 2 \end{pmatrix}$$

$$K \rightarrow \Pi^{-1} \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 3 & 6 & 1 & 5 & 2 & 4 \end{pmatrix}$$

Chiffrage par permutation / transposition

La 1ère lettre du bloc devient la 3ème, la 2ème devient la 5ème,……..

Exercice : crypter le message suivant :

LA CRYPTOGRAPHIE PREND
DE PLUS EN PLUS D 'IMPORTANCE

Intérêt de cette méthode ? Comment décrypter ?

Chiffrage par transposition

↖ Change l'ordre des lettres sans les substituer

- Exemple

B	R	I	Q	U	E	S
1	5	3	4	7	2	6
t	r	a	n	s	f	é
r	e	z	u	n	m	i
l	l	i	a	r	d	d
e	f	r	a	n	c	s
à	m	o	n	c	o	m
p	t	e	s	u	i	s
s	e	n	u	m	é	r
o	t	é	z	é	r	o
z	é	r	o	s	e	p
t	a	b	c	d	e	f

texte en clair

tranférezunmilliarddefrancsàmon
comptesuissenumérotézérozérosept

texte chiffré

TRLEAPSOZTFMDCOIEREEAZIROENERB
NUAANSUZOCRELFMTETEAEIDSMSROPF
SNRNCUMESD

Chiffrage de VIGENERE

La clé : une séquence de n caractères

$x_1 \ldots\ldots\ldots x_n$

Le texte découpé en bloc de n caractères :

$m_1\ m_2 \ldots\ldots\ldots\ldots\ m_n\ m_{n+1}$

Cryptage

$C_1 = m_1 + x_1$ mod(26)

$C_i = m_i + x_i$ mod(26)

- - - - - - - - - - - - -

$C_{n+1} = m_{n+1} + x_1$ mod(26)

Intérêt ? Mise en œuvre ?

Codage de VERNAM (« masque jetable »)

Analogue à Vigenère

mais **la clé a la même longueur que le texte**

Et souvent utilisée qu ’une seule fois (**one-time pad**)

$K = k_1 \ldots\ldots\ldots\ldots\ldots\ldots\ k_n$

$M = m_1 \ldots\ldots\ldots\ldots\ldots\ldots\ m_n$

Codage de VERNAM

$C_i = (m_i \oplus k_i)$

Utilisation

$C_i = m_i \oplus k_i$

$k_i \oplus k_i = 0$ pour $k_i = 0$ ou 1

$C_i \oplus k_i = m_i \oplus k_i \oplus k_i$

$= m_i$

ki = séquence

$\oplus$ = ou exclusif

Exemple : M = 1 1 0 0 0

K = 1 0 0 1 0

$E_k(M)$ = 0 1 0 1 0

Problème ?

Ancien mot de passe Unix, aujourd’hui DES

Transposition à base de matrices (Mix_Culumn)

- clé = une matrice + 1 séquence de lecture

- le message en clair est écrit dans la matrice

- encryptage : lecture de la matrice en respectant la séquence définit par la clé.

Exemple

clé = (6,5) , lecture par colonne

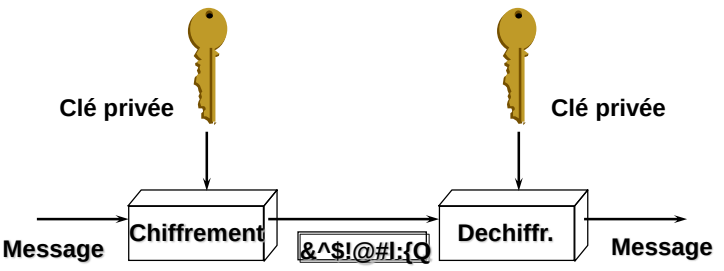
C	E	C	I	
E	S	T		U
N		M	E	S
S	A	G	E	

D(M) = CENS ES A CTMG I EE US .

Méthodes actuelles

- Bilan méthodes classiques
  - Adaptées aux textes
  - Pas rapide , mise en œuvre difficile
  - Attaquables par méthodes statistiques
  - Diffusion difficile Algo ←?→ clé
- Méthodes actuelles = Inspirées des méthodes classiques
  - **Chiffrement symétrique → 1 seule clé privée**
  - **Chiffrement asymétrique → 2 clés différentes (privée / publique)**

Chiffrage à clé privée (symétrique)

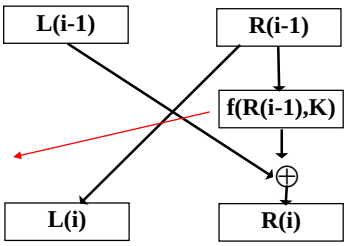


- Chiffrement et déchiffrement avec même clé
- Chiffrement et déchiffrement avec fonction mathématique
- Rapide
- Exemple: Data Encryption Standard (DES, IDEA ,AES, ...)

DES, *Data Encryption Standard*

Proposé par IBM en 1977

- Soient L_i et R_i les deux 1/2 blocs gauche et droit d'un mot.
- On répète cette opération pour i=1 (mot initial) à i=16.
- La complexité du cryptage repose sur le choix de la fonction f.
- Le code DES propose la fonction non linéaire suivante, f(R,K) :
- 1) R ➔ R' : R' est un mot de 48 bits obtenu à partir des 32 bits de R dont 16 sont dupliqués.
- 2) R' ⊕ K ➔ R" (addition modulo 2)
  -) R" ➔ R"' : on revient à 32 bits par codage des paquets de 6 bits en 4 bits grâce à une table.
- Ce système présente 2^{56} = 7,2 10^{16} clefs possibles mais il suffit de 18 caractères "clairs" pour le décrypter (mais c'est long).



• Mot de 64 bits • Clé de 56 bits • 16 rondes

231

DES

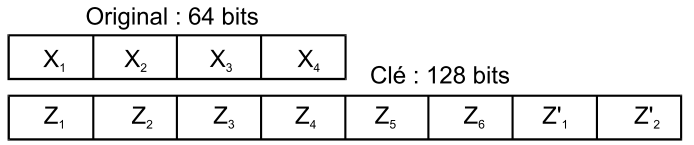
- Algorithme de déchiffrage similaire (inverse) au chiffrage
- Avec la même clé
- Cryptanalyse a montré les limites de DES :
  - Nombreux textes en clair et correspondants codés ==> décryptage
  - Pour longueur clé < 768 bits !
- Triple DES, RC2, RC5, RC6 (S_Box + P_Box)

La **clé** à échanger est à garder **secrète**

IDEA (International Data Encryption Algorithm / *Lai, Massey* 1991)

Une succession d'addition (+) , multiplication (x), et Xor (⊕)

- Mot de 64 bits
- Clé de 128 bits
- 8 rondes



- $X_1 \times Z_1 = Y_1$
- $X_2 + Z_2 = Y_2$
- $X_3 + Z_3 = Y_3$
- $X_4 \times Z_4 = Y_4$
- $Y_1 \oplus Y_3 = Y_5$
- $Y_2 \oplus Y_4 = Y_6$
- $Y_2 \times Z_5 = Y_7$
- $Y_6 + Y_7 = Y_8$
- $Y_8 \times Z_6 = Y_9$
- $Y_7 + Y_9 = Y_{10}$
- $Y_1 \oplus Y_9 = X_1'$
- $Y_3 \oplus Y_9 = X_3'$
- $Y_2 \oplus Y_{10} = X_2'$
- $Y_4 \oplus Y_{10} = X_4'$

- $X_1 \times Z_1 = X_1'$
- $X_2 + Z_2 = X_2'$
- $X_3 + Z_3 = X_3'$
- $X_4 \times Z_4 = X_4'$

DES / IDEA / AES

	Taille des blocs	Taille de clé	Nombre de rondes
DES	64	56	16
IDEA	64	128	8

- IDEA est deux fois plus rapide !
- Chip VLSI IDEA → 200 Mb/s
- IDEA le remplaçant de DES ? Et non !

❑ AES ou Rijndael normalisé en 2000 (nouvelle version de DES) :
- Longueur de clé variable 128, 192 ou 256 bits - Longueur de bloc = **128** bits
- S_box + P_box + Shift_Box + Mix_Culumn

Architecture Kerberos

- Exemple d'architecture sécurisée à base d'un chiffrement symétrique

- Protocole **Kerberos** est issu du projet « Athena » du MIT
- Version 5 du protocole Kerberos normalisée par l'IETF dans les RFC 1510 (septembre 1993) et 1964 (juin 1996).
- Basé sur DES. Kerberos partage avec chaque client du réseau une clé secrète faisant office de preuve d'identité
- Serveurs d'authentification (*AS*), permettant d'identifier des utilisateurs distants
- Serveurs de délivrement de tickets de service (*TGS*, pour *Ticket Granting System*)
- Clients = utilisateurs ou machines

Architecture Kerberos

- Les services de sécurité sont regroupés sur un même serveur, appelé **Centre de Distribution des Clés** (ou *KDC*, pour *Key Distribution Center*)

- Afin d'obtenir l'autorisation d'accès à un service, un client doit envoyer son identifiant au serveur d'authentification.

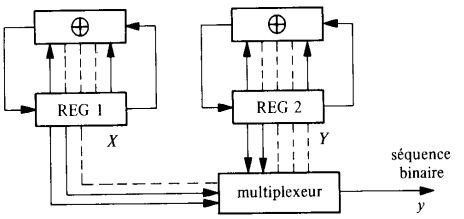
- Le serveur d'authentification vérifie que l'identifiant existe et envoie un ticket initial au client distant, chiffré avec la clé associée au client. Le ticket initial contient :
  - **un clé de session, faisant office de mot de passe temporaire pour chiffrer les communications suivantes**
  - **un ticket d'accès au service de délivrement de ticket.**

- L'authentification a une durée limitée dans le temps, on parle ainsi d'anti re-jeu.

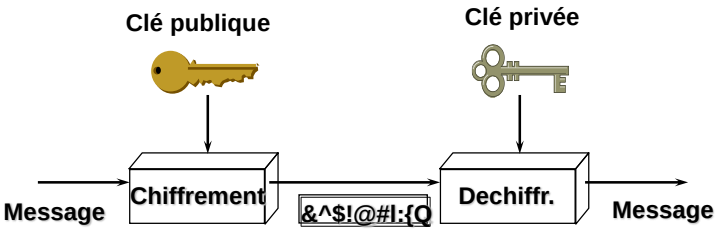
Méthodes symétriques : Challenges

- Longueur des clés
- Fréquence de renouvellement
- Création et Distribution sécurisées (HSM, …)

• Generateur de clé aléatoire



Chiffrage à clé publique



- Chiffrement et déchiffrement avec clés différentes
- Chiffrement et déchiffrement avec différentes fonctions mathématiques
- Lent
- 2 exemples principaux : RSA, Diffie-Hellman, ...

RSA (Rivest Shamir Adleman / 1978)

- **Ce système utilise la décomposition en facteurs premiers de grand nombre.**
- **Si A et B sont premiers alors la décomposition de K = AB est unique.**
- **Connaissant K, il est très dur de trouver A et B si K est très grand (si K > 10^{200} alors il faudrait plus de 10^6 années de calcul d'un gros ordinateur).**
- **Soit n un nombre calculé par produit de nombre premiers p et q. On appelle indicateur d'Euler de n, et on note Φ(n), le nombre d'entiers premiers avec n (= qui n'interviennent pas dans sa décomposition).**
- **Si n = p q (p et q premiers) alors on peut montrer que :**

$\Phi(n) = (p-1)(q-1)$ et $m^{\Phi(n)} = 1$ modulo n $\forall m$

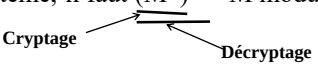
d'où :

$m^{k\Phi(n)+1} = m$ modulo n

- **Ex :** n= 3x5 = 15 → Φ(n)= 8 → {1;2;4;7;8;11;13;14;}

RSA (Rivest Shamir Adleman / 1978)

- Pour avoir un crypto-système, il faut $(M^A)^B = M$ modulo n .



- C'est à dire : A.B = k.Φ(n) +1 <=> A.B modulo Φ(n) = 1
- **Cryptage :** On choisit p,q. On en déduit n. On choisit 1 nombre premier A dans l'intervalle [3,Φ(n)] et on calcule C = E(M,A) = M^A modulo n
- **Décryptage :** On calcule tout d'abord la clef secrète B telle que A.B modulo Φ(n) = 1. On peut alors décrypter le message C par M = D(C,B) = C^B modulo n.
- A et n sont publics donc tout le monde peut crypter. Tout réside dans le fait que pour n grand, Φ(n) est très difficile à calculer rapidement (B est calculé à partir de Φ(n) et A (euclide)
- Exemple : p =3 q=11

Exemple

- p=3 et q=11 N = 33 Φ = 20
- A = 7 7.B = 1 (mod 20) B = 3 (euclide)
- C = M^3 . (mod 33) et M = C^7 . (mod 33)

Texte en clair (M)			Texte chiffré (C)		Après déchiffrage	
Caractère	Valeur	P^3	P^3 (mod 33)	C^7	C^7 (mod 33)	Caractère
S	19	6859	28	13492928512	19	S
U	21	9261	21	1801088541	21	U
Z	26	17576	20	1280000000	26	Z
A	01	1	1	1	1	A
N	14	2744	5	78125	14	N
N	14	2744	5	78125	14	N
E	05	125	26	8031810176	5	E

Calculs de l'émetteur — Calculs du récepteur

241

Comparaison Symétrique / Asym.

