

Les Langages Algébriques

S. Mazouz & B. Laichi

FEI-USTHB

Département Informatique

2019-2020

campusvirtuel.usthb.dz

Plan

- 1) Introduction
- 2) Les Grammaires Algébriques
- 3) Les Automates à Pile
- 4) Propriétés des Langages Algébriques

Introduction

Les langages réguliers :

- sont simples
- possèdent les bonnes propriétés requises théoriques et pratiques
- ...mais pouvoir d'expression faible

En effet, il n'existe pas une grammaire régulière pour générer les mots du langage $\{a^n b^n / n \geq 0\}$.

Besoin de langages plus expressifs :

LANGAGES ALGEBRIQUES

Grammaires Algébriques

- Proposées par Chomsky en 1956 pour la description des langages naturels
- Exploitées pour la description des langages de programmation

Définition (Grammaire algébrique) :

Une grammaire $G=(T, N, S, P)$ est une grammaire **algébrique** (on dit aussi à contexte libre) si et seulement si toutes ses règles sont de la forme :

$$A \rightarrow \alpha \text{ avec } A \in N \text{ et } \alpha \in (T \cup N)^*.$$

Exemple : La grammaire $G=(\{a, b\}, \{S\}, S, \{S \rightarrow aSb/\varepsilon\})$ est algébrique. Elle génère le langage $\{a^n b^n / n \geq 0\}$.

Grammaires Algébriques

Définition (Langage algébrique) : Soit X un alphabet et L un langage sur X^* . Le langage L est dit **algébrique** s'il existe une **grammaire algébrique** G telle que $L(G)=L$.

Exemple : Le langage $L=\{w \in \{a, b\}^* / w = w^R\}$ (l'ensemble des mots palindromes sur l'alphabet $\{a, b\}$) est algébrique car il est engendré par la grammaire algébrique :

$G = (\{a, b\}, \{S\}, S, \{S \rightarrow aSa / bSb / a / b / \varepsilon\})$

Remarque :

- Tout langage régulier est algébrique.
- La réciproque est fausse. En effet, le langage $\{a^n b^n / n \geq 0\}$ est algébrique mais non régulier.

Grammaires Algébriques

Définition (Dérivation gauche/droite) : On dit qu'un mot s'obtient par **dérivation gauche** (respectivement par **dérivation droite**) s'il est obtenu à partir de l'axiome en **dérivant toujours le non-terminal le plus à gauche** (respectivement le plus à droite). ($A \rightarrow \alpha$ / $A \in N$ et $\alpha \in (T \cup N)^*$).

Exemple : Soit $G = (\{a, b\}, \{S, A, B\}, S, P)$ avec P :
 $S \rightarrow bA / aB / \varepsilon$ $A \rightarrow aS / bAA$ $B \rightarrow bS / aBB$

Dérivation gauche du mot **aababb** :

$S \Rightarrow a\mathbf{B} \Rightarrow aa\mathbf{BB} \Rightarrow aab\mathbf{SB} \Rightarrow aaba\mathbf{BB} \Rightarrow aabab\mathbf{SB} \Rightarrow aabab\mathbf{B}$
 $\Rightarrow aababb\mathbf{S} \Rightarrow \mathbf{aababb}$

$S \Rightarrow a\mathbf{B} \Rightarrow aa\mathbf{BB} \Rightarrow aab\mathbf{SB} \Rightarrow aab\mathbf{B} \Rightarrow aaba\mathbf{BB} \Rightarrow aabab\mathbf{SB}$
 $\Rightarrow aabab\mathbf{B} \Rightarrow aababb\mathbf{S} \Rightarrow \mathbf{aababb}$

Grammaires Algébriques

Exemple : Soit $G=(\{a, b\}, \{S, A, B\}, S, P)$ avec $P :$
 $S \rightarrow bA / aB / \varepsilon \quad A \rightarrow aS / bAA \quad B \rightarrow bS / aBB$

Une **dérivation droite** du mot **aababb**

$S \Rightarrow a\mathbf{B} \Rightarrow aa\mathbf{B}\mathbf{B} \Rightarrow aaBb\mathbf{S} \Rightarrow aa\mathbf{B}b \Rightarrow aab\mathbf{S}b \Rightarrow aaba\mathbf{B}b$
 $\Rightarrow aabab\mathbf{S}b \Rightarrow \mathbf{aababb}$

