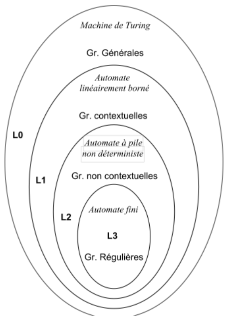
10 - Clôture des langages hors-contexte



Rappel

La classe des langages hors-contexte est close pour :

- l'union
- la concaténation
- l'opération « étoile »

La classe des langages hors-contexte n'est pas close pour :

- l'intersection
- le complémentaire

Cela dit :
Théorème l'intersection d'un langage hors-contexte et d'un langage rationnel est **hors-contexte**

On pourrait ajouter :
Théorème (admis) le complémentaire d'un langage hors-contexte reconnu par un AP **déterministe** est un langage hors-contexte.

Des langages qui ne seraient pas HC ?

- les langages hors-contexte sont engendrés par les grammaires hors-contexte et reconnus par la classe des automates à pile non-déterministes
- tout langage est-il hors-contexte (= algébrique) ?
- l'ensemble des langages hors-contexte sur un alphabet donné est dénombrable

  Une grammaire peut être perçue comme un simple mot un alphabet qui serait la réunion de ses ensembles de terminaux et de non-terminaux, auquel on ajoute un nombre fini de symboles utilitaires. Cet ensemble de mots est dénombrable.

Rappel Cours 1 : l'ensemble des langages sur un alphabet donné n'est pas dénombrable

Conclusion : il existe des langages qui ne sont pas hors-contexte ...

Problème

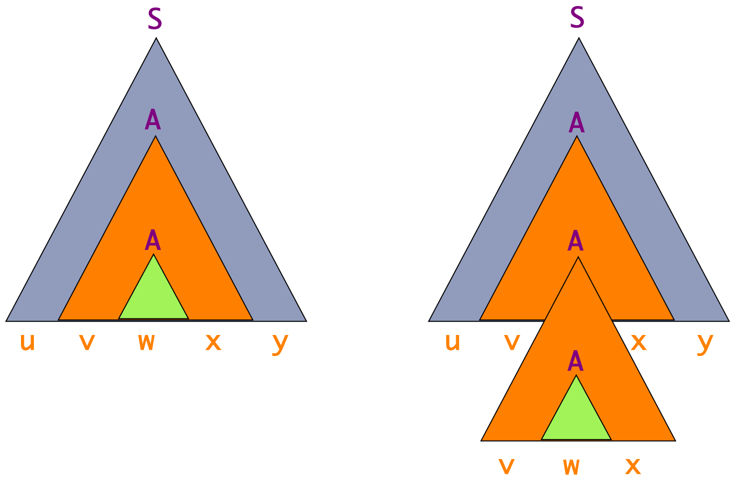
- Donnée : L un langage
- Problème : L est-il hors-contexte ?

En fait, existe-t-il un analogue au lemme de l'étoile pour les langages hors-contexte ?

- pour les langages rationnels, on a utilisé le principe des tiroirs appliqué au nombre d'états d'un automate fini
- pour les langages hors-contexte, on utilise ce même principe sur les étiquettes des nœuds de l'arbre syntaxique

  idée : un arbre de dérivation suffisamment haut va contenir plusieurs sous-arbres avec le même non-terminal en racine car les non-terminaux de la grammaire sont en nombre fini.

Idée



Lemme de l'étoile*

Lemme si L est un langage **hors-contexte**,
il existe un entier n ne dépendant que de L tel que :

- pour tout mot μ de L de longueur $\geq$ n,
- il existe une factorisation de μ en uvwxy telle que :

i) $|vx| > 0$ et $|w| > 0$

ii) $|vwx| \leq n$

iii) $\forall\ i \geq 0,\ u\,v^i\,w\,x^i\,y \in L$

* *Pumping lemma* en anglais, aussi appelé *lemme de la pompe* ou *lemme du gonflement*.

Lemme de l'étoile*

Lemme si L est un langage **hors-contexte**,
il existe un entier n ne dépendant que de L tel que :

- pour tout mot μ de L de longueur $\geq$ n,
- il existe une factorisation de μ en uvwxy telle que :

i) $|vx| > 0$ et $|w| > 0$

ii) $|vwx| \leq n$

iii) $\forall\ i \geq 0,\ u\,v^i\,w\,x^i\,y \in L$

- comme pour les langages rationnels, le lemme de l'étoile sert à montrer qu'un langage n'est pas hors-contexte

  ce lemme n'est pas une caractérisation : certains langages non HC vérifient les propriétés du lemme

- on suppose par l'absurde qu'un langage L est hors-contexte puis on cherche une contradiction

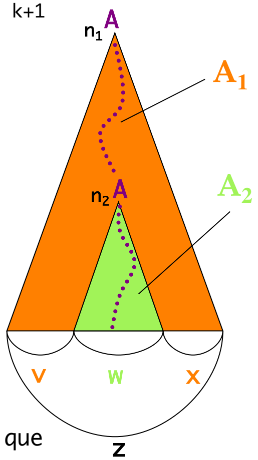
* *Pumping lemma* en anglais, aussi appelé *lemme de la pompe* ou *lemme du gonflement*.

