

总复习

赵银亮

2025

¹论域{¹¹中心概念{符号, 字母表, 符号串, 语言, 问题}}, ¹²图灵机模型{状态, 当前状态, 输入串, 当前输入符号, 状态转换规则, 初始状态, 接受状态, UTM}, ¹³Chomsky体系{形式文法, 形式语言}, ¹⁴编译过程{词法分析, 语法分析, 语义分析, 代码优化, 代码生成, 编译遍}, ¹⁵编译程序{源语言, 中间语言, 机器语言, 源程序, 中间表示, 可执行程序, 前端, 后端}, ¹⁶主文法, ¹⁷主符号系统{Lisp表, 属性表, 符号表, 实用函数, 命名惯例}},

²正则语言{²¹识别器{DFA^[12]{ Q, Σ, v, q_0, F , 完全形, 最简形, 简化型, 最小形}}, NFA^[211]{ v_N, ε 转移, ω, ε 闭集, 活动状态集, ε -NFA, g-NFA}}, RE^[11]{语言表达式, 原子语言, 语言运算, 语言模式}}, ²²判定性质{ δ ^[211-2], 模式匹配^[213], PATH[], 线索穷举树}, ²³等价性质^[21]{子集法^[21], 填表法^[211], 划分法^[211], 去除 ε 转移^[212], 去除无用状态^[21], 消除中间状态^[2127,213], 构建中间状态^[2127,213]}, ²⁴封闭性, ²⁵泵引理},

³CFL{³¹识别器{CFG^[13]{变元, 终结符, 初始符号, 产生式, 候选式, 子文法}}, PDA^[212]{ $\Gamma, Z_0, \delta, N, P, D$ }}, ³²判定性质{推导^[311]{句型, 句子, 直接推导, 最左推导, 最右推导, 确定性}, 归约^[311]{直接归约, 可归约串, 确定性}, 语法树^[311]{产物, 短语, 直接短语, 句柄}, 移动^[312]{瞬时描述, 性质, 确定性}}, ³³等价性质^[31]{文法修剪^[311]{ $a, g, \bar{e}, \bar{u}, \bar{a}, \bar{r}, \bar{c}$ }}, P2N^[312,324], N2P^[312,324], $^UN2C, ^UC2N$ }, ³⁴歧义性{定义, 来源{优先级, 结合性, 悬空else}, 固有歧义性}},

⁴词法分析{⁴¹词法记号{一符一种, 全体一种}}, ⁴² σ -DFA^[21]{ \mathcal{L}, ψ , 事实优先级, \mathbb{L} , 前缀最大化},

⁴³扫描框架{预处理{有效串}, 判定框{转移表, 识别流程}, 附加处理{缓冲区控制, 超前搜索}, 后处理{关键字识别, 值转换, 表格}}, ⁴⁴scanner(), ⁴⁵生成器{描述文件}}, ⁵LL(1)分析{⁵¹LL(1)文法^[311]{性质, 首符集, FOLLOW集, LL(1)修剪}, ⁵²LL(1)分析器{LL(1)框架, 递归下降分析程序, 预测分析表}}, ⁶SLR(1)分析{⁶¹最左归约{最左可归约串, 非确定性, 不完备性}, ⁶²SRF{符号栈, 移进, 归约, SR冲突, RR冲突}, ⁶³规范归约{活前缀, 句柄, 目标, 愿望, 积跬步, 有效项目, LR(0)规范簇^[32], 确定性}, ⁶⁴规范归约模拟器{itemNFA, itemDFA^[311,324], SLR(1)文法^[624], 冲突消解{默认冲突, 额外冲突}}, ⁶⁵SLR(1)分析器{LR(0)分析表^[632], SLR(1)分析表^[632], SLR(1)框架^[632]}, $^ULR(1)$ 分析, ULALR 分析, U 生成器},

⁷语法制导语义分析{⁷¹属性文法^[312]{属性, 属性名, 属性方程, 综合属性, 继承属性, 属性求值, 桩变量, S-属性文法, L-属性文法}, ⁷²分析框架{LL(1)制导分析框架^[72], SLR(1)制导分析框架^[72]}, ⁷³函数符号表{表头{声明宿主, 类型宽度, 参数个数, 参数表, 返回类型, 层数, 代码}, 登记项{名字, 类型, 类型特有域}, 操作{newtab(), bind(), lookup(), update[]}}, ⁷⁴声明分析^[71,73]{形参声明, 局部名声明, 属性方程设计}, ⁷⁵中间语言{三地址指令, 四元式, U 抽象语法树, U 后缀式, U 类型检查}, ⁷⁶中间代码生成^[71,75]{名字引用, 表达式, 语句, 属性方程设计}}, ⁸运行时环境{⁸¹内存映像{活动树, 栈帧{参数区, 链接区, 局部区}, 栈快照, 内存快照}, ⁸²参数传递{形参{简单变量, 数组原型, 函数原型, U 数组, U 函数签名}, 实参{算术表达式, 临时变量, 数组原型, 函数原型}, 机制{传值, 传地址, U 得结果, U 传名}}, ⁸³可执行代码^[812]{调用序列, 返回序列, 序言, 尾声, 局部名寻址}}, ⁸⁴非局部名寻址{声明宿主, 引用宿主, 访问链, U 访问链优化}, ⁸⁵目标代码{指令系统, ⁹²C指令模板, U 库{堆区管理, 预定义函数}}, U 优化{寄存器分配, 代码优化, ⁸⁷³栈帧优化{显示表, 参数传递优化}}, R 离散数学, R 数据结构, R 程序设计C, R 汇编语言{MIPS, ARM}, U 基础专业知识。



