TD 7 – Codage

Margot Catinaud margot.catinaud@lmf.cnrs.fr
Valentin Dardilhac valentin.dardilhac87@gmail.com

Dans ce TD, on considère Σ un alphabet fini.

I Codes uniquement déchiffrables

Dans cette partie, on se propose de démontrer la correction de l'algorithme de *Sardinas & Patterson* qui décide en temps polynomial si un code est uniquement déchiffrable.
Soient $u, v \in \Sigma^*$ deux mots sur l'alphabet Σ. On définit $u - v \stackrel{\text{def}}{=} \left\{u' \in \Sigma^* \mid u = v \cdot u'\right\}$. Ainsi, $u - v$ est vide ($u - v = \varnothing$) ou réduit à un seul élément. Par exemple, $\mathsf{abc} - \mathsf{a} = \{\mathsf{bc}\}$ et $\mathsf{ab} - \mathsf{ab} = \varnothing$.
Étant donné un sous-ensemble $\mathbb{S} \subseteq \Sigma^*$, on définit

$$\mathbb{T}_0 \stackrel{\text{def}}{=} \bigcup_{\substack{u,v \in \mathbb{S} \\ u \neq v}} (u - v),$$

et on définit $\mathbb{T}$ comme étant le plus petit ensemble qui contient $\mathbb{T}_0$ et qui satisfait l'inégalité

$$\bigcup_{s \in \mathbb{S},\, t \in \mathbb{T}} \big((s - t) \cup (t - s)\big) \subseteq \mathbb{T}.$$

Question 1 L'objectif de cette question est de montrer que le code $\mathcal{C} : \mathbb{X} \longrightarrow \mathbb{S}$ est uniquement déchiffrable si et seulement si $\mathbb{T}$ ne contient pas le mot vide. Pour tout entier $n \in \mathbb{N}^*$, on pose :

$$\mathbb{T}_n \stackrel{\text{def}}{=} \mathbb{T}_{n-1} \cup \bigcup_{u \in \mathbb{S}, v \in \mathbb{T}_{n-1}} \big((u - v) \cup (v - u)\big).$$

On commence par remarquer que $\mathbb{T} = \bigcup_{n \in \mathbb{N}} \mathbb{T}_n$ et que les $\mathbb{T}_n$ ne contiennent que des suffixes de mots de $\mathbb{S}$. En outre, puisque $\mathbb{T}_n \subset \mathbb{T}_{n+1}$ pour tout $n \in \mathbb{N}$, il existe un entier $n_0 \in \mathbb{N}$ tel que $\mathbb{T} = \mathbb{T}_n = \mathbb{T}_{n+1}$.

1. *Supposons qu'il existe un mot $u \in \mathbb{T}$ et deux suites $(u_i)_{i=1}^k$ et $(v_i)_{i=1}^l$ de mots de $\mathbb{S}$, pour $k, l \in \mathbb{N}$, tels que $u_1 \dots u_k = u v_1 \dots v_l$. Montrer que $\varepsilon \in \mathbb{T}$.*
2. *Supposons que, pour tout entier $n \in \mathbb{N}$, il existe un mot $u \in \mathbb{T}_n$ et deux suites $(u_i)_{i=1}^k$ et $(v_i)_{i=1}^l$ de mots de $\mathbb{S}$, pour $k, l \in \mathbb{N}$, telles que $u u_1 \dots u_k = v_1 \dots v_l$. Montrer qu'il existe un mot $v \in \mathbb{T}_0$ et deux suites $(u'_i)_{i=1}^p$ et $(v'_i)_{i=1}^m$ de mots de $\mathbb{S}$, pour $p, m \in \mathbb{N}$, telles que $v u'_1 \dots u'_p = v'_1 \dots v'_m$.*
3. *Conclure le résultat espéré.*

Question 2 *Parmi les ensembles $\mathbb{S} \subset \{0,1\}^*$ suivants, lesquels définissent des codes uniquements déchiffrables ?*

$\mathbb{S}_0 \stackrel{def}{=} \{0, 10, 11\}$ $\quad$ $\mathbb{S}_1 \stackrel{def}{=} \{0, 01, 11\}$ $\quad$ $\mathbb{S}_2 \stackrel{def}{=} \{0, 01, 10\}$ $\quad$ $\mathbb{S}_3 \stackrel{def}{=} \{0, 01\}$
$\mathbb{S}_4 \stackrel{def}{=} \{00, 01, 10, 11\}$ $\quad$ $\mathbb{S}_5 \stackrel{def}{=} \{110, 11, 10\}$ $\quad$ $\mathbb{S}_6 \stackrel{def}{=} \{110, 11, 100, 00, 10\}$

Question 3 *Donner un algorithme polynomial pour décider si un ensemble $\mathbb{S} \subseteq \Sigma^*$ définit un code uniquement déchiffrable. Estimer la complexité de l'algorithme.*

II Codage de flux de données

II.1 Découpage d'un flux infini

On veut coder un mot infini $w \in \Sigma^\omega$ par une suite de mots dans $\mathbb{S} \subset \Sigma^+$. Afin de coder tous les mots possibles, et ceci sans ambiguïté, on exige de $\mathbb{S}$ les deux propriétés suivantes

- **(Complétude)** Tout mot infini $w \in \Sigma^\omega$ admet un préfixe $v \in \mathbb{S}$;
- **(ω-déchiffrabilité unique)** $\mathbb{S}$ est *ω-uniquement déchiffrable* : Si $(u_n)_{n\in\mathbb{N}}$ et $(v_n)_{n\in\mathbb{N}}$ sont deux suites de mots de $\mathbb{S}$ telles que $u_1u_2\ldots = v_1v_2\ldots$, alors on a :

$$\forall\, n \in \mathbb{N},\ u_n = v_n.$$

Question 4 *Montrer que $\mathbb{S}$ définit ainsi un code préfixe, c'est-à-dire que pour tout mots $u, v \in \mathbb{S}$, si u est un préfixe de v, alors $u = v$.*

Dans la suite, on va s'intéresser au codage d'un flux de données décrit par une séquence $(\sigma_n)_{n\in\mathbb{N}^*}$ de lettres aléatoire (pour tout $n \in \mathbb{N}^*$, $\sigma_n \in \Sigma$). La séquence est supposée infinie, et on souhaite pouvoir coder puis décoder le message au fur et à mesure.

