Grammaires

Sommaire

1-définitions

Une **grammaire formelle**, appelée aussi **grammaire de Chomsky**, G est une description de la forme des symboles et des phrases d'un langage noté L(G). Elle est définie par un quadruplet $G = (V_T, V_N, S, R)$ où :

- V_T est un ensemble fini non vide : le **vocabulaire terminal** ou **alphabet** ;
  $L(G) \subseteq V_T^*$
- V_N est un ensemble fini non vide : le **vocabulaire non terminal** (on parle aussi de **variable** ou de **catégorie syntaxique**) ; On a $V_T \cap V_N = \emptyset$ et on note $V = V_T \cup V_N$ le **vocabulaire** de la grammaire ;
- $S \in V_N$ est un symbole non terminal particulier appelé **l'axiome** ;
- R est un ensemble de **règles**, appelées **règles de production** (ou **règles de réécriture**), de la forme $\alpha \rightarrow \beta$ avec $\alpha \in V^+$ et $\beta \in V^*$

Exemples

- G_0=({0, 1}, {S, X}, S, R) avec R :
  - R1 : S → 0X1
  - R2 : 0X → 00X1
  - R3 : X → ε

- G_1=({0, 1}, {S, X}, S, R) avec :
  - R = {S → 0X1 ; 0X → 00X1 ; 0X → 001}
  - ou R = {S → 0X1 ; 0X → 00X1 | 001} (équivalent)
- G_2=({0, 1}, {S}, S, R) avec :
  - R = {S → 0S1 | 01}
- G_3=({0, 1}, {S, X}, S, R) avec :
  - R = {S → 0S | 0X ; X → 1X | 1}

Dérivations

- Soit G = (V_T, V_N, S, R) une grammaire.
- On appelle **dérivation de longueur n** (n ∈ N) toute suite s=u_0, ...,u_n de chaînes de V* telle que u_0 =G⇒ u_1 et ... et u_{n-1} =G⇒ u_n (ce que l'on abrège par u_0 =G⇒ ... =G⇒ u_n)
- a =n⇒ b désigne une dérivation de longueur n.
- a =*⇒ b désigne une dérivation de longueur quelconque. Autrement dit, =*⇒ est la fermeture réflexive et transitive de =G⇒.
- a =+⇒ b désigne une dérivation de longueur supérieure à 1. Autrement dit, =+⇒ est la fermeture transitive de =G⇒.
- Si a =*⇒ b, on dit alors que a **engendre** b ou, symétriquement, que b **dérive de** a. Si de plus a =+⇒ b, on dit "engendre" (resp. "dérive") "de façon non triviale".

- Soit $G = (V_T, V_N, S, R)$ une grammaire :
- Une chaîne a est une forme sententielle (proto-phrase) pour la grammaire G si et seulement si $S \overset{*}{=}\Rightarrow a$.
- Le langage **engendré** par G est l'ensemble des chaînes de symboles terminaux que l'on peut dériver à partir de l'axiome S :
  $L(G)=\{x \in V_T^* \mid S \overset{+}{=}\Rightarrow x\}$.
- Deux grammaires G_1 et G_2 sont dites équivalentes si $L(G_1) = L(G_2)$.

4. Classification des grammaires

- **Type 0 : sans contraintes**

Les règles sont de la forme:

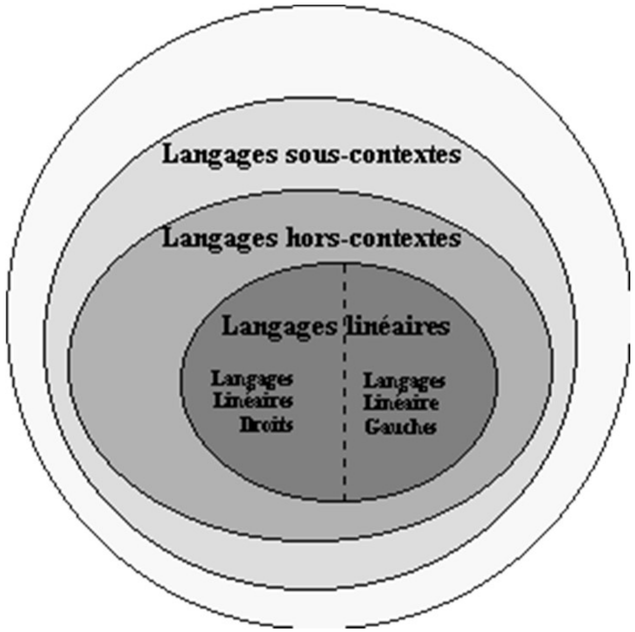
  - \- $a \rightarrow b$ avec $a \in V^+$ et $b \in V^*$.

- Type 1 (**contextuelle,** context sensitive, sensible au contexte): - $a \rightarrow b$ $|a| \leq |b|$.
- \- $aAb \rightarrow aBb$ avec $a, b \in V^*$, $A \in V_N$ et $B \in V^+$ tel que $|B| \geq 1$ (c'est-à-dire que A peut être remplacé par B dans le contexte aAb).
  l'axiome, s'il n'est pas en partie droite d'une autre règle que celle qui le définit, peut engendrer ε. Cette exception est aussi valable pour les types suivants

- **Type 2 : hors-contexte**
  (context free,
  algébrique)
- Une grammaire est de type 2 si elle est de type 1 et si toutes ces productions ont la forme suivante:
  - $A \rightarrow a$ avec $A \in V_N$ et $a \in V^+$.
- **Type 3 : linéaires**
  (régulières)
  [à] droite (resp. [à] gauche):
- Une grammaire est de type 3 si elle est de type 2 et elle est soit linéaire à droite soit linéaire à gauche.
  - $A \rightarrow x_1 \mid x_2B$ (resp. $A \rightarrow x_1 \mid Bx_2$) avec $A, B \in V_N$ et $x_1 \in V_T^+$, $x_2 \in V_T^*$.