关于主文法

- ▶ $P \rightarrow \check{D} \check{S}$
- ▶ $\check{D} \rightarrow \varepsilon \mid \check{D} D ;$
- ▶ $D \rightarrow T d \mid T d[\check{I}] \mid T d(\check{A})\{\check{D} \check{S}\}$
- ▶ $T \rightarrow \text{int} \mid \text{float} \mid \text{void}$
- ▶ $\check{I} \rightarrow i \mid \check{I}, i$
- ▶ $\check{A} \rightarrow \varepsilon \mid \check{A} A ;$
- ▶ $A \rightarrow T d \mid T d[] \mid \cancel{T d[\check{I}]} \mid \cancel{T d(\check{T})} \mid T d()$
- ▶ ~~$\check{T} \rightarrow \varepsilon \mid \check{T} T ;$~~
- ▶ 对形参更改：有数组原型和函数原型，无形参数组和函数签名。



关于主文法 (续)

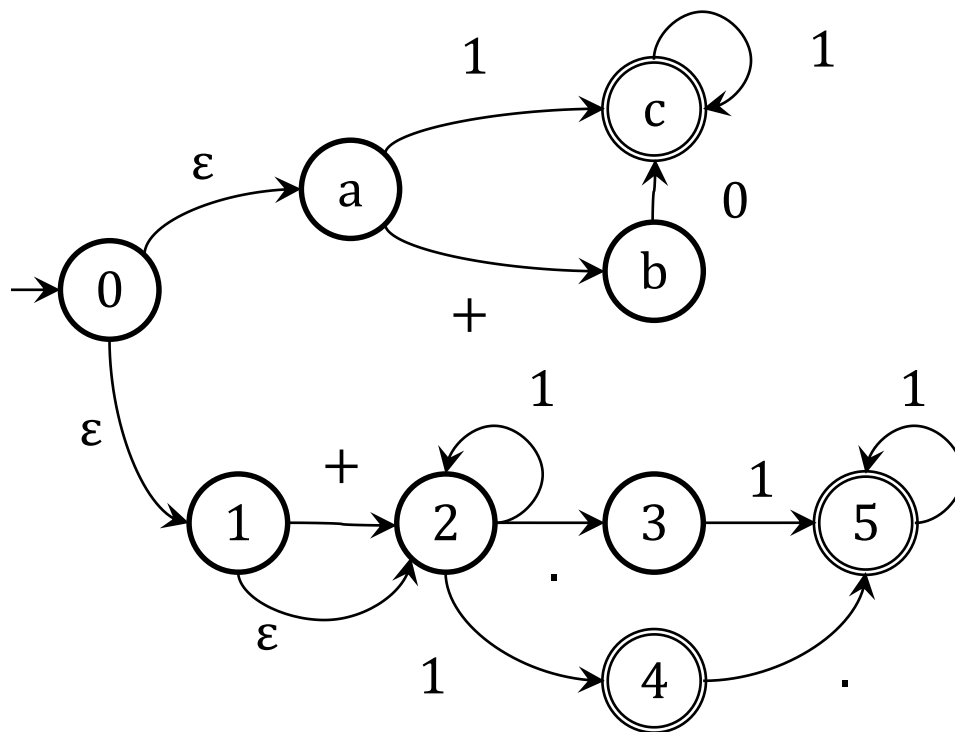
- ▶ $\check{S} \rightarrow S \mid \check{S}; S$
- ▶ $S \rightarrow d = E \mid d[\check{E}] = E \mid \text{if } (B) S \mid \text{if } (B) S \text{ else } S \mid \text{while } (B) S \mid \text{return } E \mid \{\check{S}\} \mid d(\check{R})$
- ▶ $E \rightarrow i \mid f \mid d \mid d[\check{E}] \mid E \circ E \mid u E \mid (E) \mid d(\check{R})$
- ▶ $\check{E} \rightarrow E \mid \check{E}, E$
- ▶ $B \rightarrow B \wedge B \mid B \vee B \mid ! B \mid (B) \mid E \text{ r } E \mid E$
- ▶ $\check{R} \rightarrow \varepsilon \mid \check{R} R,$
- ▶ $R \rightarrow E \mid d[] \mid d()$
- ▶ 对算术运算符即有全集一种即 \circ 也有一符一种即 $+ - * /$, 对关系运算符类似。
- ▶ 对布尔运算也有 $\&\&$ 和 \parallel



