Codage de l'information

Devoir surveillé n° 2

5 janvier 2016 - Durée 2h - Documents autorisés. Calculatrices **non autorisées**

Veuillez indiquer le numéro de votre groupe de TD sur la copie qu'il est inutile de rendre anonyme, ainsi que votre NIP (figurant sur votre carte d'étudiant).

Ce sujet contient cinq exercices indépendants. Prenez 10 mn pour lire l'intégralité du sujet avant de commencer.

Vous **justifierez** avec soin l'ensemble de vos réponses.

Il est très fortement recommandé de faire (sérieusement) ce DS avant de regarder la correction. En effet, se contenter de lire une correction, sans avoir réellement réfléchi sur les questions, n'est d'aucune aide pour comprendre un problème ou, *a fortiori*, pour réussir à savoir le résoudre soi-même.
Cette correction vous est fournie dans le seul but de montrer le type de réponses qui est attendu, ainsi que le niveau de justification qui est attendu. Il arrive que plusieurs réponses soient correctes et il est très fréquent que différents raisonnements soient corrects. Les réponses données dans cet énoncé ne sont donc pas les seules correctes. Toute question sur cet énoncé ou sur d'autres énoncés sont les bienvenues, mais il ne faut pas s'attendre à avoir une telle correction pour d'autres DS.

Exercice 1-1 *Entropie*

Question 1 Peut-on trouver une source pour laquelle l'entropie soit strictement inférieure à 1 ?

Corrigé

Pour ces deux questions, une réponse se contentant de « oui » ou de « non », n'est évidemment pas satisfaisante.
Oui on peut trouver une source dont l'entropie est strictement inférieure à 1. Par exemple une source à un seul symbole aura une entropie nulle car la quantité d'information du seul symbole est $-\log_2(1) = 0$.

Question 2 Peut-on trouver une source et un codage pour lequel la longueur moyenne sera strictement inférieure à 1 ?
Corrigé

C'est impossible car la longueur moyenne d'un codage c sur une source S est calculée par

$$\sum_{s \in S} P(s) \times |c(s)|$$

.
Or $|c(s)| > 0$ et est entier pour tout s, donc il vaut au moins 1. Ainsi au minimum la longueur moyenne sera 1 mais ne pourra pas être inférieure.

Exercice 1-2 *Représentation des entiers*

Les entiers sont généralement représentés sur un nombre fixe de bits. Néanmoins Peter Elias a proposé en 1975 plusieurs façons de représenter les entiers positifs sur un nombre variable de bits. Nous allons nous intéresser à l'une de ces représentations : le codage γ. Soit $n \in \mathbb{N}^*$ et N le nombre de minimal de bits pour la représentation binaire de n. Le résultat de $\gamma(n)$ consiste à écrire $N-1$ zéros suivis de la représentation binaire de n.

Question 1 Soit $\gamma(x) = 00000110110$, quelle est la valeur de x ?

Corrigé

$\gamma(x)$ commence par cinq zéros, donc $N = 6$. Les 6 bits suivants sont donc la représentation binaire de x.
Donc $x = \overline{110110}_2 = 2^1 + 2^2 + 2^4 + 2^5 = 2 + 4 + 16 + 32 = 54$

Question 2 Donnez le résultat de $\gamma(21)$. *Corrigé*

La représentation binaire de 21 est $\overline{10101}_2$. Alors $\gamma(21) = 000010101$.

Question 3 Soit n l'entier à encoder, quelle sera la longueur du résultat de $\gamma(n)$ en fonction de n ? *Corrigé*

La longueur de la représentation binaire de n est $\lfloor \log n \rfloor + 1$. Il y aura donc $\lfloor \log n \rfloor$ 0 suivis de $\lfloor \log n \rfloor + 1$ bits, soit $2\lfloor \log n \rfloor + 1$.

