

CH.5 Automates à pile

- 5.1 Les modèles
- 5.2 L'équivalence avec les langages non contextuels

Automates ch5 1

5.1 Les modèles

Jouent pour les langages non contextuels le rôle des automates finis pour les langages rationnels.

Structure : automate non déterministe + pile.

Formellement :

Q ensemble fini d'états

Σ alphabet fini d'entrée

Γ alphabet fini de la pile

q_0 état initial

Z_0 symbole initial de la pile

δ fonction de transition $Q \times (\Sigma \cup \{\varepsilon\}) \rightarrow 2^{Q \times \Gamma^*}$.

Transition : le symbole au sommet de la pile est dépilé, un caractère d'entrée est consommé (ou ε -transition) ; en fonction de δ , l'automate écrit sur la pile et change d'état.

Automates ch5 2

Pratique pour simuler le fonctionnement : description instantanée.

Si M un automate à pile, on note (q, w, γ) l'état de M , le mot w restant à lire et le mot γ constitué des symboles sur la pile (la dernière lettre de γ est le caractère au fond de la pile).

On note $(q, aw, Z\alpha) \vdash (p, w, \beta\alpha)$ s'il existe une transition $\delta(q, a, Z)$ vers (p, β) .

De même, pour une transition vide $(q, w, Z\alpha) \vdash (p, w, \beta\alpha)$ s'il existe une transition $\delta(q, \epsilon, Z)$ vers (p, β) .

On étend cette dérivation par transitivité : \vdash^* .

Le langage accepté par un automate à pile M est l'ensemble des mots

$$N(M) = \{w : (q_0, w, Z_0) \vdash^* (p, \epsilon, \epsilon)\}.$$

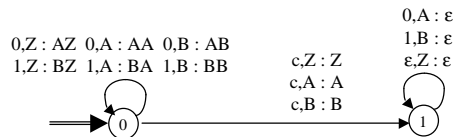
Partant de la configuration initiale, il est possible de lire tout le mot et de vider la pile.

Automates ch5 3

On pourrait aussi spécifier un ensemble d'états terminaux et décider qu'un mot est reconnu s'il peut être lu entièrement en laissant finalement l'automate dans un état terminal.

En fait, les deux notions permettent de reconnaître les mêmes langages.

Exemples : Deux états, fond de pile Z , alphabet d'entrée $\{0, 1, c\}$, alphabet de pile $\{Z, A, B\}$, reconnaissance par pile vide :

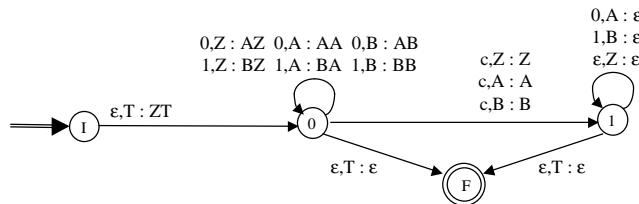


$$(q_0, 01c10, Z) \vdash (q_0, 1c10, AZ) \vdash (q_0, c10, BAZ) \vdash (q_1, 10, BAZ) \\ \vdash (q_1, 0, AZ) \vdash (q_1, \epsilon, Z) \vdash (q_1, \epsilon, \epsilon).$$

Reconnaît par pile vide l'ensemble des mots de la forme wcw^t , concaténation de w et de son image-miroir séparés par le caractère c .

Automates ch5 4

On peut le transformer pour reconnaissance par état final. On ajoute deux états I et F et un nouveau symbole de fond de pile T.

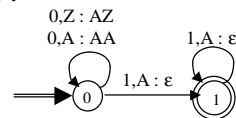


De cette façon, lorsque la pile est vide, l'automate va en F, état terminal.

Remarque :

Les deux automates ainsi définis sont déterministes : pour chaque état et chaque symbole en sommet de pile, ou bien une transition vide est spécifiée, ou bien, pour chaque lettre de l'alphabet d'entrée, au plus une transition est possible.

