



Grammaires formelles : Structure et hierarchie des grammaires

Karën Fort

karen.fort@sorbonne-universite.fr / <https://www.schplaf.org/kf/>

6 novembre 2020



Quelques sources d'inspiration

par ordre d'importance décroissant

- ▶ cours de D. Battistelli (Paris 3), grâce aux notes de C. Riquier (Master 2, Paris 4)
- ▶ cours d'A. Rozenknop (Paris 13)
- ▶ cours de B. Habert (ENS de Lyon)
- ▶ *Language as a cognitive process - Syntax* (Terry Winograd) – Addison Wesley
- ▶ *Introduction à la calculabilité* (Pierre Wolper) – InterEditions, 1991
- ▶ cours en ligne de J-F. Perrot (Paris 6), avec son accord :
<http://pagesperso-systeme.lip6.fr/Jean-Francois.Perrot/inalco/Automates/Cours17.html>

Sources

Points de vue sur les grammaires formelles

Structure

Hiérarchie des grammaires formelles

Pour finir

Différents points de vue sur les grammaires formelles

Une grammaire formelle peut être considérée sous différents aspects selon l'objectif fixé en la définissant :

- ▶ grammaire de **définition** : une grammaire définit un langage (phrases **et** structure)
- ▶ grammaire de **génération** : une grammaire peut engendrer toutes les phrases qu'elle définit
- ▶ grammaire d'**analyse** : doit pouvoir reconnaître si une phrase appartient ou pas au langage qu'elle définit

Propriétés des grammaires formelles

Une grammaire formelle est **à la fois** définitoire, générative et analytique.

C'est dans l'**utilisation** qu'on en fait qu'on peut choisir de ne s'intéresser qu'à l'une de ces propriétés.

Inadéquation d'une grammaire formelle

Une grammaire formelle peut être **inadéquate** selon le point de vue où l'on se place :

- ▶ grammaire de définition : si elle ne définit pas toutes les phrases et rien que les phrases d'un langage donné ou si les descriptions structurales qu'elle en donne sont inexactes ou insuffisantes
- ▶ grammaire de génération : si elle n'engendre pas toutes les phrases qu'elle doit définir ou si elle engendre d'autres phrases parasites ou incorrectes
- ▶ grammaire d'analyse : si elle ne reconnaît pas comme grammaticale une phrase du langage ou si elle en donne une mauvaise analyse (sous la forme d'une structure parasite) ou si elle reconnaît comme grammaticales des phrases qui n'appartiennent pas au langage



Le caractère inadéquate d'une grammaire ne la rend pas forcément inutilisable (dépend du but fixé)

Sources

Points de vue sur les grammaires formelles

Structure

- Rappel sur les arbres

- Grammaires et structures

- Grammaires et ambiguïté

- Équivalences

- Structures et arbres

Hiérarchie des grammaires formelles

Pour finir

Définition

Arbre (théorie des graphes)

« un arbre est un graphe non orienté, acyclique et connexe »

[Wikipédia, Arbre_(graphe), consultée le 12/11/14]

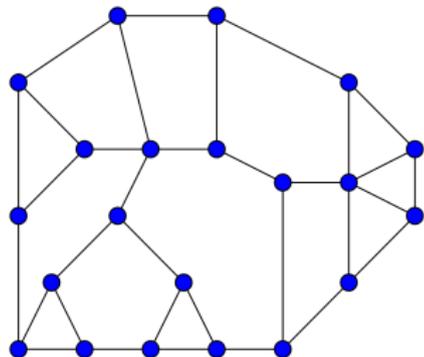
Définition

Arbre (théorie des graphes)

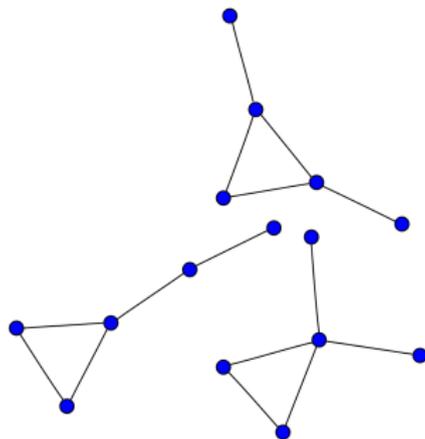
« un arbre est un graphe non orienté, acyclique et connexe »