NFA转DFA

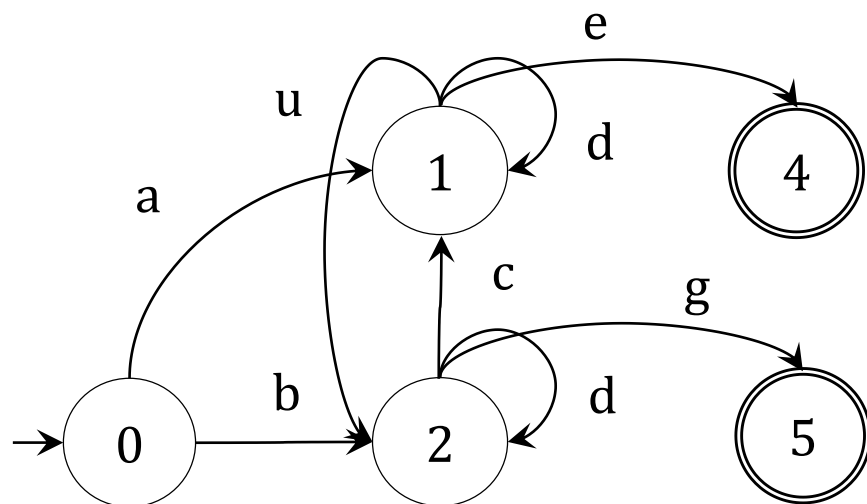
已知NFA如图所示，试完成以下3个小题。

- (1) 写出与该NFA等价的正则表达式；
- (2) 将该NFA等价地转换为DFA；
- (3) 判断所得DFA是否为最小，给出理由。





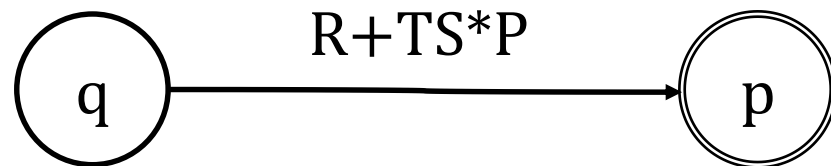
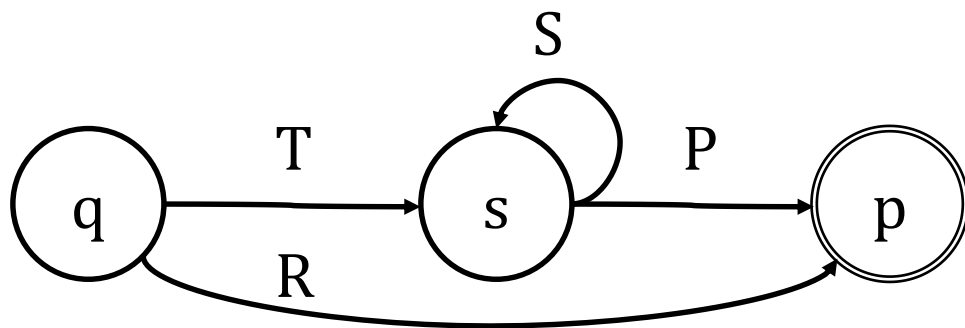
将DFA转为RE



$$\begin{aligned} & a(d+ud^*c)^*e \\ & + b(d+cd^*u)^*e \\ & + a(d+ud^*c)^*ud^*g \\ & + b(d+cd^*u)^*cd^*e \end{aligned}$$

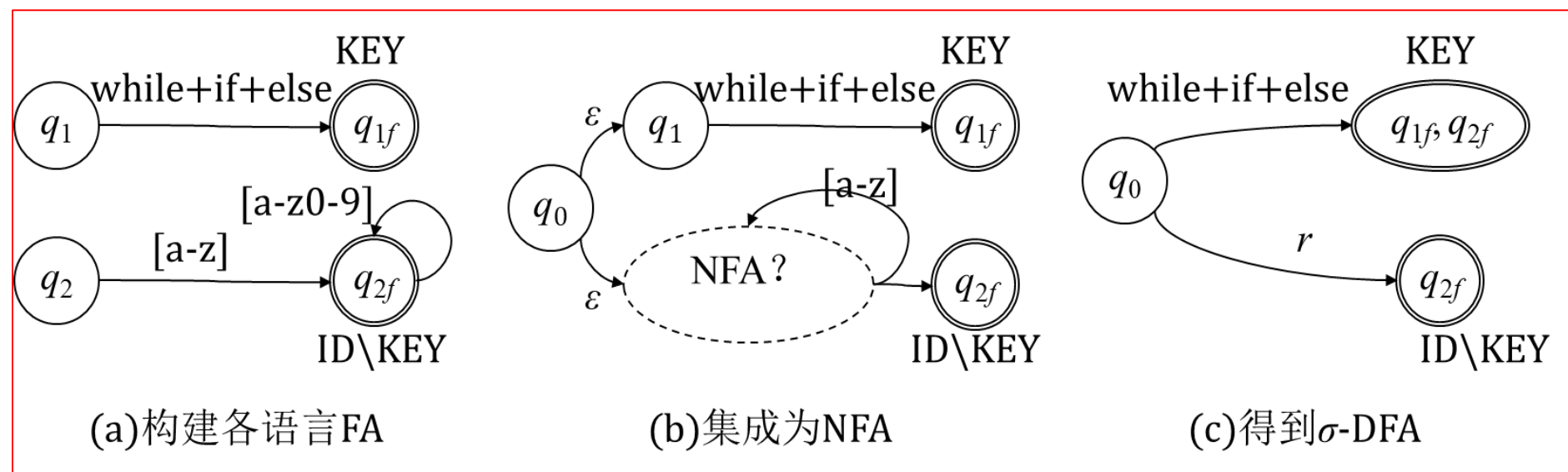
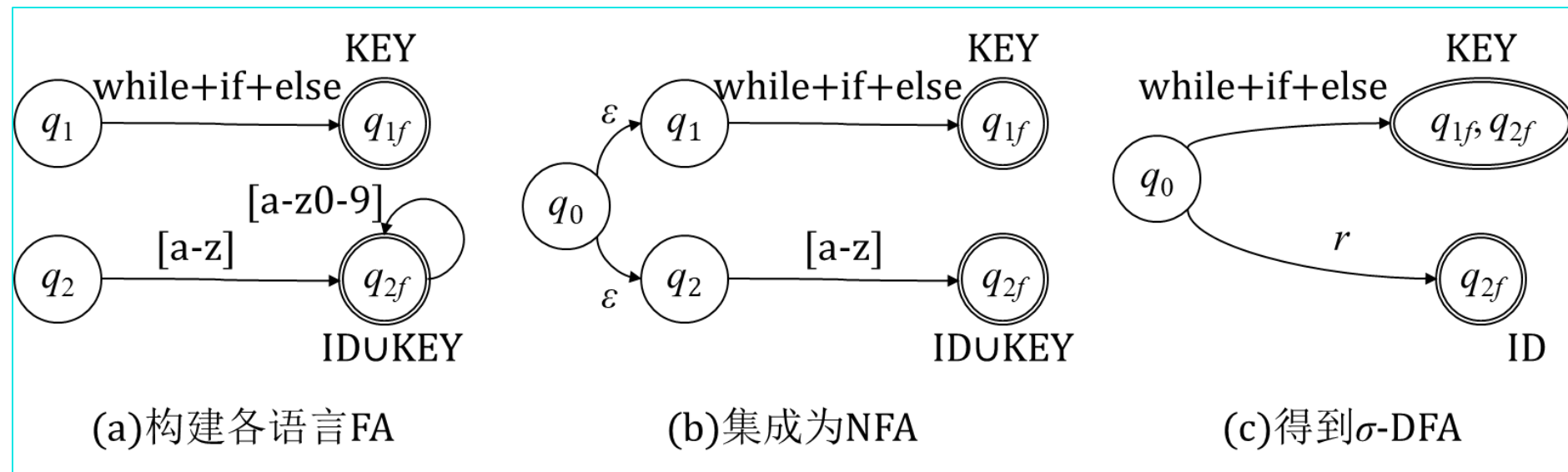
填表法/划分法

