TD 7 de Langages Formels
MIM2, 2001-2002

Laure Danthony
Eléments de correction

29 novembre 2001

1 Doublons algébriques

- Soit $L' = \{a^n b^p a^n b^p\}$. Alors $L' = L \cap a^+ b^+ a^+ b^+$, ce dernier langage étant algébrique. Si on montre que L' n'est pas algébrique, on aura donc gagné d'après un des td prédécents car l'intersection d'un langage rationnel avec un langage algébrique est algébrique. On montre que L' est algébrique à l'aide du lemme de l'étoile pour les langages algébriques en considérant le mot $a^N b^N a^N b^N$, N étant la constante du lemme de l'étoile pour le langage L'. En pompant, dans tous les cas on sort du langage.
- Montrons que le complémentaire de L est algébrique. Pour cela, on va créer un automate à pile *non déterministe* qui reconnaît $\overline{L}$ en utilisant la considération suivante : si le mot est de taille impaire, il est forcément dans le langage, sinon, il est de taille $2n$ et il existe un nombre i tel que les lettres i et $n+i$ diffèrent . Alors on empile $i-1$ lettres, on se met dans un état q_a ou q_b suivant que l'on lit a ou b, ensuite on dépile $i-1$ lettres. Ensuite, on empile $n-i$ lettres, on n'effectue pas la transition si on lit la lettre b alors qu'on a l'information que la i-ème lettre était un a. Par contre, si on lit un b, on effectue la transition et on dépile ensuite $n-i$ caractères pour se trouver en pile vide. Par le non déterminisme de l'automate on trouvera le premier i pour lequel le mot de L diffère en i et $n+i$, dans les autres cas on sera bloqué. On en déduit les transitions :
  - **Cas impair** : (i comme impair)
    - $\delta(q_0, \alpha, \gamma_0) = (q_i, \gamma_0)$,
    - $\delta(q_i, \alpha, \gamma_0) = (q_i, \alpha\gamma_0)$,
    - $\delta(q_i, n\alpha, \beta) = (q_i, \varepsilon)$, (si il existe un caractère dans la pile on dépile)
    - $\delta(q_i, \varepsilon, \gamma_0) = (q_i, \varepsilon)$.
  - **Cas pair** : (p comme pair)
    - $\delta(q_0, \alpha, \gamma_0) = (q_p, \alpha\gamma_0)$
    - $\delta(q_p \alpha, \beta) = (q_p, \alpha\beta)$
    - $\delta(q_p, a, \beta, q_a, \beta) = (q_a, \beta)$ et son symétrique pour b.
    - $\delta(q_a, \alpha, \beta) = (q_a, \varepsilon)$ et son symétrique pour b.
    - $\delta(q_a, \alpha, \gamma_0) = (q'_a, \alpha\gamma_0)$et son symétrique pour b.
    - $\delta(q'_a, \alpha, \beta) = (q'_a, \alpha\beta)$ et son symétrique pour b.
    - $\delta(q'_a, a, -) =$ rien (on veut bloquer ici) et son symétrique pour b.

- $\delta(q'_a, b\beta) = (q_d, \beta)$ et son symétrique pour b.
- $\delta(q_d, \alpha, \beta) = (q_d, \varepsilon)$.
- $\delta(q_d, \varepsilon, \gamma_0) = (q_d, \varepsilon)$.
- (on empile puis dépile jusqu'à q_a, on rempile à l'aide de q'_a et on redépile à l'aide de q_d.

Ainsi, si le mot est dans $\overline{L}$, il existe un i tels que les lettres i et $n-i$ diffèrent et il existe alors un chemin victorieux dans l'automate. La réciproque se fait de même.

2 Etoile d'un langage algébrique

3 La moitié d'un langage algébrique

1. Automate à pile reconnaissant ce langage (par pile vide) :
   - $\delta(q_0, a, \gamma_0) = (q_0, a\gamma_0)$,
   - $\delta(q_0, a, a) = (q_0, aa)$,
   - $\delta(q_0, b, a) = (q_0, \varepsilon)$,
   - $\delta(q_0, c, \gamma_0) = (q_0, ccc)$, Astuce !
   - $\delta(q_0, c, c) = (q_0, cccc)$,
   - $\delta(q_0, d, c) = (q_0, \varepsilon)$.
2. En faisant des dessins, on trouve $\frac{1}{2}L = L_1 \cup L_2 \cup L_3$ avec :
   - $L_1 = \{a^n b^i, i \leq n \text{ et i pair}\}$,
   - $L_2 = \{a^n b^n c^i, i \leq n \text{ et } n+i \text{ pair}\}$,
   - $L_3 = \{a^n b^n c^m d^i, n+i = m\}$
3. On utilise le mot $a^N b^N c^N$ et le lemme de l'étoile.

4 Mélange de langages algébriques

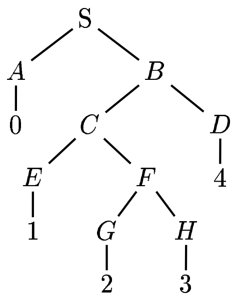
non traité.

5 Cycle d'un langage algébrique

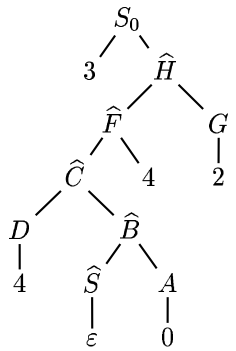
Cet exercice est très difficile. Soit $G = (V, T, P, S)$ une grammaire sous FNC (*i.e.* les productions sont uniquement de la forme $A \to BC$ et $A \to a$) engendrant le langage L. On va trouver une grammaire engendrant le langage $CYCLE(L)$.

Pour cela, considérons un arbre de dérivation pour le mot uv. On va prendre le chemin de S à la lettre la plus à gauche de v. Ensuite, on va lire ce chemin à l'envers et inverser l'ordre des productions, *i.e.* sortir les symboles sur le côté opposé d'où ils étaient à l'origine.

Faisons un exemple avec $u = 012$ et $v = 34$. On a l'arbre de dérivation de uv qui est :



La question est comment fait-on pour dériver vu ? On construit l'arbre suivant :



Alors on construit $\widehat{G} = (V \cup \{S_0\} \cup \{\widehat{A}, A \in V\}, T, \widehat{P}, S_0)$ avec $\widehat{P}$ contenant :

1. toutes les productions de P.
2. $\widehat{C} \to \widehat{A}B$ et $\widehat{B} \to C\widehat{A}$ pour $A \to BC \in P$.
3. $\widehat{S} \to \varepsilon$.
4. $S_0 \to a\widehat{A}$ si P contient $A \to a$.
5. $S_0 \to S$ (pour engendrer le langage $L \subsetneq CYCLE(L)$).

Pour montrer que $L(\widehat{G}) = CYCLE(L)$, on montre par récurrence sur la longueur de la dérivation que

$$A \vdash_G^* A_1 \dots A_n \text{ ssi } \forall i, \widehat{A_i} \vdash_{\widehat{G}}^* A_{i+1} \dots A_n \widehat{A} A_1 \dots A_{i-1}$$

Il s'ensuit alors que :

$$S \vdash_G^* A_1 \dots A_n \vdash_G^1 A_1 \dots A_{i-1} a A_{i+1} \dots A_n \text{ ssi } S_0 \vdash_{\widehat{G}}^1 a\widehat{A_i} \vdash_{\widehat{G}}^* a A_{i+1} \dots A_n \widehat{S} A_1 \dots A_{i-1}$$

et donc l'égalité des langages.

6 Forme normale

non traité.