

Jeux, Synthèse et Contrôle

Automates d'arbres

Arbres finis

Nous appelons *arbre fini* tout arbre planaire, binaire, complet, fini, étiqueté sur un ensemble fini E . Les automates envisagés ici sont non-déterministes.

Exercice 5.1

Soit $E = \{a, b, c\}$.

1. Construisez un automate d'arbres qui reconnaît l'ensemble des arbres sur E , dont la branche la plus à droite est étiquetée par $(a + b)^*abb(a + b)^*c$.
2. Construisez un automate d'arbres qui reconnaît l'ensemble des arbres sur E , dont la frontière ne contient pas le facteur abb .
3. Plus généralement, montrer qu'étant donné un langage rationnel de mots $L \subseteq \{a, b, c\}^*$, l'ensemble des arbres sur E , dont la frontière est dans L , est un langage d'arbres reconnaissable.

Exercice 5.2

Construire un automate (fini, d'arbres) qui reconnaît l'ensemble des arbres sur $\{a, b, c\}$ dont le nombre de noeuds est pair.

Exercice 5.3

Construire des a.f. d'arbres reconnaissant les langages suivants :

1. L'ensemble des arbres dont la feuille la plus à gauche est étiquetée par a .
2. L'ensemble des arbres contenant 2 noeuds u, v étiquetés par a , tels que u est ancêtre de v .
3. L'ensemble des arbres contenant 2 noeuds u, v étiquetés par a , tels que u et v ne sont pas ancêtre l'un de l'autre.
4. L'ensemble des arbres contenant un nombre pair de noeuds étiquetés par a .

Exercice 5.4

1. Montrez que l'ensemble des arbres dont toutes les feuilles ont la même profondeur, n'est pas reconnaissable.
2. Montrez que l'ensemble des arbres dont toutes les feuilles ont une profondeur paire, est reconnaissable
3. Montrez que l'ensemble des arbres de hauteur paire, n'est pas reconnaissable.

Arbres infinis

Nous appelons *arbre infini*, étiqueté sur un ensemble fini E , toute application $t : \{g, d\}^* \rightarrow E$. Autrement dit, il s'agit des arbres planaires, binaires, complets, sans feuille, étiquetés sur un ensemble fini E . Les automates envisagés ici sont non-déterministes.

Exercice 5.5

Soit $E = \{a, b\}$. Construire des automates finis reconnaissant les langages (d'arbres infinis) suivants :

1. L'ensemble des arbres ne possédant qu'un nombre fini d'occurrences de a .
2. L'ensemble des arbres possédant un nombre infini d'occurrences de a .
3. L'ensemble des arbres contenant au moins une branche étiquetée par $(ab)^\omega$.
4. L'ensemble des arbres ne contenant aucune branche étiquetée par $(ab)^\omega$.
5. L'ensemble des arbres contenant un nombre fini de branches étiquetées par $(ab)^\omega$.
6. L'ensemble des arbres possédant au moins une branche, qui possède un nombre infini d'occurrences de a .
7. L'ensemble des arbres contenant un nombre infini de branches étiquetées par $(ab)^\omega$.

Exercice 5.6

Un arbre est dit *régulier* s'il a un nombre *fini* de classes d'isomorphismes de sous-arbres. (Il s'agit d'isomorphismes d'arbres planaires étiquetés).

Un arbre $t : \{g, d\}^* \rightarrow E$ est dit *récurif* si l'application t est récursive (i.e. calculable).

Pour chaque langage d'arbres L envisagé à l'exercice précédent, donner :

- un arbre régulier appartenant à L
- un arbre non-régulier, mais récurif, appartenant à L
- un arbre non-récurif, appartenant à L .

Exercice 5.7

Tout automate \mathcal{A} d'arbres infinis peut être transformé en un jeu de parité $\mathcal{J}(\mathcal{A})$ sur une arène finie : le joueur 0 dans la position q , choisit une étiquette $e \in E$ et une transition partant de (q, e) , tandis que le joueur 1, partant d'une transition $\delta = (q, e, q_g, q_d)$, choisit une direction g (resp. d) et aboutit dans la position q_g (resp. q_d).

1- Rappeler comment on peut décider, dans un jeu de parité fini, si un sommet est gagnant pour J_0 ou pour J_1 .

2- À quels types de stratégies du joueur 0 correspondent les arbres de $L_{\mathcal{A}}$ qui sont réguliers? récurifs?

3- Donner un algorithme qui décide, pour tout automate d'arbres infinis \mathcal{A} , si $L_{\mathcal{A}} = \emptyset$?