反证如状态4和5
等价即得结论





σ -DFA(\mathcal{L})、 \mathbb{L} 、 ψ





(1) 消除文法歧义性

(2) 消除文法的无用符号

$$S \rightarrow AB|\varepsilon \quad A \rightarrow BC|a \quad C \rightarrow b$$

▶ 消去无产出变元得 $S \rightarrow \varepsilon \quad A \rightarrow a \quad C \rightarrow b$

▶ 消去不可达变元得最终结果 $S \rightarrow \varepsilon$

(3) 消除文法的 ε -产生式 $S \rightarrow AD|b \quad A \rightarrow aA|\varepsilon \quad D \rightarrow b|\varepsilon$

▶ 最终结果是 $S \rightarrow AD|D|A|b \quad A \rightarrow aA|a \quad D \rightarrow b$

▶ 另解：把D代入S中得 $S \rightarrow Ab|A|b \quad A \rightarrow aA|a$

▶ 另解：继续代入，把A代入S中得 $S \rightarrow aAb|aA|b \quad A \rightarrow aA|a$

(4) 消除文法的单位产生式 $E \rightarrow T|iT \quad T \rightarrow F|Ti \quad F \rightarrow i|(E)$

▶ 结果是 $E \rightarrow i|(E)|Ti|iT \quad T \rightarrow i|(E)|Ti \quad F \rightarrow i|(E)$

▶ 注：同一变元得候选式次序随意写；F产生式也可省略；



(5) 消除文法的左递归产生式 $\check{A} \rightarrow A | \check{A}A$; $A \rightarrow Td | Td[\check{I}]$ $\check{I} \rightarrow i | \check{I}, i$

- ▶ 最终结果是 $\check{A} \rightarrow A | A; \check{A}$ $A \rightarrow Td | Td[\check{I}]$ $\check{I} \rightarrow i | i, \check{I}$
- ▶ 另一结果 $\check{A} \rightarrow A\check{A}'$ $\check{A}' \rightarrow ;A\check{A}' | \varepsilon$ $A \rightarrow Td | Td[\check{I}]$ $\check{I} \rightarrow i\check{I}'$ $\check{I}' \rightarrow ,\check{I}' | \varepsilon$
- ▶ 另一结果 $\check{A} \rightarrow A | A\check{A}'$ $\check{A}' \rightarrow ;A\check{A}'$. . .
- ▶ 注: \check{A} 和 \check{I} 的产生式都有左递归, 都要消除;

(6) 消除文法的可回溯性

- ▶ 引入变元提取公共前缀

(7) 修剪为LL(1)文法

- ▶ 确定修剪方案包括划分为多个子文法
- ▶ 对各子文法进行必要修剪
- ▶ 合并各子文法修剪结果并去除无用符号
- ▶ 判断所得结果是否满足LL(1)文法的条件



- ▶ 给定文法，写出每个变元的首符集和**FOLLOW**集
（注意明确计算方法，计算过程不要有遗漏）

- ▶ 检查该文法是不是满足**LL(1)**文法的条件
 - 同一变元的各个候选式的首符集两两不相交
 - 每个变元的首符集和**FOLLOW**集不相交（注意前提是首符集包含 ϵ ）
 - 注意：应该说明检查了什么，而不是简单地是或否。



- CFG (V, T, P, S) 是 LL(1) 文法，它的预测分析表 M 满足，对于任意 $(A, \gamma) \in P$,
- (1) 如果 $a \in \text{FIRST}(\gamma)$ 那么 $M[A, a] = (A, \gamma)$ 且
 - (2) 如果 $\epsilon \in \text{FIRST}(\gamma)$ 且 $a \in \text{FOLLOW}(A)$ 那么 $M[A, a] = (A, \gamma)$



