Lecture 5

SLR文法

徐 辉
xuh@fudan.edu.cn

主要内容

- 一、SLR文法
- 二、更多文法

一、SLR文法

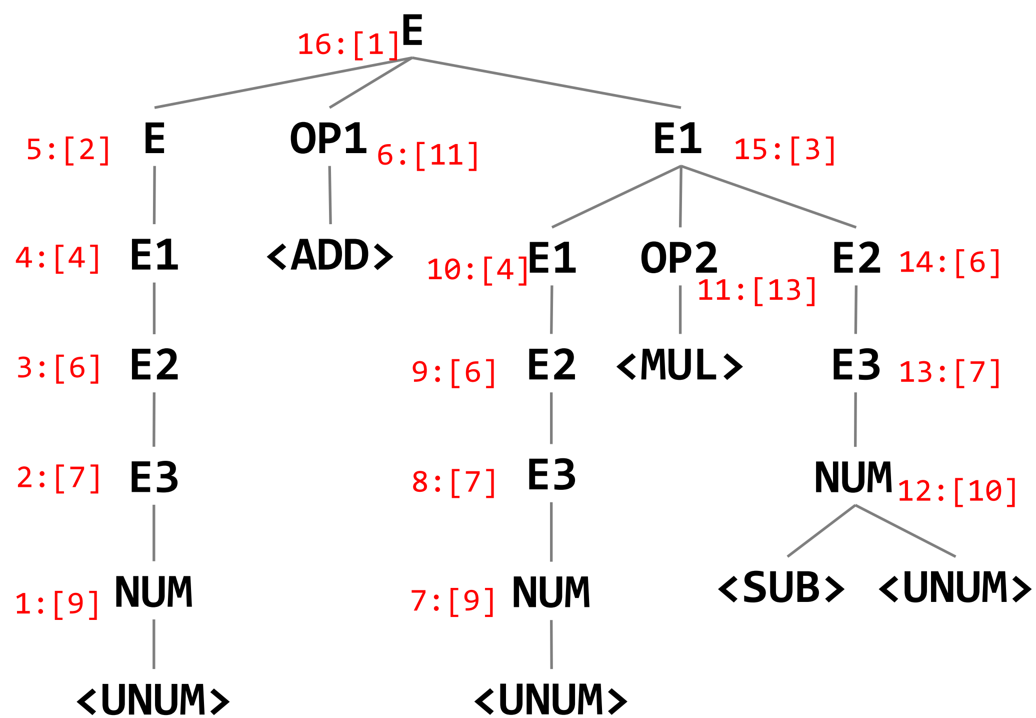
自底向上解析

语法规则：

```
[1] E → E OP1 E1
[2]   | E1
[3] E1 → E1 OP2 E2
[4]    | E2
[5] E2 → E3 OP3 E2
[6]    | E3
[7] E3 → NUM
[8]    | <LPAR> E <RPAR>
[9] NUM → <UNUM>
[10]    | <SUB> <UNUM>
[11] OP1 → <ADD>
[12]     | <SUB>
[13] OP2 → <MUL>
[14]     | <DIV>
[15] OP3 → <POW>
```

解析对象： 1+2*-3

标签流： <UNUM><ADD><UNUM><MUL><SUB><UNUM>



如何自动规约生成语法解析树？

- 两种操作：
  - 移进：读入下一个字符
  - 规约：应用语法规则规约已读入字符
- 根据句柄状态应用语法规则（从左至右）逐步规约
- 如何精准选择移进或规约？避免盲目搜索
- SLR(1)文法：Simple Left-to-Right，Rightmost，前瞻一个字符
- SLR文法的基本要求：无回溯
  - 同一个状态既可以移进，又可以规约
  - 同一个状态存在两个规约选项

句柄状态（规范项）分析

- 句柄：语法规则中已解析的字符，一般保存在栈上
- 每条语法规则$X \rightarrow \beta$可能存在$|\beta| + 1$种句柄状态

```
[1] E → E OP1 E1
[2]   | E1
[3] E1 → E1 OP2 E2
[4]    | E2
[5] E2 → E3 OP3 E2
[6]    | E3
[7] E3 → NUM
[8]    | <LPAR> E <RPAR>
[9] NUM → <UNUM>
[10]    | <SUB> <UNUM>
[11] OP1 → <ADD>
[12]     | <SUB>
[13] OP2 → <MUL>
[14]     | <DIV>
[15] OP3 → <POW>
```