Exercice 5.8

Soit E un ensemble et $\perp \notin E$. Pour tout arbre fini t étiqueté sur E , on définit l'arbre infini \hat{t} étiqueté sur $E \cup \{\perp\}$ par : pour tout $u \in \{g, d\}^*$

$$\text{si } u \in \text{Dom}(t) : \hat{t}(u) := t(u), \quad \text{sinon } \hat{t}(u) := \perp.$$

1- Montrer que, si L est un langage d'arbres finis reconnaissable. alors

$$\hat{L} := \{\hat{t} \mid t \in L\}$$

est un langage d'arbres infinis reconnaissable.

2- La réciproque de cet énoncé est-elle vraie?

ANNEXE

Definition 0.1 (Automate fini d'arbres finis) Un automate fini d'arbres est un 5-uplet, $\mathcal{A} = \langle E, Q, q_I, Q_+, \delta \rangle$ où

- E est un alphabet fini
- Q est un ensemble fini, l'ensemble des états
- $q_I \in Q$ est l'état initial
- $Q_+ \subseteq Q$ est l'ensemble des états d'arrivée
- $\delta \subseteq (Q \times E \times Q \times Q) \cup (Q \times E \times Q)$ est l'ensemble des transitions.

On appelle *calcul* (ou *course*) de l'automate \mathcal{A} sur le terme t , toute application $\rho : \text{Dom}(t) \cup (\text{Dom}(t) \cdot \{0\}) \rightarrow Q$ telle que, $\forall u \in \text{Dom}(t)$:

$$\text{si } ug, ud \in \text{Dom}(t) \text{ alors } (\rho(u), t(u), \rho(ug), \rho(ud)) \in \delta \quad (1)$$

et

$$\text{si } (ug \notin \text{Dom}(t) \text{ et } ud \notin \text{Dom}(t)) \text{ alors } (\rho(u), t(u), \rho(u0)) \in \delta \quad (2)$$

N.B. Chaque transition de la forme :

$$(q, e, q_g, q_d)$$

peut être appliquée aux noeuds internes ; chaque transition de la forme :

$$(q, e, r)$$

peut être appliquée aux feuilles.

Definition 0.2 Un terme $t \in \mathcal{T}(F)$ est accepté (ou reconnu) par \mathcal{A} , ssi il existe un calcul ρ de \mathcal{A} sur t , tel que ,

$$(AC1) \rho(\varepsilon) = q_I$$

$$(AC2) \text{ pour toute feuille } u \in \text{Dom}(t), \rho(u0) \in Q_+.$$

On appelle langage accepté (ou reconnu) par \mathcal{A} , noté $L_{\mathcal{A}}$, l'ensemble des termes acceptés (ou reconnus) par \mathcal{A} .

Un langage $L \subseteq \mathcal{T}(F)$ est dit *reconnaissable* ssi il existe un automate fini (de termes) \mathcal{A} tel que $L = L_{\mathcal{A}}$.

Definition 0.3 (Automate fini d'arbres infinis) Un automate d'arbres infinis est un 5-uplet, $\mathcal{A} = \langle E, Q, q_I, \Omega, \delta \rangle$ où

— E est un alphabet

— Q est un ensemble fini, l'ensemble des états

— $q_I \in Q$ est l'état initial

— $\Omega : Q \rightarrow \mathbb{N}$ est une application (la fonction de priorité)

— $\delta \subseteq (Q \times E \times Q \times Q)$ est l'ensemble des transitions ;

On appelle *calcul* (ou *course*) de l'automate \mathcal{A} sur le terme t , toute application $\rho : \text{Dom}(t) \rightarrow Q$ telle que, $\forall u \in \text{Dom}(t)$:

$$(\rho(u), t(u), \rho(ug), \rho(ud)) \in \delta \quad (3)$$

Definition 0.4 Un terme $t \in \mathcal{T}(F)$ est accepté (ou reconnu) par \mathcal{A} , ssi il existe un calcul ρ de \mathcal{A} sur t , tel que,

(AC1) $\rho(\varepsilon) = q_I$

(AC2) pour toute branche $u_0, u_1, \dots, u_i, \dots$ de t ,

$$\max\{c \in \mathbb{N} \mid \text{Card}\{i \in \mathbb{N} \mid \Omega(\rho(u_i)) = c\} = \infty\} \equiv 0 \pmod{2}$$

On appelle langage accepté (ou reconnu) par \mathcal{A} , noté $L_{\mathcal{A}}$, l'ensemble des arbres acceptés (ou reconnus) par \mathcal{A} .

Un langage $L \subseteq \mathcal{T}(F)$ est dit *reconnaisable* ssi il existe un automate fini (d'arbres infinis) \mathcal{A} tel que $L = L_{\mathcal{A}}$.