

TD ENS

Un automate déterministe $\mathcal{A} = (\Sigma, Q, q_0, F, \delta)$ est dit synchronisant s'il existe un mot w et un état q tels que pour tout $p \in Q$, $\hat{\delta}(p, w) = q$. Pour tout $n \geq 3$, on définit l'automate $\mathcal{C}_n = ([1, n-1], \{a, b\}, 0, 1, \delta)$ où $\delta(0, a) = 1$ et $\delta(q, a) = q$ si $q \neq 0$ et $\delta(n-1, b) = 0$ et $\delta(q, b) = q+1$ sinon.

1. Montrer que le mot $(ab^{n-1})^{n-2}a$ de longueur $(n-1)^2$ synchronise $\mathcal{C}_n$.

En fait, on peut même montrer que ce mot est le plus court mot synchronisant pour $\mathcal{C}_n$. Plus généralement on appelle délai de synchronisation la longueur du plus court mot synchronisant. Cerny a émis la conjecture qu'un automate synchronisant a un délai de synchronisation d'au plus $(n-1)^2$. Cette conjecture est toujours ouverte, mais nous allons montrer une majoration moins ambitieuse. Dans la suite, pour P un ensemble d'état et w un mot, on définit $P \cdot w = \{q \in Q \mid \exists p \in P, \hat{\delta}(p, w) = q\}$

2. Donner un algorithme permettant de décider si un automate $\mathcal{A} = (Q, \Sigma, q_0, F, \delta)$ est synchronisant. On pourra utiliser l'automate $\hat{\mathcal{A}}$ des parties donné par :

$$\hat{\mathcal{A}} = (\mathcal{P}(Q), \Sigma, Q, \{P \subseteq Q \mid |P| = 1\}, \Delta)$$

où $\Delta(P, a) = P \cdot a$.

3. Montrer qu'un automate est synchronisant si, et seulement si, pour toute paire p, p' d'états, il existe un mot w tel que $\hat{\delta}(p, w) = \hat{\delta}(p', w)$.
4. En déduire un algorithme polynomial qui décide si un automate est synchronisant.
5. Soit $\mathcal{A}$ un automate synchronisant à n états et P un ensemble de ses états de cardinal k . Montrer qu'il existe w de longueur au plus $1 + \binom{n}{2} - \binom{k}{2}$ tel que $|P \cdot w| < k$. En déduire que le délai de synchronisation de $\mathcal{A}$ est au plus $(n^3 - n^2)/3$.

Solution: TD ENS

- On remarque que $\delta(i, b) = i + 1 \bmod n$ si $i > 0$. Ainsi, pour tout i , $\hat{\delta}(i, b^{n-1}) = i - 1 \bmod n$. De ce fait, $\hat{\delta}(0, ab^{n-1}) = 0$ et pour tout $i \neq 0$, $\hat{\delta}(i, ab^{n-1}) = i - 1$. Par conséquent pour tout i , $\hat{\delta}(i, ab^{n-1}) = \max(i - 1, 0)$ donc $\hat{\delta}(i, (ab^{n-1})^{n-2}) = \max(i - (n - 2), 0)$. Si $i < n - 1$ on a alors $\hat{\delta}(i, (ab^{n-1})^{n-2}) = 0$ et donc $\hat{\delta}(i, (ab^{n-1})^{n-2}a) = 1$ alors que dans le cas $i = n - 1$, on a directement $\hat{\delta}(n - 1, (ab^{n-1})^{n-2}) = 1$ ce qui nous donne bien le résultat.
- L'automate $\hat{\mathcal{A}}$ accepte exactement les mots synchronisant $\mathcal{A}$. Il suffit alors de le construire puis de parcourir son graphe.
- Le sens direct est clair. Pour la réciproque, on se donne de tels mots $w_{p,p'}$. On montre alors par récurrence sur le cardinal de $P \subseteq Q$ qu'il existe w tel que $|P \cdot w| = 1$. Pour $|P| \leq 1$ le mot vide convient. Pour $|P| \geq 2$, on prend $p \neq p' \in P$. Alors, $|(P \cdot w_{p,p'})| < |P|$ et par hypothèse de récurrence, il existe w' tel que $|(P \cdot w_{p,p'}) \cdot w'| = 1$. Il est aisé de montrer que $(P \cdot u) \cdot v = P \cdot (uv)$.
- Il suffit de restreindre l'automate $\hat{\mathcal{A}}$ aux parties de cardinal au plus 2. Puisqu'il y a $|Q|^2$ telles parties, ceci peut être fait en temps polynomial.
- Pour trouver un mot w tel que $|P \cdot w| < k$, il suffit de trouver p, p' distincts dans P et w tel que $\hat{\delta}(p, w) = \hat{\delta}(p', w)$. On cherche donc un chemin C le plus court possible dans $\hat{\mathcal{A}}$ (restreint comme précédemment aux parties de cardinal inférieur à 2 comme dans la question précédente), depuis une partie de cardinal 2 incluse dans P vers un singleton. Les états intermédiaires de ce chemin sont nécessairement des parties $\{q, q'\}$ telles que $q \neq q'$ et $\{q, q'\} \not\subseteq P$, car sinon un chemin plus court que C existe. Il y a $\binom{n}{2} - \binom{k}{2}$ telles parties donc $|C| \leq 1 + \binom{n}{2} - \binom{k}{2}$. Pour la majoration, on considère l'algorithme suivant. On pose $P = Q$, puis tant que $|P| > 1$ on pose $P = P \cdot w$ avec w un mot de taille inférieure à $1 + \binom{n}{2} - \binom{|P|}{2}$ et tel que $|P \cdot w| < |P|$. Puisque la taille de P diminue d'au moins 1 à chaque étape, le délai de synchronisation d est alors tel que :