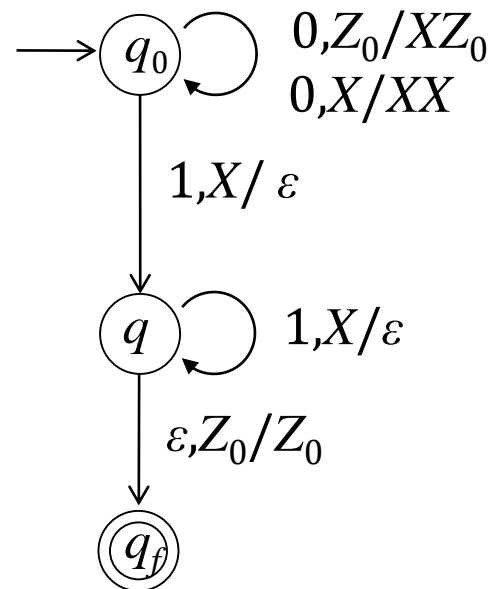
CFG判定性质

- ▶ 语法树
- ▶ 推导
- ▶ 最左推导
- ▶ 最右推导
- ▶ 规范归约
- ▶ 相关概念术语：产物、根、内节点、双亲结点、孩子结点、子树、句型、句子、短语、直接短语、句柄
- ▶ 语法树与文法对应关系
- ▶ 文法歧义性

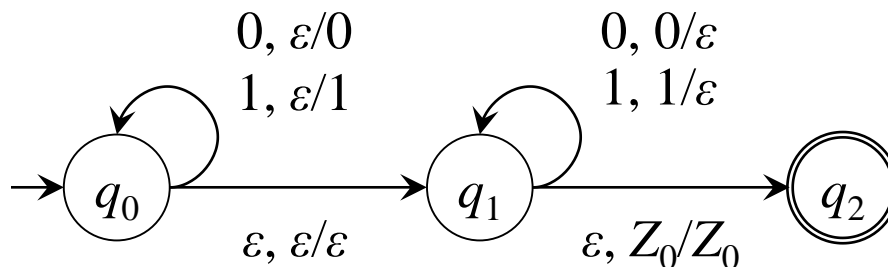


- ▶ 瞬时描述 $ID(<状态>, <剩余串>, <栈内容>)$
- ▶ 移动: 0到多步直接移动, 其中直接移动 定义为,
 $ID(q, ax, X\alpha) \vdash ID(p, x, \gamma\alpha)$ 当且仅当 $(p, \gamma) \in \delta(q, a, X)$
- ▶ 判定性质: $ID(q_0, w, Z_0) \vdash_P ID(q_f, \varepsilon, \eta)$ 或 $ID(q_0, w, Z_0) \vdash_N ID(q, \varepsilon, \varepsilon)$
- ▶ 例, $(q_0, 000111, Z_0) \vdash (q_0, 00111, XZ_0) \vdash (q_0, 0111, XXZ_0) \vdash (q_0, 111, XXXZ_0) \vdash (q, 11, XXZ_0) \vdash (q, 1, XZ_0) \vdash (q, \varepsilon, Z_0) \vdash (q_f, \varepsilon, Z_0)$
- ▶ 其他移动线索

$$\begin{aligned}\delta(q_0, 0, Z_0) &= \{(q_0, XZ_0)\} \\ \delta(q_0, 0, X) &= \{(q_0, XX)\} \\ \delta(q_0, 1, X) &= \{(q, \varepsilon)\} \\ \delta(q, 1, X) &= \{(q, \varepsilon)\} \\ \delta(q, \varepsilon, Z_0) &= \{(q_f, Z_0)\}\end{aligned}$$



是否为DPDA？例7.3的图7-5不是DPDA



$\forall q \in Q, a \in \Sigma, X \in \Gamma \cdot$

$$(|\delta(q, a, X)| + |\delta(q, a, \varepsilon)| + |\delta(q, \varepsilon, X)| = 1 \\ \wedge \delta(q, \varepsilon, \varepsilon) = \emptyset)$$

$$\forall q \in Q \cdot (|\delta(q, \varepsilon, \varepsilon)| = 1 \rightarrow \\ (\forall a \in \Sigma, X \in \Gamma \cdot \delta(q, a, X) = \\ \delta(q, \varepsilon, X) = \delta(q, a, \varepsilon) = \emptyset))$$

$$\delta(q_0, 0, \varepsilon) = \{q_0, 0\}$$

$$\delta(q_0, 1, \varepsilon) = \{q_0, 1\}$$

$$\delta(q_0, \varepsilon, \varepsilon) = \{q_1, \varepsilon\}$$

$$\delta(q_1, 0, 0) = \{q_1, \varepsilon\}$$

$$\delta(q_1, 1, 1) = \{q_1, \varepsilon\}$$

$$\delta(q_1, \varepsilon, Z_0) = \{q_2, Z_0\}$$



itemDFA

- ▶ 规范归约模拟器
- ▶ 给定文法构建itemDFA（也称识别活前缀DFA）：
初始状态为 $\omega[S' \rightarrow .S]$ ；转移函数为 $v(q, X)=p$ ，如果
 $p = \omega \cup [A \rightarrow \rho.X\eta] \in q \bullet [A \rightarrow \rho X.\eta]$
 ω 含义为： $\omega[A \rightarrow \rho.N\eta] = \{[A \rightarrow \rho.N\eta]\} \cup \omega\{[N \rightarrow .\gamma] \mid (N, \gamma) \in P\}$ ，其中 ρ 和 η 为任意文法符号串。
注意不关心是否标注接受状态，即含有完全项目的状态。
- ▶ 概念术语：右句型、句柄、活前缀、有效项目、增广文法
文法项目、初始项目、完全项目、移进项目
- ▶ SLR(1)文法，默认冲突消解规则、额外冲突消解规则。
- ▶ SLR(1)分析表。



