**Elimination des règles $A \to B$**

$A, B \in V$, 1-*production* $A \to B$ : renommer $A \to A$, 1-*production circulaire*

Soient $A, A_1, \ldots, A_n, B \in V$
$A \Rightarrow A_1 \Rightarrow \cdots \Rightarrow A_n \Rightarrow B$ : *chaîne.*

déterminer toutes les variables apparaissant dans une chaîne débutant en $A \in V$.

$C_0 = \{A\}$ et $C_{-1} = \emptyset$.
Pour $i \geq 0$,

$$C_{i+1} = C_i \cup \{C \in V \mid \exists B \in C_i \setminus C_{i-1} : B \to C \in P\}.$$

La procédure s'arrête lorsque
$C_i = C_{i+1} := \mathcal{C}(A)$

**Proposition** Soit $G = (V, \Sigma, P, S)$ une grammaire essentiellement monotone. Il existe une grammaire équivalente $G'$ ne contenant aucune 1-production. De plus, cette grammaire peut être obtenue de manière effective.

**Preuve**

Si $A \to B \in P$, alors dans $G'$ les règles qui ont pour premier membre $A$ sont de la forme $A \to w$ où

$$w \notin V \text{ et } \exists B \in \mathcal{C}(A) : B \to w \in P.$$

Remarque : $A \in \mathcal{C}(A)$. Si $A \to w \in P$ avec $w \notin V$, alors $A \to w$ est une règle de $G'$.

Si $A$ est une variable n'apparaissant dans aucun premier membre des 1-productions de $G$, les règles correspondantes de $G$ et de $G'$ sont identiques.

Les variables, l'alphabet des symboles terminaux et le symbole initial de $G'$ coïncident avec ceux de $G$.

## Exemple

$$\begin{array}{rcl} S' & \rightarrow & S \mid \varepsilon \\ S & \rightarrow & ACA \mid CA \mid AA \mid CA \mid A \mid C \\ A & \rightarrow & aAa \mid B \mid C \mid aAa \mid aa \\ B & \rightarrow & bB \mid b \\ C & \rightarrow & cS \mid c. \end{array}$$

$$\mathcal{C}(S') = \{S', S, A, B, C\},\ \mathcal{C}(S) = \{S, A, B, C\}$$

$$\mathcal{C}(A) = \{A, B, C\}$$

$$\begin{array}{rcl} S' & \rightarrow & \underbrace{\varepsilon}_{S'} \mid \underbrace{ACA \mid CA \mid AA \mid CA}_{S} \mid \underbrace{aAa \mid aAa \mid aa}_{A} \mid \\ & & \underbrace{bB \mid b}_{B} \mid \underbrace{cS \mid c}_{C} \\ S & \rightarrow & \underbrace{ACA \mid CA \mid AA \mid CA}_{S} \mid \underbrace{aAa \mid aAa \mid aa}_{A} \mid \\ & & \underbrace{bB \mid b}_{B} \mid \underbrace{cS \mid c}_{C} \\ A & \rightarrow & \underbrace{aAa \mid aAa \mid aa}_{A} \mid \underbrace{bB \mid b}_{B} \mid \underbrace{cS \mid c}_{C} \\ B & \rightarrow & bB|b \\ C & \rightarrow & cS|c. \end{array}$$

**Elimination des symboles inutiles**
$x \in V \cup \Sigma$ est *utile* si

$$S \Rightarrow^* uxv \Rightarrow^* w \in \Sigma^*$$

**Première partie** : $T \subseteq V$ variables générant des symboles terminaux

$$T_0 = \{A \in V \mid \exists w \in \Sigma^* : A \to w \in P\}.$$

Pour $i \geq 0$,

$$T_{i+1} = T_i \cup \{A \in V \mid \exists w \in (T_i \cup \Sigma)^* : A \to w \in P\}.$$

**Deuxième partie** : symboles accessibles

$A \in V$ est *accessible* si $S \Rightarrow^* uAv$

$A_0 = \{S\}$ et $A_{-1} = \emptyset$.
Pour $i \geq 0$,

$$A_{i+1} = A_i \cup \{B \in V \mid \exists C \in A_i \setminus A_{i-1}, u, v \in (V \cup \Sigma)^* : C \to uBv\}.$$

Soit $G = (V, \Sigma, P, S)$ une grammaire. Il existe une grammaire équivalente $G'$ ne contenant aucun symbole inutile. De plus, cette grammaire peut être obtenue de manière effective.

