semaine du 1er février 2021

Automates et Applications
M1–MPRI Orsay

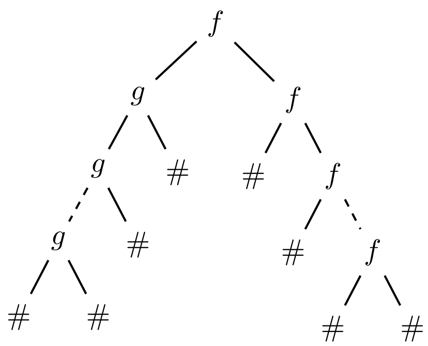
TD no 2

1 Automates

1. Donner un automate d'arbre ascendant reconnaissant le langage de tous les arbres sur $\mathcal{T}(\Sigma)$, avec $\Sigma = \{f^2, g^2, \#\}$ ayant les propriétés suivantes :
   — un nœud ayant le symbole g possède deux fils f.
   — la racine est un f.

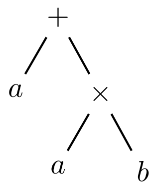
> **Réponse:** On propose l'automate $\mathcal{A} = \{\{q_0, q_1, q_2\}, \Sigma, \delta, \{q_1\}\}$ avec δ :
>
> $$\begin{array}{lcl} \# & \to & q_0 \\ f(q_0,q_0) & \to & q_1 \\ f(q_1,q_0) & \to & q_1 \\ f(q_0,q_1) & \to & q_1 \\ f(q_1,q_1) & \to & q_1 \\ g(q_1,q_1) & \to & q_2 \end{array} \quad \begin{array}{lcl} f(q_2,q_0) & \to & q_1 \\ f(q_2,q_1) & \to & q_1 \\ f(q_0,q_2) & \to & q_1 \\ f(q_1,q_2) & \to & q_1 \\ f(q_2,q_2) & \to & q_1 \end{array}$$
>
> Intuitivement, q_0 indique que l'on a vu une feuille, q_1 que l'on a vu un f et q_2 que l'on a vu un g.

2. Donner un automate d'arbre ascendant reconnaissant le langage de tous les arbres sur $\mathcal{T}(\Sigma)$, avec $\Sigma = \{f^2, g^2, \#\}$ ayant la forme suivante :



> **Réponse:** On propose l'automate $\mathcal{A} = \{\{q_0, q_1, q_2, q_3\}, \Sigma, \delta, \{q_3\}\}$ avec δ :
>
> $$\begin{array}{lcl} \# & \to & q_0 \\ g(q_0,q_0) & \to & q_1 \\ g(q_1,q_0) & \to & q_1 \\ f(q_0,q_0) & \to & q_2 \\ f(q_0,q_2) & \to & q_2 \\ f(q_1,q_2) & \to & q_3 \end{array}$$
>
> Intuitivement, q_0 indique que l'on a vu une feuille, q_1 que l'on a vu un peigne g et q_2 que l'on a vu un peigne f et q_2 demande une étiquette un nœud f avec les bons sous-arbres. Les transitions non écrites pour q_2 font que l'automate est bloqué si on lit quelque chose au dessus du f « racine ».

3. On considère $\Sigma = \{+^2, \times^2, a, b\}$, on considère le langage sur Σ des expressions arithmétiques non ambiguës. Une expression est non ambiguë si l'affichage de son arbre de syntaxe ne nécessite pas de parenthèses. Par exemple l'arbre :



Représente une expression non ambiguë. Ce langage est il reconnaissable par un automate ascendant ? Et par un automate descendant déterministe ?

