Généralisations

Philippe Quéinnec

3 janvier 2011

Plan

Automate avec sortie (transducteur déterministe)

Transducteur déterministe

sextuplet $T = (Q, X, Y, \delta, q_I, F)$ où :

- Q : ensemble fini d'états
- X : alphabet d'entrée
- Y : alphabet de sortie
- $q_I \in Q$: l'état initial de l'automate
- $F \subseteq Q$: les états finals (ou terminaux)
- $\delta \in Q \times X \mapsto Q \times Y^*$: *fonction de transition* de l'automate.

Configuration

triplet (q, m, y) avec $q \in Q,\ \ m \in X^*,\ \ y \in Y^*$

Transitions

Relation $\vdash$ entre configurations :
$(q, am, y) \vdash (q', m, yy')$ si $(q', y') = \delta(q, a)$

Langage accepté

$L(T) = \{m \in X^* \mid \exists q_F \in F : (q_I, m, \Lambda) \vdash^* (q_F, \Lambda, y)\}$

Traduction d'un mot et d'un langage

$$T(m) = y \text{ si } \exists q_F \in F : (q_I, m, \Lambda) \vdash^* (q_F, \Lambda, y)$$

$$T(L) = \bigcup_{m \in L} T(m)$$

Modélisation

Le protocole du bit alterné

Reprendre le protocole du bit alterné, pour ajouter les actions de délivrance de message (mise à disposition du client d'un message reçu).

L'ascenseur

Un ascenseur sans mémoire permet de desservir trois étages. Lors d'une requête cliente (appui-1, appui-2 ou appui-3), il faut déterminer les actions de déplacement que doit réaliser l'ascenseur à partir des actions monter-un-étage et descendre-un-étage. Donner le schéma de fonctionnement de l'ascenseur sous la forme d'un transducteur.

Plan

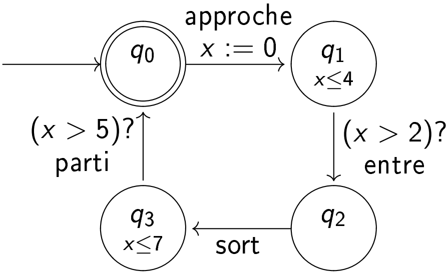
1. Automate avec sortie
2. Automates et le temps
3. Automates et l'infini
4. Automates avec contrôle auxiliaire
   - Ajout de variables auxiliaires
   - Automates à pile
   - Machine de Turing

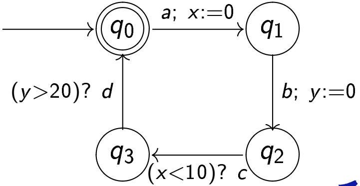
Automate temporisé

- horloges, qui avancent toutes au même rythme
- transition = garde, symbole, reset (= remise à zéro d'horloges)
- garde = contraintes sur les horloges :

$$g ::= h \left\{ \begin{matrix} < \\ = \\ > \end{matrix} \right\} c \quad | \quad h - h' \left\{ \begin{matrix} < \\ = \\ > \end{matrix} \right\} c \quad | \quad g \wedge g$$

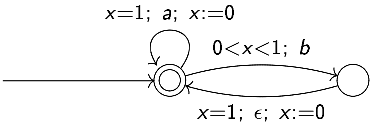
- état + invariant d'horloges (notion d'urgence)



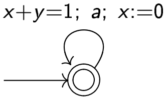


Attention à la généralisation :

- les automates temporisés déterministes sont *décidables*, comme les automates à états finis (vérifier l'égalité de deux automates, ou la nullité d'un automate)
- les automate temporisés non déterministes sont strictement plus expressifs (mais encore décidables) :



- les contraintes arbitraires sont strictement plus expressives et non décidables :



$$L = (a, \tfrac{1}{2})(a, \tfrac{3}{4})(a, \tfrac{7}{8}) \cdots (a, 1 - \tfrac{1}{2^i}) \cdots$$

Plan

1. Automate avec sortie
2. Automates et le temps
3. **Automates et l'infini**
4. Automates avec contrôle auxiliaire
   - Ajout de variables auxiliaires
   - Automates à pile
   - Machine de Turing

Automate sur des mots infinis

α est un mot infini sur X si

- tout préfixe (fini) de α est dans X^*,
- tout préfixe (fini) de α est distinct de α.

On note X^ω l'ensemble des mots infinis écrits sur l'alphabet X.

Automate de Büchi

$A = (Q, X, \delta, q_I, F)$ où A est défini comme un automate à états finis, auquel on rajoute l'acceptation des ω-mots par :
Soit $\alpha = \alpha_1\alpha_2\ldots$ un mot de X^ω, soit $\sigma = q_I\sigma_1\sigma_2\ldots \in Q^\omega$ la séquence d'états engendrés (ie tels que $\sigma_{i+1} = \delta(\sigma_i, \alpha_i)$), alors α est reconnu par A s'il existe une infinité de σ_i dans F.