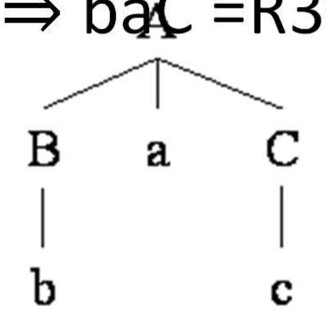
Exemples

- règles $S \rightarrow 0S \mid 1S \mid \varepsilon$ est linéaire droite.
- G_0 et G_4 sont de type 0,
- G_1 est de type 1, G_2 est de type 2 et G_3 est de type 3.



4-3 Arbres de dérivation

Toute phrase du langage L(G) peut être obtenu après un ensemble de dérivation à partir de l'axiome: $S \stackrel{+}{\Rightarrow} a$ ($a \in L(G) \subseteq V_T^*$). Une telle dérivation peut être aussi représentée par un arbre de dérivation (appelé aussi « arbre syntaxique » ou « arbre d'analyse », exemple :Par exemple, en considérant les règles R1 : $A \rightarrow BaC$, R2 : $B \rightarrow b$ et R3 : $C \rightarrow c$, à la dérivation $A \stackrel{R1}{\Rightarrow} BaC \stackrel{R2}{\Rightarrow} baC \stackrel{R3}{\Rightarrow} bac$ correspond l'arbre suivant :



la représentation des dérivations sous forme d'arbres ne mémorise pas l'ordre dans lequel les règles sont appliquées. Par exemple, l'arbre ci-dessus correspond également à la dérivation $A \stackrel{R1}{\Rightarrow} BaC \stackrel{R3}{\Rightarrow} Bac \stackrel{R2}{\Rightarrow} bac$.

Remarques: La racine de l'arbre est l'axiome. Les feuilles sont les symboles terminaux de la phrase.
Une grammaire est dite ambigüe si au moins une phrase de son langage a plus d'un arbre de dérivation.

Exemple

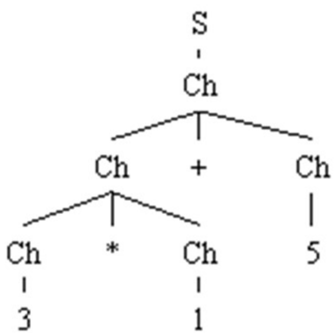
soit la grammaire G_6=({0, 1, 2, 3, 4, 5, 6, 7, 8, 9, +, *}, {S, Ch},S, R}), avec R :

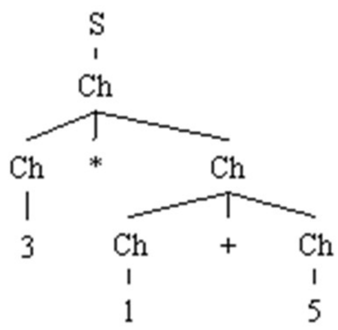
S → Ch

Ch → Ch + Ch | Ch * Ch

Ch → 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

Par cette grammaire, nous pouvons obtenir pour le mot "3*1+5" les arbres de dérivation suivants :



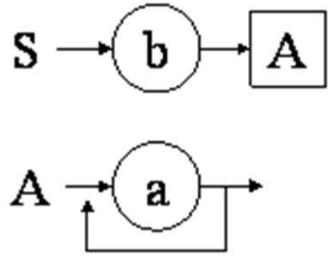


5- représentation des grammaires

- 5.1 **Notation BNF**
- Elle est particulièrement utile pour analyser un langage défini par une grammaire hors-contexte
- Les symboles terminaux sont encadrés par des guillemets ("..." ou <<...>>), les symboles non terminaux ne sont pas distingués
- L'axiome est la partie gauche de la première règle énoncée. La disjonction est notée par une barre verticale (|). La flèche des règles est remplacée par le signe "::=". Par exemple, le tableau suivant illustre la forme BNF pour les règles $\{S \rightarrow bA\ ;\ A \rightarrow aA\ ;\ A \rightarrow a\}$.
  - S ::= "b"A
  - A::="a"A | "a"

5.2 les diagrammes de conway

- Elle décrit graphiquement les grammaires hors-contexte
- La partie gauche des règles est notée à gauche du schéma. Les symboles terminaux sont écrits dans des cercles et les symboles non terminaux sont écrits dans des rectangles.



Exercices et tests :

Exercice 5.1. Pour chacune des grammaires suivantes :

1.Les écrire sous forme BNF

2.Donner les arbres de dérivation associés à certaines phrases de chaque grammaire. dans le cas où la grammaire est de type 2 ou 3. Les représenter à l'aide des diagrammes de Conway

Gex1 = ({a,b,c}, {S}, S, {S → aSbSa | c})

Gex2 = ({a,b}, {S,A}, S, {S → Aa | bA ; A → Sa | bS})

Exercice 5.2. Soit la grammaire ({0,1,2,3,4,5,6,7,8,9,;,-,+},{S,Exp,Ch}, S, R) définie par les règles suivantes :

- S → Exp ; S | Exp
- Exp → Exp + Exp | Exp - Exp | Ch
- Ch → 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

1.Ecrire ces règles sous forme BNF

2.Donner les diagrammes de Conway correspondants