

4.2.4 LR分析法

□LR(0)分析法

讨论

□算符优先分析与LR分析法比较

相同点:通过分析栈的栈顶项和当前输入符号找当

前句型句柄的右端;

不同点:优先分析法为找句柄的头必须对栈进行搜索;

LR分析法只根据栈顶状态和当前输入符号就

可判断;

□问题: LR(0)分析器如何转换成LR(0)分析表?

LR(0)分析器如何构造?

4.2.4 LR分析法

总结

□基本概念：活前缀、可归前缀

规范句型的活前缀可以为FA M所识别

□LR(0)分析法

LR(0)分析器

LR(0)分析方法

□进一步要解决的问题

由LR(0)分析器构造LR(0)分析表

构造LR(0)分析器

4.2.4 LR分析法

□LR(0)分析法

4、LR(0)分析器的构造 --DFA构造

1)项目:给定文法的一个项目是一个在右部符号串中标
有一圆点的产生式,

形式: $A \rightarrow \alpha_1 \cdot \alpha_2$

$A \rightarrow \alpha_1 \alpha_2$ 为一个产生式

表示:已从输入串中看到了能由 α_1 推导出的符号串,

希望进一步看到由 α_2 推导出的符号串.

例: $E \rightarrow E+T$ 项目: $E \rightarrow \cdot E+T$ $E \rightarrow E \cdot +T$

$E \rightarrow E+ \cdot T$ $E \rightarrow E+T \cdot$

4.2.4 LR分析法

□LR(0)分析法

- 归约项目:圆点在最后的项目

$$E \rightarrow E+T \cdot$$

- 接受项目:开始符号的归约项目

$$S \rightarrow E\# \cdot$$

- 移进项目:形如 $A \rightarrow \alpha \cdot a \beta$ 项目 $a \in V_t$

$$E \rightarrow E \cdot +T$$

- 待约项目:形如 $A \rightarrow \alpha \cdot B \beta$ 项目 $B \in V_n$

$$E \rightarrow E + \cdot T$$

4.2.4 LR分析法

□LR(0)分析法

2)有效项目:

$A \rightarrow \alpha_1 \cdot \alpha_2$ 是项目, 对于某一个活前缀 $\varphi \alpha_1$

存在 $S \xRightarrow{*} \varphi A t \Rightarrow \varphi \alpha_1 \alpha_2 t \quad t \in V_t^*$

则称 $A \rightarrow \alpha_1 \cdot \alpha_2$ 是活前缀 $\varphi \alpha_1$ 的有效项目.

- 若归约项目 $A \rightarrow \beta_1 \cdot$ 对活前缀 $\alpha \beta_1$ 是有效的, 应把 β_1 归约为 A .
- 若待约或移入项目 $A \rightarrow \beta_1 \cdot \beta_2$ 对活前缀 $\alpha \beta_1$ 是有效的则句柄尚未形成, 下一步动作是移进或待约.

4、LR(0)分析器的构造

□ DFA M 的一个状态 i

---该状态识别出的所有活前缀的有效项目集 C_i

□ DFA M 的状态集 Q

$$Q = \{ C_0, C_1, C_2, \dots, C_n \} = C$$

□ C 称为文法的LR(0)有效项目集规范族

(1)三种操作:

■ 开始操作: S 为开始符号, $S \rightarrow \delta$

则 $S \rightarrow \cdot \delta \in C_0$

■ 闭包操作: $\text{closure}(C_i)$ C_i 的闭包

① C_i 的任何项目均属于 $\text{closure}(C_i)$

② 若 $A \rightarrow \alpha \cdot X \beta$ 且 $X \in V_n$ 属于 $\text{closure}(C_i)$

则 $X \rightarrow \cdot \lambda$ 属于 $\text{closure}(C_i)$

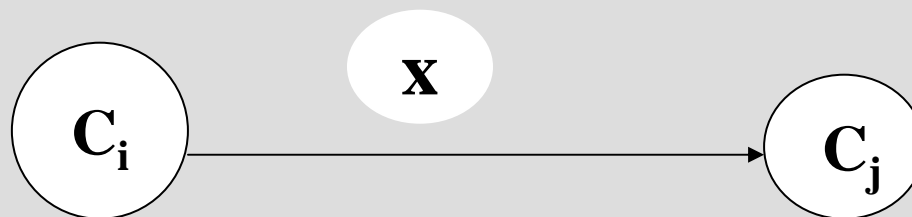
重复,直至 $\text{closure}(C_i)$ 不再增大.

