TD 7 : Grammaires
Automates d'arbres (suite)

Exercice 1 (**À préparer** - Construction de grammaires)**.** Donner des grammaires contextuelles reconnaissant les langages suivants :

1. $L_1 = \{a^{n^2} \mid n \in \mathbb{N}^*\}$
2. $L_2 = \{ww \mid w \in \{a,b\}^+\}$
3. $L_3 = \{a^n b^n c^n \mid n \in \mathbb{N}^*\}$

Exercice 2 (Grammaire linéaire gauche)**.** Montrer qu'un langage est rationnel si et seulement si il peut être engendré par une grammaire linéaire gauche.

Exercice 3 (Langages de Dyck)**.** Soit $\Sigma_n = \{a_1, \dots, a_n\} \cup \{\bar{a}_1, \dots, \bar{a}_n\}$ l'alphabet formé de n paires de parenthèses. Soit $G_n = (\Sigma_n, V, P_n, S)$ la grammaire définie par $S \to a_1 S \bar{a}_1 S + \dots + a_n S \bar{a}_n S + \varepsilon$. Le langage $D_n^* = \mathcal{L}_{G_n}(S)$ est appelé langage de Dyck sur n paires de parenthèses.

1. Montrer que $D_1^* = \{w \in \Sigma_1^* \mid |w|_{a_1} = |w|_{\bar{a}_1}$ et $|v|_{a_1} \geq |v|_{\bar{a}_1}$ pour tous $v \leq w\}$
2. On considère le système de réécriture (de type 0) $R_n = (\Sigma_n, P'_n)$ dont les règles sont $P'_n = \{(a_i \bar{a}_i, \varepsilon \mid 1 \leq i \leq n\}$.
   Montrer que $D_n^* = \{w \in \Sigma_n^* \mid w \xrightarrow{*} \varepsilon$ dans $R_n\}$.
3. Soit Γ un alphabet disjoint de Σ_n, $\Sigma = \Sigma_n \cup \Gamma$ et $L \subset \Sigma^*$ un langage.
   On définit la clôture $\mathrm{clot}(L) = \{v \in \Sigma^* \mid \exists w \in L, w \xrightarrow{*} v$ dans $R_n\}$.
   Montrer que si L est reconnaissable, alors $\mathrm{clot}(L)$ l'est aussi.
   On définit la réduction $\mathrm{red}(L) = \{v \in \mathrm{clot}(L) \mid v \not\to$ dans $R_n\}$.
   Montrer que si L est reconnaissable, alors $\mathrm{red}(L)$ l'est aussi.

Exercice 4 (Associativité)**.** Soit $\Sigma_2 = \{f\}$ et Σ_0 un ensemble fini de symboles d'arité 0. On note $\mathsf{A}(L)$ la clôture d'un langage L de $T(\Sigma)$ par $f(x, f(y,z)) = f(f(x,y),z)$.

1. Soient $t_1 = f(f(a, \Box), b)$ et $t_2 = f(a,b)$; montrer que $\mathsf{A}(t_1^* t_2)$ n'est pas reconnaissable.
2. Montrer que pour tout L de $T(\Sigma)$, on a $\mathrm{Fr}(L) = \mathrm{Fr}(\mathsf{A}(L))$, où $\mathrm{Fr}(L)$ dénote le langage des feuilles de L.
3. Montrer que pour tous L_1, L_2 de $T(\Sigma)$, $\mathrm{Fr}(L_1) \cap \mathrm{Fr}(L_2) = \mathrm{Fr}(\mathsf{A}(L_1) \cap \mathsf{A}(L_2))$.
4. Montrer que la famille des $\mathsf{A}(L)$ où L est un langage reconnaissable d'arbres n'est pas fermée par intersection.

Exercice 5 (Clôture ascendante et descendante)**.** On dit qu'une formule MSO $\varphi(x, y)$ définit une relation binaire R si, pour tout arbre t et pour tous noeuds u, v de t, $t, x \mapsto u, y \mapsto v \models \varphi$ ssi $(u, v) \in R$.

1. Montrer que la relation $R_1 = \{(x, y) \mid x \text{ est un descendant de } y\}$ est définissable par une formule logique MSO.
2. Soit une relation binaire R définie par une formule MSO $\varphi(x, y)$. Montrer qu'il existe une formule MSO $\varphi^{\uparrow^*}(x, y)$ qui définit la clôture reflexive et ascendante $R^{\uparrow^*}$ de R, c'est-à-dire, $(x, y) \in R^{\uparrow^*}$ si et seulement si, soit :
   – $x = y$.
   – $\exists z_1, \ldots, z_n, z_1 = x, z_n = y$ et, $\forall i < n, (z_i, z_{i+1}) \in R \cap R_1$.
3. Même question pour la clôture reflexive et descendante $R^{\downarrow^*}$ de R.
4. Qu'en est-il de la clôture reflexive et transitive R^* de R ?

Exercice 6 (Reconnaissance généralisée)**.** On considère le problème de décision suivant :

instance un terme t de $T(\mathcal{F}, \mathcal{X})$ et un automate d'arbres $\mathcal{A}$

question est-ce qu'il existe une instance close de t acceptée par $\mathcal{A}$?

1. Montrer que si t est un terme linéaire, alors ce problème est dans PTime.
2. Montrer que si t n'est pas linéaire mais $\mathcal{A}$ est déterministe, alors ce problème est NP-complet (on pourra chercher une réduction depuis SAT).