

Communication
Numérique

TD4

Exo 1

* Code compact : Rappel

On peut envoyer 1, 2 ou 3 bits par caractère.
On associe à chaque caractère, un code de redondance.
Celui qui arrive souvent $\rightarrow$ code court.

$\rightarrow$ Codage à longueur variable qui permet de diminuer le débit.

* Méthode d'Huffman

Elle permet de déterminer les caractères à envoyer

$\rightarrow$ Etape 1 : association des probas aux symboles

$\rightarrow$ Etape 2 : classement par ordre de proba décroissante

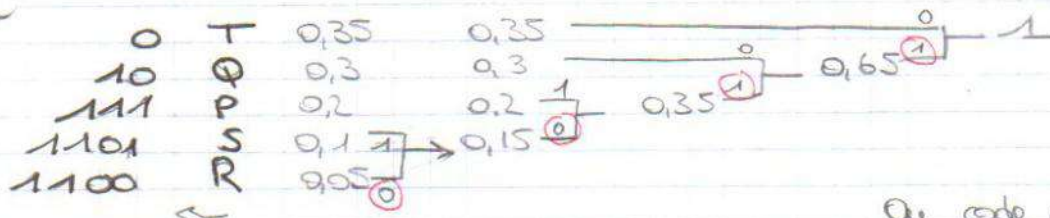
$\rightarrow$ Etape 3 : construction d'un arbre binaire en rassemblant 2 feuilles de proba les + faibles, (éventuellement classement à nouveau)

$\rightarrow$ Etape 4 : codage des branches par "0" ou "1"

$\rightarrow$ Etape 5 : lecture du code associé à une feuille depuis la racine

P Q R S T
0,2 0,3 0,05 0,1 0,35

($\rightarrow$ codage naturel : 3 bits)
(car on a 2 bits, on code 4 états de 3 bits par 5 états)



On code par un "1" la proba la + forte.

Proba	0,2	0,3	0,05	0,1	0,35
Caractère	P	Q	R	S	T
Code	111	10	1100	1101	0
nb bits	3	2	4	4	1

$$H = 3 \times 0,2 + 2 \times 0,3 + 4 \times 0,05 + 4 \times 0,1 + 1 \times 0,35 = 2,15 \text{ bits}$$

$\bar{n} = 2,15 \text{ bits} < 3 \text{ bits}$
↳ espérance.

Exemple

débit de caractère : 1 caractère / s.

Code naturel: 3 n bits / s

Code compact: 2,15 n bits / s.

Décodage

Application de la règle du préfixe

→ aucun mot binaire n'est le préfixe d'un autre.

Séquence reçue :

0 1 1 0 1 1 1 1 0 0 1 1 0 1 (fixée au hasard)
↓ ↓ ↓ ↓ ↓ ↓
T S P Q T S

Exercice 2 Code compact ternaire

Remarque: le code ternaire (+ 0 -) $\Rightarrow$ arbre à 3 branches

Généralisation: $n = m + k(m-1)$

$\left\{ \begin{array}{l} n = \text{nb de caractères} \\ m = \text{nb d'aires (3 aires} \Rightarrow \text{code ternaire)} \\ k \text{ entier} \in \mathbb{N} \end{array} \right.$

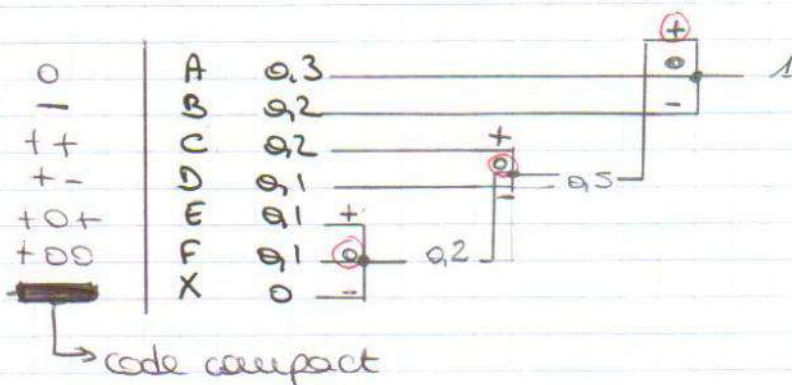
Ici, $n=6$

$$u = 3 + k(2) = 7$$

← incohérence

Solution:

ajout d'un état fictif X de proba nulle



Exercice 3 Transcodage

Hyp : source de caractères indépendants entre eux

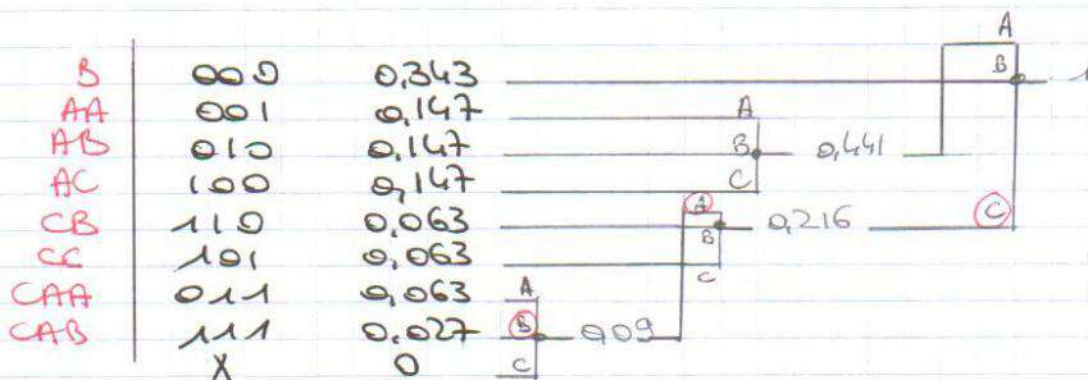
$\Rightarrow$ proba mot = produit des proba bit

	000	001	010	011	100	101	110	111
Proba	0,343	0,147	0,147	0,063	0,147	0,063	0,063	0,027

nb bits 1 2 2 3 2 2 2 3

car a a $p(0) = 0,7$ et $p(1) = 0,3$

$n = 8$
 $n = 9$ } $\rightarrow$ il faut 1 état fictif.



• Espérance : valeur moyenne / n moyen / per mot :

$$\bar{n}_{\text{mot}} = (1 \times 0,343) + 2(3 \times 0,147) + (2 \times 0,063)2 + 3 \times 0,063 + (3 \times 0,027)$$

$\underbrace{\hspace{10em}}_{\text{AA+AB+AC}}$
 $\underbrace{\hspace{10em}}_{\text{CB+CC}}$
 $\underbrace{\hspace{10em}}_{\text{CAB}}$

= 1,747 symboles (A, B, C) par mots de 3 bits

• longueur par bit :

$$\bar{n}_{\text{bit}} = \frac{\bar{n}_{\text{mot}}}{3} = 0,582 \text{ symbole / bit}$$

$$\Rightarrow R = 0,58 \text{ C} \quad \leftarrow \text{Débit (b/s)}$$

vitesse de modulation (Baud)

$\rightarrow$ + on le diminue, et + on peut réduire le nb de bits par intervalle de temps

TD4

exo 3 (212)

- limite de l'extension: (limite de Shannon)

$$\begin{aligned} \bar{n}_{\min} &= p(0) \log_m \frac{1}{p(0)} + p(1) \log_m \frac{1}{p(1)} \\ &= 0,556 \end{aligned}$$

entropie de la source

$$\log_a x = \frac{\ln x}{\ln a}$$