③ $C_i = \text{closure}(C_i)$

- 读操作: $\text{Go}(C_i, x) \quad x \in V$

$$\text{Go}(C_i, x) = C_j$$

$$\text{其中: } C_j = \{A \rightarrow \alpha x \cdot \beta \mid A \rightarrow \alpha \cdot x \beta \in C_i\}$$



(2) 算法(求文法的LR(0)有效项目集规范族C)

①拓广文法,保证唯一初态.

②生成 $C_0 = \{S \rightarrow \cdot \delta\} \cup \{S \rightarrow \cdot \delta \text{ 的闭包操作}\}$

③重复以下过程,直至C不再增大为止.

C_i 读操作,生成 $C_{j1}, C_{j2}, \dots, C_{jn}$

$C_{j1}, C_{j2}, \dots, C_{jn}$ 闭包操作

(若其中某项目集已经存在就略去)

通过上述LR(0)分析表可见,项目集中**不存在**
下述项目---**冲突项目**

(1)既含移进项目,又含归约项目

(2)含有多个归约项目

这种文法称为LR(0)文法.

不满足条件可采用SLR(1),LR(1)分析法.

4)LR(0)分析器构造另一方法

- (1) 规则含 $S \rightarrow \cdot \delta$ 为唯一初态.
- (2) 若状态 i 和状态 j 出自同一产生式而两个状态对应的项目的圆点仅差一个位置

i 为: $X \rightarrow x_1 x_2 \dots x_{i-1} \cdot x_i \dots x_n$

j 为: $X \rightarrow x_1 x_2 \dots x_i \cdot x_{i+1} \dots x_n$



- (3) 若状态 i 为 $X \rightarrow \alpha \cdot A \beta$, $A \in V_n$

则从 i 引 ε 弧到所有 $A \rightarrow \cdot r$ 的状态

- (4) 将NFA确定化为 DFA M .

5)LR(0)分析表构造---只适用于LR(0)文法

根据LR(0)文法的 $C = \{ C_0, C_1, \dots, C_n \}$

(1)若 $GO(C_i, a) = C_j$, $a \in V_t$ 置 $ACTION(i, a) = S_j$

(2)若 $A \rightarrow \alpha \cdot \in C_j$, 所有 $a \in V_t$ (含#)

置 $ACTION(i, a) = r_j$

(3)若 $S \rightarrow \delta \cdot \in C_k$, (S为拓广文法开始符号)

置 $ACTION(k, \#) = acc$

(4)若 $GO(C_i, A) = C_j$, $A \in V_n$, 置 $GOTO(i, A) = j$

(5)其它: 空白 (error)

三、SLR(1)分析法

$C_i = \{ U \rightarrow \alpha \cdot \mathbf{b} \beta, V \rightarrow \alpha \cdot, W \rightarrow \alpha \cdot \}$

含有冲突项目，LR(0)分析法无法解决

SLR(1)

向前看一个输入符号决定当前动作

产生一个最右推导(最左归约)

自左向右扫描输入串

简单Simple

1 SLR(1)分析法

SLR(1)分析法=LR(0)分析法+冲突解决办法

$$C_i = \{U \rightarrow \alpha . b \beta, V \rightarrow \alpha ., W \rightarrow \alpha .\} (b \in V_T)$$

◆对输入符号a

- $a=b$, 置 $\text{ACTION}(i,b) = \text{"移进"}$
- $a \in \text{Follow}(V)$, 则用 $V \rightarrow \alpha$ 归约
- $a \in \text{Follow}(W)$, 则用 $W \rightarrow \alpha$ 归约
- 否则, err