## Exemple

$$
\begin{array}{rcl}
S & \rightarrow & AC \mid BS \mid B \\
A & \rightarrow & aA \mid aF \\
B & \rightarrow & CF \mid b \\
C & \rightarrow & cC \mid D \\
D & \rightarrow & aD \mid BD \mid C \\
E & \rightarrow & aA \mid BSA \\
F & \rightarrow & bB \mid b.
\end{array}
$$

$$T_0 = \{B, F\},\ T_1 = \{S, A, B, F\},\ T_2 = \{S, A, B, E, F\} =$$

$$
\begin{array}{rcl}
S & \rightarrow & BS \mid B \\
A & \rightarrow & aA \mid aF \\
B & \rightarrow & b \\
E & \rightarrow & aA \mid BSA \\
F & \rightarrow & bB \mid b.
\end{array}
$$

$$A_0 = \{S\},\ A_1 = \{S, B\}.$$

$$
\begin{array}{rcl}
S & \rightarrow & BS \mid B \\
B & \rightarrow & b.
\end{array}
$$

**Remarque** On ne peut pas inverser l'ordre !

$$\begin{array}{rcl} S & \rightarrow & a \mid AB \\ A & \rightarrow & b \\ B & \rightarrow & B. \end{array}$$

$T = \{S, A\}$, ainsi une première réduction donne

$$\begin{array}{rcl} S & \rightarrow & a \\ A & \rightarrow & b. \end{array}$$

Puisque $A$ est inaccessible, $S \rightarrow a$.

Par contre, si on cherche d'abord les symboles accessibles
$A_1 = \{S, A, B\}$ : aucun symbole n'est inaccessible.
Ensuite $T = \{S, A\}$

$$\begin{array}{rcl} S & \rightarrow & a \\ A & \rightarrow & b \end{array}$$

**Forme normale de Chomsky**
$G = (V, \Sigma, P, S)$ est sous *forme normale de Chomsky* si les règles de $G$ sont toutes de l'une des formes suivantes :

- $A \to BC$ où $A \in V$, $B, C \in V \setminus \{S\}$,
- $A \to a$ où $A \in V$, $a \in \Sigma$,
- $S \to \varepsilon$.

$\Rightarrow$ Arbre d'analyse BINAIRE

Soit $G = (V, \Sigma, P, S)$ une grammaire hors contexte. On peut construire de manière effective une grammaire équivalente $G'$ mise sous forme normale de Chomsky.

**Preuve**
$G$ : essentiellement monotone ,
aucune 1-production, aucun symbole inutile.

- $S \to \varepsilon$,

- $A \to a$ où $A \in V$, $a \in \Sigma$,

- $A \to w$ où $A \in V$, $w \in ((V \cup \Sigma) \setminus \{S\})^*$ et $|w| \geq 2$.

$$A \to w = w_1 A_1 w_2 \cdots w_n A_n w_{n+1}, w_i \in \Sigma^*, A_i \in V \backslash \{S\}.$$

$$A \to W_{1,1} \cdots W_{1,\ell_1} A_1 \cdots A_n W_{n+1,1} \cdots W_{n+1,\ell_{n+1}}$$

$$\begin{array}{rcl} W_{1,1} & \to & w_{1,1} \\ & \vdots & \\ W_{n+1,\ell_{n+1}} & \to & w_{n+1,\ell_{n+1}} \end{array}$$

$$A \to A_1 \cdots A_n, \ n \geq 3,$$

$$\begin{array}{rcl} A & \to & A_1 B_1 \\ B_1 & \to & A_2 B_2 \\ & \vdots & \\ B_{n-2} & \to & A_{n-1} A_n. \end{array}$$

## Exemple

$$
\begin{array}{rcl}
S & \rightarrow & SaB \mid aB \\
B & \rightarrow & bB \mid \varepsilon.
\end{array}
$$

$$
\begin{array}{rcl}
S' & \rightarrow & S \\
S & \rightarrow & SaB \mid aB \mid Sa \mid a \\
B & \rightarrow & bB \mid b.
\end{array}
$$

$$
\begin{array}{rcl}
S' & \rightarrow & SaB \mid aB \mid Sa \mid a \\
S & \rightarrow & SaB \mid aB \mid Sa \mid a \\
B & \rightarrow & bB \mid b
\end{array}
$$

$$
\begin{array}{rcl}
S' & \rightarrow & SAB \mid AB \mid SA \mid A \\
S & \rightarrow & SAB \mid AB \mid SA \mid A \\
B & \rightarrow & B'B \mid b \\
A & \rightarrow & a \\
B' & \rightarrow & b.
\end{array}
$$

$$
\begin{array}{rcl}
S' & \rightarrow & ST \mid AB \mid SA \mid A \\
S & \rightarrow & ST \mid AB \mid SA \mid A \\
B & \rightarrow & B'B \mid b \\
A & \rightarrow & a \\
B' & \rightarrow & b \\
T & \rightarrow & AB.
\end{array}
$$

