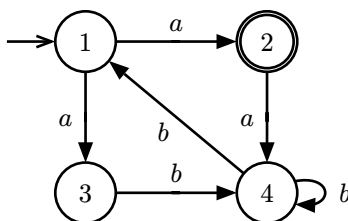


I – Algorithme de détermination

Déterminer l'automate suivant en utilisant l'algorithme du cours :



II – Clôture des langages reconnaissables

Si $m = m_1 \dots m_n$ est un mot, son miroir est $\widetilde{m} = m_n \dots m_1$. Le miroir d'un langage L est $\widetilde{L} = \{\widetilde{m} \mid m \in L\}$.

1. Montrer que le miroir d'un langage reconnaissable est reconnaissable.

Pour un langage L sur Σ , on définit :

- $\text{Pref}(L) = \{u \in \Sigma^* \mid \exists v \in \Sigma^*, uv \in L\}$, l'ensemble des préfixes ;
- $\text{Suff}(L) = \{u \in \Sigma^* \mid \exists v \in \Sigma^*, vu \in L\}$, l'ensemble des suffixes ;
- $\text{Fact}(L) = \{u \in \Sigma^* \mid \exists v, w \in \Sigma^*, vuw \in L\}$, l'ensemble des facteurs.

2. Donner des expressions régulières pour $\text{Pref}(a^*b)$ et $\text{Pref}((ab)^*)$.

3. Montrer que si L est reconnaissable, alors $\text{Pref}(L)$, $\text{Suff}(L)$ et $\text{Fact}(L)$ le sont aussi.

4. Montrer que si L est régulier, alors $\text{Pref}(L)$, $\text{Suff}(L)$ et $\text{Fact}(L)$ le sont aussi. Puisqu'on montrera que régulier et reconnaissable sont équivalents, il s'agit d'une preuve alternative.

III – Reconnaisable ou non ?

Pour chacun des langages suivants, dire s'il est reconnaissable ou non. Justifier. Pour un mot m , $|m|_a$ désigne le nombre d'occurrences de a .

1. L_1 est l'ensemble des mots sur $\{a, b\}$ sans lettres consécutives égales.
2. L_2 est l'ensemble des mots sur $\{a, b\}$ ayant un nombre pair de a et un nombre de b multiple de 3.
3. $L_3 = \{m \in \{a, b\}^* \mid |m|_a \bmod 2 = |m|_b \bmod 3\}$.
4. $L_4 = \{m \in \{a, b\}^* \mid |m|_a = |m|_b\}$.
5. L_5 est l'ensemble des écritures usuelles en base 2 des multiples de 5. Le mot 0 représente zéro ; toute autre écriture commence par 1, et le mot vide est exclu.
6. $L_6 = \{a^p \mid p \text{ est un nombre premier}\}$.

IV – Réciproque du lemme de l'étoile

Sur $\Sigma = \{a, b, c\}$, soit $L_1 = \{ab^n c^n \mid n \in \mathbb{N}\}$, $L_2 = \{a^n u \mid n \in \mathbb{N} \setminus \{1\}, u \in \{b, c\}^*\}$ et $L = L_1 \cup L_2$.

1. Soit $m \in L$ tel que $|m| \geq 2$. Montrer qu'il existe des mots x, y, z tels que :

$$m = xyz, |xy| \leq 2, y \neq \varepsilon, xy^*z \subset L.$$

2. Montrer que L n'est pas régulier. En déduire que la réciproque du lemme de l'étoile est fausse.

V – Algorithmes sur les automates

Soit $A = (\Sigma, Q, I, F, E)$ un automate à $n = |Q|$ états et $p = |E|$ transitions.

Décrire un algorithme pour résoudre chacun des problèmes suivants en précisant représentation et complexité.

1. $\mathcal{L}(A) = \emptyset$?
2. Plus petit mot de $\mathcal{L}(A)$?
3. $\mathcal{L}(A)$ fini ?
4. Pour deux automates A_1, A_2 : $\mathcal{L}(A_1) = \mathcal{L}(A_2)$?

VI – Taille discriminante

Les automates sont sans transitions ε .

1. Soit A un automate. Décrire un algorithme pour déterminer la plus petite taille d'un mot reconnu par A et préciser sa complexité.
2. Soit A un automate à n états et de langage $\mathcal{L}(A)$. Montrer que $\mathcal{L}(A) = \emptyset$ si et seulement si $\mathcal{L}(A)$ ne contient aucun mot de taille strictement inférieure à n .
3. Soient $A_1 = (\Sigma, Q_1, i_1, F_1, \delta_1)$ et $A_2 = (\Sigma, Q_2, i_2, F_2, \delta_2)$ deux ADC à n_1 et n_2 états, de langages $L_1 \neq L_2$. Notons $l(L_1, L_2)$ la plus petite taille d'un mot appartenant à l'un des deux langages mais pas à l'autre. Montrer que $l(L_1, L_2) < n_1 n_2$.

