Calculabilité - Décidabilité (ICC)

Cours n°5

Stef Graillat

Sorbonne Université

Résumé du cours précédent

- Grammaire hors-contexte : une façon de décrire un langage par des règles récursives. Un grammaire consiste en des variables, des symboles terminaux, un symbole de départ et des règles de production
- Dérivations et langages : le langage associé à une grammaire est l'ensemble des mots constitués de terminaux que l'on peut dériver à partir du symbole de départ
- Dérivations droites et gauches : on remplace à chaque fois la variable la plus à gauche (à droite)
- Arbres de dérivation : un arbre de dérivation est un arbre qui capture les informations essentielles d'une dérivation.
- Ambiguité : une grammaire est dite ambigue si on peut trouver un mot de terminaux ayant deux arbres de dérivation distincts ou bien de manière équivalente deux dérivations grauche (ou droite) distinctes

Automates à piles

Automates associés aux langages hors-contexte.

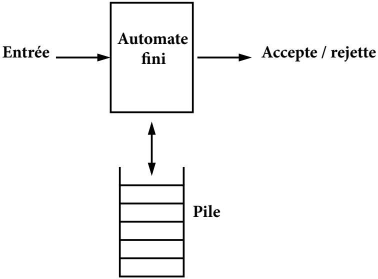
- Extension des AFN à ε-transitions auxquels on ajoute une pile.
- Reconnaissance des langages par états acceptants
- Reconnaissance des langages par pile vide
- Équivalences et langages hors-contexte
- Automates à pile déterministes

Définition d'un automate à pile

Automate fini non-déterministe qu'on dote d'une pile (structure de données classique).

Fonctionnement global :

- Consommation du symbole d'entrée à chaque transition.
- Aucun changement ou modification de l'état
- Modification de la pile ou pas (effacement du symbole au-dessus, ou modification de ce symbole, ou ajout d'un symbole).



Exemple

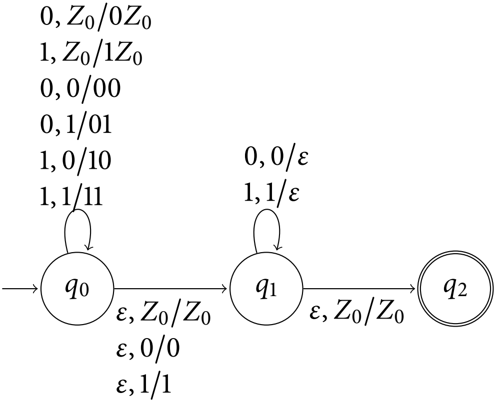
Exemple : $L_{wwr} = \{ww^R \mid w \in \{0,1\}^\star\}$

Une grammaire pour L_{wwr} est : $P \to 0P0$, $P \to 1P1$, $P \to \varepsilon$

Un AP pour L_{wwr} a 3 états et opère comme suit :

1. On suppose qu'on lit w. On reste à l'état 0 et on empile les symboles
2. On suppose qu'on est au milieu de ww^R. On passe à l'état 1
3. On suppose qu'on lit maintenant w^R. On compare alors avec le haut de la pile. Si les symboles sont identiques, on dépile, sinon on ne fait rien
4. Si la pile est vide, on va à l'état 2 et on accepte

Exemple (suite)



Définition formelle

Un automate à pile (AP) est un septuplet $P = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ où

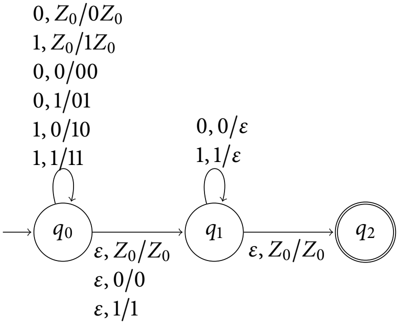
- Q un ensemble fini d'états
- Σ un ensemble fini de symboles d'entrées
- Γ un alphabet de pile (symboles pouvant être stocké dans la pile)
- $\delta : Q \times \Sigma \cup \{\varepsilon\} \times \Gamma \to \mathcal{P}(Q \times \Gamma^*)$ la fonction de transition (3 arguments) $\delta(q, a, X)$ où
  - q est un état
  - a un symbole de Σ
  - X un symbole de pile (dans Γ)