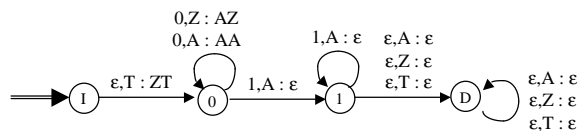
Deux états, fond de pile Z, alphabet d'entrée {0, 1}, alphabet de pile {Z, A}, reconnaissance par état final :



Cet automate reconnaît $\{0^n 1^m, n \geq m > 0\}$.

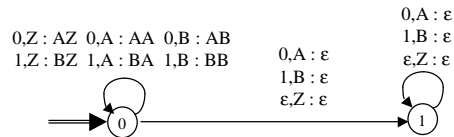
On peut le modifier pour obtenir une reconnaissance par pile vide.

On rajoute deux états I et D et un nouveau fond de pile T :



De cette manière, l'arrivée sur un état final provoque le passage dans un état qui fait se vider la pile. Mais l'automate ainsi obtenu n'est plus nécessairement déterministe (deux transitions sur A en 1). Cette technique générale ne donne souvent pas l'automate le plus simple.

Deux états, fond de pile Z, alphabet d'entrée {0, 1}, alphabet de pile {Z, A, B}, reconnaissance par pile vide :



$(q_0, 0110, Z) \vdash (q_0, 110, AZ) \vdash (q_0, 10, BAZ) \vdash (q_1, 0, AZ)$
 $\vdash (q_1, \epsilon, Z) \vdash (q_1, \epsilon, \epsilon).$

On aurait aussi pu bifurquer :

$(q_0, 10, BAZ) \vdash (q_0, 0, BBAZ)$, mais la pile ne peut plus être vidée.

Reconnaît par pile vide l'ensemble des mots de la forme ww^t ,
concaténation de w et de son image-miroir.

Cet automate n'est pas déterministe.

Automates ch5 7

5.2 L'équivalence avec les langages non contextuels

Les automates à pile sont les machines permettant de reconnaître si un mot appartient à un langage non contextuel.

Théorème : Si L est engendré par une grammaire non contextuelle G , alors il existe un automate à pile M qui reconnaît L par pile vide.

Démonstration : On suppose que L ne contient pas ϵ et que G est sous forme normale de Greibach.

Soit $G = (V, T, P, S)$ la grammaire. Une production est du type

$A \rightarrow a\gamma$ où γ ne contient que des variables. On va construire un automate dont la pile servira à conserver la liste des variables à récrire dans une dérivation à partir de l'axiome.

Automates ch5 8

$M = (\{q\}, T, V, \delta, S)$. Un seul état ; le symbole de fond de pile est S ;
la fonction de transition est $\delta(q, a, A) = \{(q, \gamma) : A \rightarrow a\gamma \in P\}$.

Montrons que :

$S \Rightarrow^* w\alpha$ est une dérivation à gauche dans G si et seulement si
 $(q, w, S) \vdash^* (q, \varepsilon, \alpha)$ est une suite de transitions dans M .

1. $S \Rightarrow^i w\alpha$. Récurrence sur la longueur i de la dérivation.

Si $i = 0$, trivial.

Supposons $S \Rightarrow^{i-1} xA\beta \Rightarrow w\alpha$. Ceci est possible seulement
si $A \rightarrow a\gamma \in P$ et $w = xa$ et $\alpha = \gamma\beta$.

Par récurrence, $(q, x, S) \vdash^* (q, \varepsilon, A\beta)$,

donc $(q, xa, S) \vdash^* (q, a, A\beta) \vdash (q, \varepsilon, \gamma\beta)$, ce qu'on veut.

2. $(q, w, S) \vdash^i (q, \varepsilon, \alpha)$. Récurrence sur le nombre i .

Si $i = 0$, trivial

Aucune production vide dans G , donc aucune transition
sur ε dans M . Si $i > 0$, on a donc $w = sa$. On a alors dans M :

$(q, xa, S) \vdash^{i-1} (q, a, \beta) \vdash (q, \varepsilon, \alpha)$.

La lettre a n'a aucune influence tant qu'elle n'est pas lue, donc

$(q, x, S) \vdash^{i-1} (q, \varepsilon, \beta)$ et par récurrence $S \Rightarrow^* x\beta$.

Par ailleurs, $(q, a, \beta) \vdash (q, \varepsilon, \alpha)$ n'est possible que si le sommet
de la pile contient une variable A (donc $\beta = A\eta$) telle que

$A \rightarrow a\gamma$ et donc $\alpha = \gamma\eta$.

