Langages rationnels (suite)

FOND1 - ENS Rennes

6 Automates déterministes complets

Définition 25 (automate déterministe complet)**.** *Un* automate déterministe complet *est un automate fini où la relation de transition* δ *est une* ***fonction totale*** $\delta : Q \times \Sigma \to Q$.

Définition 26 (exécution d'automate déterministe complet)**.** *Soit* $\mathcal{A} = (Q, \Sigma, \delta, q_0, F)$ *un automate déterministe complet. On étend* δ *aux mots en définissant* $\delta : Q \times \Sigma^* \to Q$ *par induction sur les mots :*

- *Pour* $q \in Q$, $\delta(q, \epsilon) = q$ *;*
- *Pour* $q \in Q$, $w \in \Sigma^*$, $a \in \Sigma$, $\delta(q, wa) = \delta(\delta(q, w), a)$.

On a $L(\mathcal{A}) = \{w \in \Sigma^* \mid \delta(q_0, w) \in F\}$.

6.1 Opérations union, intersection et complémentaire

Définition 27 (automates produits)**.** *Soient* $\mathcal{A}_1 = (Q_1, \Sigma, \delta_1, q_{0_1}, F_1)$ *et* $\mathcal{A}_2 = (Q_2, \Sigma, \delta_2, q_{0_2}, F_2)$ *deux automates déterministes complets. On définit l'automate produit* $\mathcal{A}_1 \times^{\cap} \mathcal{A}_2 = (Q_1 \times Q_2, \Sigma, \delta, (q_{0_1}, q_{0_2}), F)$ *avec :*

- *Pour tout* $(q_1, q_2) \in Q_1 \times Q_2$, *pour tout* $a \in \Sigma$, $\delta((q_1, q_2), a) = (\delta_1(q_1, a), \delta_2(q_2, a))$ *;*
- $F = F_1 \times F_2$.

L'automate $\mathcal{A}_1 \times^{\cup} \mathcal{A}_2$ *est défini de façon analogue mais avec* $F = (F_1 \times Q_2) \cup (Q_1 \times F_2)$.

Proposition 28 (union et intersection)**.** *Soient* $\mathcal{A}_1 = (Q_1, \Sigma, \delta_1, q_{0_1}, F_1)$ *et* $\mathcal{A}_2 = (Q_2, \Sigma, \delta_2, q_{0_2}, F_2)$ *deux automates déterministes complets.*

- $L(\mathcal{A}_1 \times^{\cap} \mathcal{A}_2) = L(\mathcal{A}_1) \cap L(\mathcal{A}_2)$ *;*
- $L(\mathcal{A}_1 \times^{\cup} \mathcal{A}_2) = L(\mathcal{A}_1) \cup L(\mathcal{A}_2)$.

Proposition 29 (passage au complémentaire)**.** *Soit* $\mathcal{A} = (Q, \Sigma, \delta, q_0, F)$ *un* ***automate déterministe complet****. L'automate* $\overline{\mathcal{A}} = (Q, \Sigma, \delta, q_0, Q \setminus F)$ *est tel que* $L(\overline{\mathcal{A}}) = \Sigma^* \setminus L(\mathcal{A})$.

6.2 Déterminisation

Définition 30 (ϵ-clôture)**.** *Soit* $\mathcal{A} = (Q, \Sigma, \delta, q_0, F)$ *un automate (non-déterministe avec* ϵ*-transitions). Pour tout* $R \in 2^Q$*, on note*

$$E(R) := \{p \in Q \mid \exists q \in R, p \text{ est accessible depuis } q \text{ par des } \epsilon\text{-transitions}\}.$$

Définition 31 (automate déterminisé)**.** *Soit* $\mathcal{A} = (Q, \Sigma, \delta, q_0, F)$ *un automate (non-déterministe avec* ϵ*-transitions). L'automate déterminisé de* $\mathcal{A}$ *est l'automate déterministe complet* $det(\mathcal{A}) = (Q', \Sigma, \delta', q'_0, F')$ *avec :*

- $Q' = 2^Q$ *;*
- $\delta'(R, a) = E(\{q' \in Q \mid \text{ il existe } q \in R \text{ tq. } (q, a, q') \in \delta\})$ *;*
- $q'_0 = E(\{q_0\})$ *;*
- $F' = \{R \in Q' \mid R \cap F \neq \emptyset\}$.

Théorème 32. $L(\mathcal{A}) = L(det(\mathcal{A}))$.

