1 Codage de source

Aucune connaissance de la source, son rôle est de minimiser la redondance

Dilemme : Si on supprime des bits dans la source, on doit en rajouter dans le canal pour augmenter la robustesse.

1.1 Entropie

Symboles $\{a_0, a_1, a_2, \cdots, a_{n-1}\}$
Probabilités : $\{P(a_0), P(a_1), P(a_2), \cdots, P(a_{n-1})\}$ Information contenue dans un message :

$$I(a_k) = -\log_2(P(a_k)) \qquad \text{[bits]}$$

Entropie de la source (moyenne du contenu d'information):

$$H = \sum_{i=0}^{n-1} P(a_i) I(a_i)$$

1.2 Méthodes de codage

1. Longueur fixe (binaire "standard")

1.3 Décodage

Décodage instantané : Aucun mot-code n'est prefix d'un autre

Inégalité de Kraft-McMillan

$$\sum_{i=0}^{n-1} 2^{-\text{longueur}(s_i)} \neq 1 \longrightarrow \text{Pas instantané}$$

Si $= 1$ cela ne veut pas forcément dire que le code est instantané

1.4 Huffman

Attention au 1 en **haut** ou en **bas** (les exemples sont données avec le 1 en haut) (les deux sont utilisés dans le cours). Ensuite on construit l'arbre avec le nouvel élément "en haut" ou "en bas" (précisé dans l'exo en principe).

1.4.1 Longueur moyenne

$$\bar{l} = \sum P(a_k) \cdot \text{longueur du code}(a_k)$$

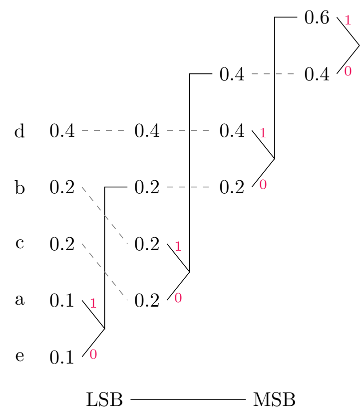
1.4.2 Variance de la longueur

$$\text{var} = \sigma^2 = \sum P(a_k) \cdot \big(\text{longueur du code}(a_k) - \bar{l}\big)^2$$

Si on doit départager deux codes, une variance plus faible est meilleure

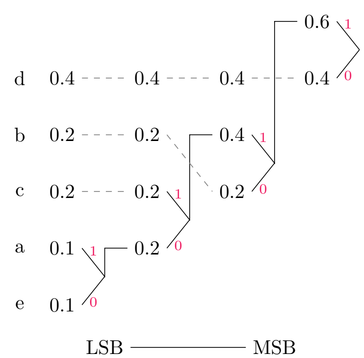
1.4.3 Huffman avec nouvel élément en haut

a	b	c	d	e
0.1	0.2	0.2	0.4	0.1



a	b	c	d	e
101	01	00	11	100

1.4.4 Huffman avec nouvel élément en bas



a	b	c	d	e
1101	10	111	0	1100

1.5 Lempel Ziv

Entrée AABABBBABAABABBBABBABB

↓

Ségmentation AA|BA|BB|BAB|AAB|AB|BBA|BBAB|B

↓

Position (Adresse)	1	2	3	4	5	6	7	8	9	10	11
Séquence	A	B	AA	BA	BB	BAB	AAB	AB	BBA	BBAB	B
Représentation			1A	2A	2B	4B	3B	1B	5A	9B	2
Code			00010	00100	00101	01001	00111	00011	01010	10011	00010