Grammaires Algébriques

Définition (Arbre de dérivation) :

Soit une grammaire $G=(T, N, S, P)$. Un **arbre de dérivation** d'un **mot w** appartenant à $L(G)$ est un arbre dont :

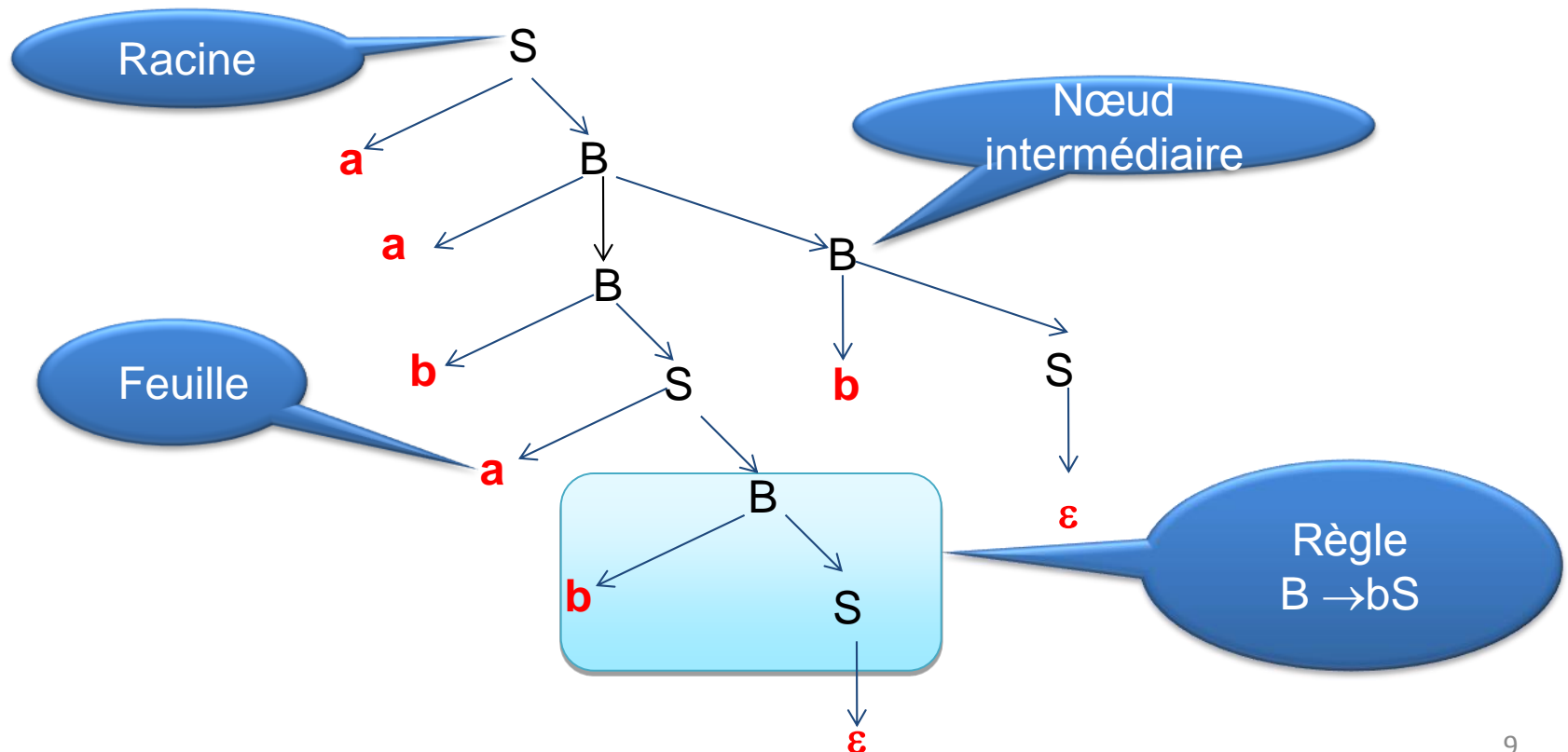
- La **racine** est étiquetée par l'axiome.
- Les **feuilles** sont étiquetées par des terminaux ou le mot vide.
- Les **nœuds intermédiaires** sont étiquetés par des non-terminaux.
- Le passage d'un nœud interne à ses fils correspond à une règle : si le **nœud interne est étiqueté** par **A** et **ses fils sont étiquetés** par **X_1, \dots, X_n** (de gauche vers la droite) alors **$A \rightarrow X_1 \dots X_n$** est une règle de P .
- La lecture des étiquettes des feuilles de **gauche à droite** donne le mot w .

