

# Grammaires et Hiérarchie de Chomsky

## Grammaires : Séance 1

Marie-Laure Potet

Grenoble INP-Ensimag

2020-2021

Notes

---

---

---

---

---

---

---

---

---

---

---

---

## Introduction

Notes

---

---

---

---

---

---

---

---

---

---

---

---

## Description de langages

Rappels :

- $V$  un vocabulaire (ensemble fini de symboles),  $L \subseteq V^*$  un langage.
- **Langages réguliers** : langages reconnus par automate ayant un nombre fini d'états, langages descriptibles par expressions régulières (union, concaténation, itération)
- Certains langages ne sont pas réguliers :  $a^n b^n$ . Preuve en montrant qu'il n'existe pas d'automate (lemme de l'étoile)
- Problème d'expressivité : peut-on décrire systématiquement plus de langages ?  
✓ **Les grammaires**
- **Le langage de programmation idéal ça serait quoi ?**

Notes

---

---

---

---

---

---

---

---

---

---

## Intérêt d'un formalisme de description

- Permet de caractériser une classe de langages. Exemples : les langages reconnus par automates d'état fini.
- Permet de donner des algorithmes agissant sur les formalismes. Exemple : algorithme de détermination.
- Permet d'étudier la décidabilité de propriétés sur les formalismes.
  - $P(x)$  Décidable : il existe un algorithme permettant de répondre oui/non à la question " $P(x)$  est vrai" pour tout  $x$ .
  - Indécidable : il n'existe pas d'algorithme

✓ **La hiérarchie de Chomsky. Expressivité versus Décidabilité et Complexité.**

**Exemples de problèmes décidables/indécidables sur les langages réguliers ? sur les programmes ?**

Notes

---

---

---

---

---

---

---

---

---

---

## Grammaires

Notes

---

---

---

---

---

---

---

---

---

---

---

---

## Grammaires : qu'est ce que c'est ?

✓ Similaire à la grammaire d'une langue. Exemple grammaire de l'anglais en français.

- **Un vocabulaire terminal** : ce qui constituera les mots (les phrases) du langage. Exemple : le vocabulaire de l'anglais.
- **Un vocabulaire non terminal** : ce qui permet de parler de la construction des mots (phrases). Les catégories grammaticales.
- **Des règles** permettant de produire les mots (phrases) du langage. Groupe\_Nominal → Article Adjectif Nom.

Notes

---

---

---

---

---

---

---

---

---

---

---

---

## Grammaires : définition formelle

### Définition: (Grammaire)

Une grammaire est un quadruplet  $G = \langle V_T, V_N, S, R \rangle$ , où :

- $V_T$  est un vocabulaire, appelé vocabulaire terminal ;
- $V_N$  est un vocabulaire, appelé vocabulaire non terminal, et tel que  $V_N \cap V_T = \emptyset$ . On pose  $V = V_T \cup V_N$ .
- $S \in V_N$  est appelé l'axiome de la grammaire.
- $R$  est un ensemble de règles de la forme  $u \rightarrow v$  avec :
  - $u \in V^+$
  - $v \in V^*$

Notes

---

---

---

---

---

---

---

---

---

---

## Exemples

Donner une grammaire pour les langages suivants :

- $G_1$  pour  $(ab)^*(ba)^+$
- $G_2$  pour  $a^n b^n$  avec  $n \geq 0$
- $G_3$  pour  $a^n b^n c^n$  avec  $n \geq 1$

Donner  $V_T$ ,  $V_N$ , l'axiome et les règles.

✓ Conventions :

- on fixe  $V_T$ .  $V_N$  est l'ensemble des symboles apparaissant dans les règles et n'appartenant pas à  $V_T$ , l'axiome est le symbole en partie gauche de la première règle.
- Si plusieurs règles  $A \rightarrow w_1, \dots, A \rightarrow w_n$  on peut noter  $A \rightarrow w_1 \mid \dots \mid w_n$ .