句柄分析 ⇒

```
[1] E → ∘ E OP1 E1
[1] E → E ∘ OP1 E1
[1] E → E OP1 ∘ E1
[1] E → E OP1 E1 ∘
[2] E → ∘ E1
[2] E → E1 ∘
[3] E1 → ∘ E1 OP2 E2
[3] E1 → E1 ∘ OP2 E2
[3] E1 → E1 OP2 ∘ E2
[3] E1 → E1 OP2 E2 ∘
...
```

规范项

语法增强

- 加入一个初始符号和初始规则，初始符号仅在初始规则出现
- 句柄为初始符号S∘, 并且后一个字符是eof则解析结束

```
[1] E → ∘ E OP1 E1
[1] E → E ∘ OP1 E1
[1] E → E OP1 ∘ E1
[1] E → E OP1 E1 ∘
[2] E → ∘ E1
[2] E → E1 ∘
[3] E1 → ∘ E1 OP2 E2
[3] E1 → E1 ∘ OP2 E2
[3] E1 → E1 OP2 ∘ E2
[3] E1 → E1 OP2 E2 ∘
...
```

语法增强

```
[0] S → ∘ E
[0] S → E ∘
[1] E → ∘ E OP1 E1
[1] E → E ∘ OP1 E1
[1] E → E OP1 ∘ E1
[1] E → E OP1 E1 ∘
[2] E → ∘ E1
[2] E → E1 ∘
[3] E1 → ∘ E1 OP2 E2
[3] E1 → E1 ∘ OP2 E2
[3] E1 → E1 OP2 ∘ E2
[3] E1 → E1 OP2 E2 ∘
...
```

构建LR(0)自动机：规范族

- 一条规范项可能由多种规范项直接或间接规约得到

```
[0] S → ∘ E
[1] E → ∘ E OP1 E1
[2] E → ∘ E1
...
```

分析规范族

While (Q has changed) //仅包含当前规范项
  for each item $[A \rightarrow \beta \circ C\delta] \in Q$
    for each production $[C \rightarrow \lambda] \in G$
      if $[C \rightarrow \circ\, \lambda] \notin Q$
        $Q \leftarrow Q \cup [C \rightarrow \circ\, \lambda]$

应用

```
[0] S → ∘ E
[1] E → E OP1 E1
[2]   | E1
[3] E1 → E1 OP2 E2
[4]    | E2
[5] E2 → E3 OP3 E2
[6]    | E3
[7] E3 → NUM
[8]    | <LPAR> E <RPAR>
[9] NUM → <UNUM>
[10]    | <SUB> <UNUM>
[11] OP1 → <ADD>
[12]     | <SUB>
[13] OP2 → <MUL>
[14]     | <DIV>
[15] OP3 → <POW>
```