  La sortie est un ensemble fini de pairs (p, γ) où $p \in Q$ et γ une chaîne de symboles qui remplace X au haut de la pile.
- q_0 l'état de départ
- Z_0 le symbole de départ
- F l'ensemble des états acceptants (ou états finaux)

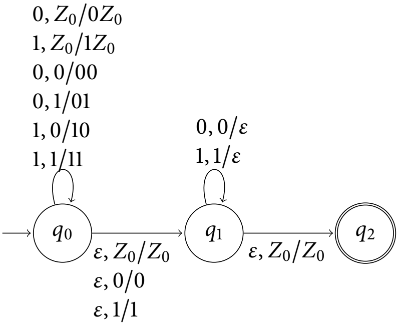
Notation graphique

Notation similaire à celle des AFD.

- Les nœuds correspondent aux états.
- Une flèche indique l'état de départ et les états finaux sont entourés de deux cercles.
- Les flèches correspondent aux transitions. Un label $a, X/\alpha$ de l'état q vers l'état p signifie que (p, α) est dans $\delta(q, a, X)$.



Table



Cet AP est le septuplet $P = (\{q_0, q_1, q_2\}, \{0, 1\}, \{0, 1, Z_0\}, \delta, q_0, Z_0, \{q_2\})$ avec δ donné par la table suivante

	$0, Z_0$	$1, Z_0$	$0, 0$	$0, 1$	$1, 0$	$1, 1$	ε, Z_0	$\varepsilon, 0$	$\varepsilon, 1$
$\rightarrow q_0$	$q_0, 0Z_0$	$q_0, 1Z_0$	$q_0, 00$	$q_0, 01$	$q_0, 10$	$q_0, 11$	q_1, Z_0	$q_1, 0$	$q_1, 1$
q_1			q_1, ε			q_1, ε	q_1, Z_0		
$\star\ q_2$									

Descriptions instantannées (configurations)

Configuration d'un AP par un triplet (q, w, γ) où

- q est l'état de l'AP
- w ce qui reste à lire de l'entrée
- γ est le contenu de la pile.

Si $\delta(q, a, X)$ contient (p, α) alors pour tout $w \in \Sigma^\star$ et $\beta \in \Gamma^\star$

$$(q, aw, X\beta) \vdash (p, w, \alpha\beta)$$

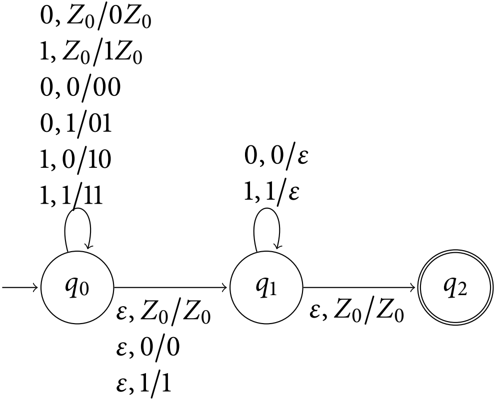
Transitions étendues : $\vdash^*$

- **Base** : $I \vdash^* I$ pour toute configuration I
- **Induction** : $I \vdash^* J$ s'il existe une configuration K telle que $I \vdash K$ et $K \vdash^* J$

Autrement $I \vdash^* J$ ssi il existe des configurations $K_1, K_2, \ldots, K_n$ telles que $I = K_1$, $J = K_n$ et $K_i \vdash K_{i+1}$ pour tout $i = 1, 2, \ldots, n-1$

