

Théorie de l'information et du codage

TD n°2 – CODAGE DE TUNSTALL

La méthode de codage décrite dans ce sujet diffère du point de vue adopté jusque ici : ce n'est plus la longueur des mots-code qui est autorisée à varier en fonction du contenu des blocs de taille fixe émis par la source, mais l'inverse. Dans tout le problème, on fixe un alphabet-source \mathcal{X} de taille $N \geq 1$, muni d'un ordre total arbitraire. On se donne également deux entiers $L \geq 1$ et $D \geq 2$.

1 Dictionnaires admissibles

Définition 1. On appelle *dictionnaire* sur l'alphabet \mathcal{X} un ensemble fini \mathcal{F} de mots non-vides de \mathcal{X}^* . Ses éléments sont appelés *lexèmes*.

Le principe général du codage *taille-variable* \rightarrow *taille-fixe* est le suivant : étant donné un dictionnaire \mathcal{F} fixé une fois pour toute, la séquence de lettres émises par la source est simplement découpée au fur et à mesure en lexèmes, puis chaque lexème est codé par le chiffre D -aire de taille L qui représente son indice lexicographique dans le dictionnaire.

Exemple 1. On désire coder les mots de l'alphabet-source $\mathcal{X} = \{a, b, c\}$ (de taille $N = 3$) à l'aide de mots-code de $L = 3$ bits chacun ($D = 2$ donc). On peut pour cela considérer le dictionnaire $\mathcal{F} = \{aaa, aab, aac, ab, ac, b, c\}$. Dans ce cas la séquence-source

$a b a a c b a a b a c a a a b c,$

pourra être découpée en :

$(a b)(a a c)(b)(a a b)(a c)(a a a)(b)(c),$

puis codée par :

011 010 101 001 100 000 101 110.

Définition 2. On dira qu'un dictionnaire \mathcal{F} sur l'alphabet \mathcal{X} est :

- **valide** si tout mot suffisamment long sur \mathcal{X} admet au moins un préfixe dans \mathcal{F} ;
- **non-ambigu** si tout mot sur \mathcal{X} admet au plus un préfixe dans \mathcal{F} ;
- **instantané** si aucun lexème n'est préfixe d'un autre lexème.

Question 1. Montrer qu'un dictionnaire est non-ambigu si et seulement s'il est instantané. Combien peut-il contenir de mots au plus si l'on veut que les séquences de mots-code D -aires de taille L obtenues soit entièrement déchiffrables ?

On ne considèrera donc désormais que des dictionnaires valides et instantanés sur \mathcal{X} de taille $|\mathcal{F}| \leq D^L$. Ces dictionnaires seront dits D^L -**admissibles**.

Question 2. Établir une bijection entre l'ensemble des dictionnaires D^L -admissibles et une certaine famille d'arbres finis que l'on définira soigneusement.

2 Facteur de compression

Dans cette partie, on fixe un dictionnaire D^L -admissible \mathcal{F} , ainsi qu'une variable aléatoire X à valeurs dans l'alphabet-source \mathcal{X} . On se donne alors une suite infinie X_1, X_2, \dots de copies indépendantes de X , et l'on note Y_1, Y_2, \dots la factorisation en lexèmes (définie récursivement) de la suite X_1, X_2, \dots .

Question 3. *Montrer que les variables aléatoires Y_1, Y_2, \dots (à valeurs dans \mathcal{F}) sont les copies indépendantes d'une même variable aléatoire Y . En déduire, lorsque le nombre n de mots-code produits tend vers l'infini, la limite $\kappa(X, \mathcal{F})$ du **facteur de compression** de la source X par le dictionnaire \mathcal{F} (nombre de lettres produites/nombre de lettres lues).*

Soit \mathcal{T} l'arbre associé au dictionnaire \mathcal{X} . Par construction, l'ensemble des sommets de \mathcal{T} peut être identifié à l'ensemble \mathcal{V} des préfixes des mots de \mathcal{F} . En particulier, la racine est le mot-vide \emptyset et les feuilles sont les lexèmes $f \in \mathcal{F}$. À tout nœud $v = v_1 \dots v_k \in \mathcal{V}$, on peut alors associer le nombre $P(v) = \mathbb{P}(X_1 = v_1, \dots, X_k = v_k)$, avec $P(\emptyset) = 1$.