Notes

---

---

---

---

---

---

---

---

---

---

## Exemple 1

Donner une grammaire pour les langages suivants :

- $G_1$  pour  $(ab)^*(ba)^+$

Donner  $V_T$ ,  $V_N$ , l'axiome et les règles.

$$\begin{aligned} S &\rightarrow AB \\ A &\rightarrow abA \\ A &\rightarrow \epsilon \\ B &\rightarrow baB \\ B &\rightarrow ba \end{aligned}$$

$V_T = \{a, b\}$ ,  $V_N = \{S, A, B\}$ ,  $S$  est l'axiome.  
ou bien :

$$\begin{aligned} S &\rightarrow AB \\ A &\rightarrow abA \mid \epsilon \\ B &\rightarrow baB \mid ba \end{aligned}$$

Notes

---

---

---

---

---

---

---

---

---

---

## Exemple 2

Donner une grammaire pour les langages suivants :

- $G_2$  pour  $a^n b^n$  avec  $n \geq 0$

Donner  $V_T$ ,  $V_N$ , l'axiome et les règles.

$$\begin{aligned} S &\rightarrow aSb \\ S &\rightarrow \epsilon \end{aligned}$$

$V_T = \{a, b\}$ ,  $V_N = \{S\}$ ,  $S$  est l'axiome.

ou bien :

$$S \rightarrow aSb \mid \epsilon$$

Notes

---

---

---

---

---

---

---

---

---

---

## Exemple 3

Donner une grammaire pour les langages suivants :

- $G_3$  pour  $a^n b^n c^n$  avec  $n \geq 1$

Donner  $V_T$ ,  $V_N$ , l'axiome et les règles.

- (1)  $S \rightarrow abc$
- (2)  $S \rightarrow aSBc$
- (3)  $cB \rightarrow Bc$
- (4)  $bB \rightarrow bb$

$V_T = \{a, b, c\}$ ,  $V_N = \{S, B\}$ ,  $S$  est l'axiome.

Notes

---

---

---

---

---

---

---

---

---

---

---

Notes

---

---

---

---

---

---

---

---

---

---

---

Langage associé à une grammaire

## Relation de dérivation

### Définition: (Relation de dérivation)

- Soit  $G = \langle V_T, V_N, S, R \rangle$  une grammaire.
- Soit  $x, y \in V^*$ . On dit que  $x$  dérive vers  $y$ , noté  $x \Rightarrow y$  si et seulement si il existe une règle  $u \rightarrow v$  et deux chaînes  $\alpha, \beta \in V^*$  telles que  $x = \alpha u \beta$  et  $y = \alpha v \beta$ .

Si on veut être précis on note  $\Rightarrow_r$ , avec  $r$  une règle ou  $\Rightarrow_{p,r}$  avec  $p$  une position dans  $x$  et  $r$  une règle.

- ✓ Ne pas mélanger  $\rightarrow$  (pour les règles) et  $\Rightarrow$  (pour les dérivations).
- ✓ Ne pas mélanger  $\Rightarrow$  (implication logique) et  $\Rightarrow$  (pour les dérivations).

Notes

---

---

---

---

---

---

---

---

---

---

## Dérivation de longueur quelconque

On note  $\Rightarrow^p$  une dérivation de longueur  $p$ . Définie par :

$$\begin{array}{l} u \Rightarrow^0 v \quad \Leftrightarrow \quad u = v \\ u \Rightarrow^1 v \quad \Leftrightarrow \quad u \Rightarrow v \\ u \Rightarrow^{p+1} w \quad \Leftrightarrow \quad \exists v . u \Rightarrow v \wedge v \Rightarrow^p w \end{array}$$

- ✓ Propriété de composition des dérivations :

$$u_1 \Rightarrow^{p_1} v_1 \quad u_2 \Rightarrow^{p_2} v_2$$

$$u_1 u_2 \Rightarrow^{p_1+p_2} v_1 v_2$$

On note  $\Rightarrow^*$  la fermeture réflexive et transitive de  $\Rightarrow$ .

$$\Rightarrow^* = \bigcup_{i \in \mathbb{N}} \Rightarrow^i$$

Notes

---

---

---

---

---

---

---

---

---

---

## Exemple 1

$$V_T = \{a, b\}$$

$$V_N = \{S\}$$

$$R = \begin{cases} S \rightarrow \varepsilon \\ S \rightarrow aSb \end{cases} \quad \text{noté également } S \rightarrow \varepsilon \mid aSb$$