Grammaires Algébriques

Exemple : Soit $G = (\{a, b\}, \{S, A, B\}, S, P)$ avec P :

$$S \rightarrow bA / aB / \varepsilon \quad A \rightarrow aS / bAA \quad B \rightarrow bS / aBB$$

Arbre de dérivation du mot **w=aababb**



Grammaires Algébriques

Définition (mot ambiguë) : Un mot est ambiguë s'il possède plusieurs arbres de dérivation différents.

Remarque : L'ambiguïté d'un mot implique l'existence de deux ou plusieurs dérivations gauche (droites) pour ce mot.

Définition (Grammaire ambiguë) : Une grammaire est ambiguë si elle génère au moins un mot ambiguë.

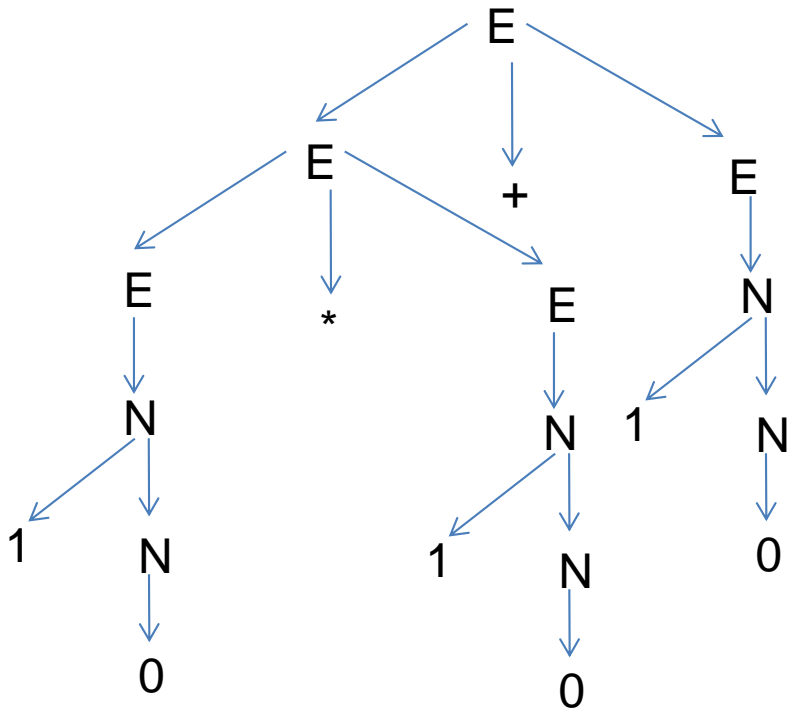
Exemple: La grammaire donnée précédemment est ambiguë car le mot **aababb** est ambiguë. Il possède deux arbres de dérivation différents.

Grammaires Algébriques

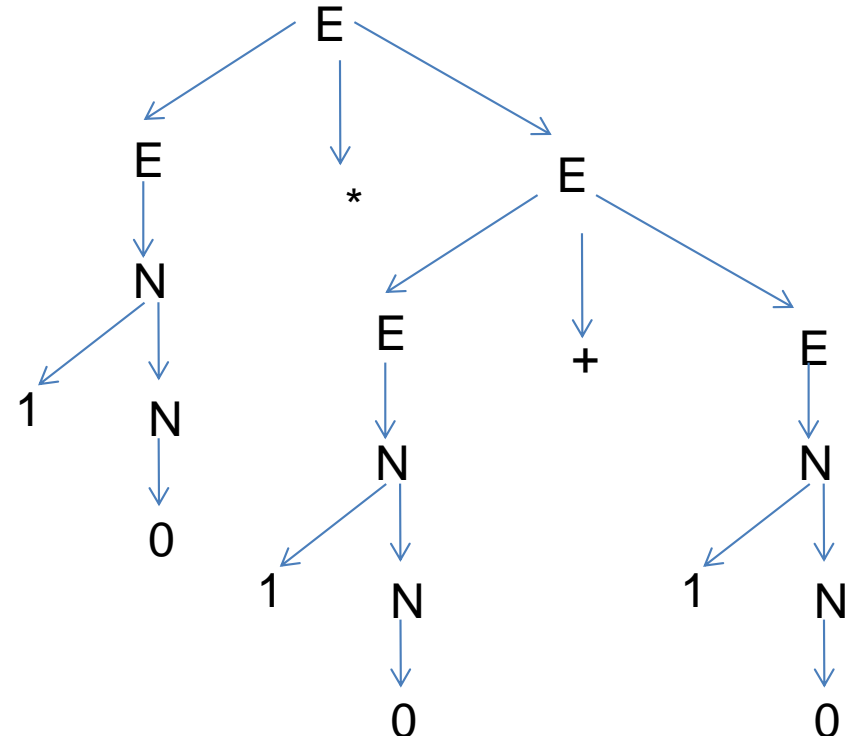
Exemple : Soit la grammaire $G = (\{+, *, 0, \dots, 9\}, \{E, N\}, E, P)$ où P :