II.2 Code de Elias

Question 5 *On note $B_0(n) \in \{0,1\}^+$ l'écriture en base 2 d'un entier $n \in \mathbb{N}^*$ en mettant le bit de poids faible à droite. Le bit le plus à gauche étant ainsi 1.*

1. *Exprimer $|B_0(n)|$ pour tout $n \in \mathbb{N}^*$.*
2. *Donner un équivalent de cette longueur lorsque n tend vers $+\infty$.*
3. *B_0 est-il un code préfixe ?*

Question 6 *Mêmes questions pour l'ensemble $\Big\{B_1(n) \overset{def}{=} 0^{|B_0(n)|-1} \cdot B_0(n) \;\Big|\; n \in \mathbb{N}^*\Big\}$, i.e. $B_1(n)$ est le mot formé de $k \overset{def}{=} |B_0(n)| - 1$ zéros suivis du mot $B_0(n)$.*

Devoir maison 6 *Mêmes questions pour l'ensemble*

$$B_2 \overset{def}{=} \Big\{B_2(n) \overset{def}{=} B_1\big(|B_0(n)|\big) \cdot B_0(n)[2\,:\,-] \;\Big|\; n \in \mathbb{N}^*\Big\}.$$

II.3 Codage par rang

On suppose dans cette sous-partie ne pas disposer des fréquences d'apparition des lettres données en entrée. Nous allons concevoir un algorithme de compression en ligne, dont le codage change au cours du temps, afin de s'adapter aux fréquences de lettres effectivement constatées. Pour cela, on conserve à chaque instant, la liste des lettres ordonnée par date de dernière apparition. On note, pour tout $n \in \mathbb{N}$,

$$w_n \overset{\text{def}}{=} x_1 \ldots x_{\mathsf{Card}(\Sigma)} \cdot \sigma_1 \ldots \sigma_n$$

le mot lu en entrée à l'instant n, préfixé par toutes les lettres de l'alphabet d'entrée. Ainsi, toute lettre de l'alphabet Σ apparaît au moins une fois dans w_n.

Question 7 *Soit $x \in \Sigma$ une lettre quelconque, et un indice $n \in \mathbb{N}^*$. On note*

$$N_n[x] \overset{def}{=} 1 + \min\left\{\mathsf{Card}(P) \;\middle|\; P \subseteq \Sigma \ \wedge\ w_n \in \Sigma^* \cdot x \cdot P^*\right\}.$$

*Expliquer ce que représente le tableau N_n, et donner un algorithme calculant N_{n+1} à partir de N_n et de σ_{n+1} en temps linéaire (*i.e. $O(\mathsf{Card}(\Sigma))$ *opérations).*

Question 8 *On se donne une fonction $\mathcal{C} : [\![1; \mathsf{Card}(\Sigma)]\!] \longrightarrow \{0,1\}^+$ injective. On code la n-ème lettre lue σ_n grâce au mot $v_n \overset{def}{=} \mathcal{C}\big(N_{n-1}[\sigma_n]\big)$. Donner un algorithme* en ligne *lisant en entrée les mots $(v_n)_{n\in\mathbb{N}^*}$ et écrivant en sortie les lettres $(\sigma_n)_{n\in\mathbb{N}^*}$.*

Question 9 *On suppose désormais que l'algorithme lit le mot infini $v = v_1 v_2 \dots$ lettre par lettre. Quelle hypothèse supplémentaire doit-on faire sur la fonction $\mathcal{C}$?*

Question 10 *Soit $n \in \mathbb{N}^*$. On pose $\Delta_n \overset{def}{=} \min\Big\{p \in \mathbb{N}^* \;\Big|\; w_n\big[|w_n| - p\big] = \sigma_n\Big\}$. Que représente l'entier Δ_n ? Montrer que l'on a :*

$$\forall\, n \in \mathbb{N}^*,\ N_{n-1}[\sigma_n] \leqslant \Delta_n.$$

Question 11 *On suppose que la variable aléatoire*

$$\begin{array}{rccc} \sigma : & \mathbb{N}^* & \longrightarrow & \Sigma \\ & n & \longmapsto & \sigma_n \end{array}$$

est indépendante et identiquement distribuée. On fixe une lettre $a \in \Sigma$. Montrer que la limite suivante existe, puis la calculer :

$$\lim_{n \to +\infty} \mathbb{E}(\Delta_n \mid \sigma_n = a).$$

Question 12 *On prend pour cette question $\mathcal{C} \overset{def}{=} B_1$. Montrer que la longueur du code d'une lettre vérifie :*

$$\limsup_{n \to +\infty} \mathbb{E}(|v_n|) \leqslant 2\mathcal{H}(\sigma) + 1.$$

Devoir maison 7 *On prend désormais $\mathcal{C} \overset{def}{=} B_2$. Montrer que l'on a la borne supérieure suivante :*

$$\limsup_{n \to +\infty} \mathbb{E}(|v_n|) \leqslant \mathcal{H}(\sigma) + 2\log\big(1 + \mathcal{H}(\sigma)\big) + 1.$$

Question 13 *Comment améliorer le taux de compression de la question précédente ? À quel coût ?*