	Symmetric	Asymmetric
Number of keys	1	2
Usual key length	128 bits	512+ bits
Performance	fast	very slow
Dedicated hardware	yes	very rare
Code breaking	difficult	almost impossible

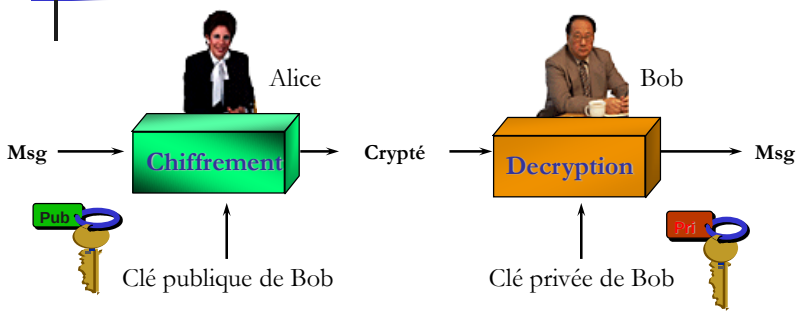
Comparaison Symétrique / Asym.

Nombre de personnes	Nombre de clés secrètes	Nombre total de clés privées ET publiques
2	1	4
3	3	6
4	6	8
5	10	10
6	15	12
7	21	14
8	28	16
9	36	18
10	45	20
15	105	30
20	190	40
50	1 225	100
100	4 950	200
500	124 750	1000
1 000	499 500	2000
10 000	49 995 000	20 000
n	n(n-1)/2	2n

Méthodes asymétriques = Usages multiples

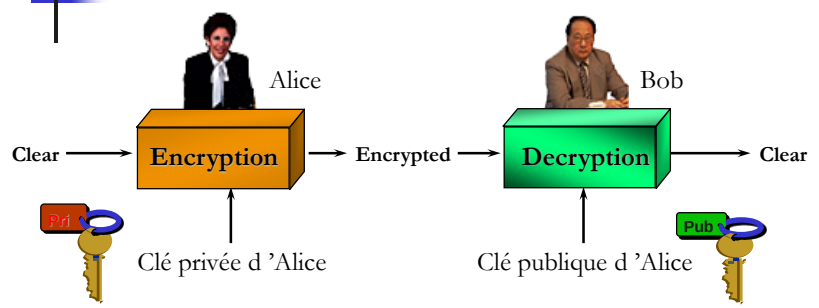
- **Les méthodes asymétriques peuvent etre utilisées pour :**
  - Confidentialité
  - Authentification
  - Confidentialité & authentification
  - Intégrité : signer des messages
  - Générer des Certificats numériques

Confidentialité



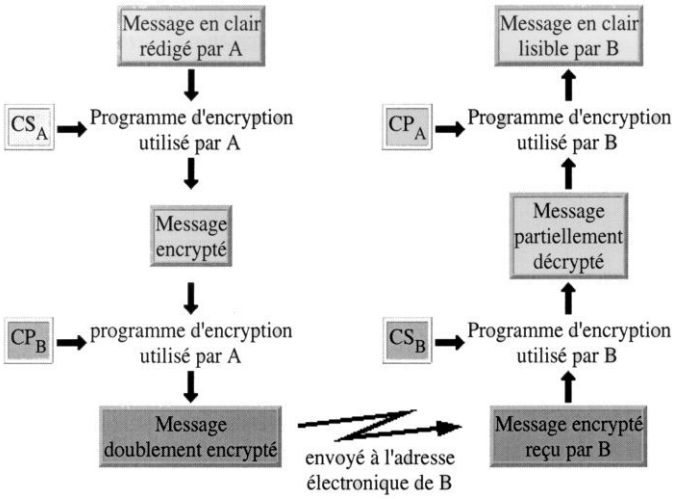
- Alice possède la clé publique de Bob pour crypter
- Bob déchiffre avec sa clé privée

Authentification



- Alice chiffre le message avec sa clé privée
- Bob déchiffre avec la clé publique d 'Alice

Confidentialité & Authentification



Signature : Authentification & Intégrité

- **Fonction de Hachage**

• Le hachage est une fonction mathématique à sens unique qui :

  • convertit une chaîne de caractères d'une longueur quelconque en une chaîne de caractères de taille fixe appelée ***digest*** **ou** ***empreinte***

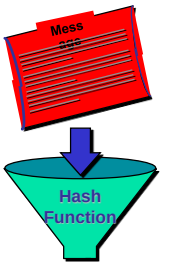
• Cette fonction est dite à sens unique pour les raisons suivantes :

  • 2 messages différents (même distinct d'un seul bit) ne produiront "jamais" la même empreinte

  • Il est très difficile de trouver le message lorsqu'on connaît l'empreinte

Signature : Authentification & Intégrité

- **Fonctions de Hachage :**
  - Algorithmes : **SHA1,** MD2, MD4, MD5, …
  - Taille typique des empreintes= 128 à 256 bits
  - Fonction = opération binaires, xor, …
- **DSS**
  - *Digital Signature Standard* conçu par le NIST
  - Utilisation de SHA-1
  - Application sur le contenu du message à signer



Hash of Message

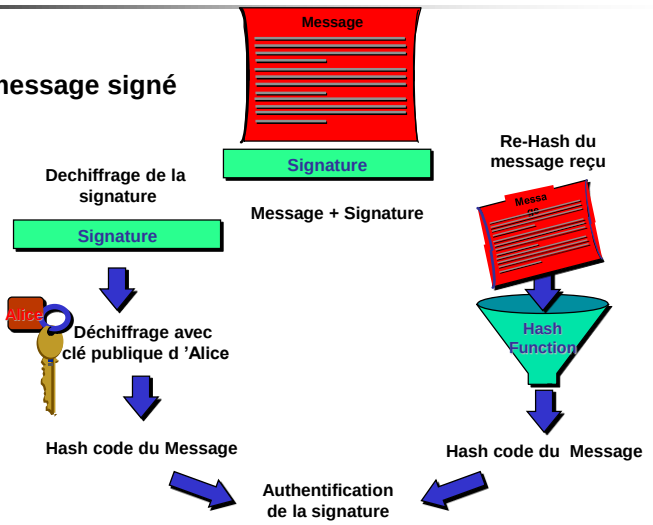
Signature : Authentification & Intégrité

MAC (Message Authentification Code)

• C'est la combinaison d'une fonction de hachage et d'une clé secrète
• L'empreinte est chiffrée à l'aide d'un algorithme à clé secrète (symétrique)
• Le destinataire ne pourra vérifier l'intégrité des données que s'il possède la clé symétrique ayant servi à la génération du MAC.
• Contrairement à la signature digitale, seul un destinataire particulier sera en mesure de faire cette opération.
• Un MAC assure l'intégrité d'un message mais pas la non-répudiation puisque émetteur et récepteur possèdent la même clé (principe du chiffrement symétrique).
• L'émetteur peut donc nier avoir signé les données puisqu'il n'est pas le seul à pouvoir le faire.
• Le MAC est très utile (rapide et efficace) à condition d'avoir mis en place un mécanisme sûr d'échange de clé secrète entre les différents protagonistes.

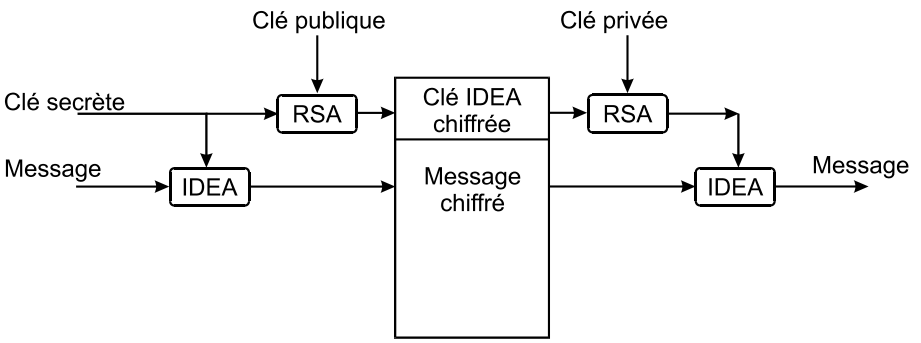
Signature : Authentification & Intégrité

Vérification du message signé



Architecture mixte : PGP (Pretty Good Privacy / 1991)

↖ Algorithme hybride : PGP = (RSA + IDEA)

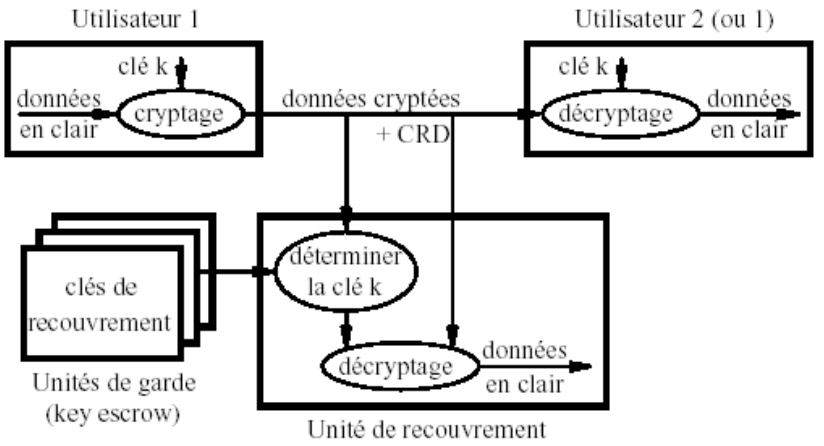


💣 Usage contrôlé en France (depuis 99) !

Architecture : Key Escrow

- **L'utilisation du cryptage pour la transmission et la sauvegarde d'informations peut poser différents problèmes, notamment :**
  - **1° En cas de perte de sa clé, le destinataire ou le propriétaire ne dispose plus d'aucun moyen d'accès aux informations.**
  - **2° L'accès aux informations par un tiers autorisé est impossible sans l'accord du destinataire/propriétaire des informations. Le tiers autorisé peut être la direction de la société dans laquelle travaille le destinataire/propriétaire (en cas d'absence ou de mauvaise volonté) ou l'autorité judiciaire (en cas d'enquête).**
- **La solution est d'associer à chaque système de cryptage une facilité de décryptage utilisable uniquement par des personnes ou organisations autorisées.**

Key Escrow



Key Escrow

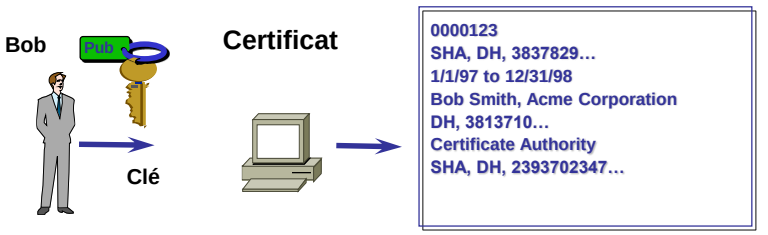
- Un Champ de Recouvrement des Données (CRD) est transmis ou stocké avec les données cryptées.
- Ce champ contient des informations suffisantes pour le décryptage par un agent autorisé (l'Unité de recouvrement), comme par exemple la clé *k* cryptée à l'aide de la clé publique (*E*, *n*) qui correspond à une clé secrète *D* de l'Unité de garde.
- Différentes précautions peuvent être prises afin d'améliorer la protection des utilisateurs face aux éventuels abus des Unités de garde ou des Unités de recouvrement, notamment :
  - 1° Unités de garde multiples et indépendantes. Chaque unité détient un seul fragment de la clé et ne peut la communiquer qu'à l'unité de recouvrement (l'agent autorisé).
  - 2° Grain fin pour les clés : le cryptage n'est pas effectué avec des clés à longue durée de vie, mais avec des clés de session. Les clés à longue durée de vie restent connues uniquement par les unités de garde. Elles permettent uniquement de récupérer les clés de session.

Méthodes asymétriques : Challenges

Distribution / Gestion des clés :

- Comment ? Qui ? A qui ? Quand ?
- **Comment Identifier les propriétaires des clés ?**

→ Certificats numériques



Certificats numériques

- Qu 'est ce qu 'un certificat numérique ?
  - Une pièce d 'identité : passeport électronique
  - notion de signatures électroniques pour l 'identification, garantir la non répudiation, l 'intégrité du message, et la confidentialité
  - fonctions d authentification et contrôle d accès
- Les différents types de certificats :
  - **Certificat Serveur** : hébergé sur un serveur (Internet), identifiant celui-ci, permettant d établir des sessions chiffrées. Lié à l '**URL du serveur**.
  - **Certificat Personnel** : hébergé sur un ordinateur, une clé USB ou une carte à puce (différents niveaux et tarifs). Lié à une **personne physique**
  - **Certificat IPsec** : hébergé sur un composant réseau (routeur, PC, …) identifiant celui-ci et permettant de chiffrer le flux entre lui et un autre équipement pour créer un VPN.

Certificats numériques

- Qui délivre les certificats ?
  - Des organisations disposant d'un certain crédit (ministère, mairie, entreprise, …) appelé les **Autorités de Certification (AC)**
- Qu'est ce qu'une AC ?
  - Organisation qui délivre des signatures électroniques
  - sert de caution morale en s 'engageant sur l 'identité d 'une personne
  - définit des conditions d 'attribution et d 'usages
  - exemple : état, banques, chambre de commerce, expert-comptable, entreprise, ..