$$E \rightarrow E + E / E * E / N$$
$$N \rightarrow 0N / \dots / 9N / 0 / \dots / 9$$

Le mot **10*10+10** possède deux arbres de dérivation différents



La valeur calculée est 110



La valeur calculée est 200

Grammaires Algébriques

Remarque : Pour un même langage, certaines grammaires sont ambiguës, d'autres non.

Un langage de programmation doit avoir une grammaire non ambiguë car l'existence d'arbres de dérivations différents pour un même programme signifie qu'il y a plusieurs façons d'analyser ce programme, avec en général des résultats très différents.

Définition (Ambiguïté inhérente) : On dit qu'un langage algébrique a une **ambiguïté inhérente** si toute grammaire qui l'engendre est ambiguë.

Remarque : Il n'existe pas d'algorithme qui permet de répondre à la question : est-ce que G est ambiguë?

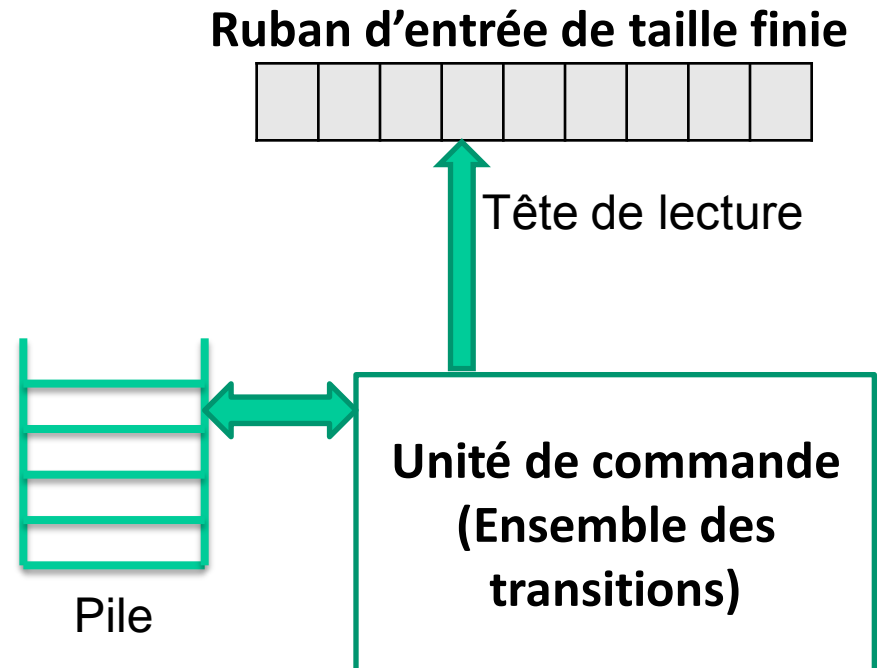
Néanmoins, il est possible de répondre à cette question pour certaines grammaires.

Automates à Pile

Les automates à pile sont les reconnaisseurs des langages algébriques.

Un automate à pile est un automate fini :

- autorisant les ε -transitions,
- avec la particularité de disposer d'une **mémoire auxiliaire** comme unité de stockage, à savoir **une pile**.