分析规范族

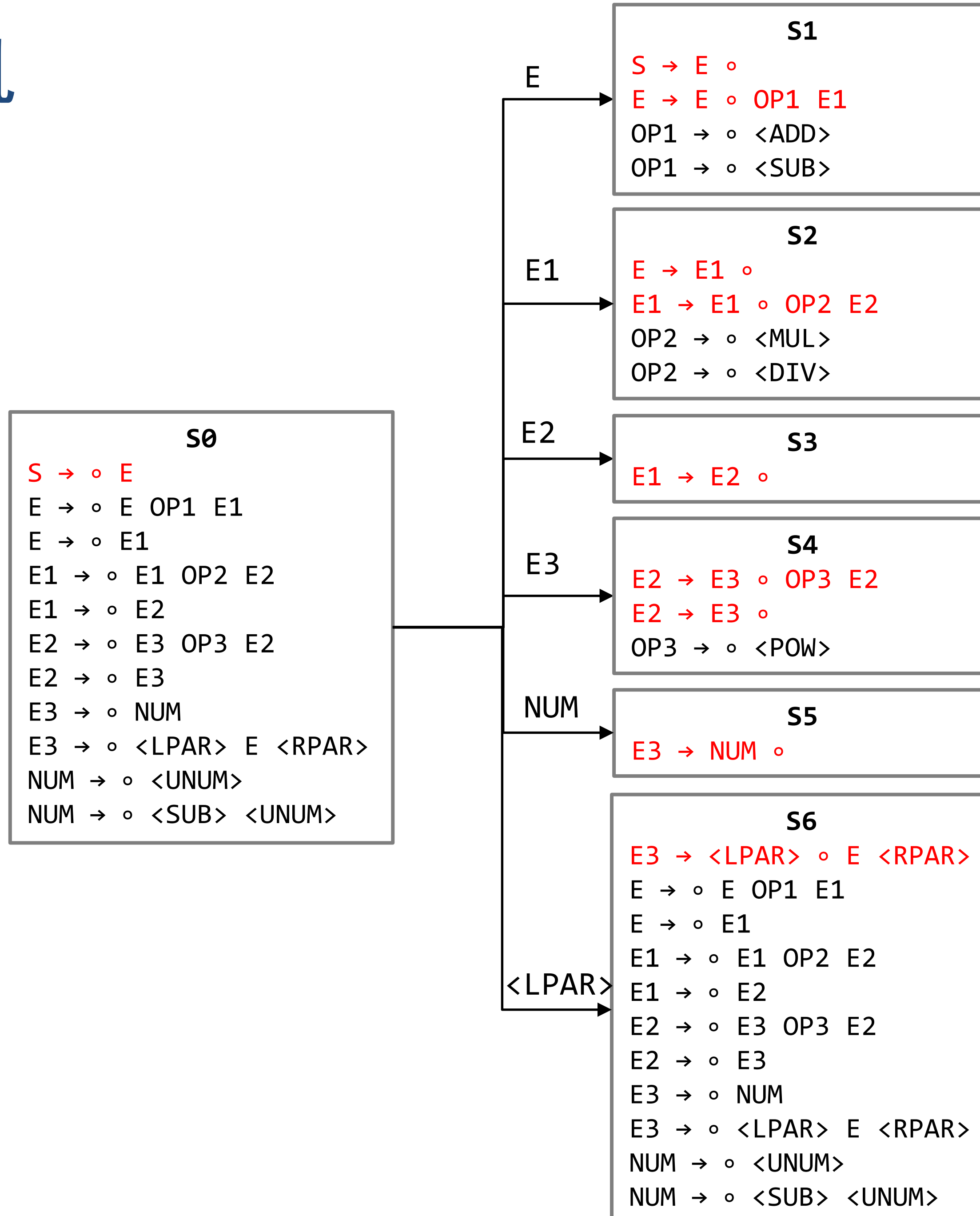
```
S0
S → ∘ E
E → ∘ E OP1 E1
E → ∘ E1
E1 → ∘ E1 OP2 E2
E1 → ∘ E2
E2 → ∘ E3 OP3 E2
E2 → ∘ E3
E3 → ∘ NUM
E3 → ∘ <LPAR> E <RPAR>
NUM → ∘ <UNUM>
NUM → ∘ <SUB> <UNUM>
```

核心项

非核心项

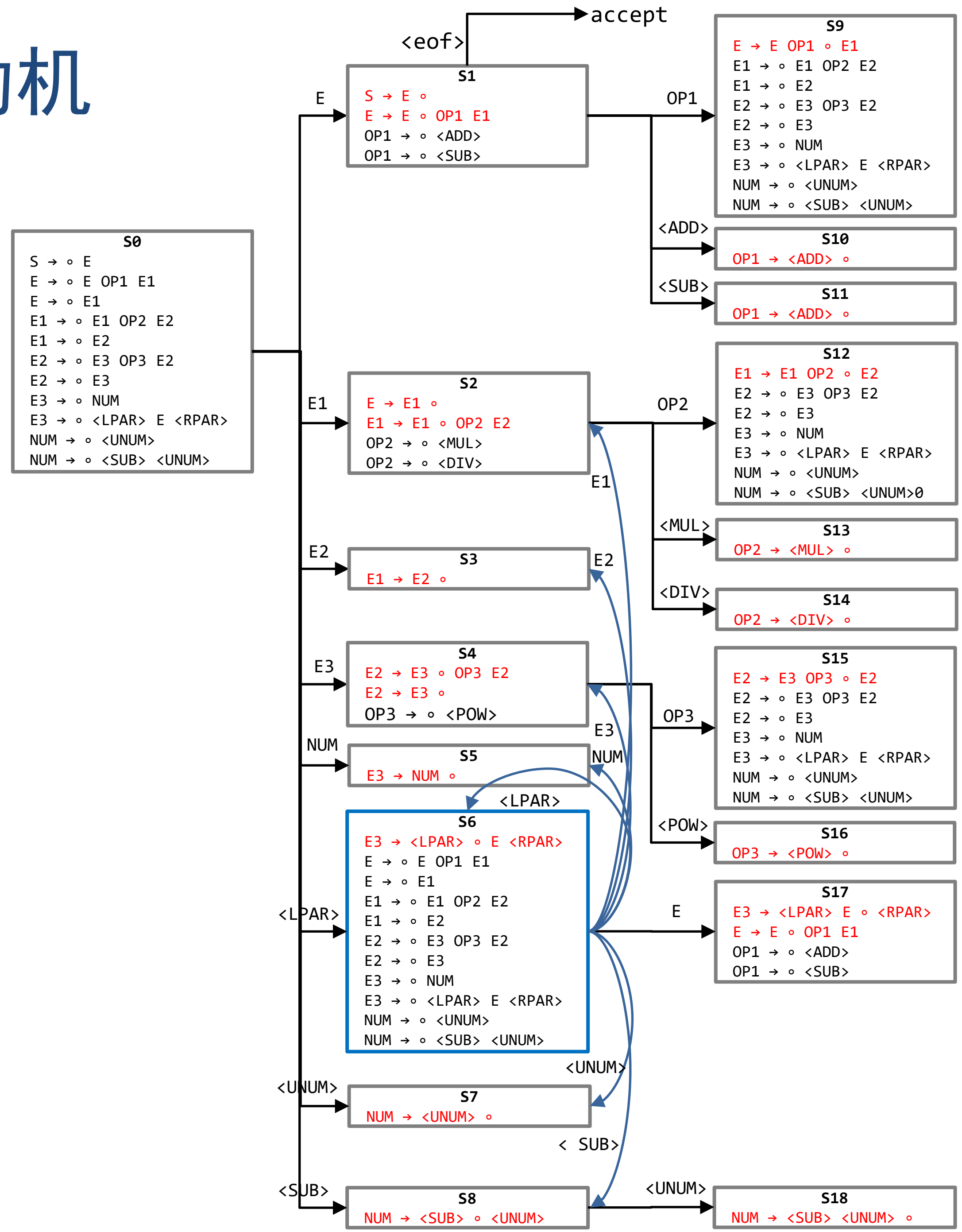
构建LR(0)自动机

```
[0] S → E
[1] E → E OP1 E1
[2]   | E1
[3] E1 → E1 OP2 E2
[4]    | E2
[5] E2 → E3 OP3 E2
[6]    | E3
[7] E3 → NUM
[8]    | <LPAR> E <RPAR>
[9] NUM → <UNUM>
[10]    | <SUB> <UNUM>
[11] OP1 → <ADD>
[12]     | <SUB>
[13] OP2 → <MUL>
[14]     | <DIV>
[15] OP3 → <POW>
```

