Langages et Automates
Grammaires

Engel Lefaucheux

Prépas des INP

Langage régulier

Définition récursive sur un alphabet $\Sigma = \{a, b\}$:

- ε, a et b sont des expressions régulières pour $\{\varepsilon\}$, $\{a\}$ et $\{b\}$
- Si r_1 et r_2 sont des expressions régulières générant L_1 et L_2, alors
  - $r_1 \cdot r_2$ génère $L_1 \cdot L_2$
  - $r_1 + r_2$ génère $L_1 \cup L_2$
  - r_1^* génère L_1^*
  - (r_1) est une expression régulière génèrant L_1

Certains langages ne sont pas réguliers.
$\rightarrow$ Lemme de l'étoile

Dépasser les expressions régulières

Les langages réguliers reconnaissent

- des mots issus de lexiques
- des structures simples.

Ils sont insuffisants pour

- les structures de type $\{a^n b^n \mid n \in \mathbb{N}\}$
- le langage naturel
- l'analyse de programmes

Outline

Définition formelle

Une grammaire est un quadruplet (T, N, R, S)

- T : symboles terminaux
  $\longrightarrow$ l'alphabet du langage que nous voulons créer
- N : symboles non-terminaux / temporaires
  $\longrightarrow$ symboles utilisés au cours de la dérivation
- R : ensemble des règles de la dérivation
- S : Axiome
  $\longrightarrow$ symbole de départ

$(\{a, b\}, \{S, A\}, R, S)$ où R est l'ensemble de règles suivant

$S \rightarrow AA$

$A \rightarrow AAA \mid b\ A \mid A\ b \mid a$

Arbre de dérivation

Représentation graphique de la dérivation

- Racine = axiome
- Noeud = Symbole non-terminal
- Feuille = symbole terminal
- Relation parent-enfant = règle

Arbre de dérivation

Représentation graphique de la dérivation

- Racine = axiome
- Noeud = Symbole non-terminal
- Feuille = symbole terminal
- Relation parent-enfant = règle

$S \rightarrow AA$

$A \rightarrow AAA \mid b\ A \mid A\ b \mid a$

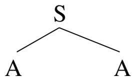
Arbre de dérivation

Représentation graphique de la dérivation

- Racine = axiome
- Noeud = Symbole non-terminal
- Feuille = symbole terminal
- Relation parent-enfant = règle

$S \rightarrow AA$

$A \rightarrow AAA \mid b\ A \mid A\ b \mid a$

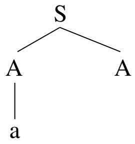


Arbre de dérivation

Représentation graphique de la dérivation

- Racine = axiome
- Noeud = Symbole non-terminal
- Feuille = symbole terminal
- Relation parent-enfant = règle

$S \rightarrow AA$
$A \rightarrow AAA \mid b\ A \mid A\ b \mid a$

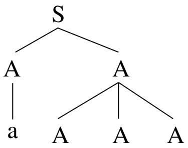


Arbre de dérivation

Représentation graphique de la dérivation

- Racine = axiome
- Noeud = Symbole non-terminal
- Feuille = symbole terminal
- Relation parent-enfant = règle

$S \rightarrow AA$
$A \rightarrow AAA \mid b\ A \mid A\ b \mid a$

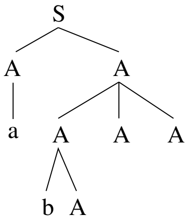


Arbre de dérivation

Représentation graphique de la dérivation

- Racine = axiome
- Noeud = Symbole non-terminal
- Feuille = symbole terminal
- Relation parent-enfant = règle

$S \rightarrow AA$
$A \rightarrow AAA \mid b\ A \mid A\ b \mid a$



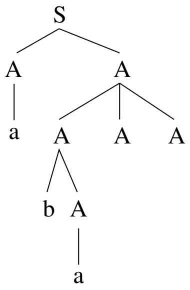
Arbre de dérivation

Représentation graphique de la dérivation

- Racine = axiome
- Noeud = Symbole non-terminal
- Feuille = symbole terminal
- Relation parent-enfant = règle