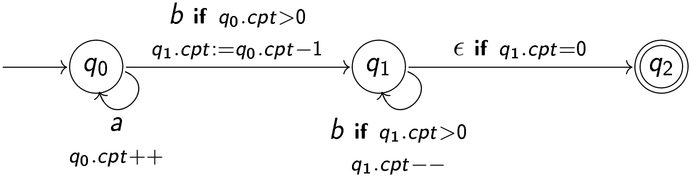
Plan

Automate à compteurs

À chaque état $\rightarrow$ compteur ≥ 0

Transition :

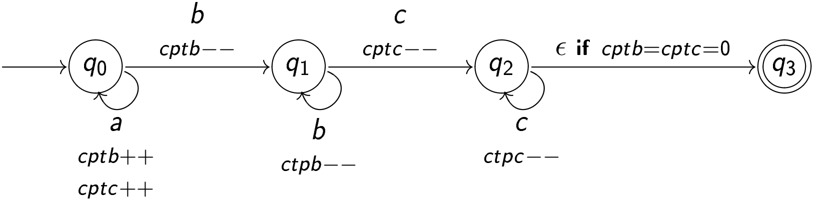
- gardée (contrôlée) par une valeur du compteur source
- modifie le compteur de l'état de départ (sans autre information)
- modifie le compteur de l'état d'arrivée (éventuellement en utilisant la valeur du compteur de départ)



$$L(A) = \{a^n b^n \mid n > 0\}$$

Automate + variables d'états

Ajout de variables arbitrairement accessibles, qui peuvent contraindre les transitions.



$$L(A) = \{a^n b^n c^n \mid n > 0\}$$

Note : on obtient les *systèmes de transitions*

Automate à pile

Automate à pile

7-uple $M = (Q, X, \Gamma, \delta, q_I, \perp, F)$ où :

- Q : ensemble fini d'états
- $q_I \in Q$: l'état initial
- $F \subseteq Q$: l'ensemble des états finals
- X : l'alphabet fini d'entrée
- Γ : l'alphabet fini de pile
- $\perp \in \Gamma$: le symbole initial de pile
- δ : application de $Q \times (X \cup \{\epsilon\}) \times \Gamma$ dans les parties finies non vides de $Q \times \Gamma^*$

Configuration

triplet (q, m, y) avec $q \in Q,\ \ m \in X^*,\ \ y \in \Gamma^*$

Transitions

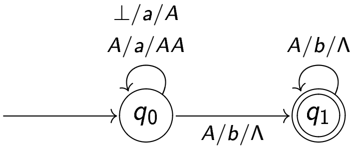
Relation $\vdash$ entre configurations :

$$
\begin{array}{ll}
(q, am, yZ) \vdash (q', m, yZ_1 \cdots Z_n) & \text{si } (q', Z_1 \cdots Z_n) \in \delta(q, a, Z) \\
(q, am, yZ) \vdash (q', am, yZ_1 \cdots Z_n) & \text{si } (q', Z_1 \cdots Z_n) \in \delta(q, \epsilon, Z) \\
(q, am, yZ) \vdash (q', m, y) & \text{si } (q', \Lambda) \in \delta(q, a, Z) \\
(q, am, yZ) \vdash (q', am, y) & \text{si } (q', \Lambda) \in \delta(q, \epsilon, Z)
\end{array}
$$

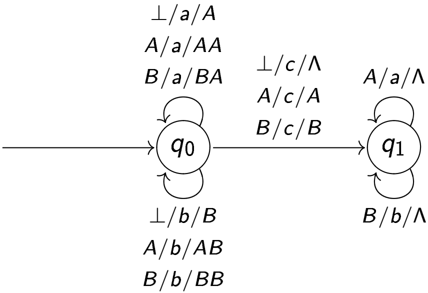
Langage accepté

- $T(M) = \{m \in X^* \mid (q_I, m, \perp) \vdash^* (q_F, \Lambda, y),\ \text{avec } q_F \in F,\ y \in \Gamma^*\}$
(langage reconnu par état terminal) ;
- $N(M) = \{m \in X^* \mid (q_I, m, \perp) \vdash^* (q, \Lambda, \Lambda),\ q \in Q\}$
(langage reconnu par pile vide) ;
- $L(M) = \{m \in X^* \mid (q_I, m, \perp) \vdash^* (q_F, \Lambda, \Lambda), q_F \in F\}$
(langage reconnu par état terminal et pile vide).

- $L(A_1) = \{a^n b^n \mid n > 0\}$



- $N(A_2) = \{mc\tilde{m} \mid m \in \{a,b\}^*\}$



Équivalences entre A.P.

Reconnaissance

Pour tout a.p. M, il existe un a.p. M' tel que $T(M') = N(M)$.
Pour tout a.p. M, il existe un a.p. M' tel que $N(M') = T(M)$.