Démonstration

- soit L un langage (supposons $L = L\backslash\{\varepsilon\}$) et G une grammaire hors-contexte sous F.N. de Chomsky pour L ayant k non-terminaux
- prenons $\mu \in L$ de longueur $|\mu| \geq n = 2^k$
- A_μ est l'arbre syntaxique associé à μ
- fait *(à montrer par simple récurrence)* :
  si A_μ a tous ses chemins de longueur $\leq$ i, $|\mu| \leq 2^{i-1}$
- comme $|\mu| > 2^{k-1}$, A_μ a un chemin C de longueur maximale au moins égale k+1
- ce chemin C comporte au moins k+2 nœuds tous étiquetés par des non-terminaux sauf sa feuille, étiquetée par un terminal
- donc il existe un non-terminal A apparaissant au moins 2 fois dans C et il existe deux nœuds n_1 et n_2 de C qui vérifient :
  - n_1 et n_2 ont la même étiquette A
  - n_1 est plus proche de la racine que n_2
  - le chemin de n_1 à la feuille est de longueur au plus k+1.

Démonstration (fin)

- A_1 engendre un facteur z de longueur $\leq 2^k$
  car les chemins de n_1 aux feuilles de longueur au plus k+1
- A_2 engendre w, un sous-facteur de z = v w x
- v et x ne peuvent être simultanément égaux à ε :
  il existe dans A_1 une règle B $\rightarrow$ C D telle que
  C et D engendrent resp. v et w x (ou bien v w et x)
  or ni C ni D ne peut se dériver dans le vide
- w est différent de ε
- on a $A \Rightarrow^* v\,A\,x$ et $A \Rightarrow^* w$
- on constate que pour tout $i \geq 0$:
  $$A \Rightarrow^* v^i\, w\, x^i$$
- on obtient une factorisation de μ en u v w x y telle que
  $x\ v^i\ w\ x^i\ y \in L$ pour tout $i \geq 0$



En pratique

- on suppose par l'absurde que L est hors-contexte
- le lemme devrait être vérifié pour un entier n
- s'il existe un mot μ de L de longueur $\geq$ n tel que :
- quelle que soit la décomposition de μ en u v w x y vérifiant :
  i) $|vx| > 0$ et $|w| > 0$
  ii) $|vwx| \leq n$
- si on trouve un $i \geq 0$ tel que :
  $$u\ v^i\ w\ x^i\ y \notin L$$
- alors :

  ➔ on peut conclure que L *n'est pas* hors-contexte.

Exemple

Supposition : le langage L = $\{0^i 1^i 2^i, i>0\}$ est hors-contexte

- le lemme est vérifié pour un n fixé (même si on ignore sa valeur)
- considérons $\mu = 0^n 1^n 2^n \in L$
- pour toute factorisat° de $\mu = u\,v\,w\,x\,y$ avec $|vx| > 0$, $|w| > 0$ et $|vwx| \leq n$
- $|vwx| \leq n \Rightarrow$ vx ne peut contenir à la fois des 0 et des 2
  - prenons $i = 0$
  - au moins un symbole parmi 0, 1 ou 2 va voir son nombre diminuer
  - au moins un symbole parmi 0, 1, 2 va voir son nombre rester intact
- cela entraîne que $z_0 = u\,v^0\,w\,x^0\,y = u\,w\,y \notin L$
- ainsi, pour toute factorisation de μ vérifiant les 3 conditions, on a trouvé un i tel que $z_i = u\,v^i\,w\,x^i\,y$ n'est pas dans L

  ➔ **contradiction** avec le lemme : notre supposition est fausse !

Conclusion : L n'est pas hors-contexte.

Une grammaire contextuelle*

G = (N, T, P, S)

- S axiome
- N = {S,T}
- T = {0,1,2}
- P = { S $\longrightarrow$ 0 S T | 0 1 2
  2 T $\longrightarrow$ T 2
  1 T 2 $\longrightarrow$ 1 1 2 2 }

L(G) = $\{ 0^i 1^i 2^i, i>0 \}$

Ce langage L(G) est donc contextuel.

* *context-sensitive* en anglais

Grammaires HC et indécidabilité

- un problème de décision est indécidable s'il n'existe pas de *procédure effective* (disons pour l'instant d'algorithme) qui donne la réponse oui ou non en un temps fini

- soient G et G' deux grammaires hors-contexte sur le même alphabet de terminaux T,
  les problèmes suivants sont indécidables :

- $L(G) \cap L(G') = \emptyset$?
- $L(G) = L(G')$?
- $L(G) = T^*$?
- G est-elle ambiguë ?