Chapitre 5 : Analyse descendante

5.3 Grammaire LL(1)

Pour certaines grammaires, la table d'analyse peut avoir des entrées qui sont définies de façon multiple. Par exemple, si G est récursive à gauche ou ambiguë, la table d'analyse aura alors au moins une de son entrée définie de façon multiple.

Exemple

Soit la grammaire G

S → iEtSS'|a

S' → eS|ε

E → b

Voici la table d'analyse pour cette grammaire.

	Symbole d'entrée					
	a	b	e	i	t	$
S	S →a			S→ iEtSS'		
S'			S' → ε S' → eS			S'→ε
E		E → b				

- L'entrée M[S',e] contient à la fois S' → eS et S' → ε, car Suivant(S') = {e,$} (donc la règle 3, si ε est dans Premier(α), ajouter A →α à M[A,b] pour chaque terminal b dans Suivant((A))
- D'autre part, premier(α) = {e} donc mettre S' → eS dans M[S',e]
- La grammaire est ambiguë (choix de la production à utiliser quand on voit e (sinon). Nous pouvons résoudre cette ambiguïté en choisissant S' →eS

Une grammaire dont la table d'analyse n'a aucune entrée définie de façon multiple est appelée LL(1). Le premier "L" de LL(1) signifie "Parcours de l'entrée de la gauche vers la droite" (left to right scanning of the input), le second "L" signifie "Dérivation gauche" (Left most Dérivation) et le "1" indique qu'on utilise un seul symbole d'entrée de prévision à chaque étape nécessitant la prise d'une décision d'action d'analyse.

Les grammaire LL(1) ont un certain nombre de propriétés distinctives.

- Aucune grammaire ambiguë ou récursive à gauche ne peut être LL(1).
- Une grammaire est LL(1) si et seulement si, chaque fois que A →α |β sont deux productions distinctes de G, les conditions suivantes s'appliquent:

1. Pour aucun terminal a, α et β ne se dérivent toutes les deux en des chaînes commençant par a.
2. Une des chaînes au plus α et β peut se dériver en la chaîne vide.
3. Si β *⇒ ε, α ne se dérive pas en une chaîne commençant par un terminal de Suivant(A). Cela causerait une entrée multiple (voir exemple précédent) car l'un des premiers de α serait égale à un suivant de A.

On peut alors vérifier que la grammaire

E→ TE'

E'→ +TE'|ε

T →FT'

T'→ *FT'|ε

F→ (E)|id

Est LL(1)

Les grammaires

S → iEtS|iEtSeS|a (1)

E →b

S → iEtSS'|a (2)

S' → eS| ε

E→b

Ne sont pas LL(1)

La grammaire (1) n'est pas LL(1) à cause de la règle (1)

La grammaire (2) n'est pas LL(1) à cause des productions S' → eS| ε qui est sous la forme A →α |β, avec β *⇒ ε mais α=eS se dérive en une chaîne commençant par e ∈ Suivant S' = {e,$}.