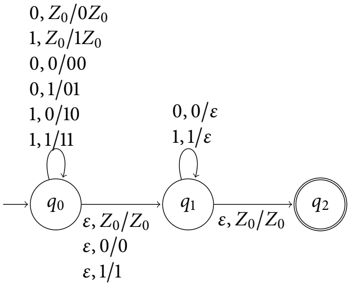
Exemple

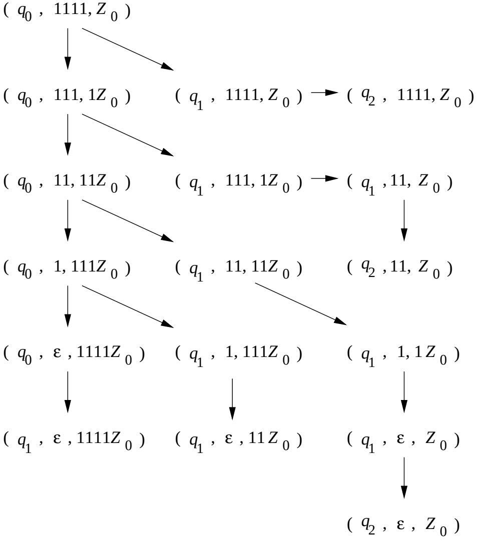
Transitions étendues pour l'AP avec 1111 en entrée



Exemple (suite)

Transitions étendues pour l'AP avec 1111 en entrée





Quelques remarques

Propriété 1

On a les propriétés suivantes :

1. *Si une suite de configuration est valide alors la suite obtenue en ajouter une chaine à la fin du composant 2 est encore valide*
2. *Si une suite de configuration est valide alors la suite obtenue en ajouter une chaine à la fin du composant 3 est encore valide*
3. *Si une suite de configuration est valide et si la fin de l'entrée n'est pas consommée alors en la retirant, la suite de configuration est encore valide*

Quelques résultats

Proposition 1

Si $P = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ *est un AP et* $(q, x, \alpha) \vdash^* (p, y, \beta)$ *alors pour toute chaîne* $w \in \Sigma^*$ *et* $\gamma \in \Gamma^*$ *on a*

$$(q, xw, \alpha\gamma) \vdash^* (p, yw, \beta\gamma)$$

Remarque 1 : si $\gamma = \varepsilon$ on obtient la propriété 1 et si $w = \varepsilon$, on obtient la propriété 2

Remarque 2 : la réciproque est fausse

Pour la propriété 3, on a

Proposition 2

Si $P = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ *est un AP et* $(q, xw, \alpha) \vdash^* (p, yw, \beta)$ *alors on a*

$$(q, x, \alpha) \vdash^* (p, y, \beta)$$

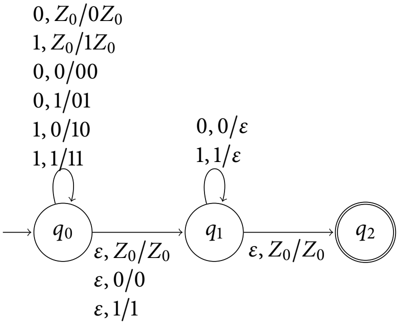
Langages d'un automate à pile par état final

Définition 1

Soit $P = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ *un AP. Alors le langage accepté par état final est*

$$L(P) = \{w : (q_0, w, Z_0) \vdash^* (q, \varepsilon, \alpha), q \in F\}$$

Exemple : l'AP suivant reconnait le langage $L_{wwr} = \{ww^R \mid w \in \{0,1\}^\star\}$ par état final



Langages d'un automate à pile par état final (suite)

Prouver que $L(P) = L_{wwr}$

"$\supset$" soit $x \in L_{wwr}$ alors $x = ww^R$ et la suite de transition est valide