Réponse: On observe que l'arbre représente une expression non ambiguë si un $+$ n'apparait jamais en dessous d'un $\times$ L'automate BU ne pose pas de difficulté. On peut reconnaître ce langage avec un automate TD det : On propose l'automate $\mathcal{A} = \{\{q_0, q_1\}, \Sigma, \delta, \{q_0\}\}$ avec δ :

$$\begin{array}{lcl} q_0, + & \to & (q_0, q_0) \\ q_0, a & \to & \{\} \\ q_0, b & \to & \{\} \\ q_0, \times & \to & (q_1, q_1) \\ q_1, \times & \to & (q_1, q_1) \\ q_1, a & \to & \{\} \\ q_1, b & \to & \{\} \end{array}$$

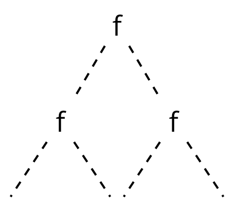
L'automate reste dans le même état tant qu'il lit des $+$, a ou b. Dès qu'il rencontre un $\times$ il change dans un nouvel état dans lequel il ne peut accepter que des $\times$ ou des a ou b.

4. Soit T un langage d'arbre. On définit l'ensemble $\mathsf{leaves}(t)$ comme le *mot* formé par tous les symboles d'arité 0 de l'arbre t rencontré dans l'ordre un parcours en profondeur d'abord. Formellement, un tel mot peut s'exprimer comme la suite $t(l_1), t(l_2), \ldots, t(l_k)$ vérifiant les propriétés :
   — $\forall i, |t(l_i)| = 0$
   — $\forall i, j, i < j \Rightarrow l_i <_{\mathsf{lex}} l_j$ où $<_{\mathsf{lex}}$ est l'ordre lexicographique sur les chemins.
   On définit ensuite $\mathsf{leaves}(T) = \{\mathsf{leaves}(T) \mid t \in T\}$. Montre que même si T est un langage d'arbre reconnaissable, $\mathsf{leaves}(T)$ n'est pas forcément un langage régulier sur les mots.

Réponse: On considère le langage $T = f(a, T, b) | g(a, b)$ donné ici par une équation récursive. Les arbres sont de la forme $f(a, f(a, f(a, ...f(a, g(a, b), b)..., b), b), b)$. Le langage $\mathsf{leaves}(T)$ est l'ensemble des mots de $\{a^n b^n \mid n \geq 1\}$. Ce dernier n'est pas régulier.

5. Montrer que le langage des arbres parfaits sur $\Sigma = \{f^2, a\}$ n'est pas reconnaissable. On rappelle qu'un arbre parfait est un arbre binaire où toutes les feuilles sont à la même profondeur.

Réponse: On veut montrer que ce langage n'est pas reconnaissable. Supposons qu'il l'est. Par le lemme de la pompe, il existe un nombre $p \geq 1$ tel que pour tout arbre du langage de hauteur au moins p, il existe C, C' des contextes, avec C' non trivial et u un sous arbre tel que $t = C[C'[u]]$ et pour tout n $C[C'^n[u]]$ est dans le langage. Considérons un arbre parfait de hauteur au moins p :



C' n'étant pas un contexte trivial, le nœud $\square$ de C' (où est enraciné u) possède un frère (gauche ou droit). Appelons $\pi 1$ le chemin menant à u et $\pi 2$ le chemin menant à son frère (on suppose que u est un fils gauche sans perte de généralité). Par définition des arbres parfaits, $t|_{\pi_1} = u$ et $t|_{\pi_2}$ ont le même nombre de nœuds N. Considérons maintenant l'arbre $C[C'[C'[u]]]$. Ce dernier n'est pas un arbre parfait. En effet, Le sous-arbre enraciné en π_2 est toujours de taille N mais le sous arbre enraciné en π_1 est $C'[u]$ dont la taille est strictement plus grande que N. L'arbre n'appartient donc pas au langage, l'hypothèse que ce dernier est reconnaissable est donc absurde.

6. Soit T un langage d'arbre sur un alphabet $\Sigma = \{f^2, a, b\}$. On considère la congruence $f(x, y) \equiv f(y, x)$. Montrer que si T est reconnaissable, l'ensemble $\{t \mid \exists t' \in Tt \equiv t'\}$ est reconnaissable.

> **Réponse:** Le langage T étant reconnaissable, il existe un automate d'arbre ascendant le reconnaissant. Pour reconnaître T', il suffit, pour chaque règle de la forme $f(q, q') \to q''$ de la transformer en $f(q', q) \to q''$. (par récurrence sur la taille de t un arbre quelconque reconnu par l'automate de T).