Corollaire 33. *L est rationnel ssi il existe un automate déterministe complet $\mathcal{A}$ tel que $L = L(\mathcal{A})$.*

Corollaire 34. *$Rat(\Sigma)$ est stable par complémentaire et intersection.*

7 Type abstrait Langage rationnel

	Expressions rationnelles	Automates non-déterministes avec ϵ-transitions	Automates déterministes complets
Appartenance d'un mot à un langage L			
Tester si L est vide			
Tester si $L = \Sigma^*$			
Tester si L est fini			
$\cup$			
$\bullet$			
$*$			
$\cap$			
complémentaire			

8 Application : arithmétique de Presburger

Définition 35 (syntaxe des formules de l'arithmétique de Presburger)**.** *Le langage de l'arithmétique de Presburger est le langage engendré par la grammaire suivante :*

$$\varphi \quad ::= \quad (x = 0) \quad | \quad (x = 1) \quad | \quad (z = x + y) \quad | \quad \neg\varphi \quad | \quad (\varphi \vee \varphi) \quad | \quad \exists x \varphi$$

où x, y, z sont des symboles de variable. *$\vee$ se lit 'ou non exclusif'*

Définition 36 (variable libre)**.** *On dit qu'une variable est* libre *s'il n'est pas sous la portée d'un quantificateur $\exists$.*

Définition 37 (sémantique d'une formule)**.** *Soit φ une formule de l'arithmétique de Presburger et $\mathcal{V}$ qui contient les variables libres de φ. On définit $[\![\varphi]\!]_{\mathcal{V}}$ par induction sur φ :*

- $[\![x = 0]\!]_{\mathcal{V}} := \left\{(n_\lambda)_{\lambda \in \mathcal{V}} \in \mathbb{N}^{\mathcal{V}} \mid n_x = 0\right\}$;
- $[\![x = 1]\!]_{\mathcal{V}} := \left\{(n_\lambda)_{\lambda \in \mathcal{V}} \in \mathbb{N}^{\mathcal{V}} \mid n_x = 1\right\}$;
- $[\![z = x + y]\!]_{\mathcal{V}} := \left\{(n_\lambda)_{\lambda \in \mathcal{V}} \in \mathbb{N}^{\mathcal{V}} \mid n_z = n_x + n_y\right\}$;
- $[\![\neg\varphi]\!]_{\mathcal{V}} := \mathbb{N}^{\mathcal{V}} \setminus [\![\varphi]\!]_{\mathcal{V}}$;
- $[\![(\varphi_1 \vee \varphi_2)]\!]_{\mathcal{V}} := [\![\varphi_1]\!]_{\mathcal{V}} \cup [\![\varphi_2]\!]_{\mathcal{V}}$;
- $[\![\exists x \varphi]\!]_{\mathcal{V}} := \left\{(n_\lambda)_{\lambda \in \mathcal{V}} \in \mathbb{N}^{\mathcal{V}} \mid \textit{il existe } n_x \in \mathbb{N} \textit{ tel que } (n_\lambda)_{\lambda \in \mathcal{V} \cup \{x\}} \in [\![\varphi]\!]_{\mathcal{V} \cup \{x\}}\right\}$;

Modélisation

- On représente un élément de $\mathbb{N}^{\mathcal{V}}$ par un mot sur $\{0,1\}^{\mathcal{V}}$ avec le bit de poids faible à gauche.
- On représente un sous-ensemble de $\mathbb{N}^{\mathcal{V}}$ par un langage sur $\{0,1\}^{\mathcal{V}}$.

Définition 38 (projection)**.** *Soit $\mathcal{V}$ un ensemble de variables et $x \notin \mathcal{V}$. Soit L un langage sur $\Sigma^{\mathcal{V} \cup \{x\}}$. La projection de L sur la variable x est le langage suivant sur $\Sigma^{\mathcal{V}}$:*

$$\left\{(w_{1,\lambda})_{\lambda \in \mathcal{V}} \dots (w_{\ell,\lambda})_{\lambda \in \mathcal{V}} \in (\Sigma^{\mathcal{V}})^* \mid \textit{ il existe } w_{1,x}, \dots, w_{\ell,x} \in \Sigma \textit{ tel que } (w_{1,\lambda})_{\lambda \in \mathcal{V} \cup \{x\}} \dots (w_{\ell,\lambda})_{\lambda \in \mathcal{V} \cup \{x\}} \in L\right\}.$$

précondition : φ est une formule de l'arithmétique de Presburger où les variables libres sont dans $\mathcal{V}$
postcondition : **retourner** le langage qui représente $[\![\varphi]\!]_{\mathcal{V}}$
fonction ENS(φ, $\mathcal{V}$)
  cas sur φ
    $(x = 0)$: **retourner** le langage rationnel qui représente $[\![(x = 0)]\!]_{\mathcal{V}}$
    $(x = 1)$: **retourner** le langage rationnel qui représente $[\![(x = 1)]\!]_{\mathcal{V}}$
    $(x + y = z)$: **retourner** le langage rationnel qui représente $[\![(x + y = z)]\!]_{\mathcal{V}}$
    $(\psi_1 \vee \psi_2)$: **retourner** ENS(ψ_1, $\mathcal{V}$) $\cup$ ENS(ψ_2, $\mathcal{V}$)
    $\neg\psi$: **retourner** $\Sigma^{\mathcal{V}} \setminus$ ENS(ψ, $\mathcal{V}$)
    $\exists x \psi$: **retourner** la projection du langage rationnel ENS(ψ, $\mathcal{V} \cup \{x\}$) sur la variable x

Théorème 39. *L'algorithme* ENS *est correct.*