

Chapitre 6 : Analyse ascendante

6.5 Analyse LALR(1)

La méthode LALR(1) (Look Ahead LR) est souvent utilisée en pratique car c'est une méthode ayant un coût et une efficacité intermédiaire en comparaison avec SLR(1) et LR(1). Pour une grammaire donnée, la table d'analyse LALR(1) occupe autant de place (même nombre d'états) que la table SLR(1) (moins que la table LR(1)) et satisfait une grande classe de grammaires. Ainsi, il est plus intéressant, plus facile et plus économique de construire des tables SLR ou LALR que des tables LR canoniques. A titre d'exemple, les méthodes SLR et LALR donnent un nombre d'états de plusieurs centaines pour un langage tel que Pascal alors qu'on arrive à plusieurs milliers d'états avec la méthode LR canonique.

6.5.1 Algorithme de construction de tables LALR(1)

Cet algorithme se base sur la construction de la collection d'ensembles d'items LR(1) ainsi que la table d'analyse LR(1). Il utilise la notion de coeur d'item LR(1), sachant que : **item LR(1)=[coeur, symbole de pré-vision]**.

L'idée générale de cet algorithme est de construire les ensembles d'items LR(1) et de fusionner les ensembles ayant un coeur commun. La table d'analyse LALR(1) est construite à partir de la collection des ensembles d'items fusionnés. Signalons que cet algorithme est le plus simple, mais il existe d'autres algorithmes plus efficaces pour la construction de tables LALR(1), qui ne se basent pas sur la méthode LR(1).

Algorithme ConstructionTableLALR;

Début

1/ Construire la collection d'ensembles d'items LR(1) pour la grammaire augmentée G' .

2/ Pour chaque coeur présent parmi les ensembles d'items LR(1), trouver tous les états ayant ce même

coeur et remplacer ces états par leur union.

3/ La table LALR(1) est obtenue en condensant la table LR(1) par superposition des lignes correspondant aux états regroupés.

Fin ;

Pendant la superposition des lignes, il y a un risque de conflit :

Cas 1 : Conflit décaler/décaler ou shift/shift (di/dj)

Ce cas de conflit ne peut pas se produire car le décalage se base sur l'élément après le point.

Cas 2 : Conflit décaler/réduire ou shift/reduce (d/r)

Ce cas de conflit ne peut se produire que s'il existe dans la table LR(1) pour l'un, au moins des états d'origine.

Cas 3 : Conflit réduire/réduire ou reduce/reduce (ri/tj)

C'est le seul conflit possible. Il se produit quand deux items LR(1) ayant respectivement la $[A \rightarrow C.,x]$ et

$[B \rightarrow C.,x]$ appartiennent à un ensemble obtenu après regroupement d'états.

Conclusion: Pendant la superposition des lignes, le seul cas de conflit qui peut se produire est réduire/réduire.

Remarque

S'il n'y a pas de conflit dans la table LALR(1), la grammaire sera considérée LALR(1) ou LALR.

6.5.2 Exemple de construction de tables d'analyse LALR(1)

On se propose de construire la table d'analyse LALR(1) pour la grammaire G :

$S \rightarrow CC$

$C \rightarrow cC \mid d$

Pour la détermination de la collection d'ensembles d'items LALR(1), on recherche les ensembles d'items LR(1) pouvant être fusionnés, on en trouve trois paires : I3 avec I6, I4 avec I7 et I8 avec I9 qui seront remplacés par leurs unions respectives :

$I_{36} = C \rightarrow c.C, c/d/\$$

$C \rightarrow .cC, c/d/\$$

$C \rightarrow .d, c/d/\$$

$I_{47} = C \rightarrow d., c/d/\$$

$I_{89} = C \rightarrow cC., c/d/\$$

Les autres ensembles d'items (non concernés par les fusions) restent inchangés. La construction de la table d'analyse LALR est obtenue par superposition des lignes correspondant aux états fusionnés:

Etat	Action			Successeur	
	c	d	\$	S	C
0	d_{36}	d_{47}		1	2
1			acc		
2	d_{36}	d_{47}			5
36	d_{36}	d_{47}			89
47	r_3	r_3	r_3		
5			r_1		
89	r_2	r_2	r_2		

6.5.3 Analyse de chaînes par la méthode LALR(1)

Si la chaîne analysée est correcte syntaxiquement (appartient au langage), l'analyse LALR(1) progressera de la même manière que LR(1). Les seules différences sont dans l'appellation ou la numérotation des états de transition.

Si la chaîne analysée est erronée syntaxiquement (n'appartient pas au langage), l'erreur sera détectée plus rapidement par l'analyseur LR(1) alors que l'analyseur LALR(1) peut effectuer plusieurs réductions avant de signaler l'erreur.