条件: $\{b\}, \text{follow}(v), \text{follow}(w)$ 两两互不相交

定义SLR(1)文法

LR(0)有效项目集规范族C中，若一个项目含有m个移进项目，同时n个归约项目，即

$$C_i = \{ U_1 \rightarrow \alpha . b_1 \beta, U_2 \rightarrow \alpha . b_2 \beta, \dots, U_m \rightarrow \alpha . b_m \beta,$$

$$V_1 \rightarrow \alpha ., V_2 \rightarrow \alpha ., \dots, V_n \rightarrow \alpha . \}$$

若 $\{b_1, b_2, \dots, b_m\}, \text{FOLLOW}(V_1), \dots, \text{FOLLOW}(V_n)$ 两两互不相交，满足此条件的文法为SLR(1)文法，可用SLR(1)分析方法解决。

解决办法：

(1) $a \in \{b_1, b_2, \dots, b_m\}$ 移进

(2) $a \in \text{Follow}(V_i)$ 归约

(3) 其他出错

2) SLR(1)分析表构造

SLR(1)分析表构造=
LR(0)分析表+SLR(1)分析法

SLR(1)分析表构造

- (1) 拓广文法 G'
- (2) 构造(对 G')LR(0)有效项目集族 C 和 GO 函数
- (3) 若 $GO(C_i, a)=C_j$ $a \in V_t$, 置 $ACTION(i,a)=S_j$
- (4) 若 $A \rightarrow \alpha \cdot \in C_j$, 且 $a \in FOLLOW(A)$,
 $a \in V_t$ 置 $ACTION(i,a)=r_j$
($A \rightarrow \alpha$ 为第 j 个产生式)
- (5) 若 $S \rightarrow \delta \cdot \in C_k$, (S 为拓广文法开始符号)
置 $ACTION(k,\#)=acc$
- (6) 若 $GO(C_i, A)=C_j$, $A \in V_n$, 置 $GOTO(i,A)=j$
- (7) 其它: 空白 (error)

- 若一个文法是SLR(1)文法, 则按上述方法构造的分析表一定不含多重定义.
- 如果不满足SLR(1)文法的要求, 则
可采用 LR(1)分析法
LALR(1)分析法

例2：算术表达式文法

$S \rightarrow E$ [0]

$E \rightarrow E + T$ [1]

$E \rightarrow T$ [2]

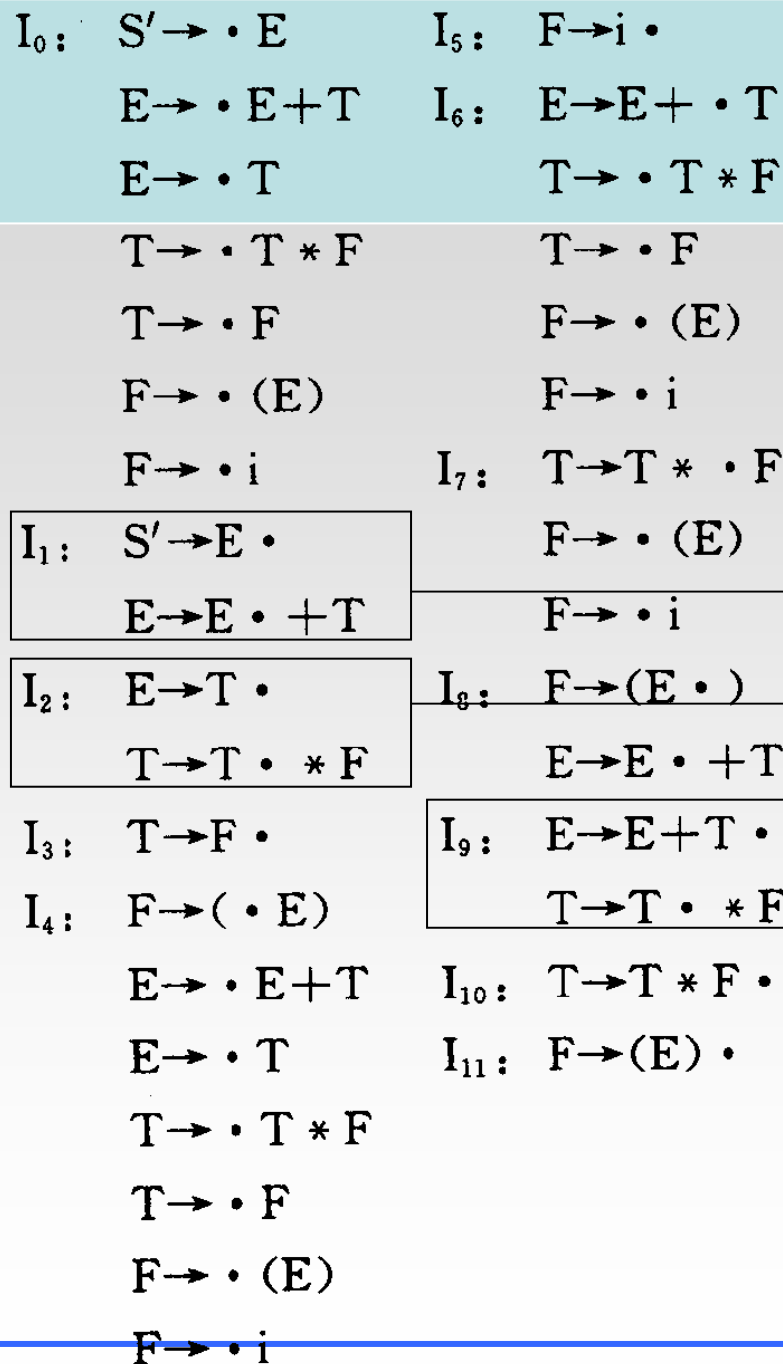
$T \rightarrow T * F$ [3]