Certificats numériques

- Comment s'organise une AC ?
  - Fonction d 'organisation (traitement des demandes, contrôle des informations, validation / rejet, révocation des certificat, …)

  ➔pour cela elle s 'appuie sur une **Autorité d 'Enregistrement (AE)**

  - Fonction technique : gestion des clés et algorithmes de chiffrement

  ➔pour cela elle s 'appuie sur un **Opérateur de Service de Certification** (Certplus, Verisign, …)

Certificats numériques

- Qu'est ce qu'une Infrastructure à Clés Publiques (ICP, PKI) ?
  - Désigne l 'ensemble des éléments nécessaires à une AC pour émettre des certificats et permettre leur administration
  - elle est constituée :
    - d 'une clé d 'autorité de certification
    - des outils nécessaire aux fonctions d 'AE
    - d 'un dispositif technique de production des certificats
- problème: durée de vie
- Voir format X509 normalisé
- Sociétés : Verisign, HP, Sun, ...

0000123
SHA,DH, 3837829....
1/1/93 to 12/31/98
Alice Smith, Acme Corp
DH, 3813710...
Acme, Security Dept.
SHA,DH, 2393702347 ...

Certificats (souvent serveurs)

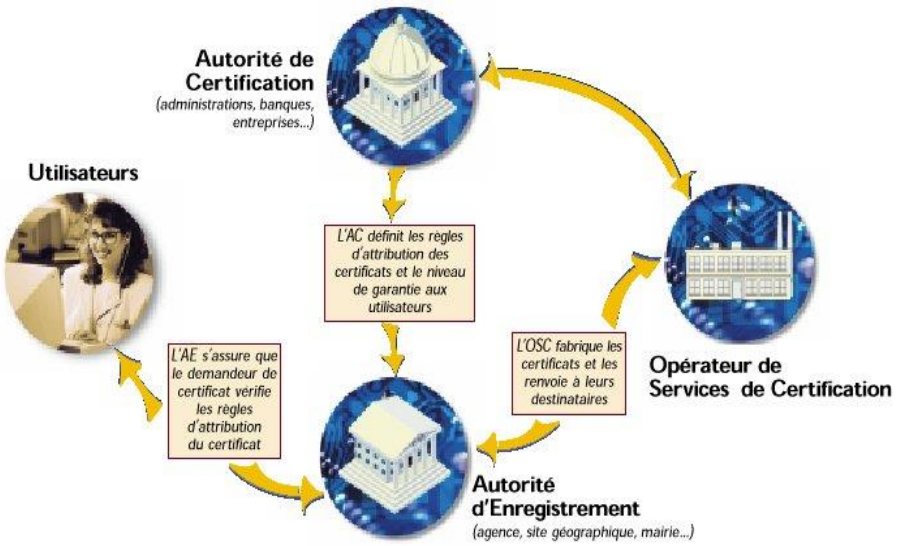
- Qu 'est ce qu 'un certificat ?
  - établit le lien entre une machine/transaction et une personne morale
  - Lien certificat → propriétaire (authentifie le serveur et son proprietaire)
  - Permet de sécuriser les échanges (entre le serveur et le client)
  - Prix : environ 300Euros/an – Certificat personnel = 80Euros/an
- Contient:
  - Clé publique du propriétaire (pour chiffrer)
  - Nom (de domaine) à certifier
  - Nom et coordonnées du propriétaire
  - Date d'expiration de la clé publique
  - Numéro de série de la signature digitale
  - Nom du distributeur (AC)
  - Signature digitale du Distributeur (AC)

Name: "www.societe.fr"
Adress: xxxxxxxxxx
Expires: 6/10/2005
Exchange Key : public
Signature Key : public
Serial#: 29483756
OtherData: 10236283025273
Signed: CA's Signature : private

Certificats X509v1, v2, v3, ...

- **Contenu minimal :**
  - Version X509 (1,2, …)
  - Numéro de série du certificat
  - Date d'expiration (debut-fin) de la clé publique
  - Algorithmes de signature utilisés par l'AC (RSA+SHA1)
  - Identifiant AC : Nom du distributeur
  - Identifiant Propriétaire : Nom et coordonnées
  - Clé publique du propriétaire + Algo (utiles pour chiffrer)
  - Autres Identifiants (facult): Nom de domaine à certifier
  - Extensions (Facultatif)
    - Liste des extensions...
  - Signature digitale du Distributeur (AC) : Champs précédents hachés et chiffrés avec la clé secrète de AC
- La clé secrète n'est pas dans le certificat (public) !

Certificats numériques



Protocoles réseaux sécurisés

- **SSL (Secure Socket Layer)**
- **Secure HTTP**
- **SSH (secure shell) :** serveur « telnet » avec clé public de 256 bits)
- **Secure TCP/IP (IPsec) → IP v.6 (Cours Réseaux)**
- **SET (Secure Electronic Transaction)**
- **...**

SSL : Services et moyens (Netscape)

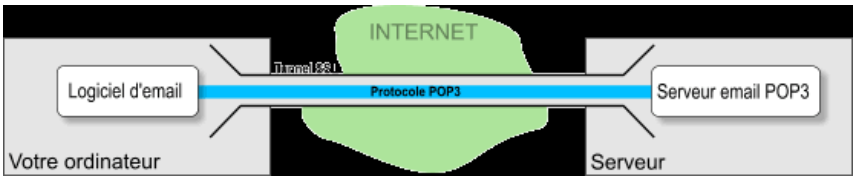
- **Communication sécurisée sur média non sécurisé**
  - **Protection de la connexion (principe de PGP)**
    - Handshake initial pour définir le codage
    - Compression possible
    - Puis cryptage symétrique des données (DES)
  - **Authentification optionnelle**
    - Les 2 entités peuvent s'authentifier en utilisant un cryptage asymétrique (Clé public/privée - RSA)
  - **Sécurisation des échanges (intégrité via signature)**
    - Authentification du message échangé (MAC = Message Authentication Code)

De SSL à TLS v1, v2, v3, openSSL …

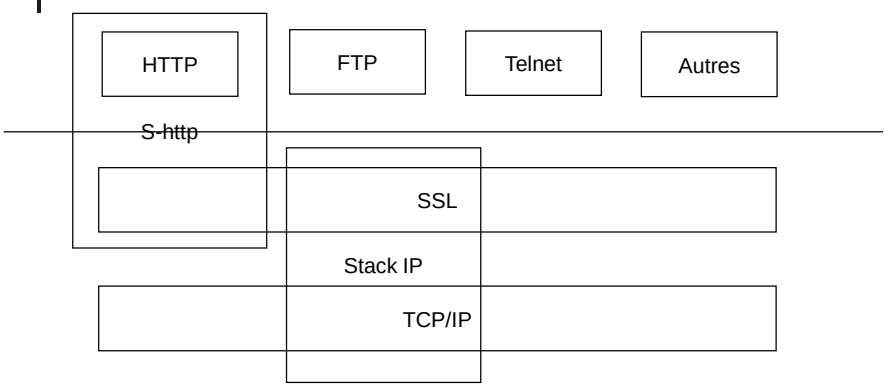
- Rebaptisée TLS (Transport Layer Security) en 2001
  - Le client se connecte au serveur sécurisé par SSL et lui demande de s'authentifier.
  - Le client envoie également la liste des cryptosystèmes qu'il supporte, triée par ordre décroissant selon la longueur des clés.
  - Le serveur envoie un certificat au client, contenant la clé publique du serveur, signée par une autorité de certification
  - Plus le nom du cryptosystème le plus évolué avec lequel il est compatible (clé de chiffrement de 40 bits ou 128 bits)
  - Le client vérifie la validité du certificat, puis crée une clé secrète aléatoire, chiffre cette clé à l'aide de la clé publique du serveur, puis lui envoie le résultat (la clé de session).
  - Le serveur est en mesure de déchiffrer la clé de session avec sa clé privée.
  - Grace à la clé commune, le reste des transactions peut se faire en garantissant l'intégrité et la confidentialité des données échangées.
  - **Extended Validation (EV) : Certification encore plus sécurisée**

De SSL à TLS v1, v2, v3, openSSL …

- Utilisé avec plein de protocoles :
  - HTTPS
  - POPS
  - SSH
  - FTPS
  - Principe = Tunnel SSL = encapsulation de protocoles



SSL vs S_HTTP

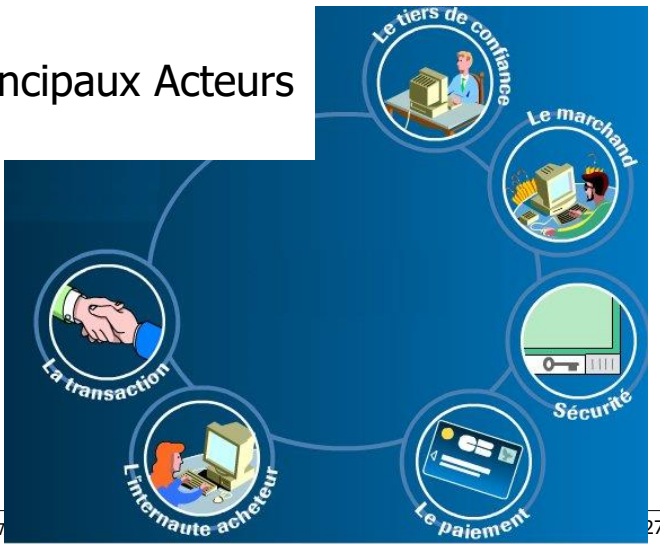


Challenge : Protéger sa clé privée

- Problème : le certificat est stocké sur le disque dur
  - **sécurité ?**
  - **certificat lié à une machine ?**
- Solutions :
  - **Utiliser une carte à puce ou clé USB pour stocker le certificat**
  - **Protection par un mot de passe : utiliser par IE, Outlook et Netscape**

Le Commerce Electronique

Les Principaux Acteurs



Commerce électronique

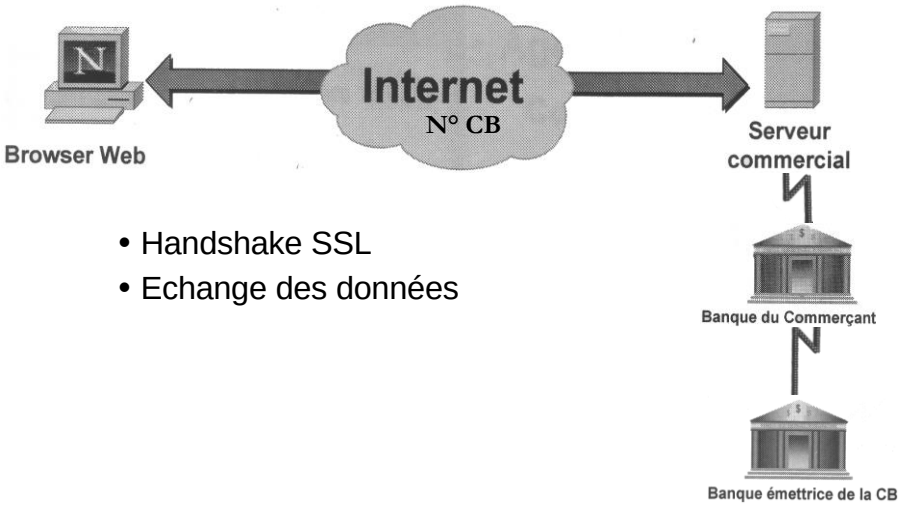
- **Aujourd'hui**
  - **Possible grâce au numéro de carte bancaire :**
    → Transfert sécurisé : S_HTTP, SSL
- **MAIS** peur du vol de numéro de carte bancaire :

  → **logiciels d'identification : SET, C_SET**
    - **Boitier de saisie de code pour PC : E_COMM**
  - **Porte monnaie électronique : PayPal, Cyber_Cash, KLELine**

Commerce électronique

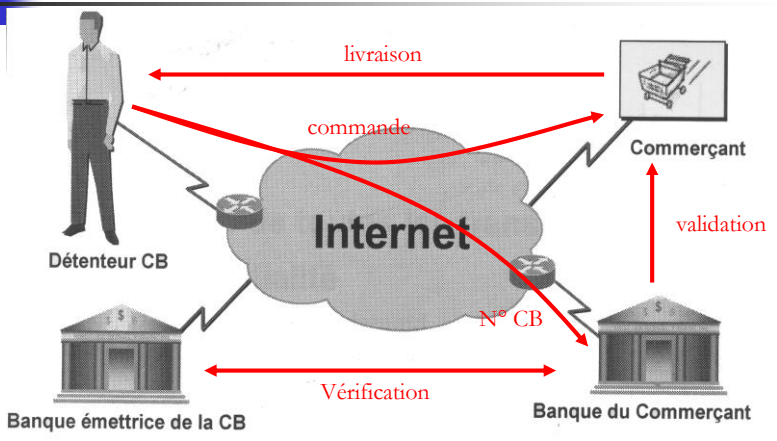
- Evolution exponentielle, initiée par les professionnels, tirée par les particuliers
- Pose tous les problèmes traités par la cryptologie
  - Authentification
  - Intégrité
  - Confidentialité
  - Non répudiation
- 2 voies principales
  - Acheteur / Vendeur → SSL
  - Acheteur / Vendeur + Banques → SET

1. Commerce Acheteur / Vendeur via SSL



- Handshake SSL
- Echange des données

2. Secure Electronic Transaction (SET)



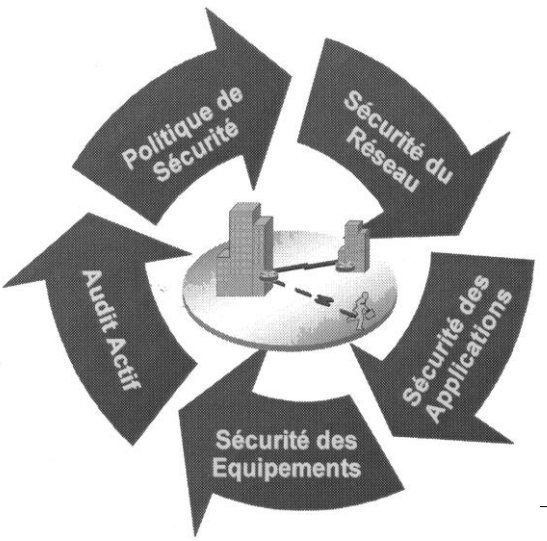
Avec SET, le commerçant ne reçoit pas le numéro de CB du client

SET

- SET prend en charge
  - Authentification acheteur & Vendeur
  - Intégrité des transmissions
  - Confidentialité du paiement et de la commande

- Déroulement d'une vente SET
  - Demande de l'acheteur
  - Vendeur vérifie la commande
  - Les banques vérifient Vendeur & Acheteur
  - Acquittement de l'ordre

Cryptologie = élément essentiel du cycle de sécurité



Législation & Cryptologie

Pas de législation internationale + évolution rapide
→ Difficulté de standardisation des protocoles

💣 Les logiciels de chiffrage ne sont pas comme les autres !