$$\begin{aligned}
 d &\leq \sum_{k=2}^n \left(1 + \binom{n}{2} - \binom{k}{2} \right) \\
 &= (n-1) \left(1 + \frac{n(n-1)}{2} \right) - \sum_{k=2}^n \frac{k(k-1)}{2} \\
 &= n-1 + \frac{n(n-1)^2}{2} - \frac{n(n+1)(2n+1)}{12} + \frac{n(n+1)}{4} \\
 &= \frac{(n-1)(n^2-2n+3)}{3} \\
 &\leq \frac{n^3-n^2}{3} \text{ car } n \geq 2
 \end{aligned}$$

Mines-Télécom Facile

1. Donner le nombre minimal et maximal d'arêtes dans un graphe biparti.
2. Montrer qu'un graphe est biparti si et seulement si il ne contient aucun cycle de longueur impaire.
3. Donner un algorithme qui permet de déterminer si un graphe est biparti. Donner sa complexité.

Mines-Télécom Plus Dur

Soit A un automate à un seul état initial q_{init} . On notera Q l'ensemble de ses sommets, $R = Q \times \Sigma^* \times Q$ les transitions et F l'ensemble de ses états finaux. On définit la relation $\preceq$ sur $\Sigma^* \times Q \times (\Sigma \cup \{\varepsilon\}) \Sigma^*$ telle que :

$$(u, q_1, av) \preceq (u', q_2, v') \text{ si, et seulement si } (q_1, a, q_2) \in R \wedge uv = u'v'$$

On définit $\preceq^*$ comme l'union finie des $\preceq$, c'est à dire $x \preceq^* y \Leftrightarrow \exists x_1, \dots, x_n, x \preceq x_1 \preceq \dots \preceq x_n \preceq y$. C'est la clôture transitive de $\preceq$. On définit de plus $L_{\ddagger}$ le langage à saut de l'automate comme :

$$L_{\ddagger}(A) = \{uv \in \Sigma^* \mid \exists f \in F, (u, q_{init}, v) \preceq^* (\varepsilon, f, \varepsilon)\}$$

1. Montrez que $L_{a,b} = \{u \in \Sigma^* \mid |u|_a = |u|_b\}$ est reconnu par le langage à saut d'un automate. $L_{a,b}$ est-il régulier ?
2. Soit L régulier, est-ce que le langage $Perm(L) = \{u \in \Sigma^* \mid \exists \sigma \in \mathfrak{S}_{|u|}, \exists v \in L, \sigma(v) = u\}$ est régulier ?

Mines-Télécom Plus Dur

Un automate de Büchi est un automate (Q, Σ, T, I, F) permettant de reconnaître des mots infinis. Soit X un ensemble de mots. On note X^ω l'ensemble des mots infinis de X . Un mot infini x de Σ^ω est reconnu par un automate infini dans cet automate étiqueté par x , commençant par un état initial et passant une infinité de fois par un état final.

1. Déterminer le langage reconnu par l'automate de Büchi non-déterministe où $q_0 \xrightarrow{0,1} q_0, q_0 \xrightarrow{0} q_1, q_1 \xrightarrow{0} q_1, q_0$ est initial et q_1 est final.
2. Proposer un automate de Büchi reconnaissant $(01)^\omega$.
3. Soient A et B deux automates de Büchi. Montrer que $L(A) \cup L(B)$ est reconnu par un automate de Büchi.
4. Soit K un langage régulier. Montrer que $KL(A)$ est reconnu par un automate de Büchi.
5. Montrer que $L(A) \cap L(B)$ est reconnu par un automate de Büchi.