## Forme normale de Greibach

$G = (V, \Sigma, P, S)$ est *sous forme normale de Greibach* si les règles de $G$ sont toutes de l'une des formes suivantes :

- $A \to aA_1 \cdots A_n$ avec $A \in V$, $a \in \Sigma$, $A_i \in V \setminus \{S\}$,
- $A \to a$ avec $A \in V$ et $a \in \Sigma$,
- $S \to \varepsilon$.

**Lemme de la pompe**

Bar-Hillel : Soit $L$ un langage hors contexte. Il existe $p \in N \setminus \{0\}$ tel que tout mot $z \in L$ de longueur $|z| \geq p$ peut s'écrire $z = uvwxy$, $u, v, w, x, y \in \Sigma^*$ avec $|vwx| < p$, $vx \neq \varepsilon$ et pour tout $n \in N$,

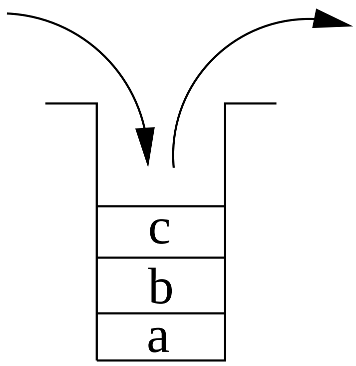
$$uv^n wx^n y \in L.$$

<u>Lemme</u> : Soit $G = (V, \Sigma, P, S)$ une grammaire hors contexte mise sous forme normale de Chomsky. Si la dérivation $A \Rightarrow^* w$, $w \in \Sigma^*$, a un arbre d'analyse de hauteur $n$, alors $|w| \leq 2^{n-1}$.

<u>Corollaire</u> : Soit $G = (V, \Sigma, P, S)$ une grammaire hors contexte mise sous forme normale de Chomsky. Si $S \Rightarrow^* w$ avec $w \in \Sigma^*$ et $|w| \geq 2^n$, alors l'arbre d'analyse associé à cette dérivation est de hauteur au moins $n + 1$.

**Automates à pile**

$\Pi$




tester si la pile est vide

empiler

dépiler

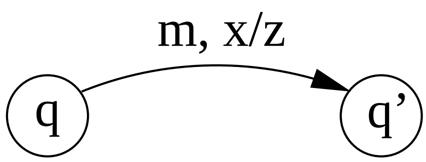
$A = (Q, \Sigma, \Pi, \delta, q_0, F)$

$$\delta \subset Q \times \Sigma^* \times \Pi^* \times Q \times \Pi^*$$

*configuration* : $[q, w, p]$ de $Q \times \Sigma^* \times \Pi^*$

$$[q, w, p] \vdash [q', w', p']$$

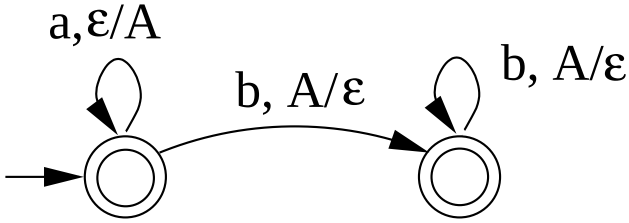
$$w = mw',\ p = xy,\ p' = z^R y,\ (q, m, x, q', z) \in \delta.$$




$\vdash^*$

$$L(\mathcal{A}) = \{w \in \Sigma^* \mid \exists q \in F : [q_0, w, \varepsilon] \vdash^* [q, \varepsilon, \varepsilon]\}.$$

**Exemple**




$$[1, aabb, \varepsilon] \vdash [1, abb, A] \vdash [1, bb, AA] \vdash [2, b, A] \vdash [2, \varepsilon, \varepsilon].$$

Automate à pile *déterministe* : au plus une transition résultant de chaque configuration.

Automate à pile est *atomique* ou *élémentaire*

- $(p, \varepsilon, \varepsilon, q, \varepsilon)$ : changement d'état

- $(p, \sigma, \varepsilon, q, \varepsilon)$ : lecture d'une lettre $\sigma \in \Sigma$

- $(p, \varepsilon, \alpha, q, \varepsilon)$ : dépilement d'$\alpha \in \Pi$

- $(p, \varepsilon, \varepsilon, q, \alpha)$ : empilement d'$\alpha \in \Pi$.