写出itemDFA

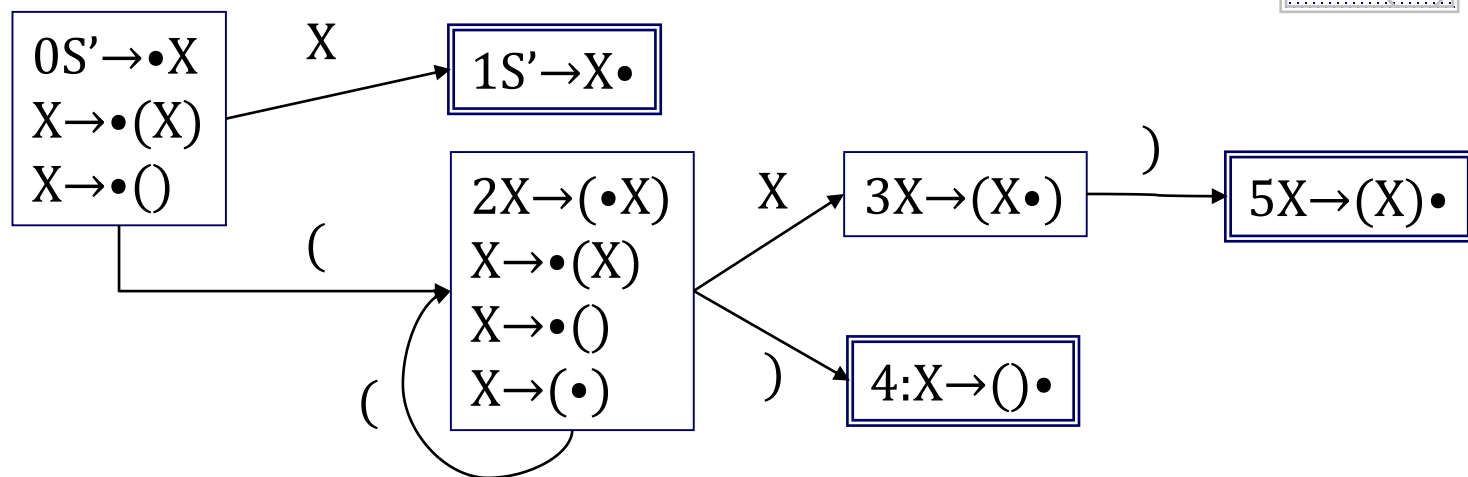
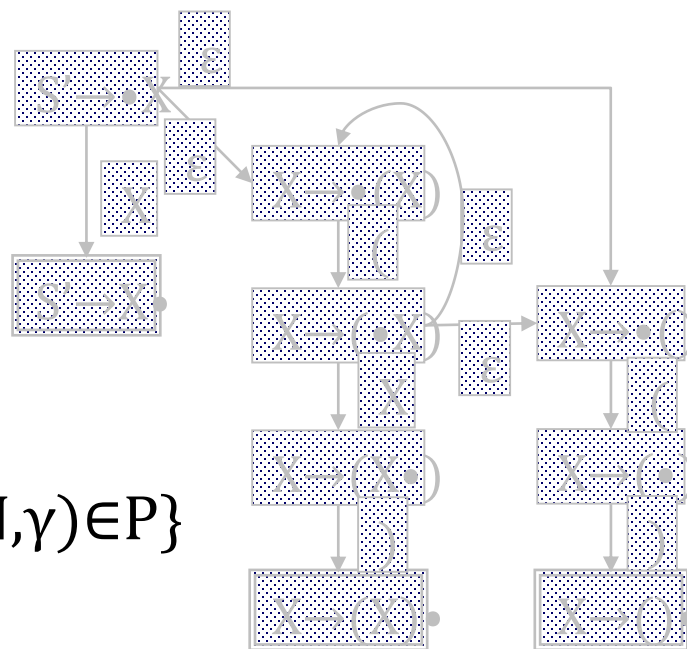
① ~~构建itemNFA并转itemDFA;~~

② 直接写出itemDFA;

➤ 初始状态为 $\omega[S' \rightarrow \cdot S]$; 转移函数为 $v(q, X) = p$, 如果

$$p = \omega\{[A \rightarrow \rho X \eta] \mid [A \rightarrow \rho \cdot X \eta] \in q\}$$

➤ $\omega[A \rightarrow \rho \cdot N \eta] = \{[A \rightarrow \rho \cdot N \eta]\} \cup \omega\{[X \rightarrow \cdot \gamma] \mid (N, \gamma) \in P\}$