On a donc, dans G :

$S \Rightarrow^* xA\eta \Rightarrow xa\gamma\eta = w\alpha$, ce qu'on désire.

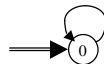
On applique l'équivalence démontrée avec $\alpha = 0$:

$S \Rightarrow^* w$ est une dérivation à gauche dans G si et seulement si
 $(q, w, S) \vdash^* (q, \varepsilon, \varepsilon)$ est une suite de transitions dans M .

Exemple : $S \rightarrow 0S1 \mid 01$. On la met sous forme normale de Greibach
 par $S \rightarrow 0SA \mid 0A \quad A \rightarrow 1$.

Automate obtenu :

$0,S : SA \quad 0,S : A \quad 1,A : \varepsilon$



On a la réciproque :

Théorème : Si L est reconnu par pile vide par un automate à pile
 M , alors il existe une grammaire non contextuelle G qui engendre
 le langage M .

Démonstration : Un peu plus délicate que la précédente.

Supposons que $M = (Q, \Sigma, \delta, q_0, Z_0)$. La grammaire G construite
 a comme variables des triplets $[q, A, p]$, où p et q sont des états
 de M , où A est un symbole de pile, plus un axiome S .

Les productions sont :

Pour chaque état q de M , on a $S \rightarrow [q_0, Z_0, q]$.

Chaque fois que $\delta(q, a, A)$ contient $(q_1, B_1 B_2 \dots B_m)$, on a les productions

$[q, A, q_{m+1}] \rightarrow a[q_1, B_1, q_2][q_2, B_2, q_3] \dots [q_m, B_m, q_{m+1}]$ pour toutes
 les suites q_2, q_3, \dots, q_{m+1} d'états de M . Ceci s'applique lorsque
 $m > 0$, même si $a = \varepsilon$ (Par la suite, a signifie une lettre ou ε .)

Si $m = 0$, alors on a la production $[q, A, q_1] \rightarrow a$.

On montre que $[q, A, p] \Rightarrow^* w$ si et seulement si $(q, w, A) \vdash^* (p, \varepsilon, \varepsilon)$.

1. $[q, A, p] \Rightarrow^i w$. Récurrence sur i (nécessairement $i > 0$).

Si $i = 1$, on a $[q, A, p] \rightarrow a$ seule production possible et $\delta(q, a, A)$ contient (p, ϵ) .

Si $i > 1$, on a

$[q, A, p] \Rightarrow a[q_1, B_1, q_2][q_2, B_2, q_3] \dots [q_m, B_m, p] \Rightarrow^{i-1} w$, et $\delta(q, a, A)$ contient $(q_1, B_1 B_2 \dots B_m)$

On a donc $w = ax_1 x_2 \dots x_m$ où $[q_1, B_1, q_2] \Rightarrow^* x_1, \dots, [q_m, B_m, p] \Rightarrow^* x_m$, chaque dérivation étant de longueur inférieure ou égale à $i - 1$.

Par récurrence, $(q_1, x_1, B_1) \vdash^* (q_2, \epsilon, \epsilon), \dots, (q_m, x_m, B_m) \vdash^* (p, \epsilon, \epsilon)$.

Par conséquent,

$(q, ax_1 x_2 \dots x_m, A) \vdash (q_1, x_1 x_2 \dots x_m, B_1 B_2 \dots B_m) \vdash^* (q_2, x_2 \dots x_m, B_2 \dots B_m) \vdash^* \dots \vdash^* (q_m, x_m, B_m) \vdash^* (p, \epsilon, \epsilon)$.

Cela démontre la première partie.

2. $(q, w, A) \vdash^i (p, \epsilon, \epsilon)$. Récurrence sur i (nécessairement $i > 0$).

Si $i = 1$, alors $w = a$ donc $\delta(q, a, A)$ contient (p, ϵ, ϵ) et $[q, A, p] \rightarrow a$.

Soit $i > 1$ et $w = ax$ avec $(q, ax, A) \vdash (q_1, x, B_1 B_2 \dots B_m) \vdash^{i-1} (p, \epsilon, \epsilon)$.