**Grammaire $\Rightarrow$ Automate à pile**

Soit $G = (V, \Sigma, P, S)$ une grammaire hors contexte. L'automate à pile $A = (Q, \Sigma, \Pi, \delta, q_0, F)$

- $Q = \{q_0, f\}$,

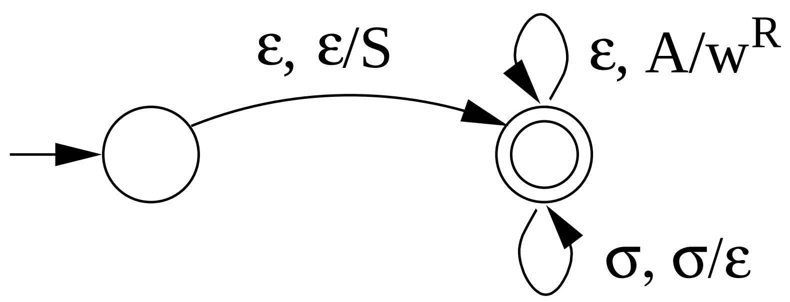
- $\Pi = V \cup \Sigma$,

- $F = \{f\}$ et

- la relation de transition $\delta$ est donnée par

$$\{(q_0, \varepsilon, \varepsilon, f, S)\} \cup \{(f, \varepsilon, A, f, w^R) \mid A \to w \in P\}$$
$$\cup \{(f, \sigma, \sigma, f, \varepsilon) \mid \sigma \in \Sigma\},$$

accepte exactement le langage $L(G)$.



**Preuve**

**1.** Si $u \in \Sigma^*$, $v \in (V \cup \Sigma)^*$ et

$$[f, u, S] \vdash^* [f, \varepsilon, v],$$

alors il existe une dérivation à gauche $S \Rightarrow^* uv$
Récurrence sur le nombre $m$ de transitions.

Si $m = 0$, alors $u = \varepsilon$ et $S = v$, $S \Rightarrow^* S$.

OK pour $m = t$, OK ? pour $m = t + 1$ ?

transitions : $(\sigma, \sigma/\varepsilon)$ ou $(\varepsilon, A/w^R)$,
pour passer de $[f, u, S]$ à $[f, \varepsilon, v]$, au moins une transition $(\varepsilon, A/w^R)$. La dernière fois :

$$[f, u, S] \vdash^* [f, r, Av'] \vdash [f, r, wv'] \vdash^* [f, \varepsilon, v]$$

$u = u'r$ et $wv' = rv$

$$[f, u, S] = [f, u'r, S] \vdash^* [f, \varepsilon\, r, Av'] = [f, r, Av']$$

$$[f, u', S] \vdash^* [f, \varepsilon, Av'].$$

par hypothèse de récurrence : il existe une dérivation à gauche

$$S \Rightarrow^* u'Av'.$$

Or $u'$ appartient à $\Sigma^*$. Ainsi, si on applique la règle $A \to w$, on conserve une dérivation à gauche et on obtient

$$S \Rightarrow^* u'Av' \Rightarrow u'wv' = u'rv = uv.$$

**2.** S'il existe une dérivation à gauche $S \Rightarrow^* uv$ avec $u \in \Sigma^*$ et $v \in V(V \cup \Sigma)^*$,
alors $[f, u, S] \vdash^* [f, \varepsilon, v]$. Récurrence sur la longueur de la dérivation (idem).

**3.** $L(A) \subset L(G)$.
$[q_0, w, \varepsilon] \vdash^* [f, \varepsilon, \varepsilon]$.
$[q_0, w, \varepsilon] \vdash [f, w, S] \vdash^* [f, \varepsilon, \varepsilon]$. Vu **1**, il existe une dérivation à g. tq $S \Rightarrow^* w$

**4.** $L(G) \subset L(A)$.
Il existe une dérivation à g. tq $S \Rightarrow^* w$.
Dernière règle de la forme $A \to y$
$S \Rightarrow^* xAz \Rightarrow xyz = w$ avec $x, z \in \Sigma^*$.
Vu **2**, $[f, x, S] \vdash^* [f, \varepsilon, Az]$ donc
$[q_0, w = xyz, \varepsilon] \vdash [f, xyz, S] \vdash^* [f, yz, Az]$
et puisque $(\varepsilon, A/y^R)$ est une transition
$[f, yz, Az] \vdash [f, yz, yz] \vdash^* [f, \varepsilon, \varepsilon]$

**Automate à pile $\Rightarrow$ Grammaire**

$A = (Q, \Sigma, \Pi, \delta, q_0, F)$

$\delta \subset Q \times (\Sigma \cup \{\varepsilon\}) \times (\Pi \cup \{\varepsilon\}) \times Q \times (\Pi \cup \{\varepsilon\})$.