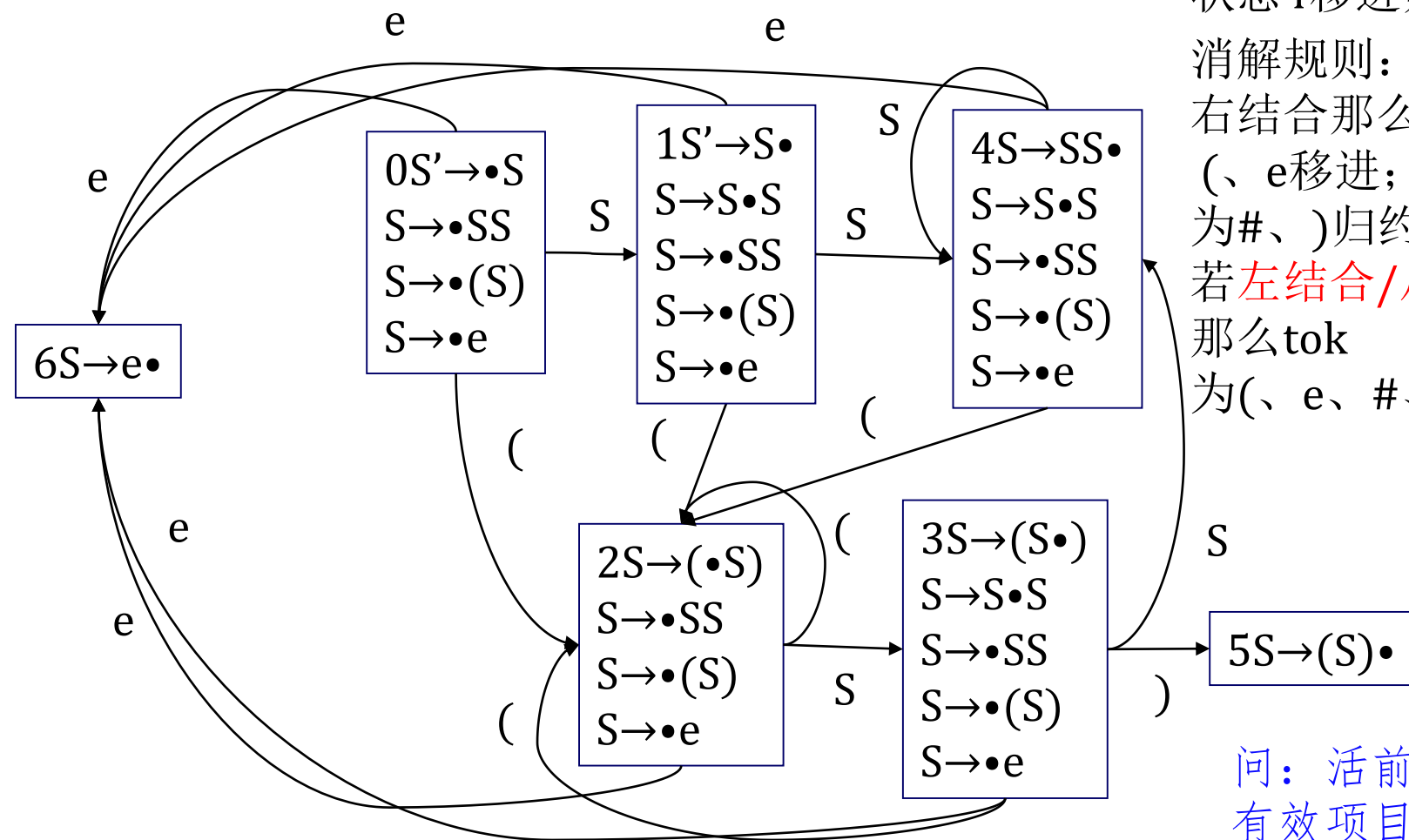
CFG G: $X \rightarrow (X) \quad X \rightarrow ()$

或G增广文法: $S' \rightarrow X \quad X \rightarrow (X) \quad X \rightarrow ()$



冲突消解

- $FOLLOW(S) = \{ \#,), (, e \}$ $FIRST(S) = \{ (, e \}$
- tok为(则移进, 为#或)则用 ϵ 候选式进行归约



状态1移进归约冲突
消解规则: tok为
(、e移进; #归约。
状态4移进归约冲突
消解规则: 若
右结合那么tok为
(、e移进;
为#、)归约。
若左结合/从左往右
那么tok
为(、e、#、)归约

问: 活前缀((的
有效项目?2所示



► 例 对于下列程序试完成：

(1) 写出该程序的符号表

(2) 当foo(3)活动结束后但还未结束之时刻的栈快照，假定按字编址，且栈底单元地址为500，并假设a数组在执行之前已初始化。

```
int x;
```

```
int a[2, 3]; //a[i,j]=j-i
```

```
int foo(int x;){x=x+a[0,2]; return x};
```

```
int g(int r(); int y; int b[];){
```

```
    if(y<=x^b[3])y=g(r(), r(x,), b[],) else y=r(y,);  
    return y};
```

```
x=1;
```

```
g(foo(), x, a[],)
```