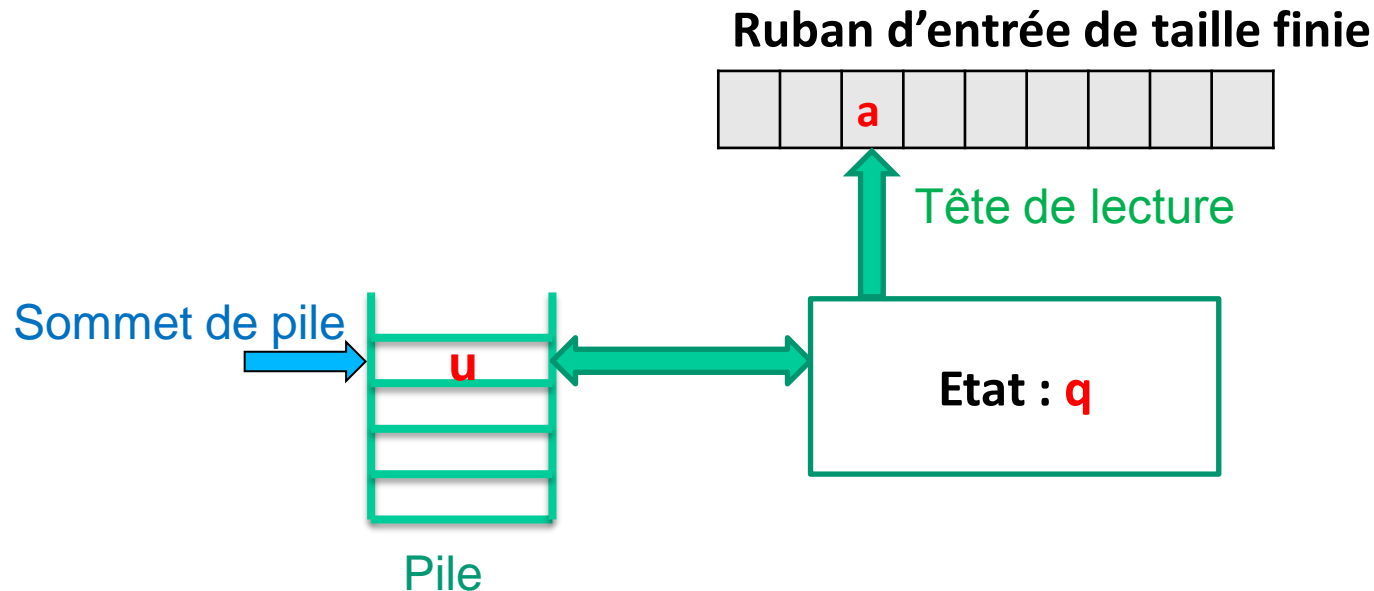
Automates à Pile

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

- Σ est un ensemble fini de l'alphabet d'entrée.
- Γ est un ensemble fini de l'alphabet auxiliaire.
- $Z_0 \in \Gamma$ est le symbole initial de la pile. $\Sigma \cup \Gamma$ constitue l'alphabet du langage de la pile.
- Q est un ensemble fini d'états et $q_0 \in Q$ est l'état initial
- $F \subseteq Q$ est l'ensemble des états finaux.

Automates à Pile

- δ est une fonction de transition qui prend comme argument un triplet (u, q, a) où :
- u est le **sommet de la pile**
 - q est **un état de Q**
 - a est un **symbole d'entrée de Σ ou le mot vide.**



Remarque Si **$a = \epsilon$** alors l'automate effectue une **transition spontanée** : c'est une transition sans lecture.

Automates à Pile

- δ est une fonction de transition qui prend comme argument un triplet (u, q, a) où :
- u est le sommet de la pile
 - q est un état de Q
 - a est un symbole d'entrée de Σ ou le mot vide.

qui retourne une **paire** (α, p) où

- α est un mot sur $(\Sigma \cup \Gamma)^*$
- et p le nouvel état

L'exécution d'une transition $\delta(u, q, a) = (\alpha, p)$ modifie le contenu de la pile selon le mot α comme suit :

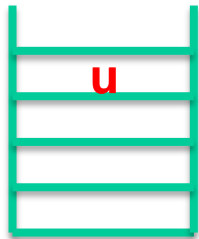
Automates à Pile

1) Si $\alpha = v_1 \dots v_n$ ie $\delta(u, q, a) = (v_1 \dots v_n, p)$ où $v_i \in \Sigma \cup \Gamma$

