
*Grammaires hors-contexte :
simplification et formes normales*

Simplification de grammaires

Une grammaire hors-contexte pourrait être non optimale.

- Symboles inutiles
- Productions redondantes ou inutiles

Il existe principalement trois méthodes (*éliminations*) pour simplifier des grammaires hors-contextes.

- Élimination des *symboles inutiles* (*ceux qui n'apparaissent dans aucune dérivation $S \Rightarrow^* w$*).
- Élimination des *ϵ -productions* (*de la forme $A \rightarrow \epsilon$*).
- Élimination des *productions unitaires* (*de la forme $A \rightarrow B$*).

Symboles inutiles

Définition 15. Un symbole $\Phi \in \{V \cup T\}$ est *utile* pour une grammaire $G = (V, T, P, S)$ s'il existe une dérivation

$$S \Rightarrow_G^* \alpha\Phi\beta \Rightarrow_G^* w$$

pour un mot $w \in T^*$.

- Un symbole Φ est *générateur* si $\Phi \Rightarrow_G^* w$ pour un certain $w \in T^*$.
- Un symbole Φ est *accessible* si $S \Rightarrow_G^* \alpha\Phi\beta$ pour un certain $\{\alpha, \beta\} \in (V \cup T)^*$.

Définition 16. Les symboles qui ne sont pas utiles sont dits *inutiles*.

Symboles non générateurs et inaccessibles

Exemple 35. Soit G une grammaire telle que

$$S \rightarrow AB \mid a$$

$$A \rightarrow b$$

- Tous les symboles sauf B sont générateurs : a et b génèrent eux-mêmes ; S génère a ; A génère b .
- Si l'on élimine B , il faut éliminer toute production contenant B comme $S \rightarrow AB$, en conséquence G devient

$$S \rightarrow a$$

$$A \rightarrow b$$

dont A et b sont inaccessibles.

Élimination des symboles inutiles

Théorème 4. Soient $G = (V, T, P, S)$ une grammaire hors-contexte telle que $L(G) \neq \emptyset$ et $G' = (V', T', P', S)$ une grammaire obtenue en étapes :

1. éliminer toutes les variables d'où ne dérive aucun mot en symboles terminaux (*éliminer les symboles non générateurs*) ;
2. éliminer tous les symboles (terminaux ou non) n'appartenant à aucun corps dérivé de l'axiome (*éliminer les symboles inaccessibles*).

Alors G' ne contient aucun symbole inutile et $L(G') = L(G)$.

Calcul des symboles générateurs

Algorithme 1. Soit $G = (V, T, P, S)$ une grammaire hors-contexte, l'induction suivante calcule l'ensemble de tous les symboles générateurs, noté $g(G)$.

1. **Base** : $g(G) = T$.
2. **Induction** : si $X \rightarrow C_1 C_2 \dots C_k \in P$ et $\{C_1, C_2, \dots, C_k\} \subseteq g(G)$, alors $g(G) = g(G) \cup \{X\}$.

À la saturation, $g(G)$ contient tous et seuls les symboles générateurs de G .

Exemple 36. Soit G la grammaire $\{S \rightarrow AB \mid a, A \rightarrow b\}$, alors d'abord $g(G) = \{a, b\}$. Puisque $S \rightarrow a$, on met S dans $g(G)$; car $A \rightarrow b$, on y rajoute A . Ainsi, $g(G) = \{a, b, A, S\}$.

Calcul des symboles accessibles

Algorithme 2. Soit $G = (V, T, P, S)$ une grammaire hors-contexte, l'induction suivante calcule l'ensemble de tous les symboles accessibles, noté $r(G)$.

1. *Base* : $r(G) = \{S\}$.
2. *Induction* : si $A \in r(G)$ et $A \rightarrow \alpha \in P$, alors ajouter tous les symboles dans α à $r(G)$.

À la saturation, $r(G)$ contient tous et seuls les symboles accessibles de G .

Exemple 37. Soit G la grammaire $\{S \rightarrow AB \mid a, A \rightarrow b\}$, alors d'abord $r(G) = \{S\}$. Puisque $S \rightarrow AB \mid a$, on étend $r(G)$ par $\{A, B, a\}$; car $A \rightarrow b$, on y rajoute b . Ainsi, $r(G) = \{a, b, A, B, S\}$.