Ajout de règles :
si $(q_i, s, \varepsilon, q_j, p) \in \delta$, $s \in \Sigma \cup \{\varepsilon\}, p \in \Pi \cup \{\varepsilon\}$, alors pour tout $A \in \Pi$, on ajoute les transitions $(q_i, s, A, q_j, Ap)$.

Construction d'une grammaire :
$G = (V, \Sigma, P, S)$

$V = \{S\} \cup \{\langle q_i, A, q_j \rangle \mid q_i, q_j \in Q, A \in \Pi \cup \{\varepsilon\}\}$.

- $\mathbf{S} \rightarrow \langle \mathbf{q_0}, \varepsilon, \mathbf{q} \rangle$, pour tout $q \in F$,

- pour chaque transition de la forme

  $(q_i, s, A, q_j, B),\ s \in \Sigma \cup \{\varepsilon\}, A, B \in \Pi \cup \{\varepsilon\}$,

  on considère les règles

  $$\langle \mathbf{q_i}, \mathbf{A}, \mathbf{q} \rangle \rightarrow \mathbf{s} \langle \mathbf{q_j}, \mathbf{B}, \mathbf{q} \rangle$$

  pour tout $q \in Q$,

- pour chaque transition de la forme

  $$(q_i, s, A, q_j, AB),\ s \in \Sigma \cup \{\varepsilon\}, A, B \in \Pi,$$

  on considère les règles

  $$\langle \mathbf{q_i}, \mathbf{A}, \mathbf{q} \rangle \rightarrow \mathbf{s} \langle \mathbf{q_j}, \mathbf{B}, \mathbf{q'} \rangle \langle \mathbf{q'}, \mathbf{A}, \mathbf{q} \rangle$$

  pour tous $q, q' \in Q$,

- $\langle \mathbf{q}, \varepsilon, \mathbf{q} \rangle \rightarrow \varepsilon$ pour tout $q \in Q$.

**Exemple**

$$(1, a, \varepsilon, 1, A),\ (1, b, A, 2, \varepsilon),\ (2, b, A, 2, \varepsilon).$$

On ajoute $(1, a, A, 1, AA)$

| | | | |
|---|---|---|---|
| $S$ | $\to$ | $\langle 1, \varepsilon, 2\rangle$ | |
| $\langle 1, \varepsilon, 1\rangle$ | $\to$ | $a\ \langle 1, A, 1\rangle$ | $(1, a, \varepsilon, 1, A)$ |
| $\langle 1, \varepsilon, 2\rangle$ | $\to$ | $a\ \langle 1, A, 2\rangle$ | |
| $\langle 1, A, 1\rangle$ | $\to$ | $b\ \langle 2, \varepsilon, 1\rangle$ | $(1, b, A, 2, \varepsilon)$ |
| $\langle 1, A, 2\rangle$ | $\to$ | $b\ \langle 2, \varepsilon, 2\rangle$ | |
| $\langle 2, A, 1\rangle$ | $\to$ | $b\ \langle 2, \varepsilon, 1\rangle$ | $(2, b, A, 2, \varepsilon)$ |
| $\langle 2, A, 2\rangle$ | $\to$ | $b\ \langle 2, \varepsilon, 2\rangle$ | |
| $\langle 1, A, 1\rangle$ | $\to$ | $a\ \langle 1, A, 1\rangle\ \langle 1, A, 1\rangle$ | $(1, a, A, 1, AA)$ |
| $\langle 1, A, 1\rangle$ | $\to$ | $a\ \langle 1, A, 2\rangle\ \langle 2, A, 1\rangle$ | |
| $\langle 1, A, 2\rangle$ | $\to$ | $a\ \langle 1, A, 1\rangle\ \langle 1, A, 2\rangle$ | |
| $\langle 1, A, 2\rangle$ | $\to$ | $a\ \langle 1, A, 2\rangle\ \langle 2, A, 2\rangle$ | |
| $\langle 1, \varepsilon, 1\rangle$ | $\to$ | $\varepsilon$ | |
| $\langle 2, \varepsilon, 2\rangle$ | $\to$ | $\varepsilon$ | |