Ruban d'entrée de taille finie

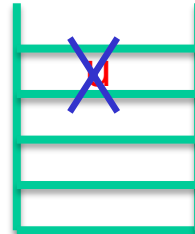


Tête de lecture

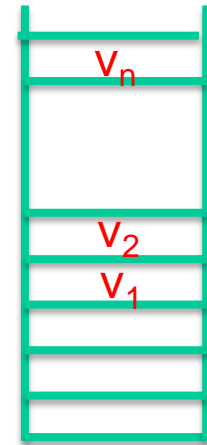


Pile

Dépiler le
sommet de pile



Empiler le mot
 $v_1 \dots v_n$



Le sommet de pile est v_n

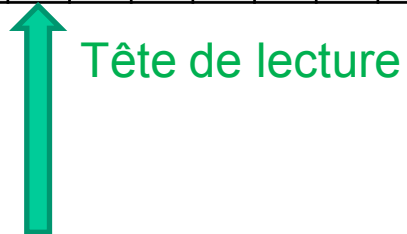
Si $\alpha = v_1 \dots v_n$ où $v_i \in \Sigma \cup \Gamma$ alors le sommet de pile est dépilé
ensuite on empile v_1, \dots et v_n dans cet ordre.

Le sommet de pile contiendra v_n .