Question 4 Supposons que nous ayons le choix entre stocker les entiers sur 16 bits ou en utilisant le codage γ. Pour quels entiers le codage γ sera plus économique en place que le codage à longueur fixe ? Est-il possible de stocker certains entiers avec le codage γ qui ne pourraient être stockés avec ce codage à longueur fixe ? Et inversement ?

Corrigé

La question est finalement de savoir quand $2\lfloor \log n \rfloor + 1 < 16$ (ou que $\lfloor \log n \rfloor < 7{,}5$). C'est donc vrai jusqu'à $n = 2^8$ (exclus). De 1 à 255 le codage γ est plus économique. Les deux codages ne pourront jamais utiliser autant de place puisque le nombre de bits utilisés par le codage γ est forcément impair. Il est possible de coder n'importe quel entier strictement positif avec le codage γ ce qui n'est pas possible sur 16 bits. Mais il est possible de coder la valeur 0 sur 16 bits.

Question 5 Soit L l'ensemble des mots produits par le codage γ jusqu'à un entier n fixé, quelconque. Pourquoi L est bien un code ?

Corrigé

L est un code car il s'agit d'un langage préfixe. En effet, $\forall u, v \in L, u \neq v$ plaçons nous dans le cas où $|u| < |v|$, notons $n_u = |u|$ et $n_v = |v|$. Il y a $\lfloor n_u/2 \rfloor$ zéros suivis d'un 1 au début de u et $\lfloor n_v/2 \rfloor$ zéros suivis d'un 1 au début de v. Donc u ne peut être préfixe de v et v ne peut pas être préfixe de u puisque v est plus gran. Dans le cas où $|u| = |v|$, u ne peut pas être préfixe de v (ni l'inverse). Enfin le cas où $|u| > |v|$ est symétrique au premier cas. Donc L est un langage préfixe.

Exercice 1-3 *Langages et codes*

Soit $\mathcal{A}$ un alphabet contenant au moins deux symboles, et $c \in \mathcal{A}$ l'un de ces symboles. Nous étudions dans cet exercice le cas des langages dont la virgule n'est ni au début ni à la fin des mots du langage.

Question 1 Soient $u, v \in \mathcal{A}^*$ tels que $u \neq v$ et contenant tout deux une seule fois le symbole c, ni en première ni en dernière position.

Le langage $L = \{u, v\}$ est-il un code ?

Corrigé

On peut se souvenir qu'un langage de deux mots est un code ssi u et v ne sont pas commutants, ce qui est le cas ici : ils ne sont pas puissance d'un même mot. Il faut cependant faire la démonstration qu'un langage composé de deux mots est un code ssi les mots ne sont pas commutants.
On peut prendre une autre approche. On décompose u et v de la façon suivante : $u = u_1 c u_2$ et $v = v_1 c v_2$.

- Soit u et v sont de même taille et L est un code à longueur fixe
- Soit $u_1 \neq v_1$ et L est un code préfixe (car $u_1 c \neq v_1 c$ et c n'apparaissant qu'une seule fois, u ne pourra être préfixe de v).
- Il reste alors le cas où $u_1 = v_1$. Supposons que $|u| > |v|$. Pour avoir une double factorisation il faut que v soit préfixe de u. Et donc $u = v \cdot u'_2$. En appliquant l'algorithme de Sardinas et Patterson cela donnerait alors un ensemble $L_0 = \{u'_2\}$, $u'_2 \neq \varepsilon$. Pour avoir une double factorisation, u'_2 est nécessairement préfixe de u ou de v. Comme u'_2 ne peut pas contenir de c il est nécessairement préfixe de u_1 ou de v_1 (qui sont égaux). On a alors $u = u'_2 \cdot u'_1 c u_2$ et $v = u'_2 \cdot u'_1 c v_2$. Ce qui donne, dans le déroulement de l'algorithme de Sardinas et Patterson, un ensemble $L_1 = \{u'_1 c u_2, u'_1 c v_2\}$. À nouveau au moins un des éléments de L_1 doit être préfixe d'un mot de L. Or ni $u'_1 c u_2$ ni $u'_1 c v_2$ ne peuvent être préfixes ni de u ni de v puisqu'un seul c apparaît dans u et v et $u_1 = v_1$ et $|u_1| > |u'_1|$. Donc $L_2 = \emptyset$ et L est un code.