$$[1, aabb, \varepsilon] \vdash [1, abb, A] \vdash [1, bb, AA] \vdash [2, b, A] \vdash [2, \varepsilon, \varepsilon].$$

$$\begin{aligned} S &\Rightarrow \langle 1, \varepsilon, 2\rangle \Rightarrow a\langle 1, A, 2\rangle \Rightarrow aa\langle 1, A, 2\rangle\langle 2, A, 2\rangle \\ &\Rightarrow aab\langle 2, \varepsilon, 2\rangle\langle 2, A, 2\rangle \Rightarrow aab\langle 2, A, 2\rangle \\ &\Rightarrow aabb\langle 2, \varepsilon, 2\rangle \Rightarrow aabb. \end{aligned}$$

**Preuve**
**1.** $L(A) \subseteq L(G)$. Si $[q_i, w, A] \vdash^* [q_j, \varepsilon, \varepsilon]$ avec $A \in \Pi \cup \{\varepsilon\}$, alors $\langle q_i, A, q_j\rangle \Rightarrow^* w$. Récurrence sur la longueur $\ell$ de la suite de configurations.

$\ell = 0$ : $q_i = q_j$, $w = \varepsilon$, $A = \varepsilon$ et $\langle q_i, \varepsilon, q_i\rangle \rightarrow \varepsilon \in P$.

OK pour $\ell = n$, OK ? pour $\ell = n + 1$.

$[q_i, w, A] \vdash^* [q_j, \varepsilon, \varepsilon]$ deux décomp. possibles
**Premier cas** :
$[q_i, w, A] \vdash [q_k, v, B] \vdash^* [q_j, \varepsilon, \varepsilon]$
$w = sv$ et $(q_i, s, A, q_k, B) \in \delta$, $s \in \Sigma \cup \{\varepsilon\}$.
Par hyp. de récurrence : $\langle q_k, B, q_j\rangle \Rightarrow^* v$.
à la transition $(q_i, s, A, q_k, B)$ correspond la règle $\langle q_i, A, q_j\rangle \rightarrow s\langle q_k, B, q_j\rangle$.
$\langle q_i, A, q_j\rangle \Rightarrow s\langle q_k, B, q_j\rangle \Rightarrow^* sv = w$.
**Second cas** :
$[q_i, w, A] \vdash [q_k, v, BA] \vdash^* [q_\ell, y, A] \vdash^* [q_j, \varepsilon, \varepsilon]$,
$(q_i, s, A, q_k, AB) \in \delta$, $s \in \Sigma \cup \{\varepsilon\}$ et $w = sv$, $v = xy$.

règle $\langle q_i, A, q_j\rangle \rightarrow s\langle q_k, B, q_\ell\rangle\langle q_\ell, A, q_j\rangle$. Puisque $[q_k, v, BA] \vdash^* [q_\ell, y, A]$ et que $v = xy$, on en tire que $[q_k, x, B] \vdash^* [q_\ell, \varepsilon, \varepsilon]$. Donc par hypothèse de récurrence, on a

$$\langle q_k, B, q_\ell\rangle \Rightarrow^* x \text{ et } \langle q_\ell, A, q_j\rangle \Rightarrow^* y.$$

D'où

$$\langle q_i, A, q_j\rangle \Rightarrow s\langle q_k, B, q_\ell\rangle\langle q_\ell, A, q_j\rangle \Rightarrow^* sxy = w.$$

D'où $L(A) \subseteq L(G)$ car $S \rightarrow \langle q_0, \varepsilon, q\rangle$, $q \in F$.

**2.** $L(G) \subseteq L(A)$. Si $\langle q_i, A, q_j\rangle \Rightarrow^* w$, $w \in \Sigma^*$, $A \in \Pi \cup \{\varepsilon\}$, alors $[q_i, w, A] \vdash^* [q_j, \varepsilon, \varepsilon]$. Récurrence sur la longueur de la dérivation.

long.$= 1$, les règles donnant un symbole terminal sont $\langle q, \varepsilon, q\rangle \rightarrow \varepsilon$ et $[q_i, \varepsilon, \varepsilon] \vdash^* [q_i, \varepsilon, \varepsilon]$.