$(q_0, ww^R, Z_0) \vdash^* (q_0, w^R, w^R Z_0) \vdash (q_1, w^R, w^R Z_0) \vdash^* (q_1, \varepsilon, Z_0) \vdash (q_2, \varepsilon, Z_0)$

"$\subset$" on observe que la seule façon attendre l'état q_2 est d'être à l'état q_1 avec Z_0 sur la pile
$\rightarrow$ il suffit donc de montrer que si $(q_0, x, Z_0) \vdash^* (q_1, \varepsilon, Z_0)$ alors $x = ww^R$ pour une chaîne w. Montrons par induction sur $|x|$ que

$$(q_0, x, \alpha) \vdash^* (q_1, \varepsilon, \alpha) \Rightarrow x = ww^R$$

Base : si $x = \varepsilon$ alors x est un palindrome

Induction : Supposons $x = a_1 a_2 \cdots a_n$ avec $n > 0$. À partir de (q_0, x, α), il y a 2 possiblités
Cas 1 : $(q_0, x, \alpha) \vdash (q_1, x, \alpha)$.
Mais $(q_1, x, \alpha) \vdash^* (q_1, \varepsilon, \beta)$ implique que $|\beta| < |\alpha|$ et donc $\beta \neq \alpha$

Langages d'un automate à pile par état final (suite)

Cas 2 : $(q_0, a_1 a_2 \cdots a_n, \alpha) \vdash (q_0, a_2 \cdots a_n, a_1\alpha)$.
On doit donc avoir

$$(q_0, a_1 a_2 \cdots a_n, \alpha) \vdash (q_0, a_2 \cdots a_n, a_1\alpha) \vdash \cdots \vdash (q_1, a_n, a_1\alpha) \vdash (q_1, \varepsilon, \alpha)$$

Par conséquent, $a_1 = a_n$ et

$$(q_0, a_2 \cdots a_n, a_1\alpha) \vdash^* (q_1, a_n, a_1\alpha)$$

Donc

$$(q_0, a_2 \cdots a_{n-1}, a_1\alpha) \vdash^* (q_1, \varepsilon, a_1\alpha)$$

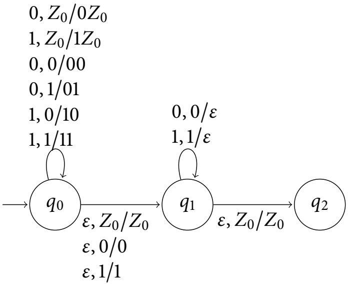
Par hypothèse de récurrence $a_2 \cdots a_{n-1} = yy^R$ et donc $x = a_1 yy^R a_n$ est un palindrome

Langages d'un automate à pile par pile vide

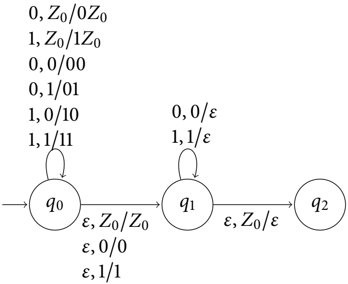
Définition 2

Soit $P = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ *un AP. Alors le langage accepté par* *pile vide* *est*

$$N(P) = \{w : (q_0, w, Z_0) \vdash^* (q, \varepsilon, \varepsilon), q \in Q\}$$



$$N(P) = \varnothing$$



$$N(P) = L_{wwr}$$

Pile vide → État final

Théorème 1

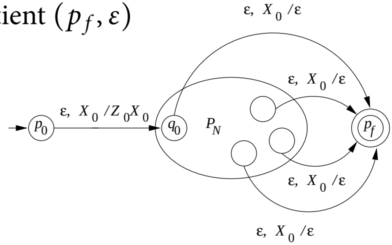
Si $L = N(P_N)$ où $P_N = (Q, \Sigma, \Gamma, \delta_N, q_0, Z_0)$ est un AP alors il existe un AP P_F tel que $L = L(P_F)$.