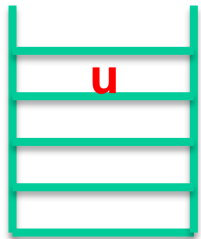
Automates à Pile

2) Si $\alpha = u$ ie $\delta(u, q, a) = (u, p)$

Ruban d'entrée de taille finie

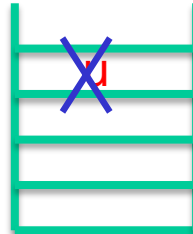


Tête de lecture

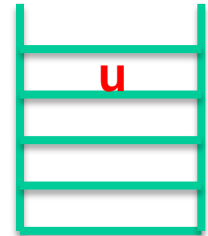


Pile

→
Dépiler le
sommet de pile



→
Empiler le mot u

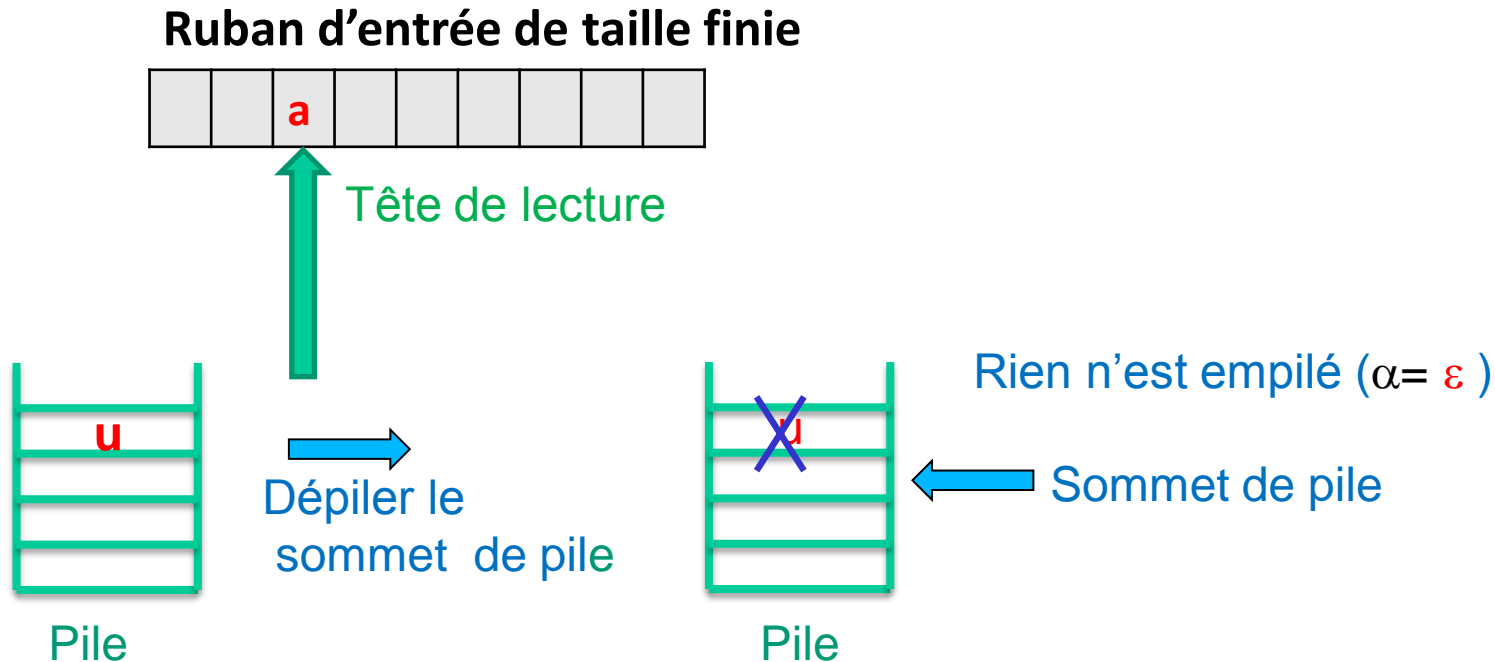


La pile est inchangée

Si $\alpha = u$ alors la pile est inchangée

Automates à Pile

3) Si $\alpha = \epsilon$ ie $\delta(u, q, a) = (\epsilon, p)$



Si $\alpha = \epsilon$ alors on dépile seulement le sommet de pile

Notation : La transition $\delta(u, q, a) = (\alpha, p)$ sera notée $uqa \rightarrow \alpha p$

Automates à Pile

Exemple :

Trouver un automate à pile reconnaissant le langage $\{a^n b^n / n \geq 0\}$.

Principe général :

- Stocker les a dans la pile au fur et à mesure de la lecture.
- Dès qu'il trouve un b, une correspondance doit être faite avec les a stockés dans la pile : chaque b lu est associé à un a dans la pile autrement à chaque lecture d'un b, on dépile un a.
- S'il ne reste aucun a dans la pile ie que le sommet de pile est Z_0 alors le nombre de a empilé est égal au nombre de b lu et transiter vers un état final.

Automates à Pile

Exemple :

Un automate à pile $P=(\Sigma, \Gamma, Z_0, Q, q_0, F, \delta)$

reconnaissant $\{a^n b^n / n \geq 0\}$ est tel que :

$\Sigma=\{a, b\}$, $\Gamma=\{Z_0\}$, $Q=\{q_0, q_1, q_2\}$, $F=\{q_2\}$ et δ est définie par :

N°	Instruction	Commentaire
1	$Z_0 q_0 a \rightarrow Z_0 a q_0$	Lire et empiler le 1 ^{er} a
2	$a q_0 a \rightarrow a a q_0$	Lire et empiler tous les a suivants
3	$a q_0 b \rightarrow q_1$	dépiler le dernier a empilé à la rencontre du 1 ^{er} b et on change d'état pour ne plus autoriser la lecture de a
4	$a q_1 b \rightarrow q_1$	dépiler un a à chaque lecture d'un b
5	$Z_0 q_1 \rightarrow Z_0 q_2$	le nombre de a qui a été empilé dans l'état q_0 est égal au nombre de b lu dans l'état q_1 et on change d'état
6	$Z_0 q_0 \rightarrow Z_0 q_2$	la reconnaissance du mot vide

Automates à Pile

Les étapes de la reconnaissance du mot **aaabbb** sont données dans la tableau suivant :

Pile	État	Sous-mot d'entrée	N° Transition
Z_0	q_0	aaabbb	1
Z_0a	q_0	aabbb	2
Z_0aa	q_0	abb	2
Z_0aaa	q_0	bbb	3
Z_0aa	q_1	bb	4
Z_0a	q_1	b	4
Z_0	q_1		5
Z_0	q_2		

Automate à pile P :

$$Z_0q_0a \rightarrow Z_0aq_0 \quad (1)$$

$$aq_0a \rightarrow aaq_0 \quad (2)$$

$$aq_0b \rightarrow q_1 \quad (3)$$

$$aq_1b \rightarrow q_1 \quad (4)$$

$$Z_0q_1 \rightarrow Z_0q_2 \quad (5)$$

$$Z_0q_0 \rightarrow Z_0q_2 \quad (6)$$

Automates à Pile

Définition (Configuration d'une pile) :

- si un automate à pile P est dans l'état q ,
- le contenu de sa pile est $\alpha \in (\Gamma \cup \Sigma)^*$
- et il lui reste à lire le sous-mot w ($w \in \Sigma^*$),

on dira que cet automate se trouve dans la configuration (α, q, w) .