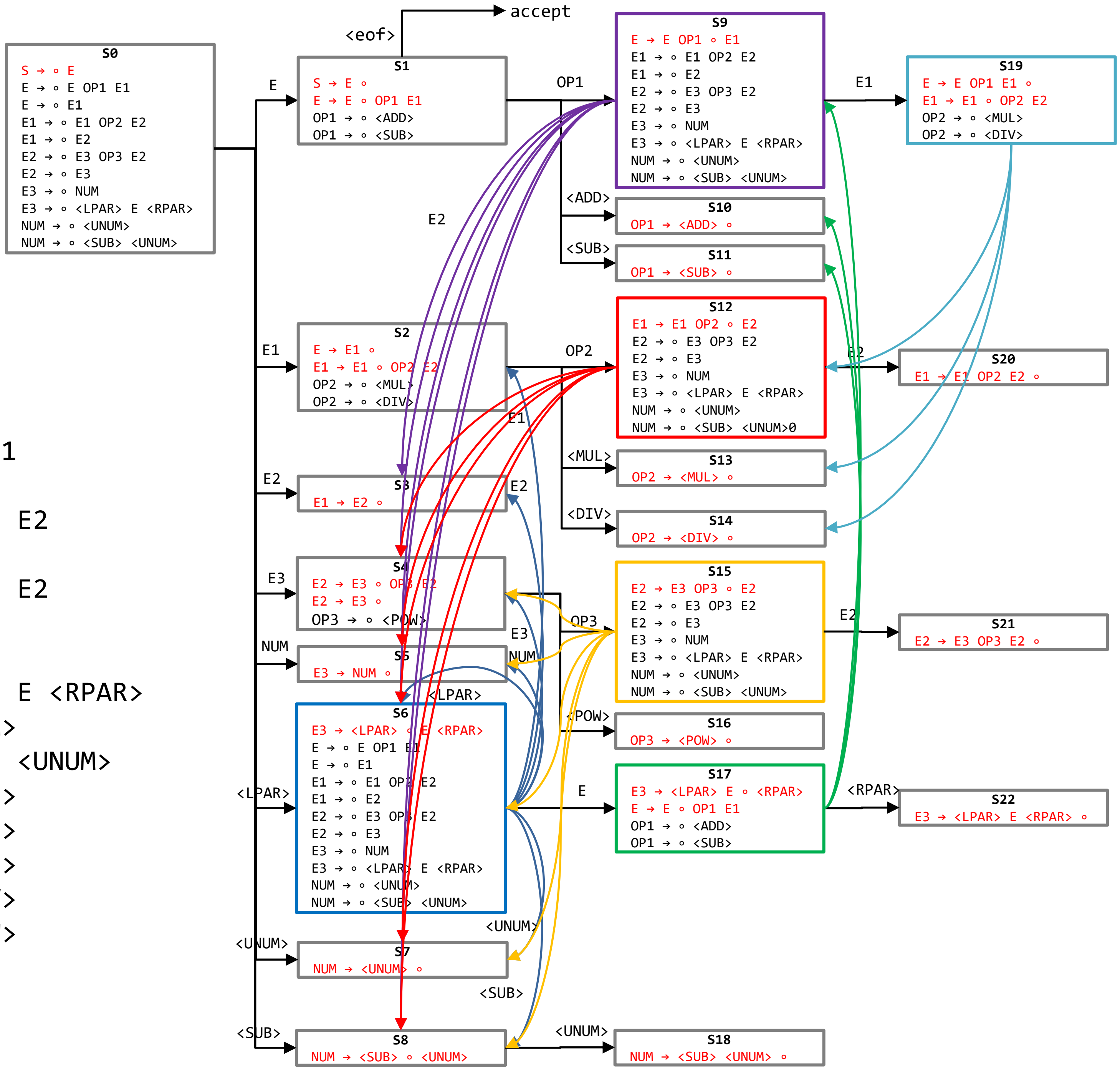


构建LR(0)自动机

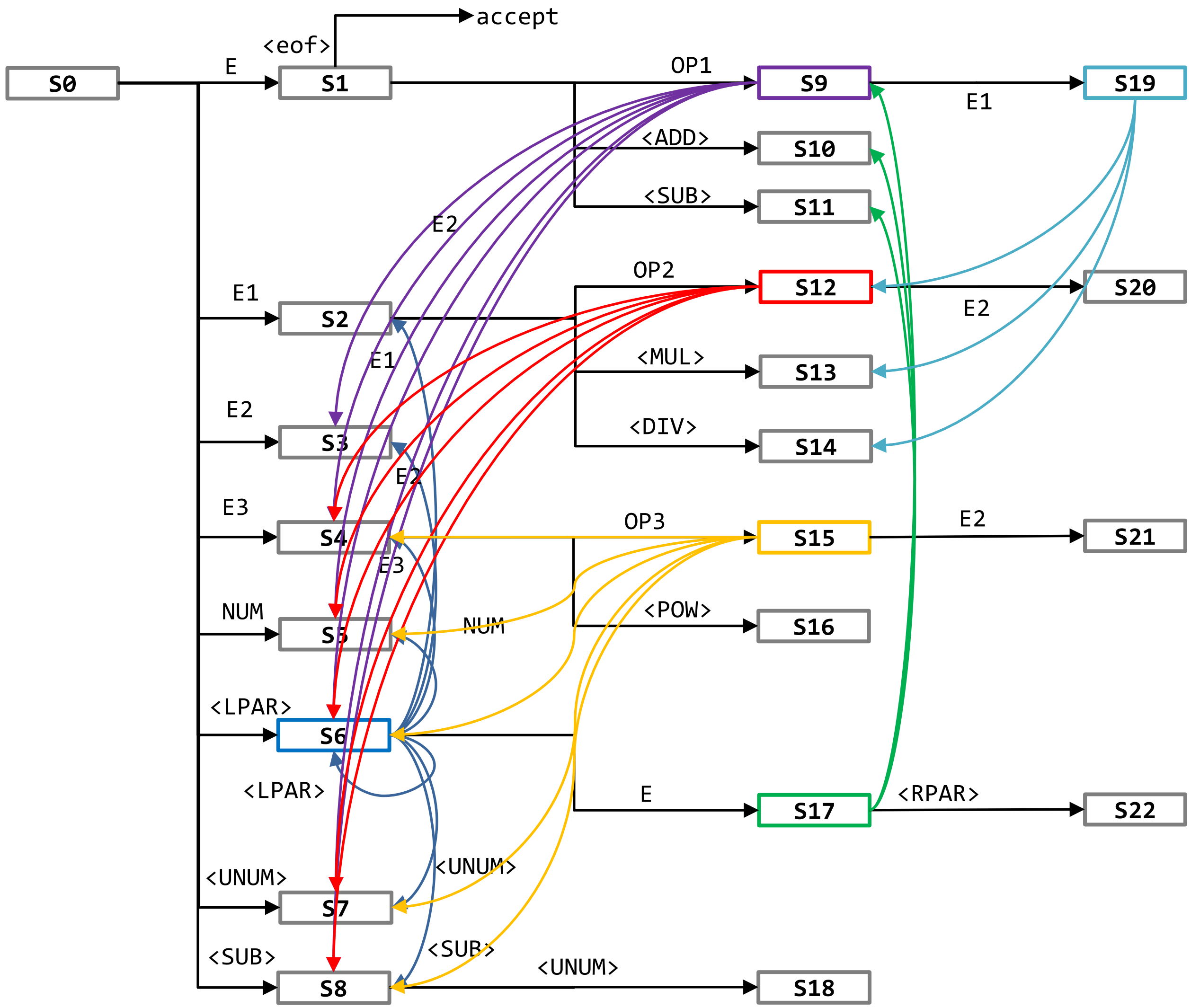
[0] S → E
[1] E → E OP1 E1
[2] | E1
[3] E1 → E1 OP2 E2
[4] | E2
[5] E2 → E3 OP3 E2
[6] | E3
[7] E3 → NUM
[8] | <LPAR> E <RPAR>
[9] NUM → <UNUM>
[10] | <SUB> <UNUM>
[11] OP1 → <ADD>
[12] | <SUB>
[13] OP2 → <MUL>
[14] | <DIV>
[15] OP3 → <POW>



