

Langages Unaires

Expliciter la forme des langages rationnels unaires.

Commutation

Pour tout alphabet Σ , donner l'ensemble des mots $(x, y) \in (\Sigma^*)^2$ tels que $xy = yx$

Solution: Langages Unaires

Ce sont les langages qui sont des unions finies pour $k \in \mathbb{N}$ et $p \in \mathbb{N}^*$ de langages de la forme $\{a^i\}$ pour $i \in \mathbb{N}, i < k$ et $\{a^j \mid j \in \mathbb{N}, j \geq k \wedge j \cong r \pmod{p}\}$ pour $r < p$. La preuve se fait par double inclusion. La réciproque étant triviale, on ne fait que le sens direct. On considère un automate fini déterministe complet émondé sans ε -transition. Puisque le langage est unaire, de chaque état sort une et une seule transition. Ainsi, un tel automate est une chaîne dont on note k la taille, suivie d'une boucle de taille p de laquelle aucune transition ne part, donnant la structure décrite.

Solution: Commutation

Il s'agit de l'ensemble des paires de mots telles qu'il existe $u \in \Sigma^*$ et $i, j \in \mathbb{N}$ tels que $x = u^i$ et $y = u^j$. La démonstration se fait par récurrence sur $|xy|$ en observant que si $x, y \neq \varepsilon$, alors $|x| \geq |y|$ et on peut montrer qu'alors $x = yz$ pour un certain $z \in \Sigma^*$ ce qui implique $zy = yz$ avec $|zy| < |xy|$.

Stabilité sous Opérations

Soit L un langage rationnel sur un alphabet. Montrer que les langages suivants sont rationnels :

1. $Init(L) = \{u \in \Sigma^* \mid \exists v \in \Sigma^*, uv \in L\}$
2. $Min(L) = \{w \in L \mid \nexists u \in L, u \text{ préfixe propre de } w\}$
3. $Max(L) = \{w \in L \mid \forall u \in \Sigma^*, wu \in L \Rightarrow u = \varepsilon\}$
4. $Cycle(L) = \{uv \in \Sigma^* \mid vu \in L\}$
5. $\frac{L}{2} = \{u \in \Sigma^* \mid \exists uv \in L \wedge |v| = |u|\}$

Solution: Stabilité sous Opérations

Soit $\mathcal{A} = (Q, \Sigma, q_0, F, \delta)$ un automate fini déterministe qui reconnaît L .

1. Si F' est l'ensemble des états co-accessibles dans $\mathcal{A}$, $(Q, \Sigma, q_0, F', \delta)$ reconnaît $Init(L)$.
2. On supprime toutes les transitions sortant des états de F pour obtenir un automate fini non-déterministe reconnaissant $Min(L)$.
3. On limite les états finaux de $\mathcal{A}$ aux états finaux qui ne sont pas co-accessibles autrement que par le mot vide pour obtenir un AFD qui reconnaît $Max(L)$.
4. On suppose que $\varepsilon \notin \Sigma$ et on définit $Q' = \{q_0\} \cup Q \times Q \times \{0, 1\}$. On pose $\delta' : Q' \times (\Sigma \cup \{\varepsilon\}) \rightarrow PQ'$ telle que :
 - Pour $a \in \Sigma$, $\delta'(q_0, a) = \emptyset$ et $\delta'(q_0, \varepsilon) = \cup_{q \in Q} \{(q, q, 0)\}$
 - Pour $a \in \Sigma$, $q, p \in Q$, $\delta'((q, p, 0), a) = \{(\delta(q, a), p, 0)\}$ et :

$$\delta'((q, p, 0), \varepsilon) = \begin{cases} \emptyset & \text{si } q \notin F \\ (q_0, p, 1) & \text{sinon} \end{cases}$$

- Pour tout $a \in \Sigma$, $q, p \in Q$, $\delta'((q, p, 1), a) = \{(\delta(q, a), p, 1)\}$ et $\delta'((q, p, 1), \varepsilon) = \emptyset$.

L'automate $(Q', \Sigma, q_0, \cup_{q \in Q} \{(q, q, 1)\}, \delta')$ reconnaît alors $Cycle(L)$.

5. On définit $Q' = \{q_0\} \cup Q^3$. On définit $\delta' : Q' \times (\Sigma \cup \{\varepsilon\}) \rightarrow PQ'$ de la manière suivante :
 - Pour $a \in \Sigma$, $\delta'(q_0, a) = \emptyset$ et $\delta'(q_0, \varepsilon) = \cup_{q \in Q} \{(q_0, q, q)\}$
 - Pour $a \in \Sigma$, $(q, p, r) \in Q^3$, $\delta'((q, p, r), a) = \cup_{b \in \Sigma} \{(\delta(q, a), p, \delta(r, b))\}$ et $\delta'((q, p, r), \varepsilon) = \emptyset$.
- L'AFN $(Q', \Sigma, q_0, \cup_{(q,r) \in Q \times F} \{(q, q, r)\}, \delta')$ reconnaît alors $\frac{L}{2}$.

Un opérateur sur $L(\Sigma^*)^2$

On s'intéresse à un alphabet $\Sigma = \{a, b\}$. Soit L, K deux langages. On définit $L \triangleright K$ par :

$$L \triangleright K = \left\{ uvw \mid (u, v, w) \in (X^*)^3, uw \in K, v \in L \right\}$$

1. Déterminer les langages suivants :

$$a^* \triangleright b^* \qquad (ba)^* \triangleright (ab)^* \qquad a^* \triangleright \left\{ a^p b a^q \mid (p, q) \in \mathbb{N}^2, p \leq q \right\}$$

2. Soit L, K_1, K_2 trois langages. Exprimer à l'aide de $K_1, K_2, L \triangleright K_2, L \triangleright K_2$ les langages :

$$L \triangleright (K_1 + K_2) \qquad L \triangleright (K_1 K_2) \qquad L \triangleright (K_1^*)$$

3. Si L et K sont deux langages réguliers, montrer que $L \triangleright K$ est régulier.
4. Construire un automate reconnaissant $L \triangleright K$.

Solution: Un opérateur sur $L(\Sigma^*)^2$

Cet exercice vient du début du sujet 2023 d'Option Informatique du Concours Centrale.

1. On a :

- $a^* \triangleright b^* = b^* a^* b^*$
- $(ba)^* \triangleright (ab)^* = (ab)^* (ba)^* (ab)^*$
- $a^* \triangleright \{a^p b a^q \mid (p, q) \in \mathbb{N}^2, p \leq q\} = a^* b a^*$

2. On a :

- $L \triangleright (K_1 + K_2) = (L \triangleright K_1) + (L \triangleright K_2)$
- $L \triangleright (K_1 K_2) = (K_1 \cdot (L \triangleright K_2)) + ((L \triangleright K_1) \cdot K_2)$
- $L \triangleright K_1^* = L + K_1^* (L \triangleright K_1) K_1^*$

3. On procède par induction structurelle sur K :

- Si $K = \emptyset$, $L \triangleright K = \emptyset$.
- Si $K = \{\varepsilon\}$ alors $L \triangleright K = L$ est régulier.
- Si $K = \{c\}$ avec $c \in \{a, b\}$ alors $L \triangleright K = c \cdot L \cup L \cdot c$ est régulier.
- Les opérateurs d'union, de concaténation et d'étoile se traitent en se basant sur la question précédente.

4. Je n'ai pas envie de taper ceci, trouvez un corrigé en ligne. Il suffit de faire l'élimination des ε -transitions sur la réponse de la question 5.