Langage algébrique

L est algébrique $\equiv \exists A$ automate à pile : $L = L(A)$
Alg = ensemble des langages algébriques

Fermeture

Alg est fermé par union, produit, étoile :
Soient $L_1, L_2 \in \text{Alg}$, alors :

$$L_1 \cup L_2 \in \text{Alg} \qquad L_1 \bullet L_2 \in \text{Alg} \qquad {L_1}^* \in \text{Alg}$$

Alg n'est pas fermé par intersection et complémentaire (mais Alg est fermé par intersection avec Rat).

Automate à pile déterministe

Automate à pile déterministe

Un a.p. $M = (Q, X, \Gamma, \delta, q_I, \perp, F)$ est déterministe si pour tout $(q, a, Z) \in Q \times (X \times \{\epsilon\}) \times \Gamma$,

- $\text{card}(\delta(q, a, Z)) \leq 1$;
- Si $\delta(q, \epsilon, Z) \neq \varnothing$, alors $\delta(q, a, Z) = \varnothing, \forall a \in X$.

Remarque : il existe des langages algébriques non déterministes (par exemple, $\{m\tilde{m} \mid m \in \{a, b\}^*\}$).

Automate à n piles

Automate à n piles

$M = (Q, X, \Gamma, \delta, q_I, \perp, F)$ où
$\delta :\ Q \times (X \cup \{\epsilon\}) \times \Gamma^n$ dans les parties finies de $Q \times (\Gamma^*)^n$.

Si M est un automate à n piles (où $n > 2$), alors il existe un automate à 2 piles équivalent.
Les automates à deux piles forment un sur-ensemble *distinct* des automates à une pile.
Les automates à deux piles sont équivalents aux machines de Turing.

Machine de Turing

Machine de Turing

Sextuplet $\mathcal{M} = (Q, X, \Gamma, \delta, q_I, F)$ où :

- Q : ensemble fini d'états
- X : alphabet (fini)
- Γ : alphabet de bande, tel que $X \subset \Gamma$, et $B \in \Gamma \setminus X$ (le blanc)
- $q_I \in Q$: l'état initial de l'automate
- $F \subseteq Q$: les états finals (ou terminaux)
- $\delta \in Q \times \Gamma \mapsto Q \times \Gamma \times \{G, D, I\}$: fonction de transition.

Une machine de Turing possède en outre une structure de stockage qui est un ruban linéaire *non borné*.

Configuration

couple $(q, u\underline{a}v)$ avec $q \in Q, u \in \Gamma^*, a \in \Gamma, v \in \Gamma^*$
Par abus, on note $\underline{m}$ pour $\underline{a}m'$ où $m = am', a \in \Gamma$.

Transitions

Relation $\vdash$ entre configurations :

$$
\begin{array}{ll}
(q, uc\underline{a}dv) \vdash (q', uc\underline{b}dv) & \text{si } (q', b, I) \in \delta(q, a) \\
(q, uc\underline{a}dv) \vdash (q', ucb\underline{d}v) & \text{si } (q', b, D) \in \delta(q, a) \\
(q, uc\underline{a}dv) \vdash (q', u\underline{c}bdv) & \text{si } (q', b, G) \in \delta(q, a)
\end{array}
$$

Langage accepté

$$L(\mathcal{M}) = \{m \in (X \cup \{B\})^* \mid \exists q_F \in F : (q_I, \underline{m}) \vdash^* (q_F, m')\}$$

Valeur calculée

$$\mathcal{M}(m) = m' \text{ si } \exists q_F \in F : (q_I, \underline{m}) \vdash^* (q_F, m')$$

On définit aussi $\mathcal{M}$ par un sextuplet $(Q, X, \Gamma, \delta, q_I, \varnothing)$

- $\varnothing$: acceptation
- $\delta \in Q \times \Gamma \mapsto (Q \times \Gamma \times \{G, D, I\}) \cup \varnothing$

Langage accepté

$$L(\mathcal{M}) = \{m \in (X \cup \{B\})^* \mid (q_I, \underline{m}) \vdash^* \varnothing\}$$

Valeur calculée

Comme précédemment : état du ruban à l'acceptation.

	B	a	b	
q_0	$\varnothing$	q_0, a, D	q_1, b, D	lit les a
q_1	$\varnothing$		q_1, b, D	lit les b

$$L(\mathcal{M}) = \{a^i b^j \mid i, j \geq 0\}$$

	B	0	1	
q_0	q_1, B, G	$q_0, 0, D$	$q_0, 1, D$	va à la fin
q_1	$q_F, 1, I$	$q_2, 1, G$	$q_1, 0, G$	incrémente avec retenue
q_2	q_F, B, D	$q_2, 0, G$	$q_2, 1, G$	retourne au début
q_F		$\varnothing$	$\varnothing$	

$$L(\mathcal{M}) = \{0, 1\}^*$$

$$\mathcal{M}(n) = n + 1$$