- $S \Rightarrow^1 \varepsilon$
- $S \Rightarrow^1 aSb \Rightarrow^1 ab$  ( $S \Rightarrow^2 ab$ )
- $S \Rightarrow aSb \Rightarrow aaSbb \Rightarrow aaasbbb \Rightarrow aaabbb$  ( $S \Rightarrow^* aaaabbbb$ )

Notes

---

---

---

---

---

---

---

---

---

---

## Exemple 2

$$(1) S \rightarrow abc$$

$$(2) S \rightarrow aSBc$$

$$(3) cB \rightarrow Bc$$

$$(4) bB \rightarrow bb$$

$$\begin{aligned} S &\Rightarrow_2 aSBc \\ &\Rightarrow_1 abcBc \\ &\Rightarrow_3 aabBcc \\ &\Rightarrow_4 aabbcc \end{aligned}$$

Notes

---

---

---

---

---

---

---

---

---

---



# Langage engendré

## Définition: (Langage engendré par une grammaire )

- Soit une grammaire  $G = \langle V_T, V_N, S, R \rangle$ . Le langage engendré par  $G$  est  $L(G) = L(S) = \{x \in V_T^* \mid S \Longrightarrow^* x\}$

**Généralisation.** Soit  $w \in V^*$ . On pose :

$$L(w) = \{x \in V_T^* \mid w \Longrightarrow^* x\}$$

Notes

---

---

---

---

---

---

---

---

---

---

# Langage engendré - Exemples

Exemple. Soit  $V_T = \{a, b\}$  et les règles suivantes :

$$\begin{aligned} S &\rightarrow AB & A &\rightarrow a & A &\rightarrow aA \\ B &\rightarrow \epsilon & B &\rightarrow bB \end{aligned}$$

*$L(A)$  ?  $L(B)$  ?  $L(S)$  ?  $L(ABa)$  ?*

✓ **Propriété :** Deux grammaires  $G_1$  et  $G_2$  sont dites équivalentes ssi elles engendrent le même langage :

$$L(G_1) = L(G_2)$$

Notes

---

---

---

---

---

---

---

---

---

---

## Grammaire : un processus énumératif

⇒ On peut voir les grammaires comme un processus permettant d'énumérer les mots du langage :

1. On part de l'axiome
2. On applique toutes les règles possibles sur toutes les occurrences possibles
3. on réitère le pas 2 sur les mots de dérivation obtenus

⇒ une procédure de semi-décision pour le problème  $w \in L(G)$ .

Exemple grammaire précédente :

pas 0 : S  
pas 1 : AB  
pas 2 : aB, aAB, A, AbB  
pas 3 : a, abB, aaB, aaAb, aA, aAbB, a, aA, abB, aaAB  
pas 4 : ...

aa ∈ L(G) ? ba ∈ L(G) ? Comment décider de l'arrêt ?

Notes

---

---

---

---

---

---

---

---

---

---

Notes

---

---

---

---

---

---

---

---

---

---

## Hiérarchie de Chomsky

## Hierarchie de Chomsky

Noam Chomsky (1928) : linguiste et philosophe. 1950 : théorie des grammaires génératives.

- Une classification des grammaires (et des langages) qui permettra l'étude du compromis expressivité/décision.
- Une restriction sur la forme des règles
- 4 classes de grammaires :
  - Grammaires régulières (classe 3)
  - Grammaires hors-contexte (classe 2)
  - Grammaires sous-contexte (classe 1)
  - Grammaires générales (classe 0)

Notes

---

---

---

---

---

---

---

---

---

---

## Grammaire régulière

Soit  $G = \langle V_T, V_N, S, R \rangle$  une grammaire.  $G$  est dite régulière si et seulement si les règles sont d'une des formes suivantes :

- $A \rightarrow \epsilon$
- $A \rightarrow aB$

avec  $A \in V_N, B \in V_N$  et  $a \in V_T$ .