[0] S → E
[1] E → E OP1 E1
[2] | E1
[3] E1 → E1 OP2 E2
[4] | E2
[5] E2 → E3 OP3 E2
[6] | E3
[7] E3 → NUM
[8] | <LPAR> E <RPAR>
[9] NUM → <UNUM>
[10] | <SUB> <UNUM>
[11] OP1 → <ADD>
[12] | <SUB>
[13] OP2 → <MUL>
[14] | <DIV>
[15] OP3 → <POW>

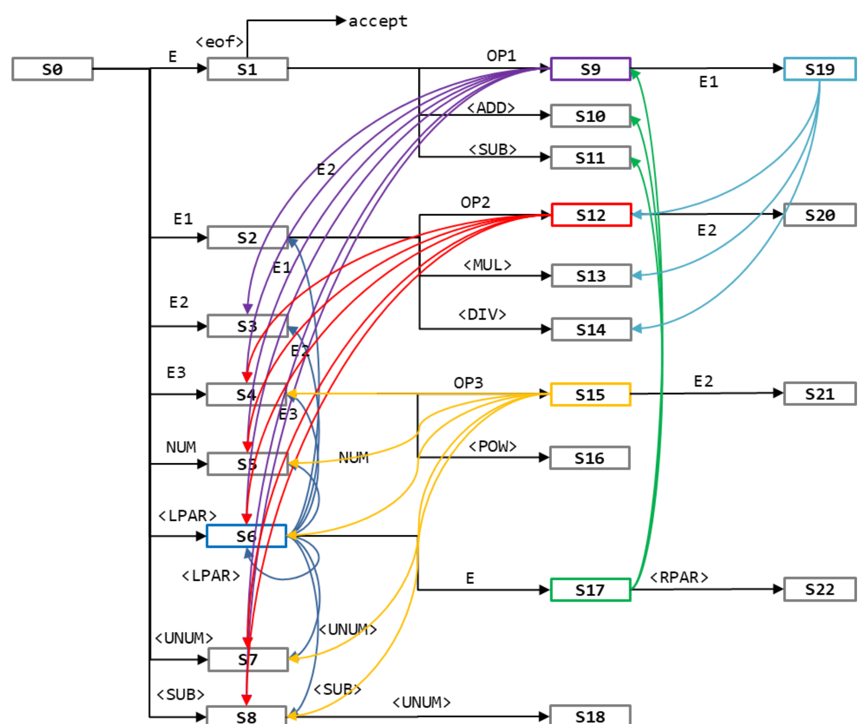


LR(0)自动机：状态转移关系



LR(0)自动机的状态转移关系表

规范族	E	E1	E2	E3	OP1	OP2	OP3	NUM	<UNUM>	<ADD>	<SUB>	<MUL>	<DIV>	<POW>	<LP>	<RP>	<eof>
S0	S1	S2	S3	S4				S5	S7		S8				S6		
S1					S9					S10	S11						accept
S2						S12						S13	S14				
S3																	
S4							S15							S16			
S5																	
S6	S17	S2	S3	S4				S5	S7		S8				S6		
...																	
S22																	