Construction de P_F :

$$P_F = (Q \cup \{p_0, p_f\}, \Sigma, \Gamma \cup \{X_0\}, \delta_F, p_0, X_0, \{p_f\})$$

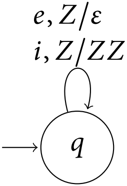
où δ_F est ainsi définie :

- $\delta_F(p_0, \varepsilon, X_0) = \{(q_0, Z_0X_0)\}$
- Pour tout $q \in Q$, $a \in \Sigma$ et $Y \in \Gamma$, $\delta_N(q, a, Y) \subset \delta_F(q, a, Y)$
- $\delta_F(q, \varepsilon, X_0)$ contient (p_f, ε)

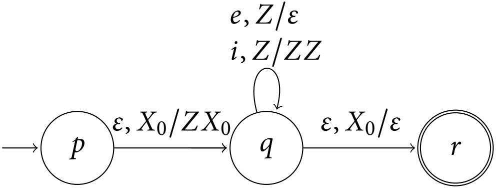


Pile vide → État final : exemple

Convertissons l'AP $P_N = (\{q\}, \{i, e\}, \{Z\}, \delta_N, q, Z)$



L'APD P_F accepte le même langage mais par état final



État final → Pile vide

Théorème 2

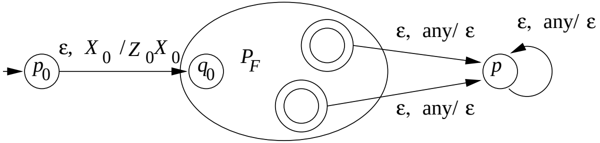
Si $L = L(P_F)$ pour un AP $P_F = (Q, \Sigma, \Gamma, \delta_F, q_0, Z_0, F)$, il existe un AP P_N tel que $L = N(P_N)$.

Construction de P_N :

$$P_N = (Q \cup \{p_0, p\}, \Sigma, \Gamma \cup \{X_0\}, \delta_F, p_0, X_0, \{\})$$

où δ_N est ainsi définie :

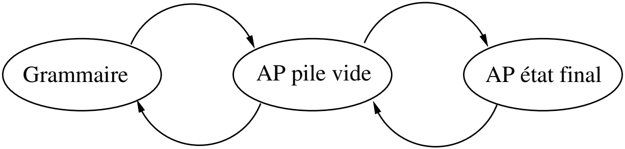
- $\delta_N(p_0, \varepsilon, X_0) = \{(q_0, Z_0X_0)\}$.
- Pour tout $q \in Q$, $a \in \Sigma$ et $Y \in \Gamma$, $\delta_N(q, a, Y)$ contient $\delta_F(q, a, Y)$.
- Pour tout q dans F, $Y \in \Gamma \cup \{X_0\}$, $\delta_N(q, \varepsilon, Y)$ contient (p, ε).
- Pour tout $Y \in \Gamma \cup \{X_0\}$, $\delta_N(p, \varepsilon, Y) = \{(p, \varepsilon)\}$.



Équivalence entre automates à piles et langages hors-contexte

Équivalence des trois classes de langage :

- Les langages hors-contexte
- Les langages acceptés par état final
- Les langages acceptés par pile vide



Des grammaires aux AP

Idée : construire un AP qui simule $\Rightarrow_g^*$.

Une forme syntaxique gauche s'écrit sous la forme $xA\alpha$ où A est la variable la plus à gauche.

Si $A \to \beta$ alors $xA\alpha \Rightarrow_g x\beta\alpha$. D'un point de vue de l'AP, cela va correspondre à avoir lu x et à avoir $A\alpha$ dans la pile : en lisant ε, l'AP va dépiler A et empiler β.

Plus formellement, si $w = xy$ alors $(q, y, A\alpha) \vdash (q, y, \beta\alpha)$

Dans la configuration $(q, y, \beta\alpha)$, l'AP se comporte comme précédement sauf s'il y a des terminaux dans les préfixes de β. Dans ce cas, on les dépile s'ils correspondent au symboles lus en entrée.