OK pour long.$= n$, OK ? pour long.$= n + 1$

**Premier cas** :
Si la première règle appliquée est de la forme $\langle q_i, A, q_j\rangle \rightarrow s\langle q_k, B, q_j\rangle$, alors on a

$$\langle q_i, A, q_j\rangle \Rightarrow s\langle q_k, B, q_j\rangle \Rightarrow^* sv = w.$$

Par hyp. de récurrence, $[q_k, v, B] \vdash^* [q_j, \varepsilon, \varepsilon]$.
La règle $\langle q_i, A, q_j\rangle \rightarrow s\langle q_k, B, q_j\rangle$ provient de $(q_i, s, A, q_k, B) \in \delta$ et $[q_i, sv, A] \vdash [q_k, v, B]$.

**Second cas** :
première règle $\langle q_i, A, q_j\rangle \rightarrow s\langle q_k, B, q_m\rangle\langle q_m, A, q_j\rangle$.

$$\langle q_i, A, q_j\rangle \Rightarrow s\langle q_k, B, q_m\rangle\langle q_m, A, q_j\rangle \Rightarrow^* w$$

avec

$$\langle q_k, B, q_m\rangle \Rightarrow^* x,\ \ \langle q_m, A, q_j\rangle \Rightarrow^* y \text{ et } w = sxy.$$

On conclut en appliquant deux fois l'hypothèse de récurrence.

**Corollaire** :
Un langage est algébrique si et seulement si il est accepté par un automate à pile.

## Intersection de langages algébriques

$$L = \{a^n b^n \mid n \in N\} c^*, \; M = a^* \{b^n c^n \mid n \in N\}$$

$$L \cap M = \{a^n b^n c^n \mid n \in N\}$$

## Complémentaire

$$L \cap M = \Sigma^* \setminus ((\Sigma^* \setminus L) \cup (\Sigma^* \setminus M)),$$

Théorème : Soient $R \subseteq \Sigma^*$ un langage régulier et $L \subseteq \Sigma^*$ un langage algébrique. Le langage $L \cap R$ est algébrique.

## Preuve

$A = (Q, q_0, F, \Sigma, \delta)$ AFD
et $A' = (Q', \Sigma, \Pi, \delta', q'_0, F')$ élémentaire
Soit l'automate à pile

$$\mathcal{P} = (Q \times Q', \Sigma, \Pi, \tau, (q_0, q'_0), F \times F')$$

$$((q_i, q'_i), \sigma, x, (q_j, q'_j), y) \in \tau$$

$$\text{si } \delta(q_i, \sigma) = q_j \text{ et } (q'_i, \sigma, x, q'_j, y) \in \delta'$$

et

$((q_i, q'_i), \varepsilon, x, (q_i, q'_j), y) \in \tau$ si $(q'_i, \varepsilon, x, q'_j, y) \in \delta'$.

L'ensemble des langages algébriques est stable par morphisme.

Soient $L \subseteq \Sigma^*$ un langage algébrique généré par $G = (V, \Sigma, P, S)$ et $h : \Sigma \to \Gamma^*$ un morphisme. Nous supposerons de plus que les trois alphabets $V$, $\Sigma$ et $\Gamma$ sont deux à deux distincts. Considérons la grammaire $G' = (V \cup \Sigma, \Gamma, P', S)$ où

$$P' = P \cup \{\sigma \to h(\sigma) \mid \sigma \in \Sigma\}.$$

$A = (Q, \Sigma, \Pi, \delta, q_0, F)$ élémentaire
$\Pi = \{\pi_1, \ldots, \pi_k\}$

$$\Phi = \Sigma \cup \{e_1, \ldots, e_k, d_1, \ldots, d_k\}$$

Si $p \in \Pi^*$

- $\forall \sigma \in \Sigma$, $\sigma.p = p$,
- $\forall i \in \{1, \ldots, k\}$, $e_i.p = \pi_i\, p$,
- si $p = \pi_i\, p'$, alors $d_i.p = p'$.

Si $x = x_1 \cdots x_n \in \Phi^*$,

$$x.p = x_1.(\cdots(x_{n-1}.(x_n.p))\cdots)$$

En particulier, $xy.p = x.(y.p)$, $x, y \in \Phi^*$.
Remarque : $d_2 e_1.p\ \nexists$.

$$D_A = \{x \in \Phi^* \mid x.\varepsilon = \varepsilon\}.$$

**Proposition** : Le langage $D_A$ est algébrique et généré par

$S \to Sd_1Se_1S \,|\, \cdots \,|\, Sd_kSe_kS \,|\, \sigma_1 S \,|\, \cdots \,|\, \sigma_m S \,|\, \varepsilon.$

**preuve**
**1.** Soit $w \in D_A$. Montrons que $S \Rightarrow^* w$ par récurrence sur $|w|$.

Si $w = \varepsilon$ OK.