Simplification par éliminer les symboles inutiles (1)

Exemple 38. Simplifier la grammaire G ci-dessous en éliminant les symboles inutiles :

$$S \rightarrow AB \mid CA$$

$$A \rightarrow a$$

$$B \rightarrow AB \mid EA$$

$$C \rightarrow aB \mid b$$

$$D \rightarrow aC$$

$$E \rightarrow BA$$

Simplification par éliminer les symboles inutiles (2)

Exemple (suite). L'ensemble de variables génératrices de la grammaire G : $g_V(G) = \{S, A, C, D\}$.

$$S \rightarrow CA \quad (\Rightarrow ba)$$

$$A \rightarrow a$$

$$C \rightarrow b$$

$$D \rightarrow aC \quad (\Rightarrow ab)$$

L'ensemble de variables accessibles : $r_V(G) = \{S, A, B, C, E\}$.

L'ensemble de variables utiles : $u_V(G) = g_V(G) \cap r_V(G) = \{S, A, C\}$.

Ainsi, il reste finalement les productions selon $u_V(G)$:

$$S \rightarrow CA \quad A \rightarrow a \quad C \rightarrow b$$

Variables annulables

Définition 17. Une variable A est dite *annulable* si $A \Rightarrow^* \epsilon$.

Exemple 39. Soit A une variable annulable (c'est-à-dire qu'il existe une production $A \rightarrow \epsilon$). Une production du type

$$A \rightarrow BAD$$

peut être remplacée par deux productions

$$A \rightarrow BAD \mid BD$$

en enlevant toutes les productions ayant le corps ϵ .

Calcul des symboles annulables

Algorithme 3. Soit $G = (V, T, P, S)$ une grammaire hors-contexte, l'induction suivante calcule l'ensemble de tous les symboles annulables, noté $n(G)$.

1. **Base** : $n(G) = \{X \mid X \rightarrow \epsilon \in P\}$.
2. **Induction** : si $X \rightarrow C_1 C_2 \dots C_k \in P$ et $\{C_1, C_2, \dots, C_k\} \subseteq n(G)$, alors $n(G) = n(G) \cup \{X\}$.

À la saturation, $n(G)$ contient tous et seuls les symboles annulables de G .

Exercice 3. Calculer l'ensemble $n(G)$ de symboles annulables pour la grammaire G au-dessous :

$$S \rightarrow AB \quad A \rightarrow aAA \mid \epsilon \quad B \rightarrow bBB \mid \epsilon$$

Simplification par éliminer les ϵ -productions

Exemple 40. On a $n(G) = \{A, B, S\}$. En éliminant les ϵ -productions, les trois productions deviendront

$$S \rightarrow AB \mid A \mid B$$

$$A \rightarrow aAA \mid aA \mid aA \mid a$$

$$B \rightarrow bBB \mid bB \mid bB \mid b$$

Ainsi, la grammaire simplifiée est :

$$S \rightarrow AB \mid A \mid B$$

$$A \rightarrow aAA \mid aA \mid aA \mid a$$

$$B \rightarrow bBB \mid bB \mid bB \mid b$$

Élimination des ϵ -productions

Soit L un langage hors-contexte, alors le langage $L \setminus \{\epsilon\}$ a une grammaire hors-contexte sans aucune ϵ -production.

Théorème 5. Soit G' la grammaire construite à partir d'une grammaire G en éliminant toutes les ϵ -productions, alors

$$L(G') = L(G) \setminus \{\epsilon\}.$$

Productions unitaires

Définition 18. Une *production unitaire* est une production de la forme

$$A \rightarrow B$$

où A et B sont deux variables.

Définition 19. Une paire (A, B) est une *paire unitaire* si la dérivation

$$A \Rightarrow^* B$$

n'utilise que des productions unitaires.

Construction de l'ensemble de paires unitaires

Algorithme 4. Soit $G = (V, T, P, S)$ une grammaire hors-contexte, l'induction suivante construit l'ensemble de toutes les paires unitaires, noté $u(G)$.

1. *Base* : $u(G) = \{(X, X) \mid X \in V\}$.
2. *Induction* : si $\{(A, B)\} \in u(G)$ et $B \rightarrow C \in P$, alors
 $u(G) = u(G) \cup \{(A, C)\}$.

À la saturation, $u(G)$ contient toutes et seules les paires unitaires de G .