Question 2 Prenons maintenant un langage $L = \{u, v, w\}$ avec les mêmes contraintes sur ces trois mots qu'à la question précédente. C'est-à-dire que u, v et w contiennent tous les trois une seule occurrence d'un symbole c de l'alphabet et cette occurrence n'est ni au début ni à la fin des mots u, v ou w.

Le langage $L = \{u, v, w\}$ est-il un code ? *Corrigé*

Non ce n'est pas nécessairement un code : $L = \{\text{aaba, aba, abaa}\}$ n'est pas un code car abaa·aba=aba·aaba. Si on ne trouvait pas de contre-exemple, il était possible de dérouler le même genre de raisonnement qu'à la question précédente, ce qui nous conduisait à pouvoir construire trois mots tels que L ne soit pas un code.

Exercice 1-4 *Codages optimaux*

Question 1 Soit un langage composé de cinq mots, sur un alphabet binaire. Donnez une distribution de longueurs des mots de ce langage pour que la somme de Kraft du langage soit égale à 1. *Corrigé*

2, 2, 2, 3, 3 (par exemple) car

$$\frac{3}{2^2} + \frac{2}{2^3} = \frac{3}{4} + \frac{2}{8} = 1$$

Question 2 Donnez un exemple de code binaire **non** préfixe, composé de cinq mots, dont la somme de Kraft soit égale à 1.

Corrigé

Pour être sûr qu'il s'agisse d'un code, il faut faire l'algorithme de Sardinas et Patterson (on ne peut pas prendre un code préfixe ni, donc, un code à longueur fixe, un code à virgule à gauche n'est pas préfixe mais il ne saurait être optimal).
Par exemple $L = \{00, 11, 01, 110, 010\}$. L'algorithme de Sardinas et Patterson donne :

$$\begin{aligned} L_0 &= L^{-1} \cdot L \setminus \{\varepsilon\} \\ &= \{0\} \\ L_1 &= L_0^{-1} \cdot L \cup L^{-1} \cdot L_0 \\ &= \{0, 1, 10\} \cup \emptyset \\ &= \{0, 1, 10\} \\ L_2 &= L_1^{-1} \cdot M \cup L^{-1} \cdot L_1 \\ &= \{0, 1, 10\} \cup \emptyset \\ &= \{0, 1, 10\} \end{aligned}$$

On a donc $L_2 = L_1$. On aura donc pour tout $n \geq 2 \quad L_n = L_{n-1}^{-1} \cdot L \cup {}^{-1} \cdot L_{n-1} = L_1$ et donc pour tout $n \geq 2 \quad \varepsilon \notin L_n$. Donc L est bien un code et on a bien $K(L) = 1$.

Question 3 Soit un alphabet $\mathcal{S}$={a, b, c, d, e}. Donnez une distribution de probabilités pour l'alphabet $\mathcal{S}$ ainsi qu'un codage utilisant le code obtenu à la question précédente de manière à ce que votre codage soit optimal pour la source constituée de votre distribution de probabilités et de l'alphabet $\mathcal{S}$.

Corrigé

Pour le code L trouvé précédemment, on peut prendre la source dont les probabilités d'apparition des symboles sont :

s	a	b	c	d	e
$P(s)$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{8}$	$\frac{1}{8}$

Dans ce cas-là nul besoin de faire l'algorithme de Huffman : pour chaque symbole s on a $P(s) = 1/2^{|c(s)|}$ et donc $I(s) = |c(s)|$, alors la longueur moyenne de ce codage est égale à l'entropie. D'après le théorème du codage sans bruit, il est impossible d'avoir un codage avec une longueur moyenne plus faible. Donc ce codage est optimal.