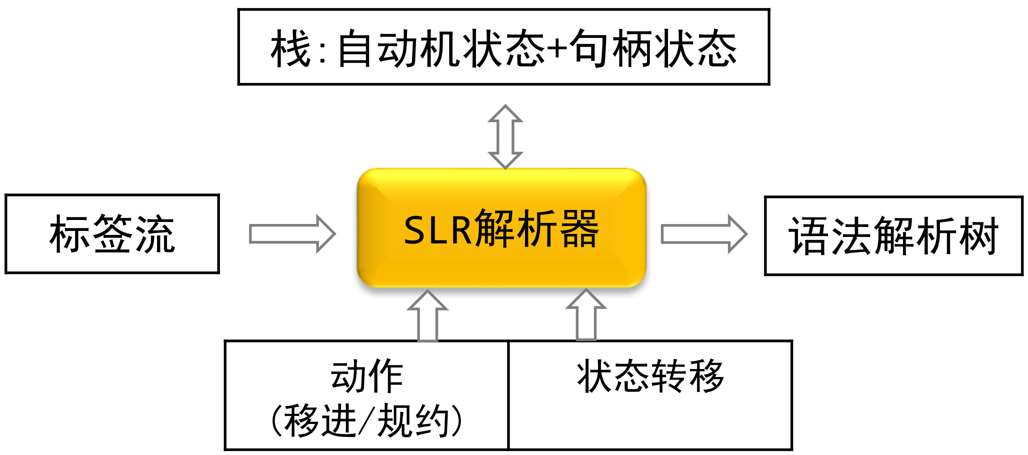


LR(0)自动机的状态转移关系表

规范族	E	E1	E2	E3	OP1	OP2	OP3	NUM	<UNUM>	<ADD>	<SUB>	<MUL>	<DIV>	<POW>	<LP>	<RP>	<eof>
S0	S1	S2	S3	S4				S5	S7		S8				S6		
S1					S9					S10	S11						accept
S2						S12						S13	S14				
S3																	
S4							S15							S16			
S5																	
S6	S17	S2	S3	S4				S5	S7		S8				S6		
S7																	
S8									S18								
S9		S19	S3	S4				S5	S7		S8				S6		
S10																	
S11																	
S12			S20	S4				S5	S7		S8				S6		
S13																	
S14																	
S15			S21	S4				S5	S7		S8				S6		
S16																	
S17					S9					S10	S11					S22	
S18																	
S19						S12						S13	S14				
S20																	
S21																	
S22																	

构建SLR解析器

- 移进条件：如果$A \to \alpha \circ a\beta \in S_i$，并且$Goto(S_i, a) = S_j$，设置$Action(S_i, a) = \text{“}Shift\ j\text{”}$
- 规约条件：如果$A \to \alpha \circ \in S_i$，$\forall a \in Follow(A)$，设置$Action(S_i, a) = \text{“}Reduce\ A \to \alpha\text{”}$



SLR解析表

规范族	GOTO								Action (Shift-Reduce)								
	E	E1	E2	E3	OP1	OP2	OP3	NUM	<UNUM>	<ADD>	<SUB>	<MUL>	<DIV>	<POW>	<LP>	<RP>	<eof>
S0	S1	S2	S3	S4				S5	S7		S8				S6		
S1					S9					S10	S11						acc
S2						S12				R[2]	R[2]	S13	S14			R[2]	R[2]
S3										R[4]	R[4]	R[4]	R[4]			R[4]	R[4]
S4							S15			R[6]	R[6]	R[6]	R[6]	S16		R[6]	R[6]
S5										R[7]	R[7]	R[7]	R[7]	R[7]		R[7]	R[7]
S6	S17							S5	S7		S8				S6		
S7										R[9]	R[9]	R[9]	R[9]	R[9]		R[9]	R[9]
S8									S18								
S9		[illegible]						S5	S7		S8				S6		
S10									R[11]		R[11]				R[11]		
S11									R[12]		R[12]				R[12]		
S12								S5	S7		S8				S6		
S13									R[13]		R[13]				R[13]		
S14									R[14]		R[14]				R[14]		
S15								S5	S7		S8				S6		
S16									R[15]		R[15]				R[15]		
S17										S10	S11					S22	
S18										R[10]	R[10]			R[10]		R[10]	R[10]
S19										R[1]	R[1]	S13	S14			R[1]	R[1]
S20										R[3]	R[3]	R[3]	R[3]			R[3]	R[3]
S21										R[5]	R[5]					R[5]	R[5]
S22										R[8]	R[8]	R[8]	R[8]			R[8]	R[8]

S2: E → E1 。
S3: E1 → E2 。
S4: E2 → E3 。
S5: E3 → NUM 。

[0] S → E
[1] E → E OP1 E1
[2] | E1
[3] E1 → E1 OP2 E2
[4] | E2
[5] E2 → E3 OP3 E2
[6] | E3
[7] E3 → NUM
[8] | <LPAR> E <RPAR>
[9] NUM → <UNUM>
[10] | <SUB> <UNUM>
[11] OP1 → <ADD>
[12] | <SUB>
[13] OP2 → <MUL>
[14] | <DIV>
[15] OP3 → <POW>