Paires unitaires de la grammaire G_{expr} (1)

Exemple 41. Trouver toutes les paires unitaires de la grammaire G_{expr} (soit E l'axiome) :

$$I \rightarrow a \mid b \mid Ia \mid Ib \mid I0 \mid I1$$

$$F \rightarrow I \mid (E)$$

$$T \rightarrow F \mid T * F$$

$$E \rightarrow T \mid E + T$$

Paires unitaires de la grammaire G_{expr} (2)

Exemple (suite). D'abord, on a les paires unitaires (E, E) , (T, T) , (F, F) et (I, I) . Ensuite, on peut effectuer les inférences suivantes :

1. (E, E) et la production $E \rightarrow T$ donne la paire unitaire (E, T) .
2. (E, T) et la production $T \rightarrow F$ donne la paire unitaire (E, F) .
3. (E, F) et la production $F \rightarrow I$ donne la paire unitaire (E, I) .
4. (T, T) et la production $T \rightarrow F$ donne la paire unitaire (T, F) .
5. (T, F) et la production $F \rightarrow I$ donne la paire unitaire (T, I) .
6. (F, F) et la production $F \rightarrow I$ donne la paire unitaire (F, I) .

Il n'y a pas d'autres paires qui peuvent être inférées.

Élimination des productions unitaires

Algorithme 5. Soit $G = (V, T, P, S)$ une grammaire hors-contexte, l'algorithme suivant construit une grammaire $G' = (V, T, P', S)$ qui ne contient aucune production unitaire.

1. Trouver toutes les paires unitaires de G .
2. Pour chaque paire unitaire (A, B) , ajouter à P' toutes les productions $A \rightarrow \alpha$ telles que $B \rightarrow \alpha \in P$ est une production non unitaire.

Simplification par éliminer les productions unitaires (1)

Exemple 42. Éliminer toutes les productions unitaires de la grammaire G_{expr} :

$$I \rightarrow a \mid b \mid Ia \mid Ib \mid I0 \mid I1$$

$$F \rightarrow I \mid (E)$$

$$T \rightarrow F \mid T * F$$

$$E \rightarrow T \mid E + T$$

Simplification par éliminer les productions unitaires (2)

Exemple (suite).

Paire	Productions
(E, E)	$E \rightarrow E + T$
(E, T)	$E \rightarrow T * F$
(E, F)	$E \rightarrow (E)$
(E, I)	$E \rightarrow a \mid b \mid Ia \mid Ib \mid I0 \mid I1$
(T, T)	$T \rightarrow T * F$
(T, F)	$T \rightarrow (E)$
(T, I)	$T \rightarrow a \mid b \mid Ia \mid Ib \mid I0 \mid I1$
(F, F)	$F \rightarrow (E)$
(F, I)	$F \rightarrow a \mid b \mid Ia \mid Ib \mid I0 \mid I1$
(I, I)	$I \rightarrow a \mid b \mid Ia \mid Ib \mid I0 \mid I1$

Simplification par éliminer les productions unitaires (3)

Exemple (suite).

$$E \rightarrow E + T \mid T * F \mid (E) \mid a \mid b \mid Ia \mid Ib \mid I0 \mid I1$$

$$T \rightarrow T * F \mid (E) \mid a \mid b \mid Ia \mid Ib \mid I0 \mid I1$$

$$F \rightarrow (E) \mid a \mid b \mid Ia \mid Ib \mid I0 \mid I1$$

$$I \rightarrow a \mid b \mid Ia \mid Ib \mid I0 \mid I1$$

Forme normale de Chomsky

Définition 20. Une grammaire hors-contexte $G = (V, T, P, S)$ est sous la *forme normale de Chomsky* (*CNF*) si toute production est de la forme

$$A \rightarrow BC \tag{1}$$

$$A \rightarrow a \tag{2}$$

où A , B et C sont des variables et a est un terminal.

L'arbre de dérivation correspondant à un mot dans un langage généré par une grammaire CNF est un *arbre binaire*.

Mise en forme normale de Chomsky (1)

Exemple 43. Convertir la grammaire G_{expr} en forme normale de Chomsky puis construire un arbre de dérivation pour le mot

$$a * (a + b00)$$

par rapport à la grammaire de résultat.

$$I \rightarrow a \mid b \mid Ia \mid Ib \mid I0 \mid I1$$

$$F \rightarrow I \mid (E)$$

$$T \rightarrow F \mid T * F$$

$$E \rightarrow T \mid E + T$$

Indication : il faut d'abord simplifier la grammaire G_{expr} .