USA
- Cryptologie, armes et munitions → Même cadre juridique
- ITAR (International Traffic Arm Regulation) → Export (40 bits)

Législation & Cryptologie

- En cours d'harmonisation au niveau européen
- en voie de libéralisation en France (loi du 26/7/96, décret + arrêté du 17/3/1999)
- **Usage** de services ou moyens cryptographiques :
  - totalement libre pour l 'authentification
  - libre pour la confidentialité si clé < 128 bits et enregistrement auprès d 'un tiers de confiance
- **Fourniture** de services ou moyens cryptographiques :
  - enregistrement et demande d 'autorisation préalable (SCSSI)
- SCSSI : Service central de la sécurité des systèmes d 'information
  - http://www.scssi.gouv.fr/

En France en 2004

- Sanctions encourues :
  - Import sans autorisation : 6 mois & 30 000 €
  - Tiers de confiance illégal : 2 ans & 40 000 €
  - Fourniture pour crime & délit : 3 ans & 60 000 €

Conclusion sur la cryptographie

- Indispensable aux réseaux de communication
  → Sécurité Intranet / Extranet / Internet
- Moteur de développement du @Business
- Conséquences juridiques

Exercices

1. Un navigateur web classique désire mettre en place une communication sécurisée mais rapide avec un serveur web disposant d'un certificat X509 grâce au protocole PGP. Indiquez les differentes etapes et échanges d'informations devant avoir lieu afin de mettre en place le canal sécurisé entre les 2 machines.

Chapitre 5 :

Voies & Supports de transmission

Caractéristiques du signal

- Fonction périodique => Fourier

$$g(t) = \frac{1}{2}c + \sum_{n=1}^{\infty} a_n \sin(2\pi nft) + \sum_{n=1}^{\infty} b_n \cos(2\pi nft)$$

$$a_n = \frac{2}{T}\int_0^T g(t)\sin(2\pi nft)dt$$

$$b_n = \frac{2}{T}\int_0^T g(t)\cos(2\pi nft)dt$$

$$c = \frac{2}{T}\int_0^T g(t)dt$$

Spectre de puissance :

$$c_n = \sqrt{a_n^{\,2} + b_n^{\,2}}$$

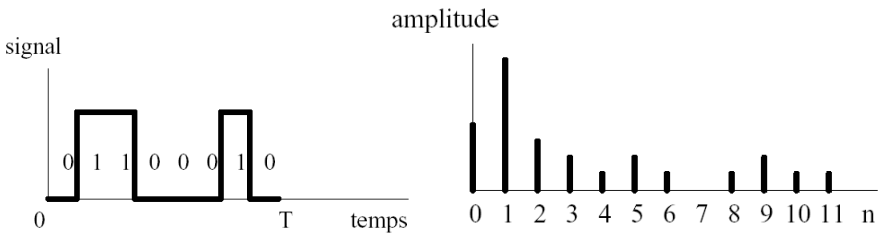
Caractéristiques du signal

- Séquence binaire : 01100010

$$a_n = \frac{1}{n\pi}\left[\cos\left(\frac{n\pi}{4}\right) - \cos\left(\frac{3n\pi}{4}\right) + \cos\left(\frac{6n\pi}{4}\right) - \cos\left(\frac{7n\pi}{4}\right)\right]$$

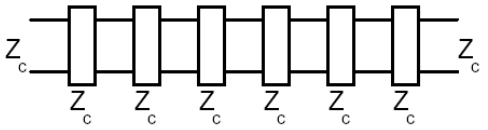
$$b_n = \frac{1}{n\pi}\left[-\sin\left(\frac{n\pi}{4}\right) + \sin\left(\frac{3n\pi}{4}\right) - \sin\left(\frac{6n\pi}{4}\right) + \sin\left(\frac{7n\pi}{4}\right)\right]$$

C=3/4

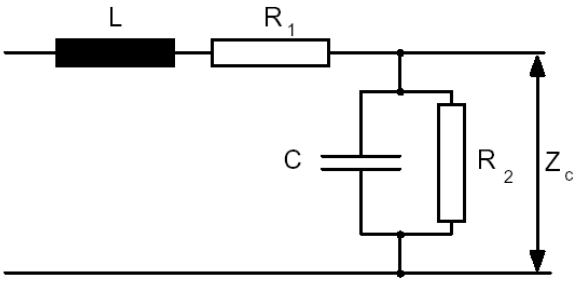


Voie de transmission

ligne = suite de cellules élémentaires



Cellule élémentaire - circuit équivalent :



Voie de transmission

- **Définition :** Une voie de transmission permet l'acheminement d'ondes électromagnétiques et électriques.
  - Modèle de référence : $a(t) = A \sin(2\pi ft + \phi)$

    où A désigne l'amplitude, f la fréquence et ϕ la phase
- **Comparaison en fréquences :**
  - Affaiblissement (ou atténuation) : $A(f) = 10 \log_{10} \left| \frac{A_1}{A_2} \right|$ en décibels
  - Déphasage = $\phi(f) = \phi_1 - \phi_2$. Le déphasage induit un retard.

Voies de transmission

- **Le bruit**
  - Le bruit blanc
  - Le bruit impulsionnel

(SNR: *Signal Noise Ratio*) = 10 $\log_{10} \left| \frac{P_s}{P_n} \right|$ db

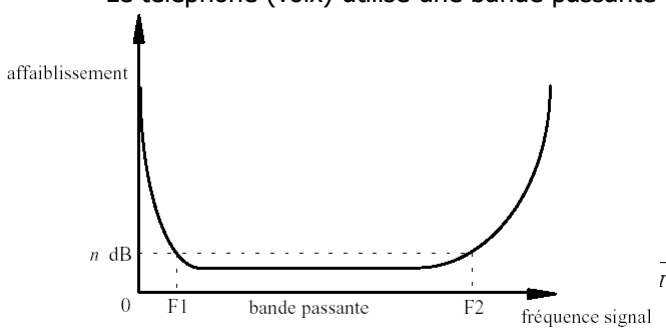
où P_s et P_n sont les puissances du signal et du bruit

Voies de transmission

- **Bande passante W :** La bande passante d'un canal de transmission est la bande de fréquence dans laquelle les signaux sont soumis à un affaiblissement inférieur ou égal à n db.

n=3 ➔ puissance de sortie > 0.5 x Puissance d'entrée

Le téléphone (voix) utilise une bande passante de [300,3400] Hz.



W = F2-F1

Voies de transmission

- **Capacité :** La capacité d'un canal est la quantité d'information que peut transmettre un canal par unité de temps (Débit binaire max.)

$$C = W \log_2 (1 + P_S/P_N) \text{ en bits/s}$$

où W est la largeur de la bande passante en Hz et $P_S/P_N = S/N$

- **Exemples :**
  - Réseau MAP, version large bande :

Le SNRmin = 26 db , 1 bande a une largeur de 12 MHz.

$C_{max} = 12\ 10^6 \log_2 (1 + 398) = 103{,}7$ Mb/s

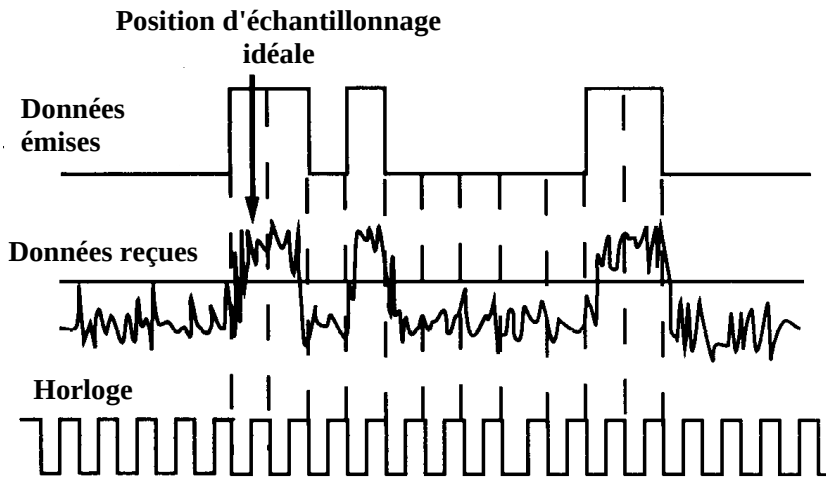
(SNR = 26db = 10 $\log_{10}$ (S/N) ➔ S/N= 398)

En pratique, le débit d'utilisation est de l'ordre de 10 Mb/s

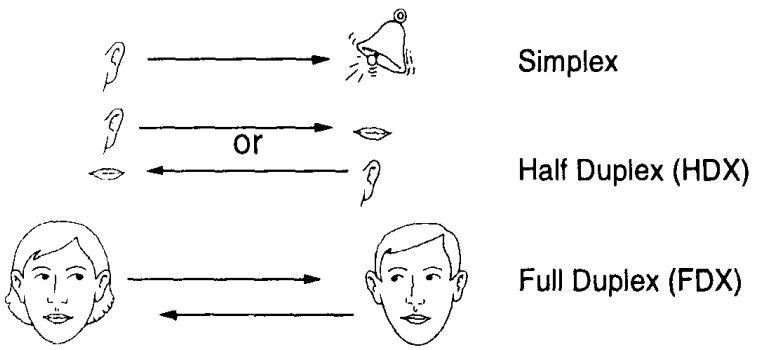
Caractéristiques de transmission

- **Instant significatif :** instant choisi pour évaluer l'état du signal transmis (instant d'échantillonnage)
- **Intervalle significatif** Δ : intervalle entre 2 instants significatifs
- **Valence V** : nombre d'états significatifs distincts pour caractériser les états du signal à transmettre
- **Rapidité de modulation R: R = 1 / Δ Bauds** (Δ en s)
- **Débit binaire :** quantité d'information émise. D = R.$\log_2$ V (bit/s)
- **Le temps de transmission (emission)** est fonction du débit du canal. Il s'exprime donc par : **T_t = N/D** où N est le nombre de bits à transmettre et D le **débit binaire** exprimé en bits/s.
- **Temps de transfert** = Temission + Tpropagation + Ttraitement

Caractéristiques de transmission



Caractéristiques de transmission



half duplex = bi-directionnelle à alternat
full duplex = bi-directionnelle intégrale

Caractéristiques de transmission

- **Type de transmission :**
  - **bande de base :** suite de bits = une suite de niveaux de tension ou de courant avec une amplitude particulière.

    La durée d'un bit T = durée d'un niveau de tension ou courant Δ (rapidité = $1/\Delta$ bauds = débit = bit/s).

  - **large bande** : adaptation de la voie de communication à la bande passante : modulation d 'une porteuse ayant une fréquence > fréquence du bit (plusieurs voies de communication sur un seul support physique)

bande de base / large bande

- Lorsque l'on veut utiliser un même médium/support pour plusieurs canaux, il faut décomposer la bande de fréquences en plusieurs bandes → **Transmission large bande.**
- Principe du Full Duplex : utilisation d'une tête de station qui reçoit tout signal dans une bande et le réémet dans une autre bande. *En mode appelant :*
  - station vers tête : basses fréquences (5 - 116 MHz)
  - tête vers station : hautes fréquences (168 - 300/400 MHz)
- La bande 116-168 MHz est volontairement inutilisée pour faciliter le repérage des signaux par des filtres passe-bas ou passe-haut (pratique aussi pour les répéteurs/amplificateurs). La tête de câble est donc un simple modulateur/démodulateur (MODEM) qui transpose la fréquence.

Caractéristiques de transmission

- **Le temps de propagation** caractérise le support physique.
- Il n'est pas négligeable pour les grandes distances ni pour les lignes chargées. Il dépend entre autre de la nature de l'isolant. Sur les supports classiques on l'estime par T_p = 5 µs/km
- **Ex** : 2 stations sont distantes de 1 km. Combien de bits peut émettre A avant que le 1er bit n'arrive en B. Débit = 10Mb/s.