$T \rightarrow F$ [4]

$F \rightarrow (E)$ [5]

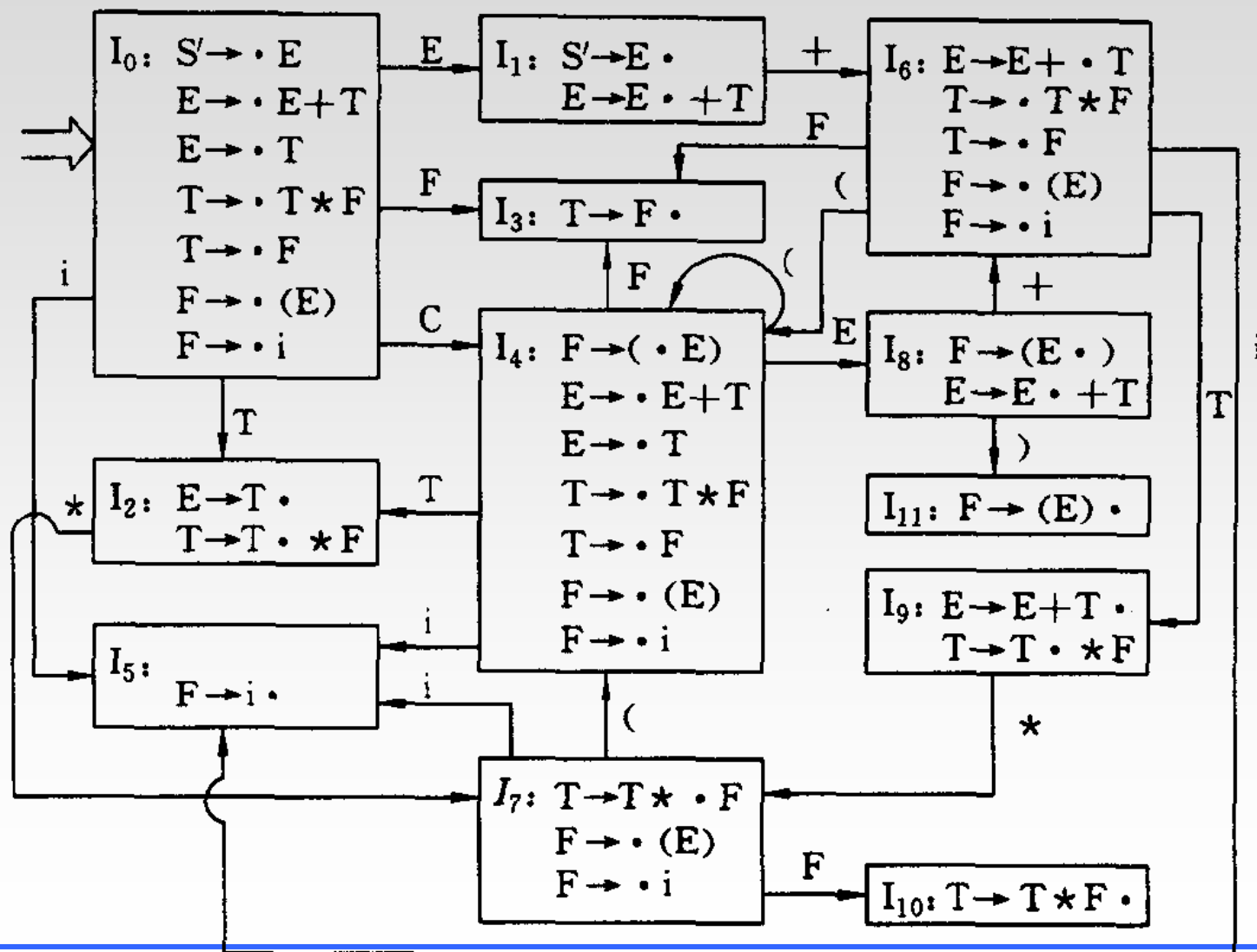
$F \rightarrow i$ [6]