Mise en forme normale de Chomsky (2)

Exemple (suite). Étape 1 :

$$E \rightarrow E + T \mid T * F \mid (E) \mid a \mid b \mid Ia \mid Ib \mid I0 \mid I1$$

$$T \rightarrow T * F \mid (E) \mid a \mid b \mid Ia \mid Ib \mid I0 \mid I1$$

$$F \rightarrow (E) \mid a \mid b \mid Ia \mid Ib \mid I0 \mid I1$$

$$I \rightarrow a \mid b \mid Ia \mid Ib \mid I0 \mid I1$$

Mise en forme normale de Chomsky (3)

Exemple (suite). Étape 2 :

$$A \rightarrow a$$

$$B \rightarrow b$$

$$Z \rightarrow 0$$

$$U \rightarrow 1$$

$$P \rightarrow +$$

$$M \rightarrow *$$

$$G \rightarrow ($$

$$D \rightarrow)$$

Mise en forme normale de Chomsky (4)

Exemple (suite). Étape 3 :

$$E \rightarrow EPT \mid TMF \mid GED \mid a \mid b \mid IA \mid IB \mid IZ \mid IU$$

$$T \rightarrow TMF \mid GED \mid a \mid b \mid IA \mid IB \mid IZ \mid IU$$

$$F \rightarrow GED \mid a \mid b \mid IA \mid IB \mid IZ \mid IU$$

$$I \rightarrow a \mid b \mid IA \mid IB \mid IZ \mid IU$$

$$A \rightarrow a$$

$$B \rightarrow b$$

$$Z \rightarrow 0$$

$$U \rightarrow 1$$

$$P \rightarrow +$$

$$M \rightarrow *$$

$$G \rightarrow ($$

$$D \rightarrow)$$

Mise en forme normale de Chomsky (5)

Exemple (suite). Étape 4 :

$$E \rightarrow EC_1 \mid TC_2 \mid GC_3 \mid a \mid b \mid IA \mid IB \mid IZ \mid IU$$

$$T \rightarrow TC_2 \mid GC_3 \mid a \mid b \mid IA \mid IB \mid IZ \mid IU$$

$$F \rightarrow GC_3 \mid a \mid b \mid IA \mid IB \mid IZ \mid IU$$

$$I \rightarrow a \mid b \mid IA \mid IB \mid IZ \mid IU$$

$$A \rightarrow a ; B \rightarrow b ; Z \rightarrow 0 ; U \rightarrow 1$$

$$P \rightarrow +$$

$$M \rightarrow *$$

$$G \rightarrow ($$

$$D \rightarrow)$$

$$C_1 \rightarrow PT$$

$$C_2 \rightarrow MF$$

$$C_3 \rightarrow ED$$

Mise en forme normale de Chomsky (6)

Exemple (suite). Arbre de dérivation pour le mot $a * (a + b00)$ en utilisant la forme normale de Chomsky de la grammaire G_{expr} .

Propriétés de la forme normale de Chomsky

Entre la forme normale de Chomsky et les grammaires hors-contexte :

- toute grammaire écrite en forme normale de Chomsky est une grammaire hors-contexte ;
- toute grammaire hors contexte peut être transformée en une grammaire équivalente en forme normale de Chomsky.

La dérivation d'un mot de longueur $n > 0$ se fait toujours en $2n - 1$ étapes :

- $n - 1$ étapes de type $A \rightarrow BC$;
- n étapes de type $A \rightarrow a$.

Un arbre de dérivation basé sur une grammaire en forme normale de Chomsky est un arbre binaire avec $2n - 1$ nœuds internes et n feuilles.

Forme normale de Greibach

Définition 21. Une grammaire hors-contexte $G = (V, T, P, S)$ est sous la *forme normale de Greibach* (*GNF*) si toute production est de la forme

$$A \rightarrow aA_1A_2 \dots A_n$$

$$A \rightarrow a$$

où A et A_i sont des variables et a est un terminal.

À chaque dérivation d'un mot dans un langage généré par une grammaire GNF, on détermine un préfixe de plus en plus long formé uniquement de symboles non terminaux.

Mise en forme normale de Greibach (1)

Exemple 44. Convertir la grammaire suivante en forme normale de Greibach.

$$E \rightarrow a$$

$$E \rightarrow (E)$$

$$E \rightarrow a + E$$

$$E \rightarrow a * E$$

Mise en forme normale de Greibach (2)

Exemple (suite). La grammaire en forme normale de Greibach.

$$E \rightarrow a$$

$$E \rightarrow (ED$$

$$E \rightarrow aPE$$

$$E \rightarrow aME$$

$$D \rightarrow)$$

$$P \rightarrow +$$

$$M \rightarrow *$$