Codage en Bande de base

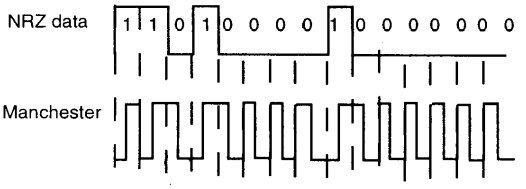
- Le codage NRZ (*Non Return to Zero*) :
  - C'est le codage le plus simple.
  - On associe deux niveaux de signes opposés pour les deux valeurs à coder :
    0 → -a
    1 → +a
- Le codage NRZI (*Non Return to Zero Inverted*) :
  - Codage causal → $c_i = d_i . c_{i-1}$
  - d_i vaut -1 (respectivement 1) si le bit à envoyer vaut 1 (respectivement 0).

Codage en Bande de base

- **<u>Le codage Manchester I</u> (biphasé : 1 bit == 1 transition)**
  - Un bit b_i correspond à une durée physique T que l'on décompose en deux intervalles de temps , auxquels sont associés deux états différents $q_{i,1}$ et $q_{i,2}$.
  - La règle de codage (Manchester I) est alors la suivante:

$$\mathbf{b_i = 0 \Rightarrow q_{i,1} = +a \text{ et } q_{i,2} = -a}$$
$$\mathbf{b_i = 1 \Rightarrow q_{i,1} = -a \text{ et } q_{i,2} = +a}$$

  - On code donc la valeur du bit par un sens de transition.



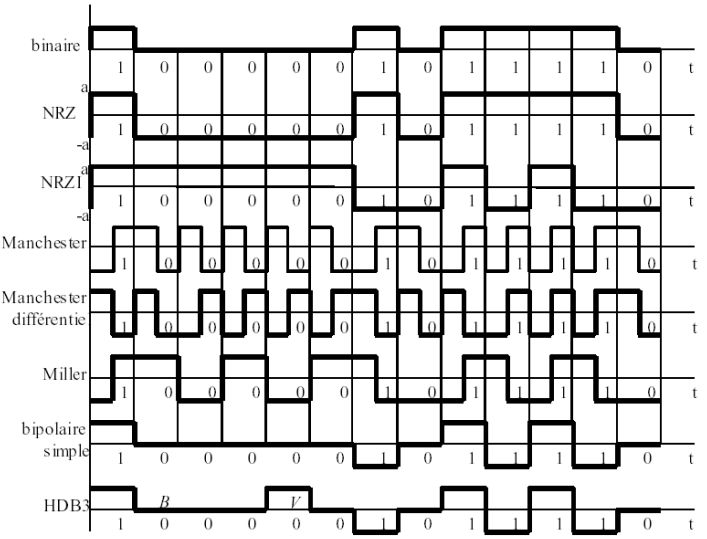
Codage en Bande de base

- **<u>Le codage bipolaire</u> (AMI: codage causal ternaire)**
- 3 niveaux : -a, O et a :

$$\mathbf{b_i = 0 \Rightarrow q_i = 0}$$
$$\mathbf{b_i = 1 \Rightarrow qi = +a \text{ si le dernier q était égale à } -a}$$
$$\mathbf{qi = -a \text{ si le dernier q était égale à } +a}$$

- **Peu utilisé dans les réseaux locaux, mais pour les transmissions rapides à longues distances.**

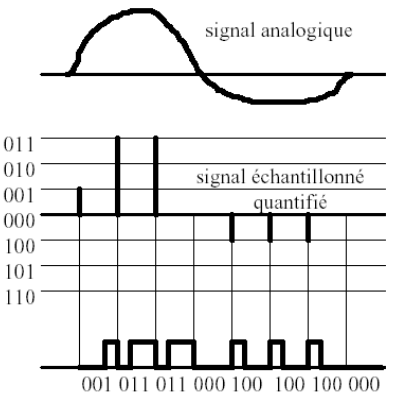
Codage en Bande de base



299

Modulation par impulsion et codage

- Transmission numérique de signaux analogiques !
- Echantillonnage + Quantification par niveau (n bits)
- Débit binaire = Fe.n = 2.Fmax.n

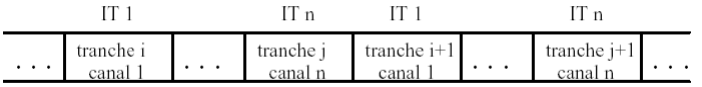


Ex : téléphone

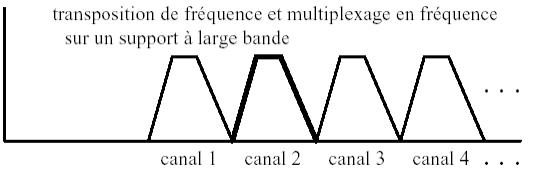
300

Multiplexage temporelle / en Fréquence

- Avoir n canaux de débit d simultanément sur un support qui accepte un débit D=n.d
  - Temporelle : Gestion des IT (statique ou dynamique)



  - Fréquence :



Canaux logiques

AMRT (Accès Multiple à Répartition dans le Temps)
(ou TDMA : *Time Division Multiple Access*)

Principe : Découpage du temps par gestion de quotas affectés à chaque station.

⊕ Pas de conflits.
⊕ Détermination **a priori** du temps maximum d'accès. Par analyse du nombre de stations, de la topologie et des quanta.
Θ Rendement très faible si les stations sont peu actives

- Utilisée dans les réseaux de terrains, pour un environnement maître-esclave avec scrutation périodique des esclaves. Un contrôleur (micro-ordinateur) qui pilote un ensemble d'automates ou de périphériques non intelligents.

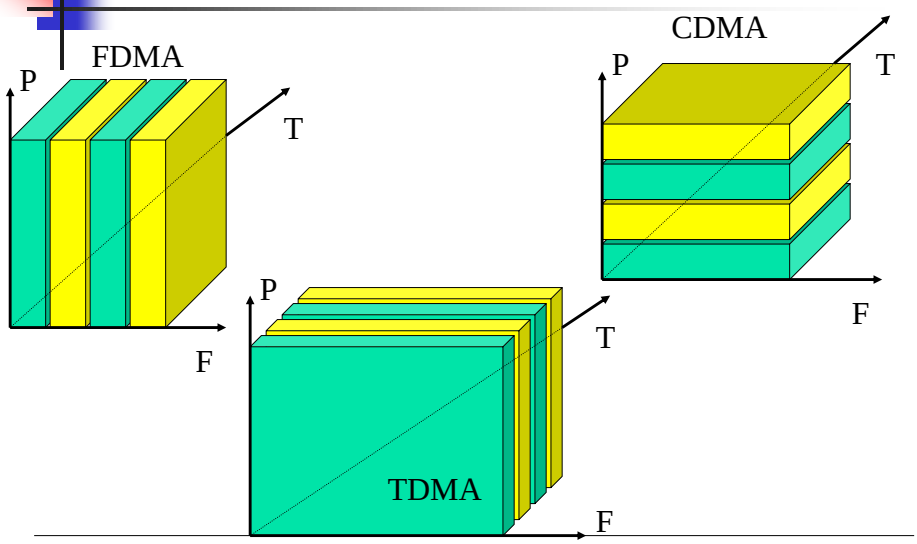
Duplexage

- Full Half Duplex
- Canaux montants/descendants

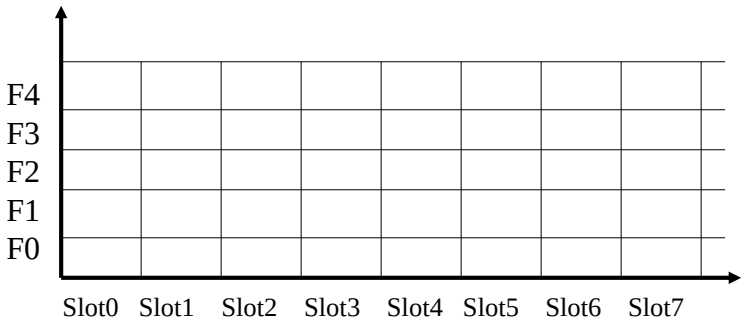
Frequency Division Duplex

Time Division Duplex

Canaux logiques



FTDMA



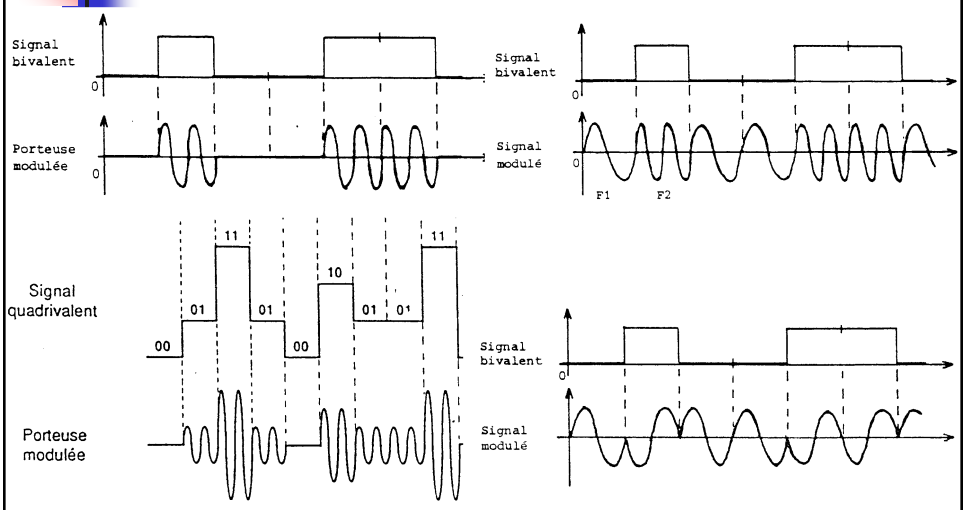
Exemple le GSM

- Montant 890-915 MHz
- Descendant 935-960 MHz
- Porteuse séparée de 240 kHz
- 124 porteuses dans chaque sens
- TDMA avec 8 ITs/porteuse
- Chaque IT dure 577 μsec

Codage par modulation (transposition de f)

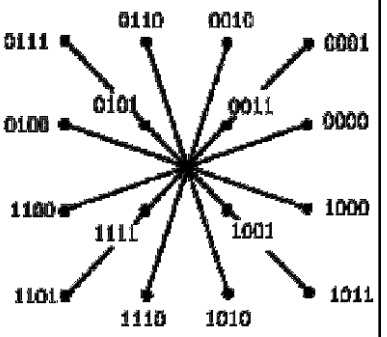
- **3 types de modulation :**
  - FSK : *Frequency Shift Keying*
  - ASK : *Amplitude Shift Keying*
  - PSK : *Phase Shift Keying*
- Les transmissions dans les réseaux publics avec modems normalisés utilisent ces principes :
  - FSK pour les faibles débits (< 1200 b/s) ;
  - PSK pour les débits intermédiaires (1200 - 4800 b/s) ;
  - ASK - AM pour les hauts débits (> 4,8 jusqu'à 7,2 Kb/s).

Codage par modulation



Des Modulation de + en + complexes …

- Phase + amplitude : 16 états auxquels on associe 4 bits. En deux périodes de modulation, on a transmis un octet
- Ce type de modulation est appelé ***QAM*** - *Quadrature Amplitude Modulation*.
- CAP Carrierless Amplitude and Phase *(Modulation Amplitude/Phase sans Porteuse)* # QAM
- PAM = Pulse Amplitude Modulation
- DMT = Discrete Multi Tone = Modulation de freq
- …



Les MODEMs

- **Il y a deux familles principales de Modems:**
  - Pour ligne commutées: Mode Asynchrone
  - Pour lignes permanentes : Mode Synchrone (point à point par des circuits loués aux opérateurs des télécoms)
- **Modems pour lignes commutées:**
  - *DCE - Data Communication Equipment*
  - *DTE - Data Terminal Equipment*
- **Configuration en Mode Appelant :**
  - Génération des tons ou des impulsions de composition du numéro de téléphone
  - Adaptation de la vitesse de transmission en fonction des conditions *(Fallback)*
  - Gestion de la réception de la porteuse *(Carrier Detect)* ou de sa perte

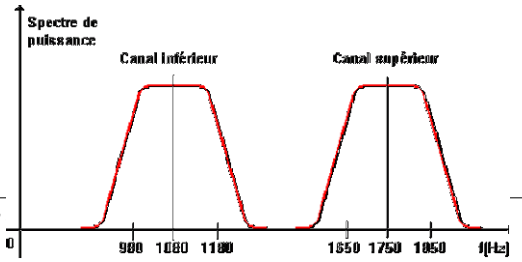
Les MODEMs

Configuration en Mode Appelé :

- Détection de sonnerie *(Ring indicator)* pour signaler la réception d'un appel
- Etablissement de la connexion avec l'appelant, échange des modes de fonctionnement
- En cas de dégradation de la qualité de la ligne: *Fallback*
- Gestion de la réception de la porteuse *(Carrier Detect)* ou de sa perte

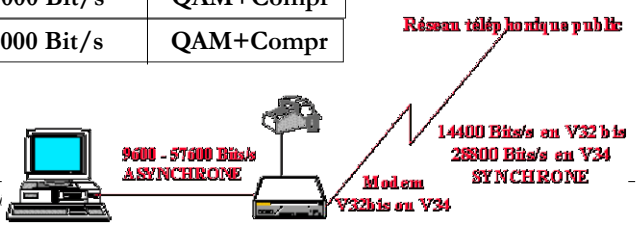
Fréquences vocales, Bande passante et Modulation :