算术表达式 文法的LR(0) 项目集族



存在移进-
归约冲突

对应识别活前缀的DFA



冲突解决

$$I_1 \text{ 中 FOLLOW } (S') = \{\#\} \cap \{+\} = \Phi \quad \checkmark$$

$$I_2 \text{ 中 FOLLOW } (E) = \{+, ,) , \#\} \cap \{*\} = \Phi \quad \checkmark$$

$$I_9 \text{ 中 FOLLOW } (E) = \{+, ,) , \#\} \cap \{*\} = \Phi \quad \checkmark$$

$$I_1 = \{S \rightarrow E., E \rightarrow E.+T\}$$

$$I_2 = \{E \rightarrow T., T \rightarrow T.*F\}$$

$$I_9 = \{E \rightarrow E+T., T \rightarrow T.*F\}$$

冲突得以解决

SLR(1)分析表

状态	ACTION						GOTO		
	i	+	*	()	#	E	T	F
0	S ₅			S ₄			1	2	3
1		S ₆				acc			
2		r ₂	S ₇		r ₂	r ₂			
3		r ₄	r ₄		r ₄	r ₄			
4	S ₅			S ₄			8	2	3
5		r ₆	r ₆		r ₆	r ₆			
6	S ₅			S ₄				9	3
7	S ₅			S ₄					10
8		S ₆			S ₁₁				
9		r ₁	S ₇		r ₁	r ₁			
10		r ₃	r ₃		r ₃	r ₃			
11		r ₅	r ₅		r ₅	r ₅			

对输入串 $i+i*i\#$ 的SLR(1) 分析过程

步骤	状态栈	符号栈	输入串	ACTION	GOTO
1	0	#	$i+i*i\#$	S_5	
2	05	#i	$+i*i\#$	r_6	3
3	03	#F	$+i*i\#$	r_4	2
4	02	#T	$+i*i\#$	r_2	1
5	01	#E	$+i*i\#$	S_6	
6	016	#E+	$i*i\#$	S_5	
7	0165	#E+i	$*i\#$	r_6	3
8	0163	#E+F	$*i\#$	r_4	9
9	0169	#E+T	$*i\#$	S_7	
10	01697	#E+T*	$i\#$	S_5	
11	016975	#E+T*i	#	r_6	10
12	01697(10)	#E+T*iF	#	r_3	9
13	0169	#E+T	#	r_1	1
14	01	#E	#	acc	