[Wikipédia, Arbre_(graphe), consultée le 12/11/14]

Graphe connexe (et cyclique) :



Graphe non connexe :



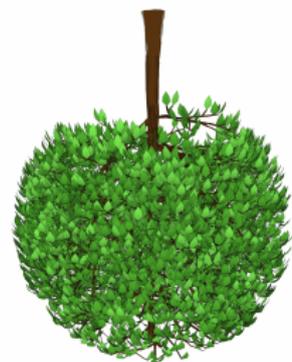
[Koko90]

[Note] De la croissance des arbres

Un arbre. . .

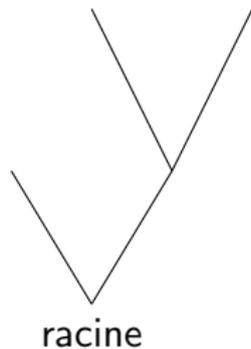


Un arbre informatique

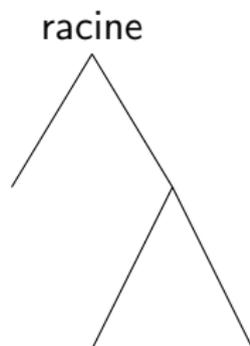


[Note] De la croissance des arbres

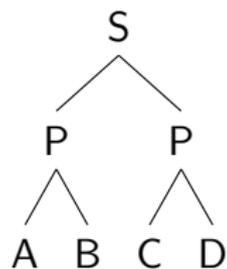
Un arbre. . .



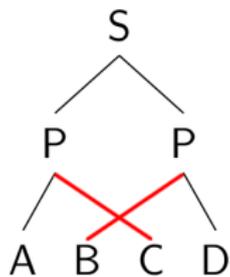
Un arbre informatique



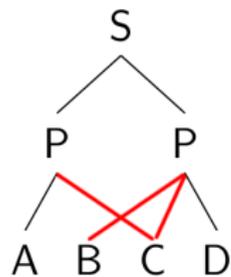
Arbre vs graphe



Arbre



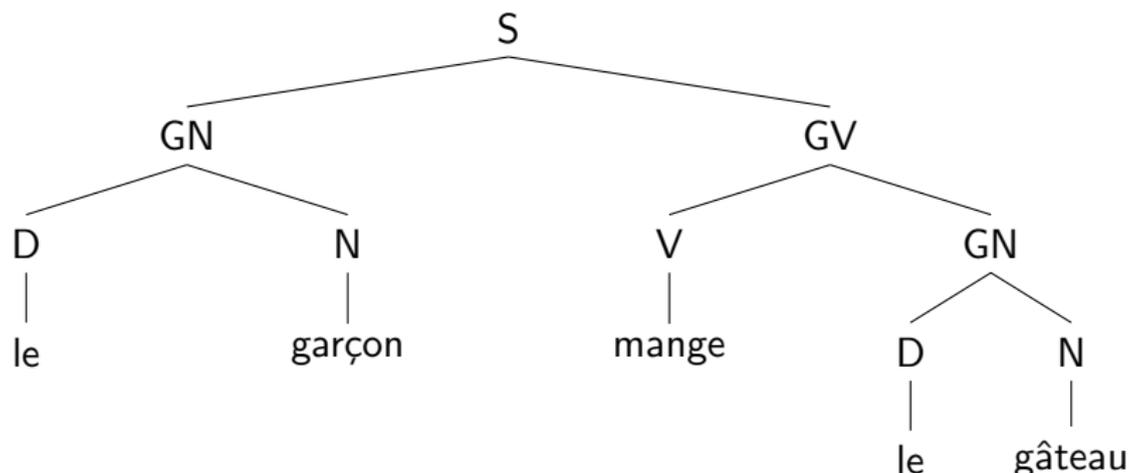
Arbre (non adjacence)



Graphe (cyclique)

Graphe de structure

On peut associer à chaque chaîne c de $L(G)$ un graphe (ici, un arbre) qui illustre la dérivation de c . On dit que ce graphe est la **structure** de c .