Si le mot à reconnaître est v alors la configuration initiale est :

(Z_0, q_0, v)

Automates à Pile

Définition (Dérivation) :

Une configuration $(\alpha u, q, aw)$ ($u \in \Sigma \cup \Gamma$, $a \in \Sigma \cup \{\varepsilon\}$)
dérive directement $(\alpha\beta, p, w)$, notée :

$$(\alpha u, q, aw) \Rightarrow (\alpha\beta, p, w)$$

si $uqa \rightarrow \beta p$ est une transition de l'automate.

La fermeture réflexive et transitive de la relation \Rightarrow
est notée \Rightarrow^* .

On parle de **dérivation indirecte**, qui correspond à
zéro, **une** ou **plusieurs** dérivations directes
consécutives.

Automates à Pile

Deux modes de reconnaissances :

- par état final
- par pile vide

Nous verrons dans la suite le mode de reconnaissance par état final.

Définition (Langage reconnu par état final) :

Soit $P=(\Sigma, \Gamma, Z_0, Q, q_0, F, \delta)$ un automate à pile. Le langage accepté par P par état final, noté $L(P)$, est :

$$L(P) = \{ w / \exists \alpha \in (\Sigma \cup \Gamma)^* \exists q_F \in F, (Z_0, q_0, w) \Rightarrow^*(\alpha, q_F, \varepsilon) \}.$$

Dans la configuration finale :

- Le contenu de la pile n'a pas d'importance.
- Le mot d'entrée a été entièrement lu.
- L'automate se trouve dans un état final.

Automates à Pile

Exemple : Soit l'automate à pile

$$P=(\{a, b\}, \{Z_0\}, Z_0, \{q_0, q_1\}, q_0, \{q_1\}, \delta) :$$

N°	Instruction	Commentaire
1	$Z_0q_0a \rightarrow Z_0aq_0$	empiler le 1 ^{er} a
2	$aq_0a \rightarrow aaq_0$	empiler tous les a suivants
3	$aq_0b \rightarrow q_1$	dépiler le dernier a empilé à la rencontre du 1 ^{er} b et on change d'état pour ne plus autoriser la lecture de a
4	$aq_1b \rightarrow q_1$	dépiler un a à chaque lecture d'un b

A l'état **q₁**, la pile peut contenir encore des **a** et donc le nombre de **a** empilé est supérieur ou égale au nombre de **b** lu.

Cet automate reconnaît par état final le langage $\{a^n b^m / n \geq m \geq 1\}$

Propriétés des langages algébriques

Théorème :

La classe des langages algébriques est fermée par rapport aux opérations régulières :

union, concaténation et l'étoile.

Si L_1 et L_2 sont des langages algébriques

Alors $L_1 \cup L_2$, $L_1.L_2$ et $(L_1)^*$ sont algébriques

Propriétés des langages algébriques

Union de deux langages algébriques :

L_1 algébrique \Rightarrow il existe un automate à pile

$P_1 = (\Sigma_1, \Gamma_1, Z_{01}, Q_1, q_{01}, \delta_1, F_1)$ reconnaissant L_1 par état final

L_2 algébrique \Rightarrow il existe un automate à pile

$P_2 = (\Sigma_2, \Gamma_2, Z_{02}, Q_2, q_{02}, \delta_2, F_2)$ reconnaissant L_2 par état final

$L_1 \cup L_2$ est reconnu par l'automate à pile $P = (\Sigma, \Gamma, Z_0, Q, q_0, \delta, F)$

- $\Sigma = \Sigma_1 \cup \Sigma_2$
- $\Gamma = \Gamma_1 \cup \Gamma_2 \cup \{Z_0\}$ $Z_0 \notin \Gamma_1 \cup \Gamma_2$
- $Q = Q_1 \cup Q_2 \cup \{q_0\}$ $q_0 \notin Q_1 \cup Q_2$
- $F = F_1 \cup F_2$
- $\delta = \delta_1 + \delta_2 + \{ \delta(Z_0, q_0, \varepsilon) = \{(Z_{01}, q_{01}), (Z_{02}, q_{02})\} \}$

Propriétés des langages algébriques

La classe des langages algébriques n'est pas fermée par rapport à l'intersection et par rapport au complément (L'intersection de deux langages algébriques n'est pas forcément algébrique. Le complément d'un langage algébrique, n'est pas forcément algébrique).

Théorème : si L est algébrique et R est régulier
alors $L \cap R$ est algébrique.

Théorème : si L est algébrique
alors L^R est aussi algébrique.