Exercice 1-5 *Codages linéaires systématiques*

On nous donne une matrice génératrice d'un codage linéaire systématique C

$$G = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 \end{pmatrix}$$

Question 1 On souhaite transmettre le message 101, quel sera le message transmis après encodage avec C ?

Corrigé

$$\begin{pmatrix}1 & 0 & 1\end{pmatrix}\begin{pmatrix}1 & 0 & 0 & 0 & 1 & 1 & 1\\ 0 & 1 & 0 & 1 & 0 & 1 & 1\\ 0 & 0 & 1 & 1 & 1 & 0 & 1\end{pmatrix} = \begin{pmatrix}1 & 0 & 1 & 1 & 0 & 1 & 0\end{pmatrix}$$

Question 2 Donnez une matrice de contrôle correspondant à cette matrice génératrice.

Corrigé

Diverses réponses possibles. En prenant partie du fait qu'il s'agit d'un codage linéaire systématique on peut facilement trouver une matrice de contrôle : on transpose les quatre dernières colonnes de la matrice génératrice et elles constituent les trois premières colonnes de la matrice de contrôle. Les quatre colonnes restantes sont une matrice identité 4×4.

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

Question 3 Donnez les valeurs n, k et d de ce $[n, k, d]$-codage linéaire.

Corrigé

[7, 3, 4] : La matrice génératrice est de taille $n \times d$. La distance est de 4 car la somme de 4 colonnes de la matrice de contrôle est nulle.
Autre manière de trouver la distance : calculer le minimum des poids de tous les mots non nuls du code (il y en a 7).

Question 4 En déduire les capacités de détection et correction de ce codage.

Corrigé

On applique les formules du cours. Si un codage a une distance minimale d alors il est $d-1$-détecteur, ici il est donc 3-détecteur, et il est $\lfloor \frac{d-1}{2} \rfloor$-correcteur, ici 1-correcteur.

Question 5 Le mot $v = 0111011$ est reçu. Quelle conclusion peut-on tirer ?

Corrigé

$$\begin{aligned} s(v) &= v \times {}^tH \\ &= \begin{pmatrix}0 & 1 & 1 & 1 & 0 & 1 & 1\end{pmatrix}\begin{pmatrix}0 & 1 & 1 & 1\\ 1 & 0 & 1 & 1\\ 1 & 1 & 0 & 1\\ 1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{pmatrix} \\ &= \begin{pmatrix}1 & 1 & 0 & 1\end{pmatrix} \end{aligned}$$

$s(v)$ est différent de 0, le mot contient donc une ou plusieurs erreurs.
Sachant que $s(v)$ correspond à la troisième ligne de ${}^tH$, en faisant l'hypothèse qu'il n'y a qu'une seule erreur dans le mot transmis, elle aurait eu lieu en troisième position de v. Donc $v = \begin{pmatrix}0 & 1 & 0 & 1 & 0 & 1 & 1\end{pmatrix}$ et le message initialement transmis serait 010.

Question 6 Le mot $v = 1001110$ est reçu. Quelle conclusion peut-on tirer ? *Corrigé*

On suit le même raisonnement qu'à la question précédente.

$$
\begin{aligned}
s(v) &= v \times {}^{t}H \\
&= \begin{pmatrix} 1 & 0 & 0 & 1 & 1 & 1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \\
&= \begin{pmatrix} 1 & 0 & 0 & 1 \end{pmatrix}
\end{aligned}
$$

Là aussi le syndrome n'est pas nul dont le mot reçu comporte une ou plusieurs erreurs. Le syndrome n'est égal à aucune ligne de la matrice de contrôle. Il y a donc eu plusieurs erreurs dans la transmission et comme le codage est au plus 1-correcteur, il ne permet pas de toute les corriger. Nous pouvons juste dire qu'il y a au moins deux erreurs (mais il pourrait y en avoir plus).