SLR查表解析应用示例

规范族	GOTO								Action (Shift-Reduce)								
	E	E1	E2	E3	OP1	OP2	OP3	NUM	<UNUM>	<ADD>	<SUB>	<MUL>	<DIV>	<POW>	<LP>	<RP>	<eof>
S0	S1	S2	S3	S4				S5	S7		S8				S6		
S1					S9					S10	S11						acc
S2						S12				R[2]	R[2]	S13	S14			R[2]	R[2]
S3										R[4]	R[4]	R[4]	R[4]			R[4]	R[4]
S4							S15			R[6]	R[6]	R[6]	R[6]	S16		R[6]	R[6]
S5										R[7]	R[7]	R[7]	R[7]	R[7]		R[7]	R[7]
S6	S17	S2	S3	S4				S5	S7		S8				S6		
S7										R[9]	R[9]	R[9]	R[9]	R[9]		R[9]	R[9]
S8									S18								
S9		S19	S3	S4				S5	S7		S8				S6		
S10									R[11]		R[11]				R[11]		

Stack	Symbols	Input	Action
S0		<UNUM><MUL><UNUM><eof>	shift <UNUM>, goto S7
S0,S7	<UNUM>	<MUL><UNUM><eof>	Reduce [9], back to S0, goto S5
S0,S5	NUM	<MUL><UNUM><eof>	Reduce [7], back to S0, goto S4
S0,S4	E3	<MUL><UNUM><eof>	Reduce [6], back to S0, goto S3
S0,S3	E2	<MUL><UNUM><eof>	Reduce [4], back to S0, goto S2
S0,S2	E1	<MUL><UNUM><eof>	Shift <MUL>, goto S13
S0,S2,S13	E1 <MUL>	<UNUM><eof>	Reduce [13], back to S2, goto S12
S0,S2,S12	E1 OP2	<UNUM><eof>	

SLR查表解析应用示例

Stack	Symbols	Input	Action
S0		<UNUM><MUL><UNUM><eof>	shift <UNUM>, goto S7
S0,S7	<UNUM>	<MUL><UNUM><eof>	Reduce [9], back to S0, goto S5
S0,S5	NUM	<MUL><UNUM><eof>	Reduce [7], back to S0, goto S4
S0,S4	E3	<MUL><UNUM><eof>	Reduce [6], back to S0, goto S3
S0,S3	E2	<MUL><UNUM><eof>	Reduce [4], back to S0, goto S2
S0,S2	E1	<MUL><UNUM><eof>	Shift <MUL>, goto S13
S0,S2,S13	E1 <MUL>	<UNUM><eof>	Reduce [13], back to S2, goto S12
S0,S2,S12	E1 OP2	<UNUM><eof>	Shift <UNUM>, goto S7
S0,S2,S12,S7	E1 OP2 <UNUM>	<eof>	Reduce [9], back to S12, goto S5
S0,S2,S12,S5	E1 OP2 NUM	<eof>	Reduce [7], back to S12, goto S4
S0,S2,S12,S4	E1 OP2 E3	<eof>	Reduce [6], back to S12, goto S20
S0,S2,S12,S20	E1 OP2 E2	<eof>	Reduce [4], back to S12, goto S2
S0	E1	<eof>	Reduce [3], back to s0, goto S2
S0,S2	E1	<eof>	Reduce [2], back to s0, goto S1
S0,S1	E	<eof>	accept

二、更多文法

思考：LL(1)和SLR哪个语法的表达能力更强？

- 如果一个语法是SLR，是否一定是LL(1)？
- 如果一个语法是LL(1)，是否一定是SLR？

[1]	$S \rightarrow AaAb$
[2]	$\quad\ \ |BbBa$
[3]	$A \rightarrow \epsilon$
[4]	$B \rightarrow \epsilon$

是LL(1)

- $First^{+}(S \rightarrow AaAb) = \{a\}$
- $First^{+}(S \rightarrow BbBa) = \{b\}$

不是SLR(1)

- $Follow(A) = Follow(B) = \{a, b\}$
- $Action(S_0, a) = reduce[3]$ 或 $reduce[4]$

LL(1)/SLR不够用怎么办？

- 表达能力太弱：解析表存在冲突
  - LR(1)>LALR>SLR：规范族构造时考虑Follow信息
  - LL(K) > LL (1)
- 通用上下文无关文法解析算法：
  - 自底向上： CYK、GLR（Generalized LR）
  - 自顶向下：Earley算法