$S' \rightarrow S[0]$

$S \rightarrow aAd[1]$

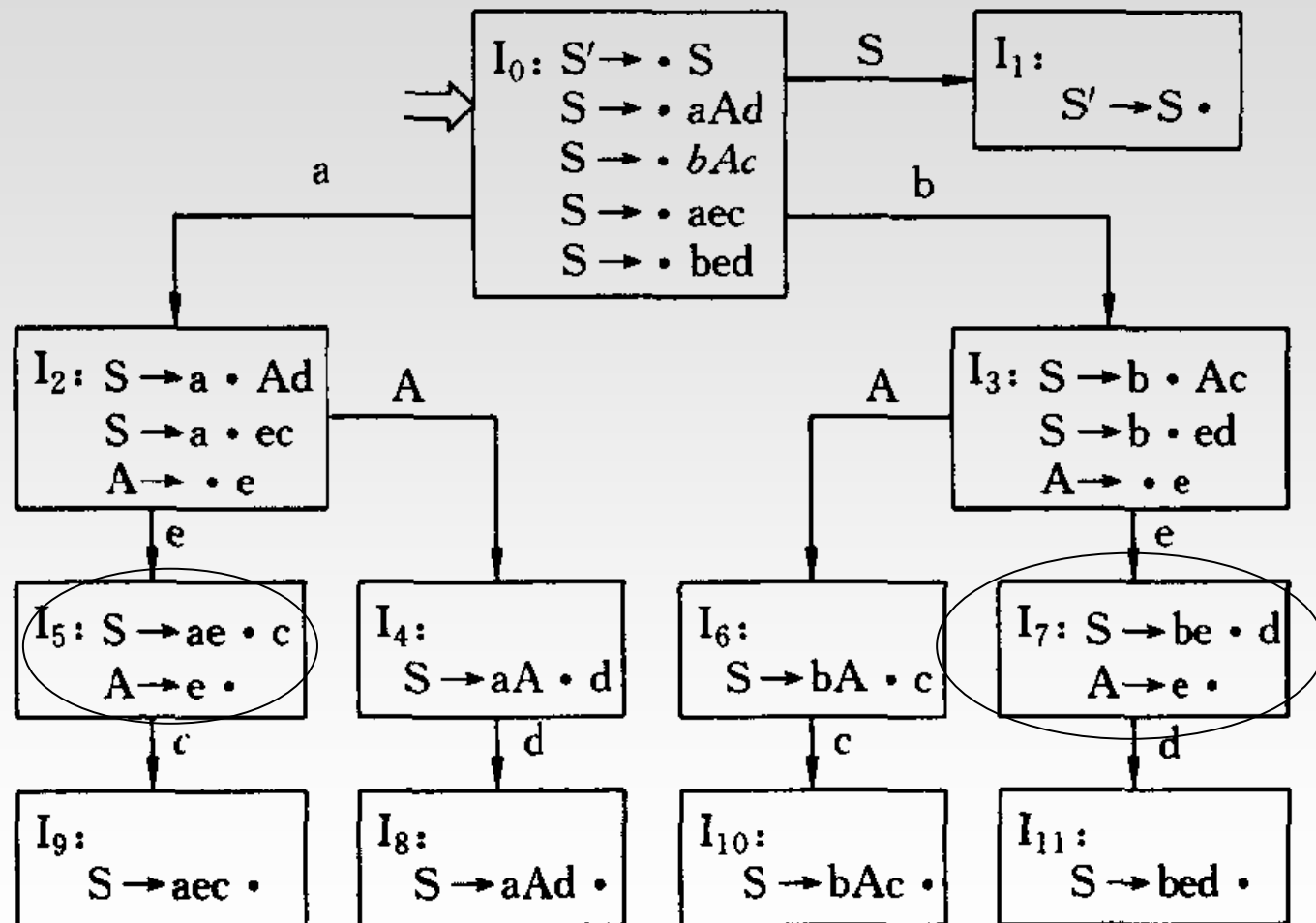
$S \rightarrow bAc[2]$

$S \rightarrow aec[3]$

$S \rightarrow bed[4]$

$A \rightarrow e[5]$

对应识别活前缀的DFA



冲突

- $I_5 \quad S \rightarrow ae \cdot c \quad A \rightarrow e \cdot$
- $I_7 \quad S \rightarrow be \cdot d \quad A \rightarrow e \cdot$
- $\text{FOLLOW}(A) = \{c, d\}$
- $\text{FOLLOW}(A) \cap \{c\} \neq \varnothing$
- $\text{FOLLOW}(A) \cap \{d\} \neq \varnothing$
- **利用SLR(1)无法解决冲突**

- 如何解决SLR(1)分析法中出现的冲突？
- 采用LR(1)、LALR(1)分析
- 提示：利用FIRST()集