Une grammaire formelle ne se limite pas à l'énumération des phrases d'un langage, elle associe à chacune une ou plusieurs structures

S'il te plaît, dessine-moi des constituants

the glorious sun will shine in the winter

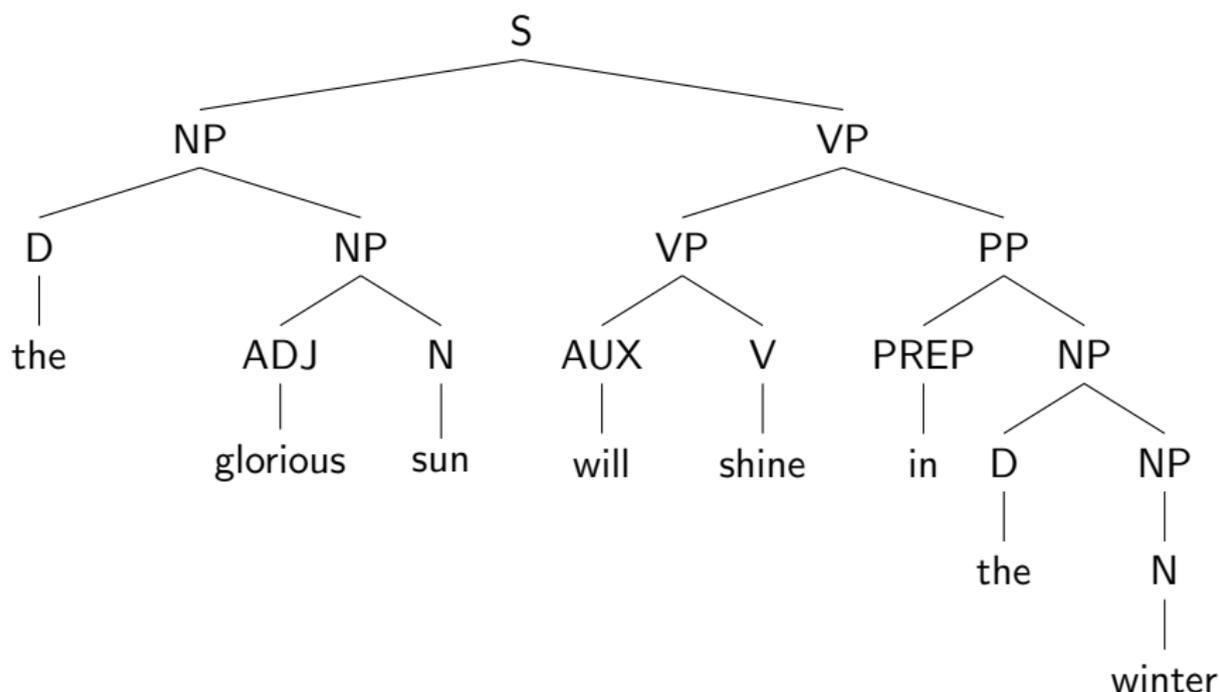
S'il te plaît, dessine-moi des constituants

the glorious sun will shine in the winter

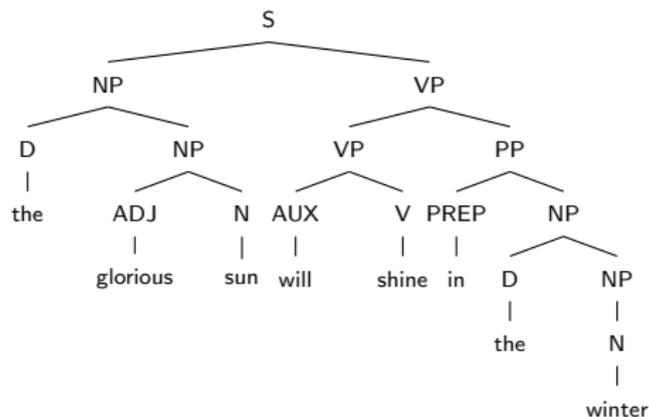
[*S*[*NP* the [*NP* glorious sun]][*VP*[*VP* will shine]][*PP* in [*NP* the [*NP* winter]]]

