Langages formels, calculabilité et complexité
TD4

20 octobre 2016

Exercice 1 Lemme d'Ogden

L'objectif de cet exercice est de démontrer le résultat suivant :

Lemme 1. *(Ogden)*
Soit L un langage algébrique. Il existe $N \in \mathbb{N}$ tel que $\forall z \in L$ dans lequel on marque au moins N positions distinctes, il est possible de décomposer z sous la forme $z = uxvyw$ avec :
- *x ou y contient au moins une position marquée,*
- *vxy contient au plus N positions marquées,*
- *$\forall i \geq 0, ux^i vy^i w \in L$.*

Definition 1. *Embranchement*
Soit T un sous-arbre d'un arbre de dérivation selon une grammaire CNF. On suppose marquées certaines feuilles de T. On appelle embranchement un noeud de T ayant deux fils, tles que chacun de ses fils contienne au moins une feuille marquée.

1. Soit T un sous-arbre d'un arbre de dérivation selon une grammaire CNF. On suppose qu'au moins 2^h feuilles distinctes de T ont été marquées. Montrer qu'il existe un chemin, d'une feuille à la racine, passant par au moins h embranchements et tel que pour tout i, le i-ème branchement ait au plus 2^i descendants marqués.
2. On considère une grammaire CNF engendrant le langage L. Montrer qu'il existe un entier N tel que :
   – pour tout mot $w \in L$ dans lequel on marque au moins N positions,
   – pour tout arbre de dérivation de w,
   il existe deux enbranchements b_1 et b_2 tels que
   1. b_1 est un ancêtre de b_2,
   2. b_1 est un ancêtre d'au plus N feuilles marquées,
   3. b_1 et b_2 sont étiquetés par la même variable.
3. En déduire le lemme d'Ogden.
4. On s'intéresse au langage $L = \{a^i b^j c^k d^l | i = 0 \text{ ou } j = k = l\}$.
   Montrer que pour tout $N \geq 1$ et tout mot $z \in L$ avec $|z| \geq N$, il existe une décomposition $z = uxvyw$ telle que :
   – $|xy| \geq 1$
   – $|xvy| \leq N$
   – $i\forall \geq 0$, $ux^i vy^i w \in L$.
5. Montrer que L n'est pas algébrique.

Exercice 2 Automates à pile

Construire un automate à pile pour chacun des langages algébriques suivants :

1. $\{w \in \{a,b\}^* \mid |w|_a = |w|_b\}$
2. $\{va^n \mid v \in \{a,b\}^* \text{ et } |v|_a = n\}$
3. $\{wc\tilde{w} \mid w \in \{0,1\}^*\}$;
4. $\{w\tilde{w} \mid w \in \{0,1\}^*\}$;
5. l'ensemble des palindromes non vides ;
6. le langage de Dyck D_n^*.

Exercice 3 Langage inhéremment ambigu

Soit $L = \{a^m b^m c^n \mid m,n \geq 0\} \cup \{a^m b^n c^n \mid m,n \geq 0\}$. On veut montrer qu'il n'existe pas de grammaire non-ambiguë qui engendre ce langage.

1. Donner une grammaire engendrant ce langage.
2. Déduire du lemme d'Ogden appliqué à $z = a^k b^k c^{k+k!}$ pour un k assez grand la forme des dérivations possible pour z.
3. En déduire que le mot $a^{k+k!} b^{k+k!} c^{k+k!}$ se dérive d'au moins deux manières différentes.

Exercice 4 Langage non algébrique

Pour tout langage L on définit :

$$\frac{1}{2}L = \{u \mid \exists v : uv \in L \text{ et } |u| = |v|\}$$

1. Montrer que $L_1 = \{a^n b^n c^m d^{3m}, \quad n,m \geq 1\}$ est algébrique.
2. Calculer $\frac{1}{2}L_1$
3. Montrer que $\frac{1}{2}L_1$ n'est pas est algébrique.