OK pour $|w| \leq l$, OK ? pour $|w| = l + 1$
Si $w \in \Sigma^{\ell+1}$, OK.
Sinon $w$ contient un symbole $e_i$ (le plus à droite)

$$w = ue_iz,\ u \in \Phi^*, z \in \Sigma^*.$$

à ce $e_i$ correspond exactement un symbole $d_i$ (car $w \in D_A$). Ainsi,

$$w = xd_iye_iz,\ \text{avec } x, y \in \Phi^*.$$

De là, $x, y, z \in D_A$. par hyp. de récurrence :
$S \Rightarrow^* x$, $S \Rightarrow^* y$, $S \Rightarrow^* z$
$S \Rightarrow Sd_iSe_iS \Rightarrow^* xd_iye_iz = w$.

**2.** Si $S \Rightarrow^* w$, $w \in \Phi^*$, alors $w \in D_A$.
$\forall p \in \Pi^*$, on pose $S.p = p$.
Ppar récurrence sur la longueur de la dérivation, si $S \Rightarrow^* w$, $w \in (\Phi \cup \{S\})^*$, alors $w.p = p$ $\forall p \in \Pi^*$.

long.$= 0$ : $S.p = p$ OK

OK pour long.$\leq k$, OK ? long.$= k+1$

En mettant en évidence la dernière règle appliquée,

$$S \Rightarrow^* w_1 S w_2 \Rightarrow w.$$

**a)** Si règle : $S \to S d_i S e_i S$, alors $w = w_1 S d_i S e_i S w_2$

$$\begin{aligned}
& w_1 S d_i S e_i S w_2.p \\
= \ & w_1 S d_i S e_i S.(w_2.p) = w_1 S d_i S e_i.(w_2.p) \\
= \ & w_1 S d_i S.(\pi_i(w_2.p)) = w_1 S d_i.(\pi_i(w_2.p)) \\
= \ & w_1 S.(w_2.p) = (w_1 S w_2).p = p
\end{aligned}$$

**b)** Si règle : $S \to \sigma_i S$, alors $w = w_1 \sigma_i S w_2$ et

$$w_1 \sigma_i S w_2.p = w_1 \sigma_i S.(w_2.p) = w_1 \sigma_i.(w_2.p)$$

$$= w_1.(w_2.p) = (w_1 S w_2).p = p.$$

**c)** Si règle : $S \to \varepsilon$, $w = w_1 w_2$ et

$$w_1 w_2.p = (w_1 S w_2).p = p.$$

**Théorème de Chomsky-Schützenberger**
Soit $L \subseteq \Sigma^*$ un langage accepté par un automate à pile $A$. Il existe un morphisme $h : \Phi \to \Sigma^*$ et un langage régulier $R \subseteq \Phi^*$ tel que

$$L = h(D_A^R \cap R).$$

En particulier, $L$ est algébrique.

**Preuve**
On suppose $A$ élémentaire.

- $(p, \varepsilon, \varepsilon, q, \varepsilon)$ devient $p \overset{\varepsilon}{\to} q$,

- $(p, \sigma_i, \varepsilon, q, \varepsilon)$ devient $p \overset{\sigma_i}{\to} q$,

- $(p, \varepsilon, \pi_i, q, \varepsilon)$ devient $p \stackrel{e_i}{\to} q$,

- $(p, \varepsilon, \varepsilon, q, \pi_i)$ devient $p \stackrel{d_i}{\to} q$.

On a donc un automate fini
$\to R \subseteq \Phi^*$ langage régulier.

Soit $w \in \Sigma^*$ accepté par l'automate à pile $A$

$$[q_0, w, \varepsilon] \vdash^* [q, \varepsilon, \varepsilon], \; q \in F.$$

A ce mot, correspond donc un chemin dans $A$ de label $W \in \Phi^*$ débutant en $q_0$ et aboutissant en $q \in F$.

Puisque, pour être accepté par l'automate, la suite de configurations débute et se finit par une pile vide, il est clair que $W^R$ appartient à $D_A$.

De plus, $W$ appartient à $R$. On retrouve $w$ en appliquant à $W$ le morphisme

$$h : \Phi \to \Sigma^* : \begin{cases} \sigma_i \mapsto \sigma_i \\ e_i \mapsto \varepsilon \\ d_i \mapsto \varepsilon. \end{cases}$$

Réciproquement, si $W \in \Phi^*$ appartient à $D_A \cap R$, alors $h(W)$ est accepté par l'automate.

**Exemple**