S'il te plaît, dessine-moi des constituants

[_S[_{NP} the [_{NP} glorious sun]][_{VP}[_{VP} will shine]][_{PP} in [_{NP} the [_{NP} winter]]

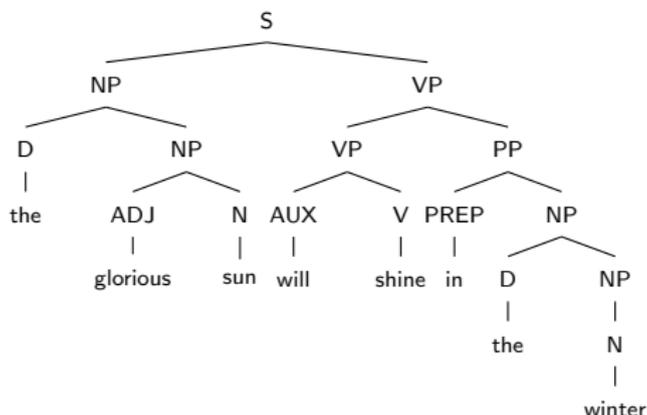


Exercice : décomposition



🗨️ Règles de réécriture permettant de générer cette phrase ?

Exercice : décomposition



{	$S \rightarrow NP VP$	$D \rightarrow the$
	$NP \rightarrow D NP$	$N \rightarrow sun winter$
	$NP \rightarrow ADJ N N$	$V \rightarrow shine$
	$VP \rightarrow VP PP$	$ADJ \rightarrow glorious$
	$VP \rightarrow AUX V$	$AUX \rightarrow will$
	$PP \rightarrow PREP NP$	$PREP \rightarrow in$

Grammaires ambiguës

Définition

Une grammaire formelle peut éventuellement permettre d'associer à une même chaîne c , élément de $L(G)$, **plusieurs structures différentes**, qui résultent de dérivations différentes (pas le résultat d'une simple permutation dans l'ordre des règles).

Dans ce cas, on dit que la grammaire est **ambiguë** et le langage qu'elle définit est lui aussi **ambigu**.

Exemple

Soit la grammaire G définie par :

$$V_N = \{S, A, B\}$$

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

$$P = \left\{ \begin{array}{ll} S \longrightarrow aSb & (1) \\ S \longrightarrow ab & (2) \\ S \longrightarrow AB & (3) \\ A \longrightarrow aA & (4) \\ B \longrightarrow Bb & (5) \\ A \longrightarrow a & (6) \\ B \longrightarrow b & (7) \end{array} \right.$$

 $L(G)$?

Exemple

Soit la grammaire G définie par :

$$V_N = \{S, A, B\}$$

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

$$P = \begin{cases} S \rightarrow aSb & (1) \\ S \rightarrow ab & (2) \\ S \rightarrow AB & (3) \\ A \rightarrow aA & (4) \\ B \rightarrow Bb & (5) \\ A \rightarrow a & (6) \\ B \rightarrow b & (7) \end{cases}$$

 $L(G)$? $L(G) = \{a^n b^m \mid n \geq 1 \text{ et } m \geq 1\}$

Exemple

Soit la grammaire G définie par :

$$V_N = \{S, A, B\}$$

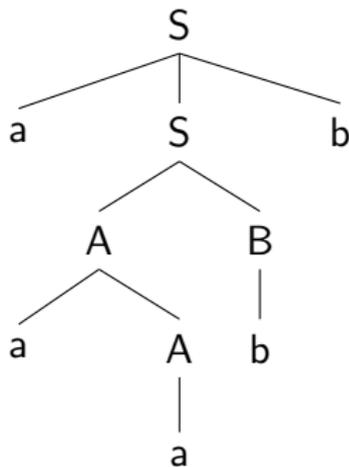
$$V_T = \{a, b\}$$

$$P = \left\{ \begin{array}{ll} S \longrightarrow aSb & (1) \\ S \longrightarrow ab & (2) \\ S \longrightarrow AB & (3) \\ A \longrightarrow aA & (4) \\ B \longrightarrow Bb & (5) \\ A \longrightarrow a & (6) \\ B \longrightarrow b & (7) \end{array} \right.$$