- La bande passante garantie par les opérateurs du téléphone se situe entre **800 et 3800 Hz**
- Mode *Full Duplex* :



Normes et vitesses de transmission

AVIS CCITT	VITESSE	MODULATION
V21/Bell 103	300 Bits/s	FSK
V22/Bell 212a	1200 Bits/s	DPSK
V23	1200/75 Bits/s	DPSK
V22bis	2400 Bits/s	QAM
V32	9600 Bits/s	QAM
V32bis	14.400 Bits/s	QAM
V34	28.800 Bits/s	QAM
V34+	33.600 Bits/s	QAM
V90	**56.000 Bit/s**	**QAM+Compr**
V92	**56.000 Bit/s**	**QAM+Compr**



Les MODEMs

- **Mode de transmission :**
  - **Contrôle de flux :** contrôle du débit entre le DTE et le DCE
  - **Correction d'erreurs :** retransmission de paquets entre-eux
  - **Compression :** optimisation par techniques de compression sur la ligne téléphonique (MNP5, V42bis, …)
- **Modems RNIS :** Modems pour lignes du *Réseau Numérique à Intégration de Services*
- **Modems pour ligne louées :**
  - Pour relier deux réseaux distants
  - Liaisons téléphoniques des PTT
  - Aucun organe de commutation → ligne point-à-point.
  - Débits : 256 Kbits/s à 54 Mbits/s
- **→Technologie xDSL** :

  ADSL et HDSL : Haut débit numérique sur cuivre (ligne d'abonné)

Technologie xDSL : Le fonctionnement global

- **La bande passante de la boucle locale est limitée à 4 KHz par des filtres mis en place par les compagnies de téléphone**
- **Cette bande passante suffit pour le transport de la voix**
- **Si on retire le filtre passe-bas, la bande réelle de la boucle locale dépasse le MHz lorsque la paire de cuivre est en bon état, et que sa longueur ne dépasse pas quelques kilomètres.**
- **Quand on dispose d'une bande de fréquence large d'un MHz, on peut la diviser en :**
- **1000/4 = 250 canaux (de 4 KHz chacun)**

- Si un canal permet de faire passer 33,6 Kbps (modem analogique traditionnel), on dispose d'un débit total de :

  250 x 33,6 x 1000 = 8,4 Mbps

- Là réside la base des procédés DSL.

Les variantes DSL … Pas toujours claires …

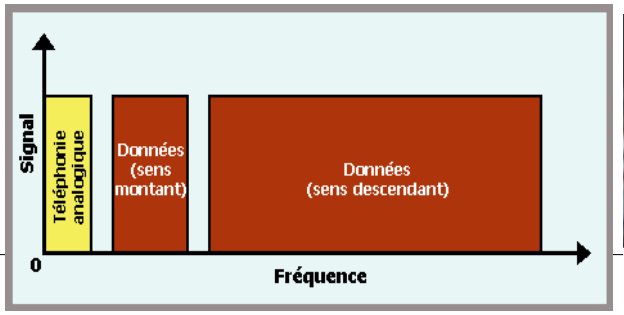
- ITU Union Internationale des Télécommuncations
- ARCEP / ART / autorité de régulation des communications

Technologie	*Signification*	*Mode de Transmission*	*Débit opérateur vers utilisateur*	*Débit utilisateur vers opérateur*	*Distance maximale*
HDSL	High-Data-Rate DSL	Symétrique (2B1Q/CAP)	1,544 Mbits/s	1,544 Mbits/s	3,6 km
SDSL	Single-Line DSL	Symétrique (2B1Q/CAP)	768 kbits/s	768 kbits/s	3,6 km
ADSL	Asymetric DSL	Asymétrique (DMT)	1,544 Mbits/s à 9 Mbits/s	16 kbits/s à 640 kbits/s	5,4 km (à 1,5 Mbits/s)
RADSL	Rate-Adaptative DSL	Asymétrique (CAP)	600 kbits/s à 7 Mbits/s	128 kbits/s à 1 Mbits/s	5,4 km (à 1,5 Mbits/s)
DSL	Digital Subscriber Line	Symétrique (CAP/DMT...)	160 kbits/s	160 kbits/s	5,4 km
IDSL	ISDN over DSL	Symétrique (2B1Q)	128 kbits/s	128 kbits/s	3,6 km
VDSL	Very-High-Data-Rate DSL	Asymétrique (CAP/DMT...)	13 Mbits/s à 53 Mbits/s	1,544 Mbits/s à 2,3 Mbits/s	1,5 km (à 13 Mbits/s)

-La technologie DSL et ses variantes-

ADSL : Asymmetrical DSL

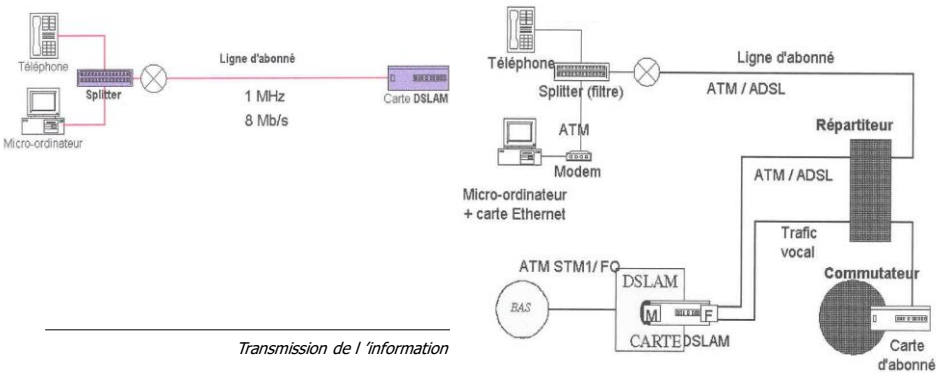
- Le procédé ADSL s'est développé fort lentement :
  - **La norme qui décrit le procédé ADSL ne fixe pas les valeurs des débits :**
  - **Le matériel n'était pas normalisé.**
  - **Les frais de raccordement étaient élevés, un technicien devait venir installer le splitter au domicile de l'usager, et le configurer.**
  - **Le prix de l'abonnement était trop élevé, du moins pour les particuliers.**



Technologie ADSL

Sans ADSL

Avec ADSL

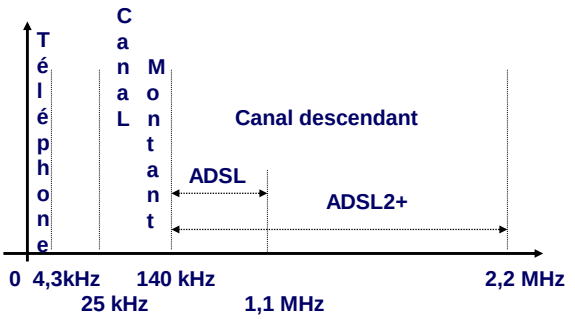


Les variantes ADSL : ADSL2, ADSL2+, ReADSL

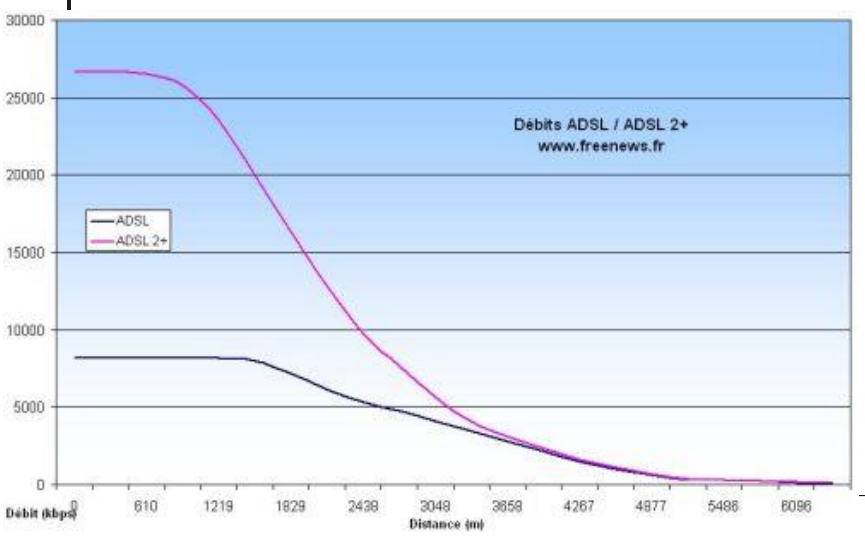
	ADSL	ADSL 2	ADSL 2+
Débits montants (max théoriques)	1 Mbps	1 Mbps	1,2 Mbps
Débits descendants (max théoriques)	8 Mbps	10 Mbps	25 Mbps
Temps d'initialisation de la connexion	10 s	3 s	3 s
Distance maximale de raccordement	5 km	(5à 10% sup)	(5à 10% sup)

ADSL 2 : Optimisation de la méthode de modulation
ADSL2+ : Elargissement de la bande passante jusqu'à 2,2 MHz
Reach Extended ADSL : On augmente encore les distances
Re ADSL 2 : Meilleures utilisation des canaux basses Freq
VDSL 2 : 100Mb/s dans les 2 sens (W=30MHz -700m)

Les variantes ADSL : ADSL2, ADSL2+, ReADSL



Les variantes ADSL : ADSL2, ADSL2+, ReADSL



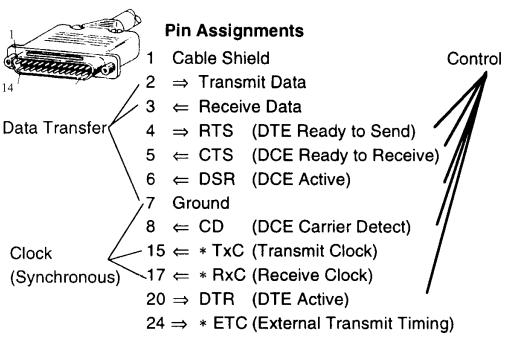
Quelques Normes : V24

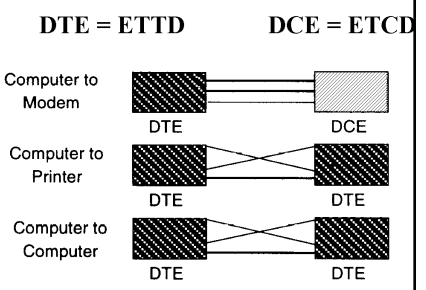
- ***RS232C, Interfaces V24 / V 28 (CCITT)***
  - Très répandue surtout entre ETCD – ETTD
  - Connecteur DB25 ou DB9
  - Ex : communication entre un micro-ordinateur et une imprimante.
    - Liaison série : une ligne dans chaque sens
    - Codage par niveaux de tension
    - Longueur maximale du réseau : 15 m.
    - Vitesse de modulation < 20 Kbauds.
    - Liaison bipoint uniquement.

→ Modifications du réseau pas évidentes une fois l'installation faite.

Quelques Normes : V24

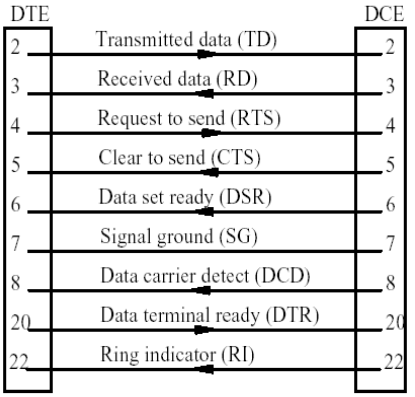
Utilisée pour relier un terminal à un modem :



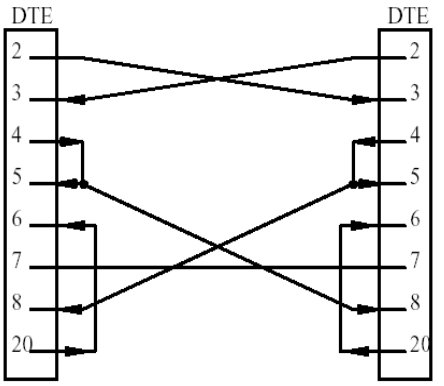


DTE = ETTD = Equipement terminal de traitement de données
DCE = ETCD = Equipement de terminaison de circuit de données

Quelques Normes : V24

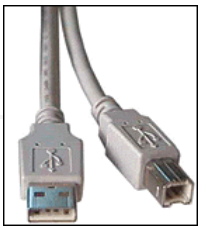


Liaison ETCD-ETTD via DB9

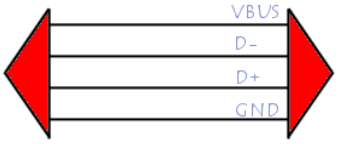


Liaison série PC - PC via DB9

Quelques Normes : USB

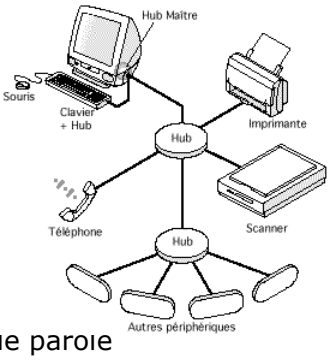


- Universal Serial Bus v1.1, née en 1995
- 2 modes de communication
  - 12 Mbps en mode haute vitesse
  - 1.5 Mbps à basse vitesse



- connexion d'une grande variété de périphériques (Plug & play)
- fournit l'alimentation électrique (pour périph. - 5 Volt)
- câble composé de quatre fils (la masse GND, l'alimentation VBUS et deux fils de données appelés D- et D+).