$S \rightarrow AA$

$A \rightarrow AAA \mid b\ A \mid A\ b \mid a$



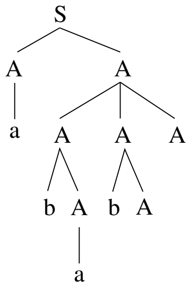
Arbre de dérivation

Représentation graphique de la dérivation

- Racine = axiome
- Noeud = Symbole non-terminal
- Feuille = symbole terminal
- Relation parent-enfant = règle

$S \rightarrow AA$

$A \rightarrow AAA \mid b\ A \mid A\ b \mid a$



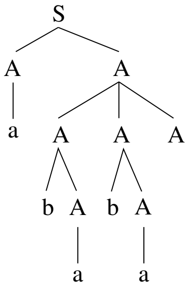
Arbre de dérivation

Représentation graphique de la dérivation

- Racine = axiome
- Noeud = Symbole non-terminal
- Feuille = symbole terminal
- Relation parent-enfant = règle

$S \rightarrow AA$

$A \rightarrow AAA \mid b\ A \mid A\ b \mid a$



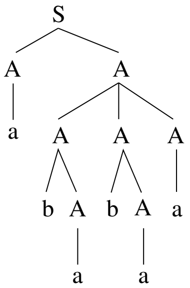
Arbre de dérivation

Représentation graphique de la dérivation

- Racine = axiome
- Noeud = Symbole non-terminal
- Feuille = symbole terminal
- Relation parent-enfant = règle

$S \rightarrow AA$

$A \rightarrow AAA \mid b\ A \mid A\ b \mid a$



Petit exemple de grammaire

On veut construire une PHRASE simple.

$$
\begin{aligned}
PHRASE &\rightarrow SUJET\ \ VERBE\ \ COMPLEMENT\\
SUJET &\rightarrow ludovic \mid pierre \mid nicolas\\
VERBE &\rightarrow mange \mid porte\\
COMPLEMENT &\rightarrow ARTICLE\ \ NOM\ \ ADJECTIF\\
ARTICLE &\rightarrow un \mid le\\
NOM &\rightarrow livre \mid plat \mid chat\\
ADJECTIF &\rightarrow delicieux \mid rouge \mid doux
\end{aligned}
$$

On part de PHRASE, et on remplace les termes en grâce aux règles

Petit exemple de grammaire

Construction d'une *IFSTRUC*

$$
\begin{aligned}
IFSTRUC &\rightarrow if\ \ TEST\ \ then\ BLOCK\ \ else\ BLOCK \\
TEST &\rightarrow VAR\ \leq\ INT \mid TEST\ \ et\ \ TEST \\
BLOCK &\rightarrow INSTANCE \mid INSTANCE\ \ BLOCK \\
VAR &\rightarrow x \mid y \\
INT &\rightarrow 0 \mid 1 \mid INT\ \ INT \\
INSTANCE &\rightarrow incr\ \ VAR \mid decr\ \ VAR \mid IFSTRUC \mid return\ \ VAR
\end{aligned}
$$

Exercices

Quel langage pour les grammaires suivantes commençant par S

1. $S \rightarrow \varepsilon \mid T$
   $T \rightarrow a\ b \mid a\ T\ b$
2. $S \rightarrow \varepsilon \mid A$
   $A \rightarrow a\ B\ C \mid a\ A\ B\ C$
   $C\ B \rightarrow B\ C$
   $A\ B \rightarrow A\ b$
   $a\ B \rightarrow A\ b$
   $B\ B \rightarrow B\ b$
   $C \rightarrow c$

Construisez une grammaire pour le langage des palindromes sur $\Sigma = \{a, b\}$

Outline

Hiérarchie de Chomsky

Type 0 Pas de restrictions sur les règles

Type 1 règles de la forme
$u\ A\ v \ \rightarrow \ u\ w\ v$ avec $A \in N$ et $u, v, w \in (N \cup T)^*$