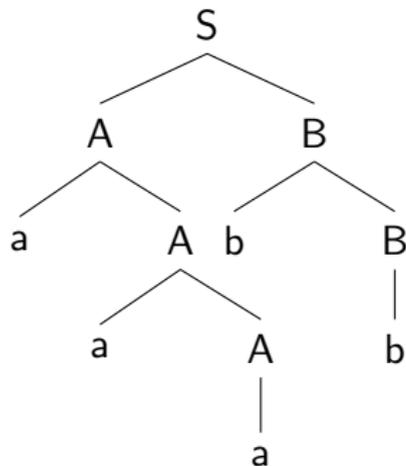
 Soit $c = aaabb$, combien de dérivations sont possibles pour l'obtenir ?

Exemple : dérivations pour $c = aaabb$

$S \rightarrow aSb \rightarrow aABb \rightarrow$
 $aAbb \rightarrow aaAbb \rightarrow aaabb$



$S \rightarrow AB \rightarrow aAB \rightarrow$
 $aaAB \rightarrow aaaB \rightarrow aaaBb \rightarrow$
 $aaabb$



Nous obtenons au moins 2 structures différentes, la grammaire est donc **ambiguë** (toutes les chaînes engendrées par cette grammaire ne sont pas forcément ambiguës)

Équivalence

Équivalence de grammaires (rappel)

On dit que deux grammaires G_1 et G_2 sont **faiblement équivalentes** si :

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

Équivalence de langages

Deux langages qui contiennent les mêmes chaînes et de plus ont les mêmes structures associées aux mêmes chaînes sont dits **fortement équivalents**.

Conclusion sur les structures

Une grammaire associe toujours une ou plusieurs structures à chacune des phrases qu'elle définit

 Selon les règles de la grammaire, ces structures ne sont **pas forcément** des arbres !

