

Compiladores

Analísadores Sintáticos LR Canônico e LALR

Analísadores LR mais poderosos.

Vamos estender a técnica de análise LR anterior incorporando nos itens o primeiro símbolo da entrada, ainda não lido. Denominamos este símbolo de **lookahead**.

Existem dois métodos LR com esta característica:

- LR Canônico ou apenas LR; e
- LookAhead LR ou LALR.

Ambos são construídos sobre itens LR(1).

Analísadores LR mais poderosos.

Itens LR(1)

- Um item LR(1) é uma tupla formada por um item LR(0) e um símbolo terminal ou \$, ou seja, é um item LR(0) com uma informação extra, chamada de lookahead.
- O lookahead não tem efeito sobre os itens com a forma $[A \rightarrow \alpha.\beta, a]$, onde β não é ϵ , mas um item com a forma $[A \rightarrow \alpha., a]$ requer uma redução pela produção $A \rightarrow \alpha$ somente se o próximo símbolo da entrada for a .

Analísadores LR mais poderosos.

- Operação closure sobre Itens LR(1)

SetOfItems closure(I) {

 repeat

 for (cada item $[A \rightarrow \alpha.B\beta, a]$ em I)

 for (cada produção $B \rightarrow \gamma$ em G')

 for (cada terminal b em $\text{First}(\beta a)$)

 adicione $[B \rightarrow \gamma, b]$ ao conjunto I

 until não conseguir adicionar mais itens em I;

}

Analísadores LR mais poderosos.

- Operação goto(I,X) sobre itens LR(1)

```
SetOfItem goto(I, X) {  
    inicializa J como um conjunto vazio  
    for ( cada item  $[A \rightarrow \alpha.X\beta, a]$  em I)  
        adicione o item  $[A \rightarrow \alpha X.\beta, a]$  em J  
    return( closure(J))  
}
```

Analísadores LR mais poderosos.

- Coleção Canônica de Conjunto de Itens LR(1)

```
void items(G') {
```

```
    initialize C = closure({[S' → .S,$]})
```

```
    repeat
```

```
        for ( cada conjunto de itens I em C)
```

```
            for ( cada símbolo X da gramática)
```

```
                if (goto(I,X) não é vazio e não está em C)
```

```
                    adicione goto(I,X) em C;
```

```
    until não haja mais conjuntos de itens a serem incluídos em
```

```
    C;
```

```
}
```

Analizador LR Canônico

- Construção da Tabela Sintática LR Canônica

Algoritmo tabelaLR

entrada: uma gramática estendida G'

saída: funções Action e Goto da tabela LR

1- construa $C = \{I_0, I_1, \dots, I_n\}$

2- o estado i do analisador é construído a partir de I_i , suas ações definidas pelas regras:

a) Se $[A \rightarrow \alpha.a\beta, b]$ está em I_i e $\text{goto}(I_i, a) = I_j$, então $\text{Action}[i, a] = \text{shift } j$

b) Se $[A \rightarrow \alpha. , b]$ está em I_i e $A \neq S'$, então $\text{Action}[i, b] = \text{reduce by } A \rightarrow \alpha$

c) Se $[S' \rightarrow S., \$]$ estiver em I_i , então $\text{Action}[i, \$] = \text{Acc}$

Se quaisquer ações de conflito ocorrer a partir das regras acima, a gramática não é LR(1) e o algoritmo encerra.

3- as funções de transições para o estado i são construídas para todos não-terminais A pela regra: Se $\text{goto}(I_i, A) = I_j$, então $\text{Goto}[i, A] = j$.

4- todas as entradas da tabela não definidas pelas regras 2 e 3 são error.