VII – Ensembles distinguants

Soit Σ un alphabet fini et $L \subset \Sigma^*$. Un ADC est un automate déterministe complet, noté $A = (Q, \Sigma, q_0, F, \delta)$. On rappelle qu'un langage est régulier si et seulement s'il est reconnu par un ADC. Pour les exemples, on fixe $\Sigma = \{a, b\}$ et $H = \mathcal{L}((a \mid b)^* ab (a \mid b)^*)$.

Un mot w est un *suffixe distinguant* pour u et v si exactement l'un des mots uw et vw appartient à L . Un ensemble $D \subset \Sigma^*$ est *distinguant* pour L si deux mots distincts de D ont toujours un suffixe distinguant.

I – Ensembles distinguants

1. Montrer que $\{\varepsilon, a, ab\}$ est distinguant pour H en donnant un suffixe distinguant pour chaque paire.
2. Montrer que, dans un ADC reconnaissant L , deux mots conduisant au même état depuis q_0 n'ont aucun suffixe distinguant. En déduire une borne sur le nombre d'états à l'aide d'un ensemble distinguant fini, puis montrer que l'existence d'un ensemble distinguant infini entraîne que L n'est pas régulier.
3. Trouver un ensemble distinguant infini pour $K = \{a^n b^n \mid n \in \mathbb{N}\}$ et en déduire que K n'est pas régulier.

Distinguer par des préfixes et des contextes. Dans les deux questions suivantes, $D \subset \Sigma^*$ est infini. On note $\widetilde{m}$ le miroir de m et $\widetilde{L} = \{\widetilde{m} \mid m \in L\}$. On rappelle que le miroir d'un langage régulier est régulier.

4. On suppose que, pour tous mots distincts $u, v \in D$, il existe un *préfixe distinguant* x tel qu'exactlyement l'un des mots xu et xv appartient à L . Montrer que L n'est pas régulier.
5. On suppose que, pour tous mots distincts $u, v \in D$, il existe un *contexte distinguant* $(x, y) \in (\Sigma^*)^2$ tel qu'exactlyement l'un des mots xuy et xvy appartient à L . Montrer que L n'est pas régulier.

Indication : dans un ADC à n états, combien de fonctions distinctes $f_u : q \mapsto \delta^*(q, u)$ les mots peuvent-ils induire ?

VIII – Automates et palindromes

ENS 2018 MP oral

On fixe un alphabet fini Σ avec $|\Sigma| > 1$. Un mot $w = a_1 \dots a_n$ est un palindrome si $a_i = a_{n-i+1}$ pour tout $1 \leq i \leq n$. On note Π le langage des palindromes et $\mathcal{L}(A)$ le langage reconnu par un automate fini A sur Σ . Les automates sont sans transitions ε .

1. Soit $\Pi_n = \Pi \cap \Sigma^n$. Montrer que, pour tout ADC A et tout $n \in \mathbb{N}$, si $\mathcal{L}(A) \cap \Sigma^{2n} = \Pi_{2n}$, alors A a au moins $|\Sigma|^n$ états.
2. En déduire que Π n'est pas régulier.
3. Étant donné un automate fini A , peut-on calculer un automate A_Π reconnaissant $\mathcal{L}(A) \cap \Pi$?
4. Pour $m = b_1 \dots b_n$, on note $\overline{m} = b_n \dots b_1$ son miroir. Étant donné A , peut-on calculer un automate A'_Π reconnaissant $\{m \in \Sigma^* \mid m\overline{m} \in \mathcal{L}(A)\}$?
5. On note $\Pi_{\text{pair}} = \bigcup_{n \in \mathbb{N}} \Pi_{2n}$. Proposer un algorithme déterminant si $\mathcal{L}(A) \cap \Pi_{\text{pair}}$ est vide, fini ou infini. Discuter ses complexités en temps et en espace.
6. Modifier l'algorithme de la question 5 pour calculer la cardinalité de $\mathcal{L}(A) \cap \Pi_{\text{pair}}$ lorsqu'elle est finie, en supposant A déterministe. Comment la complexité est-elle affectée ?
7. Modifier les algorithmes des questions 5 et 6 pour traiter $\mathcal{L}(A) \cap \Pi$.