Quelques Normes : USB



- Architecture bus ou étoile
- Adresse sur 1 octet, 5 Hub max
- Câble de 5m ➔ 25m max
- 1 maître envoie un jeton pour la prise de paroıe
- Attribution automatique d'adresse par énumération
- USB v2.0 (480 Mb/s)

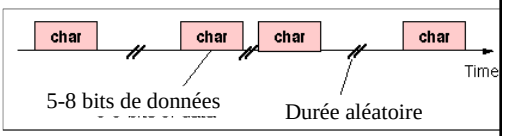
- FireWire (IEEE 1394) : fournir un système d'interconnexion permettant de faire circuler des données à haute vitesse en temps réel.

Caractéristiques de transmission

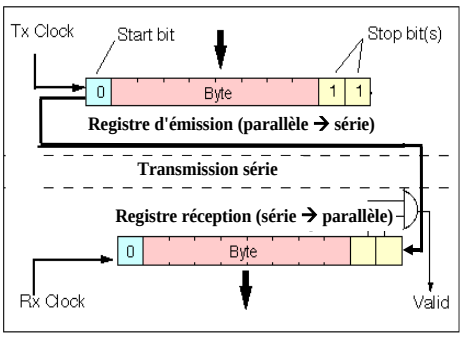
- **Mode de transmission :**
  - **En parallèle**, N bits ➔ N voies de transmission ➔ débit élevé
  - Mais sur les longues distances, désynchronisation entre lignes
  - **En mode série**, 1 seule voie ➔ plus performante à coût égal
- **Communication Synchrone / Asynchrone :**
  - **Asynchrone**: la synchronisation est réinitialisée à chaque caractère (à chaque paquet) par envoi d'un signal particulier.
  - **Synchrone**: la synchronisation est maintenue en permanence
    - **Synchronisation-bit (voir les codages)**

Communication Asynchrone

- Transmission caractère par caractère

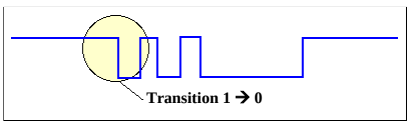


- Horloges indépendantes → caractère court (<8 bits) et débit faible (< 9,6 Kb/s)

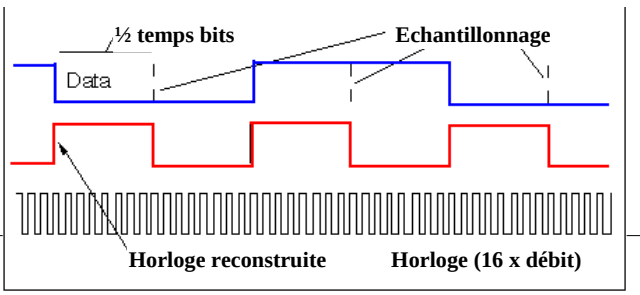


Communication Asynchrone

- Le UART (Universal Asynchronous Receiver Transmitter) détecte le bit START (transition 1→0)



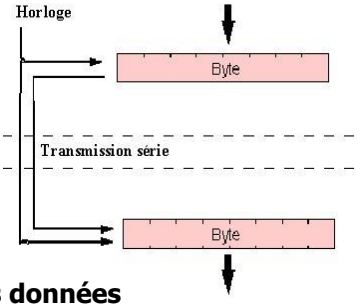
- Reconstruction de l'horloge et échantillonnage



Communication Synchrone

Horloges Emetteur/Récepteur synchronisées

- **1. L'horloge est transmise sur une voie**



- **2. L'horloge est contenue dans les données**
  - **Manchester**

Exercices

Le support physique ou Médium

- Paires torsadées
- Coaxial
- Fibre optique
- Ondes (Laser, satellites, radio)
- . . .

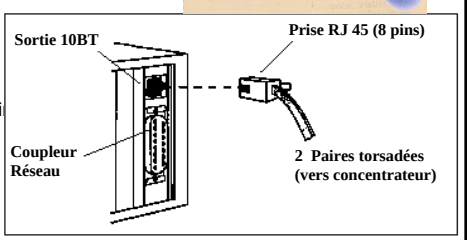
La fibre optique semble l'avenir mais cette technique a aussi des inconvénients majeurs.

- Critère de choix :
  - Coût
  - Extensibilité
  - Fiabilité :
    - Immunité électromagnétique
    - Résistance mécanique
    - Souplesse
    - Résistance thermique
    - Corrosion
    - Facilité de localisation des coupures

La paire torsadée

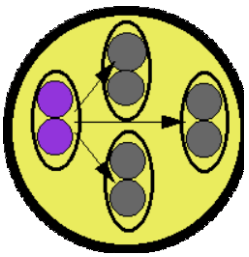
- Deux conducteurs monobrins recouvert d'un isolant et torsadés l'un par rapport

- Il existe 5 catégories :
  - catégorie 1 : aucune spécification
  - catégorie 2 : D < 4 Mb/s
  - catégorie 3 : D < 16 Mb/s
  - catégorie 4 : D < 20 Mb/s
  - catégorie 5 : D < 100 Mb/s (voi
  - catégorie 5E, 6, ...



Paire torsadée

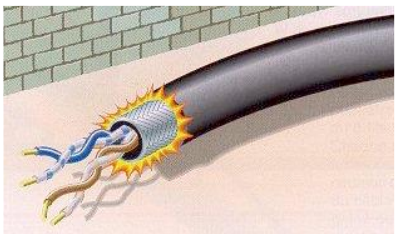
- **Points importants pour la qualité du câble Cat 5 :**
  - La résistance ohmique
  - L'atténuation
  - La paradiaphonie entre les paires



Emission parasite d'une paire sur les autres: Diaphonie

Pour diminuer le risque de diaphonie, réduire le couplage capacitif parasite entre les paires → 4 paires ayant des pas de torsade différents (pas d'imbrication)

La paire torsadée blindée



- **Aujourd'hui, on installe que du câble catégorie 5 non blindé *(UTP - Unshielded Twisted Pairs)* ou blindé *(STP - Shielded Twisted Pairs)* munis de connecteurs *RJ45*:**
- Une "guerre" entre les partisans d'*UTP* et de *STP*

La paire torsadée (blindée ou non)

⊕ Faible coût.

⊕ Transmission de tout courant (du continu à l'alternatif).

⊕ Les fréquences supportées peuvent aller jusqu'à plusieurs MHz.

⊕ Support est bien adapté aux faibles distances en point à point.

⊕ Dans le cas des réseaux locaux, ce support est utile pour les transmissions en modulation de bande de fréquences.

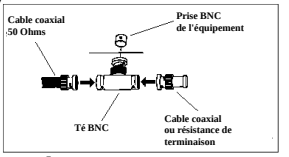
Θ Liaisons multipoints sont difficiles à réaliser

Θ Très peu sûr car l'espionnage est facile et quasi-indétectable.

Le câble coaxial

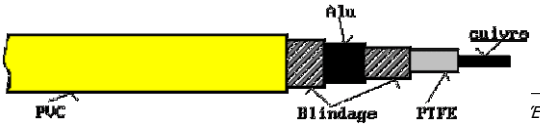
Fil conducteur mono ou multi brins, entouré d'un isolant et disposé dans l'axe d'un tube conducteur.

- Rapport optimal entre les deux diamètres : 3,6
- Impédance < 100 ohms.



Les plus connus des coaxes sont :

- **CATV (*Community Antenne Television*), 75 ohms**
- **Ethernet fin**
- **Ethernet épais**



Le câble coaxial

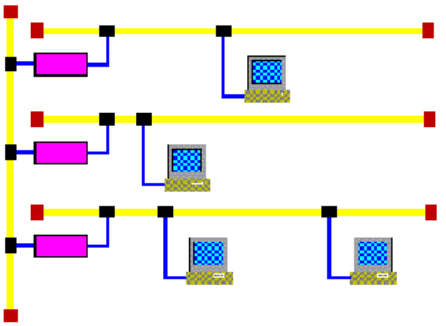
Avantages / Inconvénients :

⊕ **Supporte les normes en bande de base (RS232, RS422, …).**

Θ Θ **Bande passante de quelques dizaines à plusieurs centaines de MHz selon la distance du réseau.**

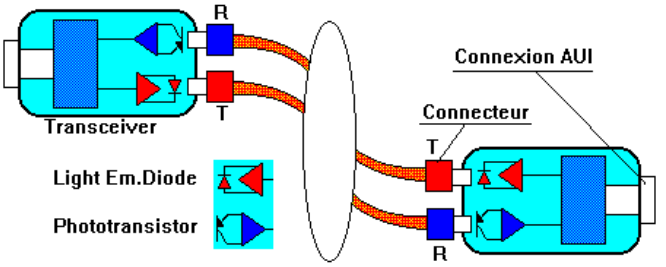
Θ **L'espionnage est aisé.**

Utilisé surtout pour les backbone (câblage XY) :

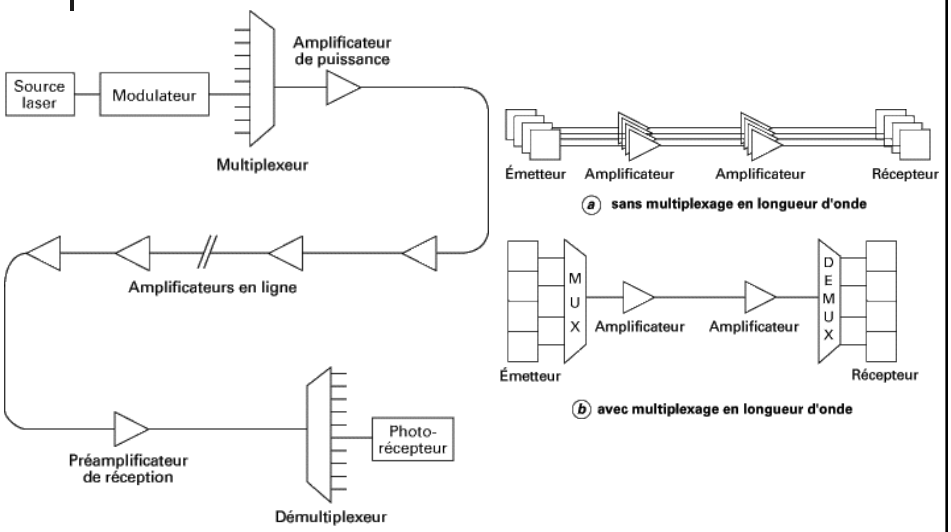


La fibre optique

- Une fibre optique est un guide d'ondes cylindriques créé dans un matériau transparent par variation de l'indice de réfraction vers la périphérie.
- Matériaux : verre (silice) ou plastique.
- **Conversion de signaux électriques en signaux optiques :**



La fibre optique



La fibre optique

Conversion de signaux électriques en sign. optiques :

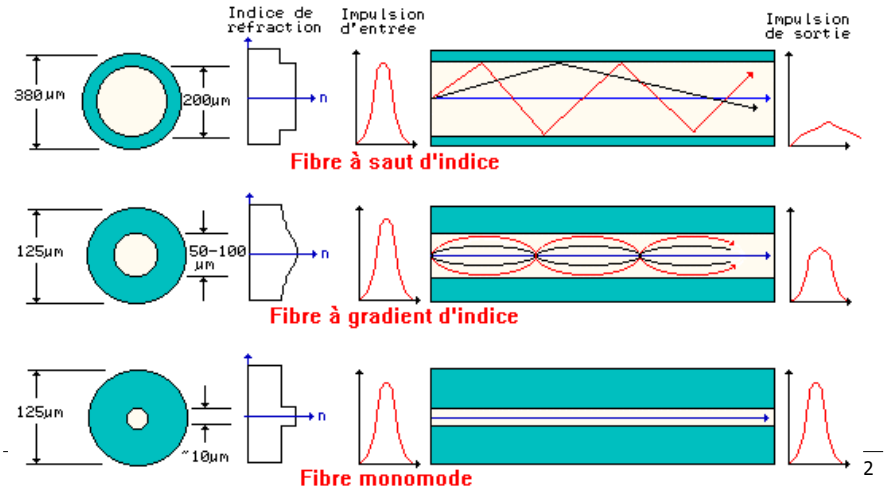
- convertir des impulsions électriques en signaux optiques véhiculés au cœur de la fibre
- les signaux électriques seront traduits en impulsions optiques par une LED et lus par un phototransistor ou une photodiode
- On utilise une fibre pour chaque direction de la transmission
- Les émetteurs utilisés sont de trois types:
  - Les LED *Light Emitting Diode* qui fonctionnent dans le rouge visible (850nM). C'est ce qui est utilisé pour le standard Ethernet FOIRL
  - Les diodes à infrarouge qui émettent dans l'invisible à 1300nM
  - Les lasers, utilisés pour la fibre monomode, dont la longueur d'onde est 1300 ou 1550nM

La fibre optique

- **Les trois types de fibre optique** :
  - **La fibre à saut d'indice** constituée d'un cœur et d'une gaine optique en verre de différents indices de réfraction. Cette fibre provoque de par l'importante section du cœur, une grande dispersion des signaux la traversant, ce qui génère une déformation du signal reçu.
  - **La fibre à gradient d'indice** dont le cœur est constitué de couches de verre successives ayant un indice de réfraction proche. On s'approche ainsi d'une égalisation des temps de propagation, ce qui veut dire que l'on a réduit la dispersion nodale. Bande passante typique 200-1500Mhz par km.
  - **La fibre monomode** dont le cœur est si fin que le chemin de propagation des différents mode est pratiquement direct. La dispersion nodale devient quasiment nulle. La bande passante transmise est preque infinie (> 10Ghz/km). Cette fibre est utilisée essentiellement pour les sites à distance.