Analizador LALR

Algoritmo tabela_LALR

Entrada: uma gramática estendida G'

Saída: tabela sintática LALR

1. Construa a coleção de conjuntos de itens LR(1) $C = \{I_0, I_1, \dots, I_n\}$
2. Para todos os núcleos presentes em conjuntos de itens LR(1), determine aqueles conjuntos que tenham o mesmo núcleo, e os substitua pela sua união;
3. Considere $C' = \{J_0, J_1, \dots, J_m\}$, a coleção de conjuntos de itens LR(1) resultante. As ações de análise do estado i são construídas a partir do conjunto J_i , da mesma maneira que no algoritmo para construção da tabela do analisador LR Canônico
Se houver conflito a tabela do analisador não pode ser construído e a gramática não é considerada LALR(1).
4. A tabela GOTO é construída

Analizador LALR

4. A tabela GOTO é construída da seguinte forma:

Se J é a união de $I_1 \cap I_2 \cap \dots \cap I_k$, então os núcleos de $\text{goto}(I_1, X)$, (I_2, X) , ..., (I_k, X) são os mesmos, desde que $I_1, I_2, \dots, I_k$ tenham todos os mesmos núcleos. Considere que K seja a união de todos os conjuntos de itens tendo o mesmo núcleo que $\text{goto}(I_i, X)$. Então $\text{GOTO}(J, X) = K$.

Analizador LALR

$I_0 = \{ [S' \rightarrow .S, \$], [S \rightarrow .DD, \$], [D \rightarrow .dD, d|e], [D \rightarrow .e, d|e] \}$

$I_1 = \{ [S' \rightarrow S. , \$] \}$

$I_2 = \{ [S \rightarrow D.D, \$], [D \rightarrow .dD, \$], [D \rightarrow .e, \$] \}$

$I_3 = \{ [D \rightarrow d.D, d|e], [D \rightarrow .dD, d|e], [D \rightarrow .e, d|e] \}$

$I_4 = \{ [D \rightarrow e. , d|e] \}$

$I_5 = \{ [S \rightarrow DD. , \$] \}$

$I_6 = \{ [D \rightarrow d.D, \$], [D \rightarrow .dD, \$], [D \rightarrow .e, \$] \}$

$I_7 = \{ [D \rightarrow e. , \$] \}$

$I_8 = \{ [D \rightarrow dD. , d|e] \}$

$I_9 = \{ [D \rightarrow dD. , \$] \}$

Analizador LALR

$I_0 = \{ [S' \rightarrow .S, \$], [S \rightarrow .DD, \$], [D \rightarrow .dD, d|e], [D \rightarrow .e, d|e] \}$

$I_1 = \{ [S' \rightarrow S. , \$] \}$

$I_2 = \{ [S \rightarrow D.D, \$], [D \rightarrow .dD, \$], [D \rightarrow .e, \$] \}$

$I_3 = \{ [D \rightarrow d.D, d|e], [D \rightarrow .dD, d|e], [D \rightarrow .e, d|e] \}$

$I_4 = \{ [D \rightarrow e. , d|e] \}$

$I_5 = \{ [S \rightarrow DD. , \$] \}$

$I_6 = \{ [D \rightarrow d.D, \$], [D \rightarrow .dD, \$], [D \rightarrow .e, \$] \}$

$I_7 = \{ [D \rightarrow e. , \$] \}$

$I_8 = \{ [D \rightarrow dD. , d|e] \}$

$I_9 = \{ [D \rightarrow dD. , \$] \}$

Analizador LALR

$J_0 = \{ [S' \rightarrow .S, \$], [S \rightarrow .DD, \$], [D \rightarrow .dD, d|e], [D \rightarrow .e, d|e] \}$

$J_1 = \{ [S' \rightarrow S. , \$] \}$

$J_2 = \{ [S \rightarrow D.D, \$], [D \rightarrow .dD, \$], [D \rightarrow .e, \$] \}$

$J_{36} = \{ [D \rightarrow d.D, d|e|\$], [D \rightarrow .dD, d|e|\$], [D \rightarrow .e, d|e|\$] \}$

$J_{47} = \{ [D \rightarrow e. , d|e|\$] \}$

$J_5 = \{ [S \rightarrow DD. , \$] \}$

$J_{89} = \{ [D \rightarrow dD. , d|e|\$] \}$

Analizador LALR

	ACTION			GOTO	
	d	e	\$	S	D
0	s36	s47		1	2
1	ACC				
2	s36	s47			5
36	s36	s47			89
47	R4	R4	R4		
5			R2		
89	R3	R3	R3		