Sources

Points de vue sur les grammaires formelles

Structure

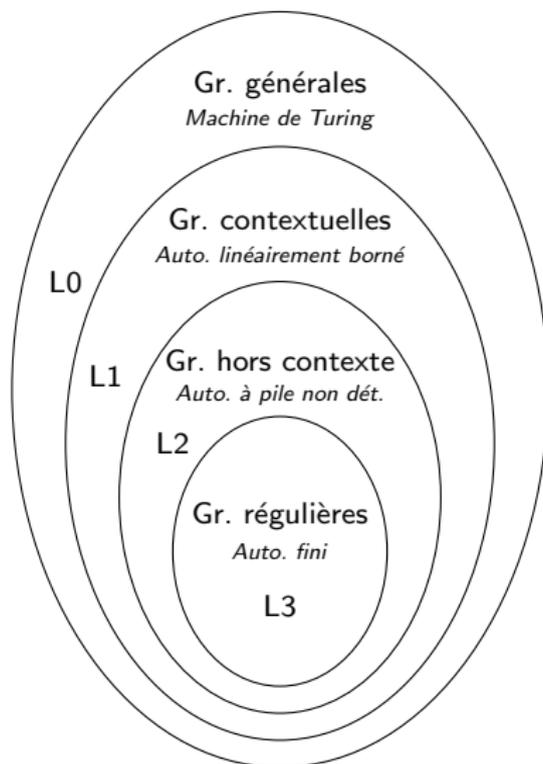
Hiérarchie des grammaires formelles

Types de grammaires

Hiérarchie des grammaires

Pour finir

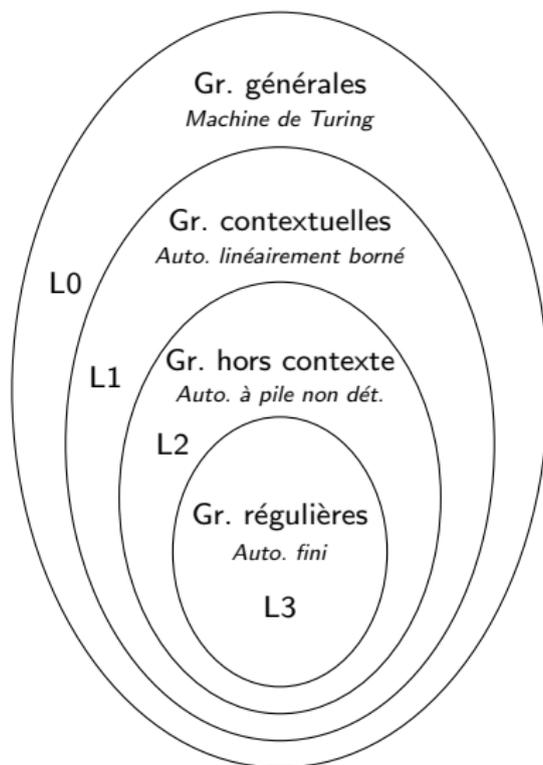
Hiérarchie ?



Types de grammaires formelles

- ▶ Type 0 : pas de restriction sur les règles
- ▶ Type 1 : grammaires sensibles au contexte (*context-sensitive*)
- ▶ Type 2 : grammaires hors-contexte (*context-free*)
- ▶ Type 3 : grammaires régulières

Hiérarchie ?



type 3 \subset type 2 \subset type 1 \subset type 0

Sources

Points de vue sur les grammaires formelles

Structure

Hiérarchie des grammaires formelles

Pour finir

CQFR : Ce Qu'il Faut Retenir

TD



Structure :

- ▶ arbre vs graphe
- ▶ définition
- ▶ passage d'une phrase du langage à sa structure
- ▶ ambiguïté

Hiérarchie des grammaires :

- ▶ types
- ▶ inclusion

Exercice 1

- ▶ La liste de variables suivante vous est donnée :
det, N, V, Adj, Adv, Prep
GN, GV, GP
- ▶ Trouver des règles de transformations qui reconnaissent les deux phrases suivantes :
Le bateau part à la dérive.
Les petits Rochellais aiment la bonne pigouille.

Exercice 2

Soit la grammaire G_{ex2} définie par :

$$V_N = \{S, VP, NP, PP, D, A, N, P, V\}$$

$$V_T = \{an, arrow, time, flies, like\}$$

$$P = \begin{cases} S \rightarrow NP VP & (1) \\ S \rightarrow VP & (2) \\ NP \rightarrow D NP & (3) \\ NP \rightarrow NP & (4) \\ NP \rightarrow NP PP & (5) \\ NP \rightarrow N & (6) \\ NP \rightarrow A NP & (7) \\ PP \rightarrow P NP & (8) \\ VP \rightarrow V \mid V NP \mid VP PP & (9) \end{cases}$$

 Montrer que le langage défini par cette grammaire est ambigu

Exercice 3

« Zoë regarde le singe avec des lunettes »

Créer une grammaire formelle qui rende compte de l'ambiguïté de cette phrase (détailler les étapes)

Conseil : commencer par dessiner les deux structures

Exercice 4 : structure qui n'est pas un arbre

Soit la grammaire $G_{nonarbre}$ définie par :

$$V_N = \{S, A, B, C\}$$

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

$$P = \left\{ \begin{array}{l} S \rightarrow AcB \quad (1) \\ A \rightarrow SA \quad (2) \\ A \rightarrow aC \quad (3) \\ B \rightarrow dA \quad (4) \\ cBAc \rightarrow cac \quad (5) \\ acdA \rightarrow acda \quad (6) \\ Ac \rightarrow bc \quad (7) \\ C \rightarrow cd \quad (8) \end{array} \right.$$

 Appliquez (1)(2)(1)(5)(4)(6)(7) et (1)(4)(3)(8)(3)(8) et dessinez les structures obtenues.