



第5章

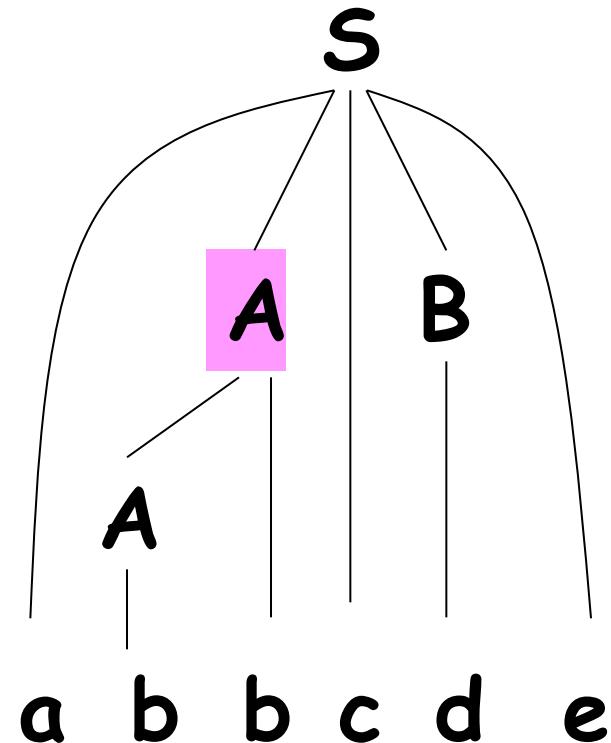
自底向上 优先分析法

5.1自底向上分析方法概述

设文法 $G[S]$:

- (1) $S \rightarrow aAcBe$
- (2) $A \rightarrow b$
- (3) $A \rightarrow Ab$
- (4) $B \rightarrow d$

符号串**abbcde**是否是
G[S]的句子？



实现方法：移进-归约

步骤	符号栈	输入符号串	动作
1)	#	abbcde#	移进
2)	#a	bbcde#	移进
3)	#ab	bcde#	归约($A \rightarrow b$)
4)	#aA	bcde#	移进
5)	#aAb	cde#	归约($A \rightarrow Ab$)
6)	#aA	cde#	移进
7)	#aAc	de#	移进
8)	# aAc d	e#	归约($B \rightarrow d$)
9)	#aAcB	e#	移进
10)	#aAcBe	#	归约($S \rightarrow aAcBe$)
11)	#S	#	接受

移进-归约过程是自顶向下最右推导的逆过程。

- 基本问题
 - 如何找到句柄？
 - 找出的句柄应归约到哪一个非终结符？

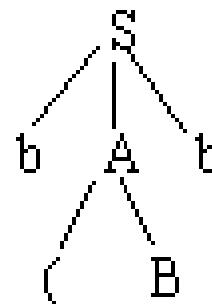
- 主要分析方法
 - 优先分析法
 - 简单优先法
 - 算符优先分析方法
- LR类分析法

5.2 简单优先分析法(自学、研讨)

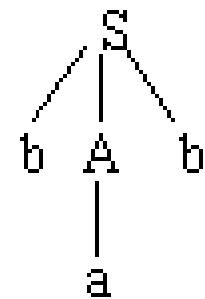
b(Bb) bab b((Aa)a)b b(aa)b

■ 引例

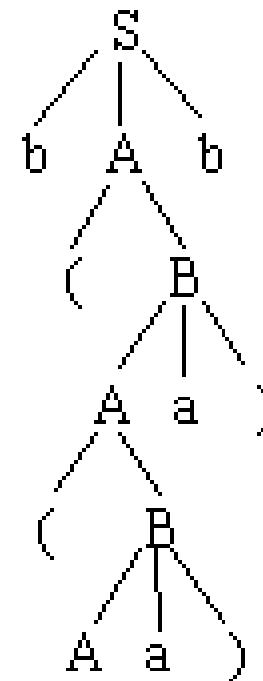
文法 **G[S]:**
S → bAb
A → (B|a
B → Aa)



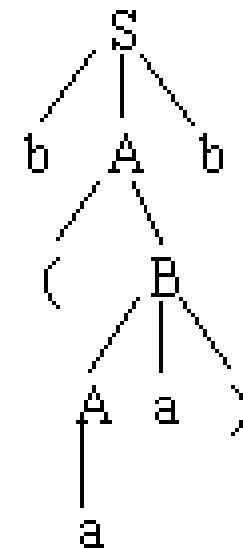
(a)



(b)



(c)



(d)

- 基本思想

按照文法符号（终结符和非终结符）的优先关系确定句柄。

简单优先分析示例

文法 $G[S]$:

- (1) $S \rightarrow bAb$
- (2) $A \rightarrow (B|a)$
- (3) $B \rightarrow Aa$

	S	b	A	(B	a)	#
S								>
b		=	<		<			>
A		=			=			
(<	<	=	<			
B		>			>			
a		>			>	=		
)		>			>			
#	<	<						

简单优先关系矩阵

对输入串 $b(aa)\#$ 的简单优先分析过程

步骤	符号栈	输入符号串	动作
1)	#	$b(aa)b\#$	#<b, 移进
2)	#b	(aa)b#	b<(), 移进
3)	#b(aa)b#	(<a, 移进
4)	#b(a	a)b#	a>a, 归约 $A \rightarrow a$
5)	#b(A	a)b#	A=a, 移进
6)	#b(Aa)b#	a=), 移进
7)	#b(Aa)	b#)>b, 归约 $B \rightarrow Aa$
8)	#b(B	b#	B>b, 归约 $A \rightarrow (B$
9)	#bA	b#	A=b, 移进
10)	#bAb	#	b>#, 归约 $S \rightarrow bAb$
11)	#S	#	接受

5.3 算符优先分析法(自学、研讨)

- 某些文法具有“算符”特性
 - 表达式运算符（优先级、结合性）。
 - 人为地规定其算符的优先顺序，即给出优先级别和同级别的结合性。
- 只考虑算符之间的优先关系来确定可归约串。
- 不是规范规约

算符优先分析示例

对输入串*i+i*i*的算符优先分析过程

步骤	符号栈	输入符号串	动作
1)	#	i+i*i#	#<i, 移进
2)	#i	+i*i#	#<i>+, 规约
3)	#F	+i*i#	#<+, 移进
4)	#F+	i*i#	+<i, 移进
5)	#F+i	*i#	+<i>*, 规约
6)	#F+F	*i#	+<*, 移进
7)	#F+F*	i#	*<i, 移进
8)	#F+F*i	#	*<i>#, 规约
9)	#F+F*F	#	+<*>#, 规约
10)	#F+T	#	#<+>#, 规约
11)	#E	#	接受

算符优先关系表

文法 $G[E]$: $E \rightarrow E+E | E-E | E^*E | E/E | E\uparrow E | (E) | i$