Exercice 1 : langages vers expressions régulières et automates

Considérons l'alphabet $\Sigma = \{a, b\}$. Pour chacun des langages suivants sur Σ, donner une expression régulière qui le dénote et un automate qui le reconnait.

1. L'ensemble des mots qui contiennent le facteur aab.
2. L'ensemble des mots qui ne contiennent pas de b.
3. L'ensemble des mots qui ne sont pas $\{a, aa, aaa\}$.
4. L'ensemble des mots dont la longueur est soit multiple de 3, soit multiple de 5.
5. L'ensemble des mots w tels que $|w|_a - |w|_b \equiv 1[3]$ (*i.e.* le nombre de a moins le nombre de b est congru à 1 modulo 3).

Exercice 2 : Régularité d'un langage

Prouver que les langages suivants ne sont pas réguliers.

1. $\{a^i b^j \mid i, j \in \mathbb{N} \wedge i \geq j\}$
2. $\{w \cdot w \mid w \in \{a, b\}^*\}$
3. $\{a^{2^n} \mid n \in \mathbb{N}\}$

Bonus : Montrer qu'aucun sous ensemble infini de $\{a^n b^n \mid n \in \mathbb{N}\}$ n'est régulier.

Exercice 3 : langages réguliers et automates

Produisez une expression régulière pour les langages suivants. Donner les étapes vous ayant permis de la construire.

— $L_1 = mirroir(\overline{(a+b)^* abb(a+b)^*})$

— $L_2 = \overline{(a+b)^* a^2 \overline{a^*}}$

Rappel : le mirroir d'un langage est le langage obtenu en inversant l'ordre des lettres ($mirroir(abcd) = dcba$) et $\overline{L}$ est le langage complément de L ($\Sigma^* \setminus L$).

On attend ici que vous utilisiez le lien entre automate et langage régulier vu lundi 22 Janvier.

Exercice 4 : Complexité de la déterminisation

— Construisez un automate non-déterministe simple acceptant le langage $(a+b)^* a(a+b)$.

— Construisez ensuite un automate déterministe pour ce langage, ainsi que pour $(a+b)^* a(a+b)^2$.

— Combien d'états possèderait votre automate non-déterministe s'il était étendu au langage $(a+b)^* a(a+b)^n$ (avec $n \in \mathbb{N}$). Combien d'états possèderait sa version déterministe ?

— Bonus : prouvez qu'il n'existe pas d'automate déterministe acceptant $(a+b)^* a(a+b)^n$ ayant moins de 2^n états.

Exercice 5 : Minimisation d'un automate

Soit L un langage et w un mot. On dénote par $w^{-1}L$ le langage des suffixes u tel que $wu \in L$. Par exemple, $(ab)^{-1}(a^*b^*) = b^*$.

— Calculez les expressions rationnelles des langages suivants : $\varepsilon^{-1}(ab^*(ab)^*), a^{-1}(ab^*(ab)^*), aab^{-1}(ab^*(ab)^*)$.

— Montrez que pour $u, v \in \Sigma^*$ et $L \subseteq \Sigma^*$, $u^{-1}(v^{-1}L) = (vu)^{-1}L$.

— Utilisant le résultat ci-dessus, listez l'ensemble des expressions rationelles possibles représentant des langages de la forme $w^{-1}(ab^*(ab)^*)$.

— Construisez l'automate A dont les états Q sont les éléments de la liste que vous avez construite. Dont l'état initial est associé à L. Dont les états finaux sont ceux associés à un langage contenant ε. Et qui possède une transition (L, z, L') ssi $L' = z^{-1}L$ (pour $z \in \{a, b\}$).

— Quel langage est produit par cet automate ?

— Peut-on diminuer le nombre d'états de cet automate (tout en restant complet) ?

Exercice 6 : Langages continuables et mots primitifs

On fixe un alphabet fini Σ et on suppose $|\Sigma| > 1$. Dans cet exercice, on considérera des automates sur l'alphabet Σ qui seront toujours supposés déterministes complets. Un mot non-vide $w \in \Sigma^*$ est dit primitif s'il n'existe pas de mot $u \in \Sigma^*$ et d'entier $p > 1$ tel que $w = u^p$.

1. Le mot *abaaabaa* est-il primitif ? Le mot *ababbaabbbababbab* (de longueur 17) est-il primitif ?
2. Proposer un algorithme naïf qui, étant donné un mot, détermine s'il est primitif ?
3. Donner un exemple d'un langage régulier infini qui ne contienne aucun mot primitif.
4. Donner un exemple d'un langage régulier infini qui ne contient que des mots primitifs.
5. Un langage régulier L est dit continuable s'il a la propriété suivante : pour tout $u \in \Sigma^*$, il existe $v \in \Sigma^*$ tel que $uv \in L$. Donner un exemple de langage régulier infini non continuable. Existe-t-il des langages réguliers continuables dont le complémentaire soit infini ?
6. Étant donné un automate A, proposer un algorithme pour déterminer si le langage L(A) qu'il reconnaît est continuable. Justifier sa correction.
7. (a) Montrer que tout langage régulier continuable contient une infinité de mots primitifs.

   (b) Étant donné un langage régulier continuable L, donner une borne supérieure sur la taille du plus petit mot primitif de L.

   (c) La réciproque de la question (a) est-elle vraie ?