写出符号表

```
@table:(outer:NIL width:52 argc:0 arglist:NIL rtype:INT level:0 code:[t10=1;
x=t10; PAR a; PAR x; PAR foo; t11=CALL g, 3] entry:(name:x type:INT
offset:4) entry:(name:a type:ARRAY base:28 etype:INT dims:2 dim[0]:2
dim[1]:3) entry:(name:foo type:FUNC offset:36 mytab:foo@table)
entry:(name:g type:FUNC offset:44 mytab: g@table) entry:(name:t10
type:TEMP offset:48) entry:(name:t11 type:TEMP offset:52))
```

```
foo@table:(outer:@table width:28 argc:1 arglist:(x) rtype:INT level:1
code:[t1=0; t2=t1*3; t3=t2+2; t4=t3*4; t5=a[t4] t6=x+t5; x=t6; RETURN x]
entry:(name:x type:INT offset:4) entry:(name:t1 type:TEMP offset:8) ...
entry:(name:t6 type:TEMP offset:28))
```

```
g@table:(outer:@table width:28 argc:3 arglist:(r y b) rtype:INT level:1)
code:[IF y<=x THEN l1 ELSE l2; LABEL l1; t7=3; t8=b[t7]; IF t8!=0 THEN l3
ELSE l4; LABEL l3; PAR x; t9=CALL r,1; PAR b; PAR t9; PAR r; y=CALL g, 3;
GOTO l5; LABEL l4; LABEL l2; PAR y; y=CALL r,1; RETURN y; LABEL l5;]
entry:(name:r type:FUNPTT offset:8 rtype:INT) entry:(name:y type:INT
offset:12) entry:(name:b type:ARRPTT offset: 16 etype:INT) entry:(name:t7
type:TEMP offset:20) ... entry:(name:t9 type:TEMP offset:28))
```



当foo(3)活动结束后还未结束时刻的栈快照

500 <访问链> NIL

499 <控制链> NIL

498 <返址>

497 x 1

496 a[1,2] 1

495 a[1,1] 0

494 a[1,0] -1

493 a[0,2] 2

492 a[0,1] 1

491 a[0,0] 0

490 foo[1] 499

489 foo[0] foo@label

488 g[1] _

487 g[0] g@label

486 t10 1

485 t11 _

484 <参数3> 491

483 <参数2> 1

482 <参数1> 477

481 <访问链> 499

480 <控制链> 499

479 <返址>

478 r[1] 499

477 r[0] r@label

476 y 1/5

475 a 491

474 t7 3

473 t8 -1

472 t9 3/5



当foo(3)活动结束后还未结束时刻的栈快照

471 <参数3> 491

470 <参数2> 3

469 <参数1> 464

468 <访问链> 499

467 <控制链> 480

466 <返址>

465 r[1] 499

464 r[0] r@label

463 y 3/5

462 a 491

461 t7 _

460 t8 _

459 t9 _

458 <参数1> 3

457 <访问链> 499

456 <控制链> 467

455 <返址>

454 x 3/5

453 t1 0

452 t2 0

451 t3 2

450 t4 8

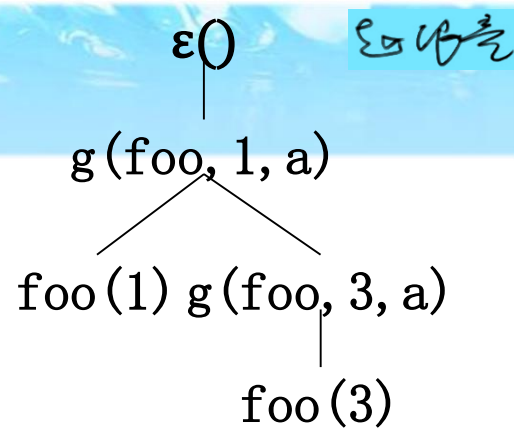
449 t5 2

448 t6 5

```

int x; int a[2, 3]; //假定初始化为a[i,j]=j-i
int foo(int x;){x=x+a[0,2]; return x;}
int g(int r(); int y; int b[];){
    if(y<=x^b[3])y=g(r(), r(x), b[],) else y=r(y,);
    return y};
x=1; g(foo(), x, a[],)

```



```

@frame:(alink:NIL; clink:NIL; raddr; x:1 a[1,2]:1 a{1,1}:0 a[1,0]:-1 a[0,2]:2 a{0,1}:1
a[0,0]:0 foo[1]:@frame foo[0]:foo@label g[1] g[0]:g@label t10:1 t11:5)

```

```

g(foo,1,a)@frame:(arg3:&b arg2:1 arg1:&r alink:@frame clink:@frame raddr r[1]:@frame
r[0]:foo@label y:1/5 a:&b t7:3 t8:-1 t9:3/5 )

```

```

foo(1)@frame:(arg1:1 alink:@frame clink:g(foo,1,a)@frame raddr x:1 t1:0 t2:0 t3:2 t4:8
t5:2 t6:3)

```

```

g(foo,3,a)@frame:(arg3:&b arg2:3 arg1:&r alink:@frame clink:g(foo,1,a)@frame raddr
r[1]:@frame r[0]:foo@label y:3/5 a:&b t7:_ t8:_ t9:_ )

```

```

foo(3)@frame:(arg1:3 alink:@frame clink:g(foo,1,a)@frame raddr x:1/5 t1:0 t2:0 t3:2
t4:8 t5:2 t6:5)

```

```

foo@code:[t1=0; t2=t1*3; t3=t2+2; t4=t3*4 t5=a[t4] t6=x+t5; x=t6; RETURN x]

```

```

g@code:[IF y<=x THEN l1 ELSE l2; LABEL l1; t7=3; t8=b[t7]; IF t8!=0 THEN l3 ELSE l4;
LABEL l3; PAR x; t9=CALL r,1; PAR b; PAR t9; PAR r; y=CALL g, 3; GOTO l5; LABEL l4;
LABEL l2; PAR y; y=CALL r,1; RETURN y; LABEL l5]

```




- **callseq:** 列出各项功能
- **retseq:** ...
- **prologue:** ...
- **epilogue:** ...
- **name reference:** 局部名、非局部名、函数原型foo、数组元素a[t]



- ▶ 答疑：线上即时答疑；线下临近安排届时通知

预祝大家考试取得好成绩！