Solution: Mines-Télécom Plus Dur

On ne donne une solution que pour l'intersection, le reste fonctionnant comme pour les langages réguliers. L'intuition est que l'automate attend de passer par un état acceptant de $\mathcal{A}$, puis, il attend de visiter un état acceptant de $\mathcal{B}$. Lorsque cela se produit, il passe par un état acceptant de lui-même et il se remet en attente. Formellement : $Q_{\cap} = Q_{\mathcal{A}} \times Q_{\mathcal{B}} \times \{0, 1, 2\}$, $F_{\cap} = Q_{\mathcal{A}} \times Q_{\mathcal{B}} \times \{2\}$ et :

$$\delta_{\cap}((p, q, i), a) = \{(p', q', j) \mid p' \in \delta_{\mathcal{A}}(p, a), q' \in \delta_{\mathcal{B}}(q, a)\}, \text{ où } j = \begin{cases} 1 & \text{si } i = 0 \wedge p' \in F_{\mathcal{A}}, \text{ ou } i = 1, q' \in F_{\mathcal{B}} \\ 2 & \text{si } i = 1 \wedge q' \in F_{\mathcal{B}} \\ 0 & \text{sinon} \end{cases}$$

Type X-ENS

Dans cet exercice, on considère des graphes non vides à sommets numérotés par des entiers et possiblement avec des boucles. Un homomorphisme d'un graphe $G = (S, A)$ vers un graphe $G' = (S', A')$ est une fonction $\varphi : S \rightarrow S'$ telle que si $(x, y) \in A$, alors $(\varphi(x), \varphi(y)) \in A'$.

1. Pour les graphes G et G' ci-dessous, donnez un homomorphisme (s'il en existe un) de G vers G' . De même, s'il en existe un, donnez un homomorphisme de G' vers G .

$$G = (\llbracket 1, 5 \rrbracket, \{(1, 3), (1, 4), (2, 3), (3, 4), (4, 2), (4, 5)\}) \quad G' = (\{a, b, c, d\}, \{(a, b), (a, c), (b, c), (b, d), (c, d), (d, a)\})$$

2. Écrire le pseudocode d'un algorithme naïf qui détermine, étant donné la matrice d'adjacence de deux graphes G et G' s'il existe un homomorphisme de G vers G' et le calcul le cas échéant. Déterminer sa complexité en temps et en espace.
3. Soient G et G' deux graphes orientés et soient $G_1, \dots, G_n$ et $G'_1, \dots, G'_m$ les composantes connexes (dans leurs représentations désorientées) de G et G' . Supposons que l'on ait déterminé pour chaque $(i, j) \in \llbracket 1, n \rrbracket \times \llbracket 1, m \rrbracket$ s'il existe un homomorphisme de G_i vers G'_j . Peut-on déterminer s'il existe un homomorphisme de G vers G' .

On appelle famille de graphes un ensemble de graphes. Une famille $\mathcal{F}$ est dite close si pour tout $G \in \mathcal{F}$, s'il y a un homomorphisme de G vers G' alors $G' \in \mathcal{F}$.

4. Montrer que la famille des graphes possédant un cycle est close.
5. Donnez un graphe $G_\perp$ tel que la famille de tous les graphes est l'unique famille contenant $G_\perp$.
6. Montrez qu'une famille close non vide est nécessairement infinie.

Un graphe G est dit minimal pour une famille close si tout graphe obtenu à partir de G en supprimant une arête n'est pas dans $\mathcal{F}$.

7. Montrer que si $\mathcal{F} \neq \emptyset$ alors $\mathcal{F}$ a un graphe minimal.

Une famille close $\mathcal{F}$ est finiment engendrée s'il existe une famille finie $\mathcal{F}'$ telle que $\mathcal{F}$ est exactement l'ensemble des graphes G tel qu'il existe un homomorphisme d'un graphe de $\mathcal{F}'$ vers G .

8. Donner un exemple de famille close finiment engendrée et de famille close qui ne l'est pas.
9. Montrer que si $\mathcal{F}$ est close mais pas finiment engendrée, alors pour tout entier $n > 0$ il existe un graphe minimal pour $\mathcal{F}$ ayant au moins n arêtes.

Solution: Type X-ENS

https://anthonylick.com/wp-content/uploads/MPI_exos_xens_corrige.pdf