Question 4. *Étant donnée une fonction de pondération arbitraire $\pi: \mathcal{V} \setminus \{\emptyset\} \rightarrow \mathbb{R}$ sur l'ensemble des sommets de \mathcal{T} (racine exceptée), on définit la hauteur pondérée $h_\pi(f)$ d'une feuille $f \in \mathcal{F}$ comme la somme des poids des nœuds le long de l'unique chemin reliant \emptyset (excluse) à f (incluse). Établir la relation :*

$$\sum_{f \in \mathcal{F}} P(f) h_\pi(f) = \sum_{v \in \mathcal{V} \setminus \{\emptyset\}} P(v) \pi(v).$$

Question 5. *En déduire :*

$$(1) \mathbb{E}[|Y|] = \sum_{v \in \mathcal{V} \setminus \mathcal{F}} P(v); \quad (2) H_D(Y) = H_D(X) \mathbb{E}[|Y|]; \quad (3) \kappa(X, \mathcal{F}) = L \frac{H_D(X)}{H_D(Y)}.$$

Quelle borne naturelle obtient-on pour le facteur de compression ?

3 Algorithme de Tunstall

Ainsi le dictionnaire D^L -admissible optimal pour une source sans-mémoire X est celui dont l'arbre maximise la somme des probabilités associées aux nœuds internes. Il est donc naturel de considérer la stratégie glotonne suivante, suggérée par Tunstall :

Algorithme 1 (Tunstall, 1968).

1. Au départ, l'arbre est constitué de la racine et des N fils de la racine.
2. Tant que le nombre de feuilles est inférieur ou égal à $D^L - (N - 1)$, choisir une feuille dont la probabilité est maximale et l'éclater en un nœud interne et $N - 1$ feuilles.

Question 6. *Construire le dictionnaire de Tunstall pour $\mathcal{X} = \{a, b\}$, $P(a) = 1 - P(b) = 0.7$, $L = 3$ et $D = 2$.*

Question 7. *Démontrer que l'algorithme de Tunstall produit un dictionnaire D^L -admissible $\mathcal{F}_{D,L}^*$ dont le facteur de compression est minimal.*

Question 8. *Démontrer que ce facteur de compression satisfait*

$$H_D(X) \leq \kappa(X, \mathcal{F}_{D,L}^*) \leq \frac{H_D(X)L}{\log_D(|\mathcal{F}_{D,L}^*| p_{min})},$$

où $p_{min} = \min_{x \in \mathcal{X}} \mathbb{P}(X = x)$, et en déduire finalement : $\kappa(X, \mathcal{F}_{D,L}^*) \xrightarrow{L \rightarrow \infty} H_D(X)$.

4 Exercice supplémentaire : codage de Shannon

On considère une source générant des lettres de l'alphabet $\mathcal{X} = \{x_1, \dots, x_k\}$ selon une loi de probabilité p . On suppose les lettres de \mathcal{X} ordonnées de telle manière que $p(x_i) \geq p(x_j)$ si $i < j$. Pour tout $i \in \{1, \dots, k\}$, on définit $S(x_i) = \sum_{j < i} p(x_j)$. On souhaite utiliser le code φ suivant : le mot $\varphi(x_i)$ codant x_i est donné par les $\lceil -\log p(x_i) \rceil$ premiers bits du développement de $S(x_i)$.

Question 9. *Montrer que le code de Shannon est instantané et que sa longueur moyenne $|\varphi|$ vérifie*

$$H(X) \leq |\varphi| \leq H(X) + 1, \text{ où } X \sim p.$$

Question 10. *Le code de Shannon est-il optimal ?*