### ✓ Exemples

Une telle grammaire est dite linéaire à droite. Il existe d'autres façons équivalentes (i.e. engendrant les mêmes langages) de décrire les grammaires régulières (voir TD).

C'est bien la même classe que celle qu'on a déjà vue ! (séance 2 ou 3)

Notes

---

---

---

---

---

---

---

---

---

---

## Grammaire hors-contexte

Soit  $G = \langle V_T, V_N, S, R \rangle$  une grammaire.  $G$  est dite **hors-contexte** si et seulement si les règles sont de la forme :

- $A \rightarrow w$

avec  $A \in V_N$  et  $w \in (V_T \cup V_N)^*$ .

### ✓ Exemples

⇒ Une classe qu'on aime bien ! Bon compromis  
Expressivité/décidabilité

Notes

---

---

---

---

---

---

---

---

---

---

## Grammaire sous-contexte

Soit  $G = \langle V_T, V_N, S, R \rangle$  une grammaire.  $G$  est dite **sous-contexte** si et seulement si les règles sont de la forme :

- $u \rightarrow v$  avec  $|u| \leq |v|$

avec  $u \in V^+$ ,  $v \in V^+$ . Rappel : par définition des grammaires  $u \neq \epsilon$ .

### ✓ Exemples

⇒ La condition sur la taille donnera un algorithme de décision pour  $w \in L(G)$ .

⇒ On énumère tous les mots  $\alpha \in (V_T \cup V_N)^*$  tel que  $S \Longrightarrow^* \alpha$  et  $|\alpha| \leq |w|$ . Si on n'a pas trouvé  $w$  on ne le trouvera pas.

Notes

---

---

---

---

---

---

---

---

---

---

## Grammaire sous-contexte (suite)

- Une définition équivalente (si, si!). Règles de la forme :
  - $uAv \rightarrow uwv$avec  $A \in V_N$  et  $u, v, w$  dans  $V^*$  et  $w \neq \epsilon$ .

Pour les 2 définitions :

Si  $\epsilon$  dans le langage on peut ajouter les règles :

- $Z \rightarrow \epsilon$
- et  $Z \rightarrow S$ ,

avec  $Z$  un nouveau symbole qui devient l'axiome.

Notes

---

---

---

---

---

---

---

---

---

---

## Grammaire générale

Pas de restriction.

✓ Résultats :

- $G$  régulière  $\Rightarrow G$  hors-contexte
- $G$  hors-contexte sans  $\epsilon$ -règle  $\Rightarrow G$  sous-contexte
- $G$  sous-contexte  $\Rightarrow G$  générale

On appelle  $\epsilon$ -règle une règle de la forme  $A \rightarrow \epsilon$ .

✓ Questions :

- Soit  $G$  une grammaire. Peut-on décider de sa classe ?
- Grammaire des grammaires ?

Notes

---

---

---

---

---

---

---

---

---

---

## Classes de langages

⇒ Extension de la notion de classes de grammaires aux langages.

Un langage  $L$  est dit régulier (hors-contexte, sous-contexte, général) si et seulement si il existe une grammaire  $G$  régulière (hors-contexte, sous-contexte, générale) telle que  $L(G) = L$ .

### ✓ Remarques :

- On s'intéresse généralement à la plus petite sous-classe d'un langage
- Prouver  $L(G) = L$  est " complexe ". On verra une manière systématique de faire cette preuve (pour les langages hors-contexte).
- Il n'y a pas d'algorithme pour décider de la classe d'un langage.

Notes

---

---

---

---

---

---

---

---

---

---

## Hierarchie de langages

### ✓ Résultats :

- Extension des implications sur les grammaires aux langages
- Plus généralement  $L$  hors-contexte  $\Rightarrow L$  sous-contexte (preuve en séance 2 ou 3)
- Inclusion stricte des classes de langages
- Les grammaires ne captent pas tous les langages

✓ Voir schéma au tableau

L Hors-contexte non régulier :

L Sous-contexte non Hors-contexte :

L général non Sous-contexte :

L non général :

Notes

---

---

---

---

---

---

---

---

---

---