Type 2 règles de la forme
$A \ \rightarrow \ w$ avec $A \in N$ et $w \in (N \cup T)^*$

Type 3 Toutes les règles sont soit de la forme
$A \ \rightarrow \ a\ B$ ou $A \ \rightarrow \ a$ (grammaire à droite)
soit de la forme
$A \ \rightarrow \ B\ a$ ou $A \ \rightarrow \ a$ (grammaire à gauche)

Hiérarchie de Chomsky

Type 0 Pas de restrictions sur les règles

Type 1 règles de la forme
$u\ A\ v \ \rightarrow\ u\ w\ v$ avec $A \in N$ et $u, v, w \in (N \cup T)^*$

Type 2 règles de la forme
$A \ \rightarrow\ w$ avec $A \in N$ et $w \in (N \cup T)^*$

Type 3 Toutes les règles sont soit de la forme
$A \ \rightarrow\ a\ B$ ou $A \ \rightarrow\ a$ (grammaire à droite)
soit de la forme
$A \ \rightarrow\ B\ a$ ou $A \ \rightarrow\ a$ (grammaire à gauche)

Grammaire de type 0 $\Longleftrightarrow$ Machine de Turing

Hiérarchie de Chomsky

Type 0 Pas de restrictions sur les règles

Type 1 règles de la forme
$u\ A\ v \ \rightarrow\ u\ w\ v$ avec $A \in N$ et $u, v, w \in (N \cup T)^*$

Type 2 règles de la forme
$A \ \rightarrow\ w$ avec $A \in N$ et $w \in (N \cup T)^*$

Type 3 Toutes les règles sont soit de la forme
$A \ \rightarrow\ a\ B$ ou $A \ \rightarrow\ a$ (grammaire à droite)
soit de la forme
$A \ \rightarrow\ B\ a$ ou $A \ \rightarrow\ a$ (grammaire à gauche)

Grammaire de type 0 $\Longleftrightarrow$ Machine de Turing
Grammaire de type ? $\Longleftrightarrow$ Expression régulière

Hiérarchie de Chomsky

Type 0 Pas de restrictions sur les règles

Type 1 règles de la forme
$u\ A\ v \ \rightarrow\ u\ w\ v$ avec $A \in N$ et $u, v, w \in (N \cup T)^*$

Type 2 règles de la forme
$A \ \rightarrow\ w$ avec $A \in N$ et $w \in (N \cup T)^*$

Type 3 Toutes les règles sont soit de la forme
$A \ \rightarrow\ a\ B$ ou $A \ \rightarrow\ a$ (grammaire à droite)
soit de la forme
$A \ \rightarrow\ B\ a$ ou $A \ \rightarrow\ a$ (grammaire à gauche)

Grammaire de type 0 $\Longleftrightarrow$ Machine de Turing
Grammaire de type 3 $\Longleftrightarrow$ Expression régulière

Exercices

Représenter les langages suivant avec une grammaire de type 3

- $baab^*$
- $b(aab)^*$

De quel type est la grammaire
$S \rightarrow aU \mid c$
$U \rightarrow Sb \mid d$
Quel est son langage ?

L_1 et L_2 langages de grammaire G_1 et G_2
Informellement, comment construire une grammaire pour
$L_1 \cup L_2$, $L_1 \cdot L_2$ et L_1^*

Exercices

Représenter les langages suivant avec une grammaire de type 3

- $baab^*$
- $b(aab)^*$

De quel type est la grammaire
$S \rightarrow aU \mid c$
$U \rightarrow Sb \mid d$
Quel est son langage ? $\{a^n cb^n, a^{n+1} db^n \mid n \in \mathbb{N}\}$

L_1 et L_2 langages de grammaire G_1 et G_2
Informellement, comment construire une grammaire pour
$L_1 \cup L_2$, $L_1 \cdot L_2$ et L_1^*