Autrement dit, $\delta(q, a, A)$ contient $(q_1, B_1 B_2 \dots B_m)$

Puisque la pile est vidée en partant de $(q_1, x, B_1 B_2 \dots B_m)$, il existe un moment où B_1 a été enlevé de la pile alors que $B_2 \dots B_m$ n'a pas été modifié. Ceci se produit après que le mot y_1 a été lu.

Donc, $x = y_1 x_1$ et $(q_1, y_1, B_1) \vdash^* (q_2, \epsilon, \epsilon)$, où la suite de transitions est de longueur au plus $i - 1$. De la sorte,

$(q_1, y_1 x_1, B_1 B_2 \dots B_m) \vdash^* (q_2, x_1, B_2 \dots B_m)$

De même, $x_1 = y_2 x_2$ et $(q_2, y_2, B_2) \vdash^* (q_3, \epsilon, \epsilon), \dots,$

$x_{m-1} = y_m$ et $(q_m, y_m, B_m) \vdash^* (p, \epsilon, \epsilon)$.

On a donc $w = a y_1 y_2 \dots y_m$.

Puisque $\delta(q, a, A)$ contient $(q_1, B_1 B_2 \dots B_m)$, il existe une production

$$[q, A, p] \Rightarrow a[q_1, B_1, q_2][q_2, B_2, q_3] \dots [q_m, B_m, p].$$

De plus, par l'hypothèse de récurrence, $(q_1, y_1, B_1) \vdash^* (q_2, \varepsilon, \varepsilon)$ donc

$$[q_1, B_1, q_2] \Rightarrow^* y_1.$$

De même, $[q_2, B_2, q_3] \Rightarrow^* y_2, \dots, [q_m, B_m, p] \Rightarrow^* y_m.$

On en déduit :

$$[q, A, p] \Rightarrow^* ay_1 y_2 y_m = ax = w.$$

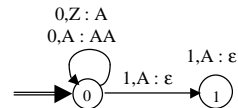
Ceci établit la deuxième partie.

On conclut maintenant en appliquant l'équivalence avec $A = Z_0$ et $q = q_0$:

$$[q_0, Z_0, p] \Rightarrow^* w \text{ si et seulement si } (q_0, w, Z_0) \vdash^* (p, \varepsilon, \varepsilon).$$

Mais $S \rightarrow [q_0, Z_0, q]$ pour n'importe quel q (donc pour p).

Exemple : Automate reconnaissant $0^n 1^n$.



La grammaire G est :

$$S \rightarrow [q_0, Z, q_0] \mid [q_0, Z, q_1]$$

$$[q_0, Z, q_0] \rightarrow 0[q_0, A, q_0] \quad [q_0, Z, q_1] \rightarrow 0[q_0, A, q_1]$$

$$[q_0, A, q_0] \rightarrow 0[q_0, A, q_0][q_0, A, q_0] \mid 0[q_0, A, q_1][q_1, A, q_0]$$

$$[q_0, A, q_1] \rightarrow 0[q_0, A, q_0][q_0, A, q_1] \mid 0[q_0, A, q_1][q_1, A, q_1]$$

$$[q_0, A, q_1] \rightarrow 1 \quad [q_1, Z, q_1] \rightarrow \varepsilon \quad [q_1, A, q_1] \rightarrow 1$$

Après élimination des variables inutiles, il reste :

$$S \rightarrow [q_0, Z, q_1]$$

$$[q_0, Z, q_1] \rightarrow 0[q_0, A, q_1] \quad [q_0, A, q_1] \rightarrow 0[q_0, A, q_1][q_1, A, q_1]$$

$$[q_0, A, q_1] \rightarrow 1 \quad [q_1, Z, q_1] \rightarrow \varepsilon \quad [q_1, A, q_1] \rightarrow 1$$

On peut remplacer les deux dernières variables par leur production :

$$S \rightarrow [q_0, Z, q_1] \quad [q_0, Z, q_1] \rightarrow 0[q_0, A, q_1]$$

$$[q_0, A, q_1] \rightarrow 0[q_0, A, q_1]1 \mid 1$$

En renommant et en supprimant la production unitaire sur S :

$$S \rightarrow 0B \quad B \rightarrow 0B1 \mid 1$$