TIC (Incomplet)

Chapitre 1 : Communication Numérique

Remarques importantes :

$$\delta(t-t_0) \otimes f(t) = f(t-t_0)$$

$$\delta(t-t_0).f(t) = f(t_0).\delta(t-t_0)$$

Translation en temps : $TF(x(t+t_0)) = X(f)e^{2j\pi t_0 f}$

Translation en fréquence : $TF(x(t)e^{2j\pi f_0 t}) = X(f-f_0)$

Dérivation en temps : $TF(\frac{dx(t)}{dt}) = 2j\pi f\, X(f)$;

$$TF(\frac{d^m x(t)}{dt}) = (2j\pi f)^m\, X(f)$$

Dilatation ou compression en temps : $TF(x(at)) = \frac{1}{|a|}X(\frac{f}{a}$

Inversion du temps : $TF(x(-t)) = X(-f)$

Conjugaison : $TF(x^*(t)) = X^*(-f)$

Dualité : soit $X(f) = TF(x(t))$ alors on a :

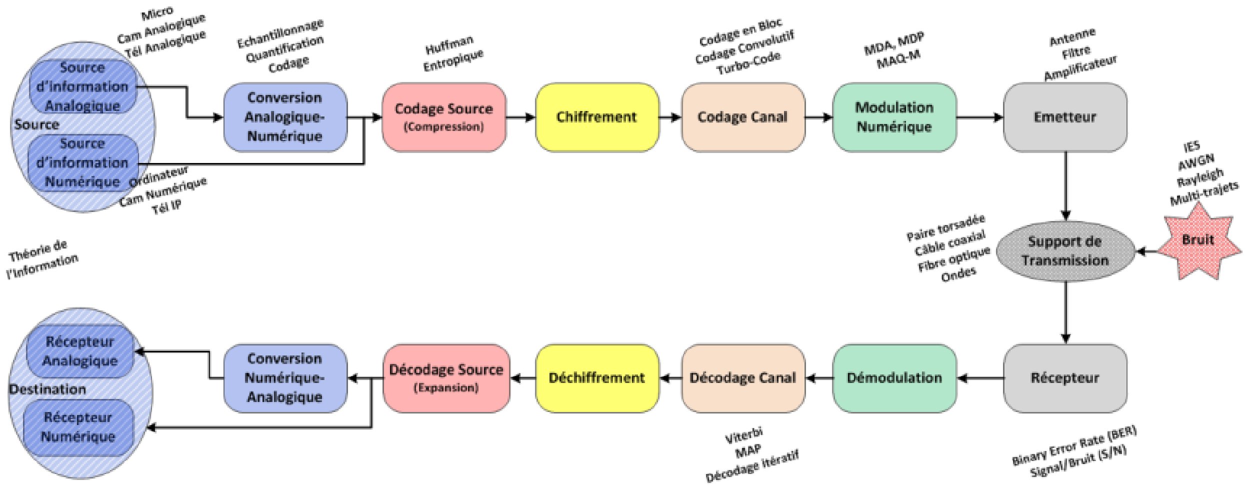
$$TF(X(t)) = x(-f) \text{ et } TF(X(-t)) = x(f)$$

TF(Acos(2pif$_0$t)) = A/2(Dirac(f-f$_0$) + Dirac(f+f$_0$))

TF(Asin(2pif$_0$t)) =A/2j(Dirac(f-f$_0$) - Dirac(f+f$_0$))

Cos(a+b) = cosa cosb - sina sinb…

Chaine de transmission:



Avantages de la transsmission numérique:

-Meilleure qualité de transmission: Une information analogique ne sera jamais retrouvé intégralement.

-Codage de source : Compression des données.

-Codage Canal : Codage correcteur et/ou detecteur d'erreurs.

-Chiffrement : Coder l'information par une clé secréte pour qu'elle ne puisse etre dechiffré que par les détenteurs de la clé.

Conversion analogique numérique:

-Echantillonnage : (REVOIR EXERCICE COURS ET CORRECTION)

Théoreme de Shanon :

Fe >= 2 fmax

-Quantification

-Codage

Chapitre 2 : Codage Source

Entropie d'une source :

$$H(X) = \sum_{i=1}^{M} p_i \log_2 \left(\frac{1}{p_i} \right).$$

Taux de codage source :

$$R_s = \frac{n}{log_2(M)} = \frac{\sum_{i=1}^{M} p_i l_i}{\log_2(M)} \tag{8}$$

où $p_i = p(E = E_i) = p(U = U_i)$ et l_i est la longueur en nombre de bits de E_i.
Bien évidemment, plus R_s est faible, plus la compression est forte.

Théoremes :

Théorème

Le théorème de Mac Millan

Ce théorème donne une condition nécessaire que doivent vérifier les longueurs l_i des étiquettes E_i pour que le codeur soit u.d.