Des grammaires aux AP (suite)

Théorème 3

Soit $G = (V, T, Q, S)$ *une grammaire hors-contexte. Soit* $P = (\{q\}, T, V \cup T, \delta, q, S)$ *un AP avec* δ *définie par*

- *Pour toute variable* A

$$\delta(q, \varepsilon, A) = \{(q, \beta) \mid A \to \beta \text{ est une production de } G\}$$

- *Pour tout symbole terminal* a, $\delta(q, a, a) = \{(q, \varepsilon)\}$

Alors, P *reconnait* $L(G)$ *par pile vide.*

Exemple : soit la grammaire

$$\begin{array}{lcl} I & \to & a \mid b \mid Ia \mid Ib \mid I0 \mid I1 \\ E & \to & I \mid E * E \mid E + E \mid (E) \end{array}$$

Des grammaires aux AP (suite)

Exemple : soit la grammaire

$$\begin{array}{lcl} I & \to & a \mid b \mid Ia \mid Ib \mid I0 \mid I1 \\ E & \to & I \mid E * E \mid E + E \mid (E) \end{array}$$

Les terminaux de l'AP sont $\{a, b, 0, 1, (,), +, *\}$ et

a) $\delta(q, \varepsilon, I) = \{(q, a), (q, b), (q, Ia), (q, Ib), (q, I0), (q, I1)\}$
b) $\delta(q, \varepsilon, E) = \{(q, I), (q, E + E), (q, E * E), (q, (E))\}$
c) $\delta(q, a, a) = \{(q, \varepsilon)\}$; $\delta(q, b, b) = \{(q, \varepsilon)\}$; $\delta(q, 0, 0) = \{(q, \varepsilon)\}$; $\delta(q, 1, 1) = \{(q, \varepsilon)\}$; $\delta(q, (, () = \{(q, \varepsilon)\}$; $\delta(q,),)) = \{(q, \varepsilon)\}$; $\delta(q, +, +) = \{(q, \varepsilon)\}$; $\delta(q, *, *) = \{(q, \varepsilon)\}$

Des AP aux grammaires

Soit $P = (Q, \Sigma, \Gamma, \delta, q_0, Z_0)$ un AP. Il existe une grammaire G telle que $L(G) = N(P)$.

$$G = (V, \Sigma, R, S)$$

où l'ensemble V de variables est consitué de :

- Le symbole spécial S de départ.
- Tous les symboles de la forme $[pXq]$ où p et q sont des états de Q et X est un symbole de la pile.

Les productions de G sont ainsi définies :

- Pour tous les états p de P, G est doté de la production $S \to [q_0 Z_0 p]$
- Si $\delta(q, a, X)$ contient la paire $(r, Y_1 \cdots Y_k)$ (il est possible que $k = 0$). Alors pour toute liste d'états $r_1, \ldots, r_k$, G contient la production

$$[qXr_k] \to a[rY_1r_1][r_1Y_2r_2] \cdots [r_{k-1}Y_kr_k]$$

Des AP aux grammaires (suite)

Intuition : la variable $[pXq]$ répresente l'ensemble des mots qui permettent de passer de l'état p à q en ayant pour effet de dépiler X

De manière plus rigoureuse, on peut montrer que

$$[pXq] \Rightarrow^* w \text{ si et seulement si } (p, w, X) \vdash^* (q, \varepsilon, \varepsilon)$$

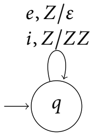
- la règle $S \to [q_0 Z_0 p]$ dit que les mots de S sont ceux qui permettent de passer de q_0 à un état p en dépilant Z_0 (il s'agit donc des mots reconnus par pile vide par l'automate)
- si $(r, Y_1 \cdots Y_k) \in \delta(q, a, X)$ alors la règle

$$[qXr_k] \to a[rY_1r_1][r_1Y_2r_2] \cdots [r_{k-1}Y_kr_k]$$

  dit que les mots permettant de passer de q à r_k en dépilant X sont les concaténations de a, de ceux permettant de passer de r à r_1 en dépilant Y_1, de ceux permettant de passer de r_1 à r_2 en dépilant Y_2, etc.