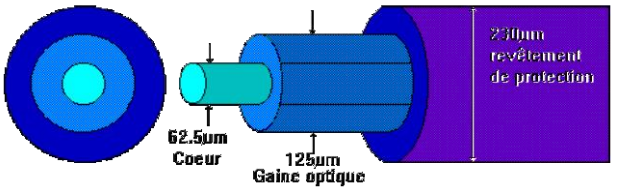
La fibre optique

- **Les trois types de fibre optique** :

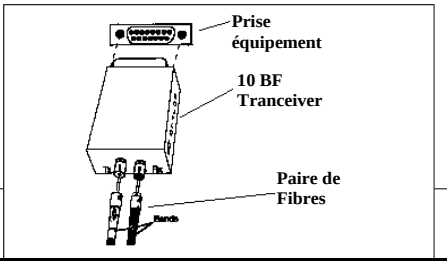


La fibre optique

- **Fibre à gradient d'indice** :



- **Fibre monomode :** Les petits diamètres du cœur (10um) nécessite une grande puissance d'émission, donc des diodes au laser qui sont relativement onéreuses.



La fibre optique

Caractéristiques majeures :

- **L'atténuation : inférieure à 1db/km => Très utile pour les longues distances**
- **Bande passante : 1GHz pour 1km.**
- **Ouverture numérique : liée au diamètre, elle permet de déterminer la fraction du rayon incident qui est admise par la fibre (important surtout pour les raccordements).**
- **Le signal véhiculé par la fibre optique est caractérisé par sa longueur d'onde.**
- **A priori, la longueur d'onde n'induit pas de contraintes.**
- **Les émetteurs et récepteurs disponibles sur le marché proposent essentiellement deux longueurs d'ondes : 850 ou 1300 nm.**
- **Les systèmes à 850 nm coûtent moins chers, mais, l'atténuation est plus faible pour les systèmes à 1300 nm (ce qui permet d'envisager des réseaux plus longs sans amplificateurs).**

La fibre optique

- **Avantages /Inconvénients :**

⊕ Taux d'erreur très faible même en milieu perturbé.

⊕ A priori, pas d'espionnage possible.

⊕ Le poids de la fibre optique est très faible ce qui en fait un support intéressant pour les réseaux embarqués.

⊕ On peut tout à fait se servir de la gamme des longueurs d'ondes pour créer plusieurs canaux de communication, de même que dans le système large bande.

Θ Le verre plus coûteux, plus fragile et plus difficile à utiliser pour les raccordements.

Θ Liaisons multipoints sont difficiles à réaliser.

Θ Support coûteux.

La fibre optique

- Wavelength Division Multiplexing :
  - **Multiplexage de longueur d 'ondes (8 longueurs d 'ondes)**
  - **Dense WDM : 160 longueurs d 'ondes => 400 Gb/s !**

DONC :

- La fibre optique est donc incontestablement le support de l'avenir. Mais …
- On estime que la fibre optique coûte cinq fois plus cher que la paire torsadée

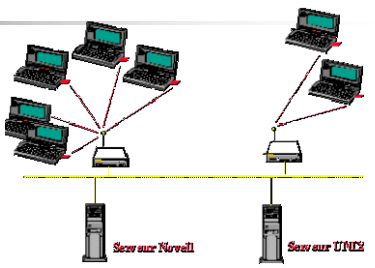
Signaux optiques : Liaison Laser

- Lorsque que l'on a pas la possibilité d'établir une liaison par fibre optique ou ligne téléphonique dédiée
- Deux sites à relier distants de moins d'un kilomètre
- Pas d'obstacle au faisceau.
- On trouve des lasers qui se comportent comme une paire torsadée et même des lasers à 155 Mbits/s pour ATM!

- **Points forts et points faibles :**
  - **L'alignement des faisceaux est difficile**
  - **En cas de déménagement, on peut récupérer une paire de lasers, pas une fibre sous la route**
  - **Sensible à la météo (dans les cas extrêmes) ….**
  - **A installer hors d'atteinte des mains des bricoleurs...**

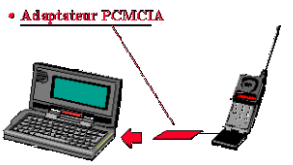
Transmission Radio

3 catégories de liaison radio :



• ***Lan to Lan***

• **Ordinateurs mobiles**



• **Téléphones portables GSM équipés de modem**

On peut ainsi communiquer avec son portable sur tout le réseau des mobiles à 9600 bits/s.

Transmission radio

- Les gouvernements sont en général le régulateur de l'utilisation des bandes de fréquences,
- ils proposent des bandes de fréquence pour une utilisation libre, c'est-à-dire ne nécessitant pas de licence de radiocommunication.
- Les organismes chargés de réguler l'utilisation des fréquences radio sont :
  - l'ETSI (*European Telecommunications Standards Institute*) en Europe
  - la FCC (*Federal Communications Commission*) aux Etats-Unis
  - le MKK (*Kensa-kentei Kyokai*) au Japon

Fréquences à travers le monde

- Les systèmes exploitant une bande non complète ne pourront pas communiquer avec les autres.
- En 1985 les Etats-Unis ont libéré 3 bandes de fréquence à destination de l'Industrie, de la Science et de la Médecine. Ces bandes de fréquence, **baptisées *ISM*** (*Industrial, Scientific, and Medical*), sont les bandes *902-928 MHz*, *2.400-2.4835 GHz*, *5.725-5.850 GHz*.
- En Europe la bande s'étalant de *890 à 915 MHz* est utilisée pour les communications mobiles (*GSM*), ainsi seules les bandes *2.400 à 2.4835 GHz* et *5.725 à 5.850 GHz* sont disponibles pour une utilisation radio-amateur.

Pays	Bande de fréquence	Canaux	Nombre
USA & Europe	2400 - 2483,5 MHZ	f = 2402 + k MHZ	k = 0, .. , 78
Japon	2471 - 2497 MHZ	f = 2473 + k MHZ	k = 0, .. , 22
France	2446,5 - 2483,5 MHz	f = 2454 + k MHz	k = 0, .. , 22
Espagne	2445 - 2475 MHz	f = 2449 + k MHz	k = 0, .. , 22

Transmission radio

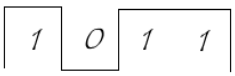
- Contraintes :
  - Le partage de la bande passante entre les différentes stations présentes dans une même cellule.
  - La propagation par des chemins multiples d'une onde radio (différentes direction, réfléchie ou réfractés réception multiples de mêmes informations ayant emprunté des cheminements différents.
- Plusieurs techniques de transmission permettent de limiter les problèmes dûs aux interférences :
  - La technique de l'étalement de spectre à saut de fréquence
  - La technique de l'étalement de spectre à séquence directe

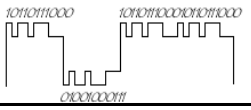
FHSS (*Frequency Hopping Spread Spectrum)*

- *Etalement de spectre par saut de fréquence*
  - découper la large bande de fréquence en un minimum de 75 canaux (*hops* ou *sauts* d'une largeur de 1MHz)
  - transmettre en utilisant une combinaison de canaux connue de toutes les stations de la cellule.
  - Dans la norme 802.11, la bande de fréquence *2.4 - 2.4835 GHz* permet de créer 79 canaux de 1 MHz.
  - La transmission se fait ainsi en émettant successivement sur un canal puis sur un autre pendant une courte période de temps (400 ms)
  - originalement conçue dans un but militaire afin d'empêcher l'écoute.
  - réduire les interférences entre les transmissions des stations d'1 cellule.

DSSS (*Direct Sequence Spread Spectrum*)

- ***Etalement de spectre à séquence directe***
  - transmettre pour chaque bit une séquence *Barker* (*bruit pseudo-aléatoire*) de bits.
  - chaque bit valant 1 est remplacé par une séquence de bits et chaque bit valant 0 par son complément.
  - La couche physique de la norme 802.11 définit une séquence de 11 bits (*10110111000*) pour représenter un 1 et son complément (*01001000111*) pour coder un 0.
  - On appelle *chip* ou *chipping code* (en français *puce*) chaque bit encodé à l'aide de la séquence. Cette technique (appelée *chipping*) revient donc à moduler chaque bit avec la séquence *barker*.





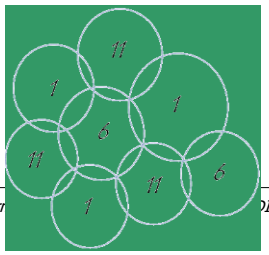
DSSS (*Direct Sequence Spread Spectrum*)

Chipping = information redondante = contrôle/correction d'erreurs

Dans le standard 802.11b, la bande de fréquence *ISM* a été découpée en 14 canaux séparés de 5MHz, dont seuls les 11 premiers sont utilisables aux Etats-Unis. Seuls les canaux 10 à 13 sont utilisables en France.

Certains canaux recouvrent partiellement les canaux adjacents.

Si deux points d'accès utilisant les mêmes canaux ont des zones d'émission qui se recoupent, des distorsions du signal risquent de perturber la transmission →



Transmission Radio

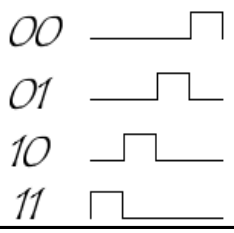
- Les moyens homologués en Europe :
  - bande de fréquence de **2.4 à 2.4835 Ghz**
  - puissance d'émission n'excède pas **100 mW**

→ limitation considérable de la portée : Entre 80 et 150m en milieu fermé et 300m en milieu ouvert

- La vitesse de transmission est typiquement de 54 Mbits/s
- La distance de transmission dépend fortement du milieu et du gain des antennes (Pb fours micro-ondes)

Transmission Infrarouge

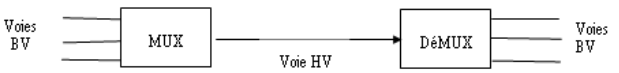
- une onde lumineuse en "vue directe" ou par réflexion.
- Le caractère non dispersif des ondes lumineuses offre un niveau de sécurité plus élevé.
- Il est possible grâce à la technologie infrarouge d'obtenir des débits allant de 1 à 2 Mbit/s en utilisant une modulation appelé **PPM** (*pulse position modulation*).
- La modulation *PPM* consiste à coder l'information suivant la position de l'impulsion.
- Le débit de 1 Mbps est obtenu avec une modulation de *16-PPM*, le débit de 2 Mbps est obtenu avec une modulation *4-PPM* :



Eléments de comparaison

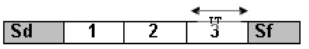
Support	Paire torsadée	Onde Radio	Coaxial	Optique Laser	Fibre optique
Distance (m)	1-5000	50-1 000	10-10 000	0.5-30	10-10 000
Débit (Mb/s)	0.3-100	1200b/s-54Mb/s	3-100	0.05-10	1-1000
Coût du Nœud ($)	10-30	50-100	30-50	20-75	75-200
Coût Installation	Faible		Moyen		Fort

Le multiplexage temporel par caractère consiste à partager dans le temps le débit binaire d'une voie Haute-Vitesse entre plusieurs voies Basse-Vitesse (canaux). La somme des débits BV ne peut dépasser le débit de la voie HV.



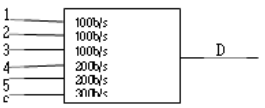
L'efficacité du multiplexage se définit par $E = \sum_{i=1}^{n} \frac{Ci.Ni}{D}$ ou Ci : débit voie BV n°i en caractère/s, Ni : nombre bits utiles dans un caractère sur la voie n°i et D : débit binaire de la voie HV.

La trame sur la voie HV est décomposée en intervalles de temps. La structure de la trame est construite en fonction du multiplexage à réaliser. En général, on choisit 1 IT = 1 caractère.



Sd et *Sf* = Signalisation = information échangée entre MUX (information non utiles).

On désire réaliser un multiplexage de 6 voies BV comme décrit ci-dessous. Les caractères émis sur les voies BV sont constitués de 10 bits = 8 bits de données + 1start bit + 1stop bit.



1. Proposez différentes solutions d'affectation des IT et comparez les débits binaires qu'elles nécessitent. Calculer leur efficacité.
2. L'IT de signalisation est affectée cycliquement à chacune des voies (voie HV = voie 0). Quel temps sépare 2 signalisations successives d'une même voie (pour la meilleure solution).

THAT 'S ALL ...

Bibliographie

- **Clavier, J., Niquil, M., Coffinet, G., Behr, F. (1972) Théorie et technique de la transmission des données, Masson et Cie, 1972**
- **Marsault, X. (1992) Compression et cryptage en informatique, Hermès, Paris, 1992**
- **Stinson, D. (1996) Cryptographie : théorie et pratique, International Thomson Publishing, Paris, 1996. 394 p.**
- **Technologie des ordinateurs et des réseaux. P.A. Goupille. Dunod. 2004.**
- **Réseaux Locaux: Normes & Protocoles. Pierre Rolin. Hermès. 1993.**
- **Réseaux locaux informatiques. Guy Pujolle, M. Schwartz. Eyrolles. 1994.**