Un code est u.d. alors $\sum_{i=1}^{M} 2^{-l_i} \leq 1$ où M est le cardinal de la source.

Théorème

Le théorème de Kraft

Si $\sum_{i=1}^{M} 2^{-l_i} \leq 1$ alors on peut construire un code instantané dont les étiquettes ont pour longueur $\{l_i\}_i$.

Ud = uniquement déchiffrable : toute concaténation d'étiquettes ne peut être interprétée que d'une seule façon

Propriété : Un code est instantané s'il vérifie la condition du préfixe : aucune étiquette ne doit être le début d'une autre. Un code instantané n'entraine pas de retards lors du décodage du moment qu'aucune étiquette n'est le début d'une autre.

Exemple :
- $E = \{E_1 = 0,\ E_2 = 10,\ E_3 = 100,\ E_4 = 101\}$ est ambigu.
- $E = \{E_1 = 10,\ E_2 = 00,\ E_3 = 11,\ E_4 = 110\}$ est u.d.
- $E = \{E_1 = 0,\ E_2 = 10,\ E_3 = 110,\ E_4 = 111\}$ est instantané.

Condition pour que le codeur soit optimal :

$$\frac{H(U)}{\log_2(M)} \leq R_s < \frac{H(U)}{\log_2(M)} + \frac{1}{\log_2(M)}$$

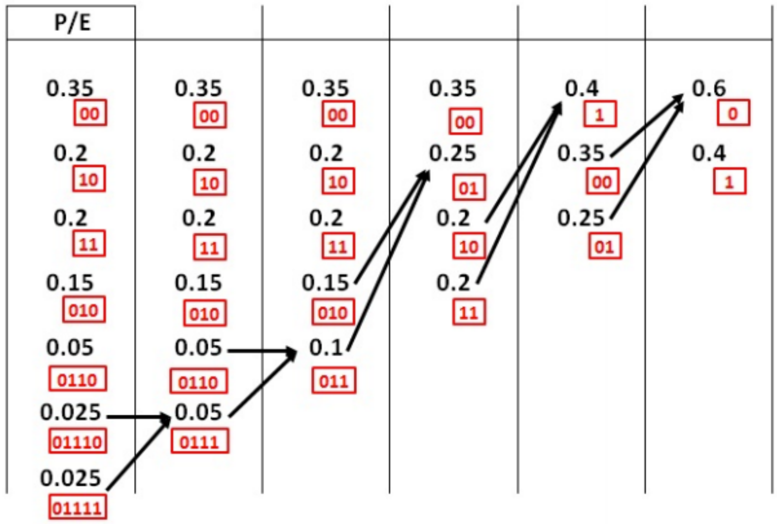
La partie a gauche de cette inégalité s'appelle limite de shanon

Exple :

On considère une source de cardinal 7 ayant la distribution de probabilité suivante :

$$p = \{0.35,\ 0.2,\ 0.2,\ 0.15,\ 0.05,\ 0.025,\ 0.025\}.$$

1- En utilisant l'algorithme de Huffman, déterminer les étiquettes E
2- Est ce que ce codeur est optimal ?
3- Vérifier si le théorème de shannon pour le codage de source est vérifié ou non.



E = {00, 10, 11, 010, 0110, 01110, 01111}

Selon les étiquettes obtenues, on a :

$$E = \{E_i\}_{1 \le i \le 7} = \{11, 01, 00, 101, 1001, 10001, 10000\}$$

$$l = \{l_i\}_{1 \le i \le 7} = \{2, 2, 2, 3, 4, 5, 5\}$$

$$p = \{p_i\}_{1 \le i \le 7} = \{0.35, 0.2, 0.2, 0.15, 0.05, 0.025, 0.025\}$$

Après calcul, on a :

$$\frac{H(U)}{\log_2(M)} = 0.538$$

$$R_s = \frac{\sum_{i=1}^{M} p_i l_i}{\log_2(M)} = 0.857$$

$$\left(\frac{H(U)}{\log_2(M)} + \frac{1}{\log_2(M)}\right) = 0.895$$

Le taux de compression obtenu $R_s = 0.857$ est optimal parceque :

$$\frac{H(U)}{\log_2(M)} \le R_s < \left(\frac{H(U)}{\log_2(M)} + \frac{1}{\log_2(M)}\right)$$

1

Chapitre 3 : Codage canal

Correction par redondance cyclique

Exemple

Au niveau de l'émission

- Prenons M = 1101011011. Le polynôme associé est noté par M(x):

$$M(x) = x^9 + x^8 + x^6 + x^4 + x^3 + x + 1$$

- C = 10011, soit $C(x) = x^4 + x + 1$
- Multiplions M par x^4 (revient à ajouter 4 "0" à la séquence M), soit:

M' = 1101011011 0000

- Divisons M'(x) par C(x) en utilisant l'opération XOR (OU exclusif)
- Le reste obtenu correspond au FCS donné par 1110
- Le message envoyé est : M' – R = 11010110111110

A la réception,

- La station effectue la division par le même CRC de la séquence entière.
- Si le reste est 0, pas d'erreur

Codage correcteur d'erreurs :

- **A l'émission**
  - $M=[m_1,\dots, m_k]$ est le mot d'information à transmettre
  - $C=[c_1,\dots, c_n]$ est le mot de code donné par :

    $$C = M\,G \quad \text{où } G=[id_k \;\; P]$$

    - G : matrice génératrice du code de dimension (k , n)
    - P : matrice de parité
- **A la réception**
  - Soit R le vecteur ligne représentant le mot de code de n éléments reçu:

    $$R = C + E$$

  - E est un vecteur ligne dont les composantes binaire représentent les éventuelles erreurs de transmission.

> • **Vecteur Erreur:** Les erreurs subies par un mot code peuvent être représentées par un vecteur $E=[\,e_1, e_2,\dots, e_n\,]$, où e_i prends la valeur 0 s'il n'y pas d'erreur sur le bit d'indice i du mot code et la valeur 1 dans le cas contraire.

Méthode de décodage

1ere étape: Calcule du vecteur syndrome S

$$S = R\,H^T = (C+E)H^T \text{ avec } H = \left[P^T \;\; Id_{n-k}\right] \text{ matrice de contrôle de parité}$$

2eme étape: Détermination des vecteurs d'erreurs *e* possibles

- Calculer la distance de Hamming minimal d_m
- Déduire les capacités de détection et de correction données respectivement par:

$$(d_m - 1) \;\; et \;\; x \le (d_m - 1)/2$$

- Déduire les vecteurs erreurs possibles de poids x

3eme étape: Détermination du vecteur erreur correspondant au syndrome S

- Relier chaque syndrome à l'erreur correspondante
- Déduire à partir du mot de code reçu le mot code envoyé

$$S = e\,H^T$$

> • **Distance de Hamming:** Étant donné deux mots de n bits m1 et m2, le nombre de bits dont ils diffèrent est appelé leur distance de Hamming.
> • **Le poids de Hamming** d'un vecteur binaire est le nombre d'éléments "1" qu'il contient.

Exemple

- A l'émission
  - Mot d'information: M= [m1 m2 m3] = [1 1 0]
  - Matrice génératrice du code: G = [Id_3 P] avec P donnée par: $P = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}$
  - Mot code C correspondant au mot information M :

    C = M G

$$C = [1 \;\; 1 \;\; 0] \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix} = [\underbrace{1 \;\; 1 \;\; 0}_{\text{Mot info}} \;\; \underbrace{0 \;\; 1 \;\; 1}_{\text{Bits de redondance}}]$$

- A la réception
  - Le mot de code reçu R : $R = [\, 1 \;\; 1 \;\; \boxed{1} \;\; 0 \;\; 1 \;\; 1 \,]$
  - On calcul le syndrome S comme: $S = [\, 1 \;\; 1 \;\; 1 \;\; 0 \;\; 1 \;\; 1 \,] \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = [\, 0 \;\; 1 \;\; 1 \,]$

Calcule de la distance de Hamming minimale d_m:

Mots informations x_1 x_2 x_3	Mots de codes x_1 x_2 x_3 a_1 a_2 a_3	Poids de Hamming
0 0 0	0 0 0 0 0 0	0
0 0 1	0 0 1 0 1 1	3
0 1 0	0 1 0 1 1 0	3
0 1 1	0 1 1 1 0 1	4
1 0 0	1 0 0 1 0 1	3
1 0 1	1 0 1 1 1 0	4
1 1 0	1 1 0 0 1 1	4
1 1 1	1 1 1 0 0 0	3

$d_m = 3$

$e \leq 1$

Dans cet exemple, le code de bloc ne peut corriger qu'une seule erreur

Relier chaque syndrome à l'erreur correspondante

Vecteur erreurs	Syndrome
1 0 0 0 0 0	1 0 1
0 1 0 0 0 0	1 1 0
0 0 1 0 0 0	0 1 1
0 0 0 1 0 0	1 0 0
0 0 0 0 1 0	0 1 0
0 0 0 0 0 1	0 0 1

Déduire à partir du mot de code reçu le mot code envoyé:

Y^T = [mot code + Vecteur erreur] mod (2)

= [1 1 1 0 1 1] + [0 0 1 0 0 0] mod (2)

= [1 1 0 0 1 1]

Chapitre 4 : Modulations numériques