Des AP aux grammaires (suite)

Convertissons l'AP $P_N = (\{q\}, \{i, e\}, \{Z\}, \delta_N, q, Z)$



avec $\delta_N(q, i, Z) = \{(q, ZZ)\}$ et $\delta_N(q, i, Z) = \{(q, \varepsilon)\}$ en une grammaire

$$G = (V, \{i, e\}, R, S)$$

avec $V = \{[qZq], S\}$ et

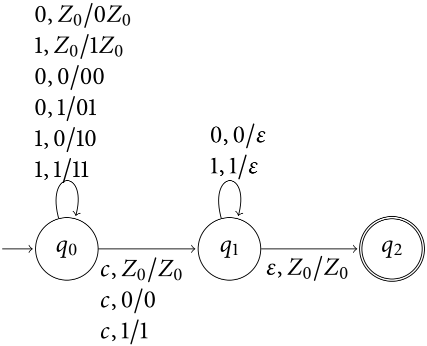
$$R = \{S \to [qZq], [qZq] \to i[qZq][qZq], [qZq] \to e\}$$

Automates à pile déterministes

Un AP $P = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ est déterministe si

1. $\delta(q, a, X)$ contient au plus un seul couple pour tout $q \in Q$, $a \in \Sigma \cup \{\varepsilon\}$ et $X \in \Gamma$.
2. Si $\delta(q, a, X)$ est non vide, alors $\delta(q, \varepsilon, X)$ est vide.

Exemple : définissons $L_{wcwr} = \{wcw^R : w \in \{0,1\}^*\}$



Théorème 4

Si L est un langage régulier alors il existe un APD P tel que $L = L(P)$

Preuve : Puisque que A est régulier, il existe un AFD $A = (Q, \Sigma, \delta_A, q_0, F)$ tel que $L = L(A)$. On définit alors l'APD

$$P = (Q_\Sigma, \{Z_0\}, \delta_P, q_0, Z_0, F)$$

avec $\delta_P(q, a, Z_0) = \{\delta_A(q, a), Z_0\}$ pour tout $q \in Q$ et $a \in \Sigma$.

On montre alors par induction sur $|w|$ que

$$(q_0, w, Z_0) \vdash^* (p, \varepsilon, Z_0) \Leftrightarrow \widehat{\delta}_A(q_0, w) = p$$

■

APD et pile vide

Définition 3

On dit que $L \subset \Sigma^*$ *est préfixe ssi*

$$(x, y) \in L^2 \text{ et } \exists w \in \Sigma^* \text{ tel que } x = yw \Rightarrow x = y (w = \varepsilon)$$

Autrement dit, un langage L est préfixe s'il n'existe pas deux mots distincts de L tel que l'un soit préfixe de l'autre

Exemples :

- L_{wcwr} est préfixe
- $\{0\}^*$ n'est pas préfixe

Théorème 5

Un langage $L = N(P)$ *pour un APD* P *ssi* L *est préfixe et* $L = L(P')$ *avec* P' *un APD.*

Classification

On a

langages réguliers $\subset$ $L(APD)$ $\subset$ $L(AP)$ = langages hors-contexte

Il s'agit d'inclusions strictes. En effet

- $L_{wcwr} \in L(APD)$ mais n'est pas régulier
- L_{wwr} est hors-contexte mais $\notin L(APD)$

APD et langages hors-contexte (ambiguité)

Théorème 6

Si $L = N(P)$ *où* P *est un APD, alors* L *a une grammaire non ambigue.*

Théorème 7

Si $L = L(P)$ *où* P *est un APD, alors* L *a une grammaire non ambigue.*