第三章　词法分析

廖力
xobjects@seu.edu.cn
3793235

第三章　词法分析

- 词法分析是编译的第一个阶段，在单词的级别上分析和翻译源程序。
- 理论基础：
  - 有限自动机理论
  - 有限自动机理论与正规文法、正规式之间在描述语言方面有一一对应的关系。
- 学习目标：
  - 掌握有限自动机与正规文法、正规式之间的转换。
  - 能够构造词法分析程序，完成实验1。

第一节　正规文法和有限自动机

第一节 正规文法和有限自动机

一、正规文法、正规集与正规式

1、正规文法

- 是Chomsky 3型文法

- 注：正规文法是描述正规集的文法，可用于描述程序设计语言的语法部分。例如：标识符这种单词可以用下面的规则描述。
  - <标识符> → <字母>|<标识符>(<字母>|<数字>)
  - <字母>表示任意英文字母，<数字>表示任意数字)

第一节　正规文法和有限自动机

一、正规文法、正规集与正规式

2、正规集

- 由正规文法产生的语言。

- 注：正规集是集合，可有穷也可无穷。可通过正规式来形式化表示。

第一节　正规文法和有限自动机

一、正规文法、正规集与正规式

3、正规式

– 设A是非空的有限字母表，$A=\{a_i \mid i=1,2,\ldots\ldots n\}$,则

1. ε，Φ和a_i $(i=1,2,\ldots\ldots n)$都是正规式。

2. 若α、β是正规式，则α|β、 α•β 、α*、 β +也是正规式。

3. 正规式只能通过有限次使用1，2规则获得。

- 注：1)“|”读作为“或”，也可写作为“+”或“，”；“•”读作连接。

  2)仅由字母表$A=\{a_i \mid i=1,2,\ldots\ldots n\}$上的正规式α所组成的语言称作正规集，记作L(α)。

  3)利用正规集相同，可用来证明相应正规式等价。

第一节　正规文法和有限自动机
一、正规文法、正规集与正规式

4、三个概念间关系

- 一个正规语言可以用正规文法定义，也可以用正规式定义，对任意一个正规文法，存在一个定义同一个语言的正规式；同样，对每个正规式，存在一个生成同一语言的正规文法；有些正规语言很容易用文法定义，有些则用正规式定义更容易；两者之间是可以转换的，结构上具有等价性。
- 由正规文法或正规式定义的正规语言的集合构成正规集。

第一节 正规文法和有限自动机
一、正规文法、正规集与正规式

- 例：证明$b(ab)^*=(ba)^*b$
- 证明：$\because L(b(ab)^*)=\{b,bab,babab,\ldots\ldots\}$

  $L((ba)^*b)=\{b,bab,babab,\ldots\ldots\}$

  又$\because$正规集的前n项相同

  $\therefore$可知它们的正规集是相等的

  故：正规式 $b(ab)^*=(ba)^*b$

第一节　正规文法和有限自动机

一、正规文法、正规集与正规式

4、定理1：

- 若α、β、γ是正规式则下述等价式成立
  - 1. $\alpha+\beta=\beta+\alpha$
  - 2. $\alpha+(\beta+\gamma)=(\alpha+\beta)+\gamma$　　$\alpha(\beta\gamma)=(\alpha\beta)\gamma$
  - 3. $\alpha(\beta+\gamma)=\alpha\beta+\alpha\gamma$　　$(\alpha+\beta)\gamma=\alpha\gamma+\beta\gamma$
  - 4. $\varepsilon\alpha=\alpha\varepsilon=\alpha$
  - 5. $(\alpha^*)^*=\alpha^*$
  - 6. $\alpha^*=\alpha^+ + \varepsilon$　　$\alpha^+ = \alpha\alpha^* = \alpha^*\alpha$
  - 7. $(\alpha+\beta)^*=(\alpha^*+\beta^*)^*=(\alpha^*\beta^*)^*$

第一节　正规文法和有限自动机

一、正规文法、正规集与正规式

5、定理2：

若α、β、γ是字母表A上的正规式，且$\varepsilon \notin L(\gamma)$，则

$\alpha = \beta | \alpha\gamma$ 当且仅当 $\alpha = \beta\gamma^*$

$\alpha = \beta | \gamma\alpha$ 当且仅当 $\alpha = \gamma^*\beta$

第一节　正规文法和有限自动机

一、正规文法、正规集与正规式

6、正规文法转换成相应正规式

其步骤为：

- 1.由正规文法G的各个产生式写出对应的正规方程式，得到联立方程组。
- 2.把方程组中的非终结符当作变元。
- 3.求此正规式方程组的解，得到关于开始符号S的解：S=w ，$w \in V_T^*$，w就是所求正规式。

第一节　正规文法和有限自动机

一、正规文法、正规集与正规式

例：已知正规文法G1的产生式，求出它所定义的正规式。

- 产生式为：$S \rightarrow aS \mid aB$
- $B \rightarrow bB \mid bA$
- $A \rightarrow cA \mid c$

解：1.由产生式写出对应的联立方程组

$$
\begin{cases}
S = aS \mid aB & \ldots\ldots \quad (1) \\
B = bB \mid bA & \ldots\ldots \quad (2) \\
A = cA \mid c & \ldots\ldots \quad (3)
\end{cases}
$$

- 2.根据定理2，
- 由（1） $S = aS \mid aB$ 得： $S=a^*aB=a^+B$ ……（4）
- 同理，由（2） $B = bB \mid bA$ 得： $B=b^+A$ ……（5）
- 同理，由（3） $A = cA \mid c$ 得： $A=c^*c=c^+$ ……（6）
- 将（6）代入（5）得：$B=b^+c^+$ ……（7）
- 将（7）代入（4）得：$S=a^+b^+c^+$ ……（8）

- 3.故：正规式为$S=a^+b^+c^+$

第一节　正规文法和有限自动机

二、有限自动机(Finite Automation,FA)

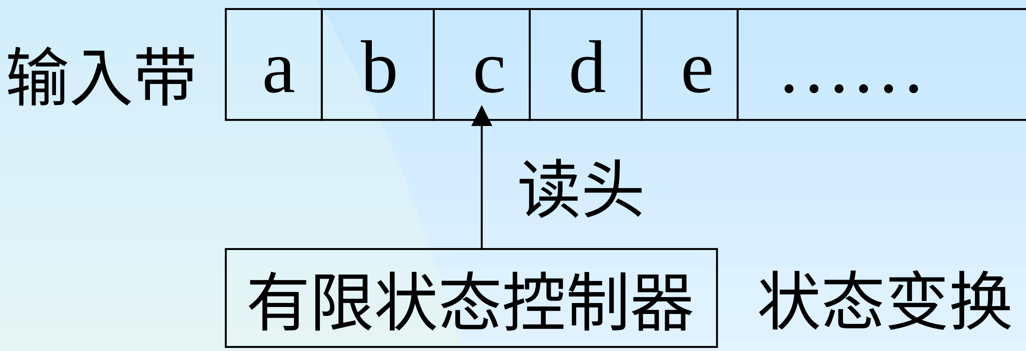
1、有限自动机

- 有限自动机是一种识别装置，它能准确地识别正规集。它为词法分析程序的构造提供了方法和工具。
- 有限自动机是具有离散输入输出系统的数学模型。它具有有限数目的内部状态，系统可以根据当前所处的状态和面临的输入字符决定系统的后继行为。其当前状态概括了过去输入处理的信息。

第一节　正规文法和有限自动机

二、有限自动机

2、有限自动机模型

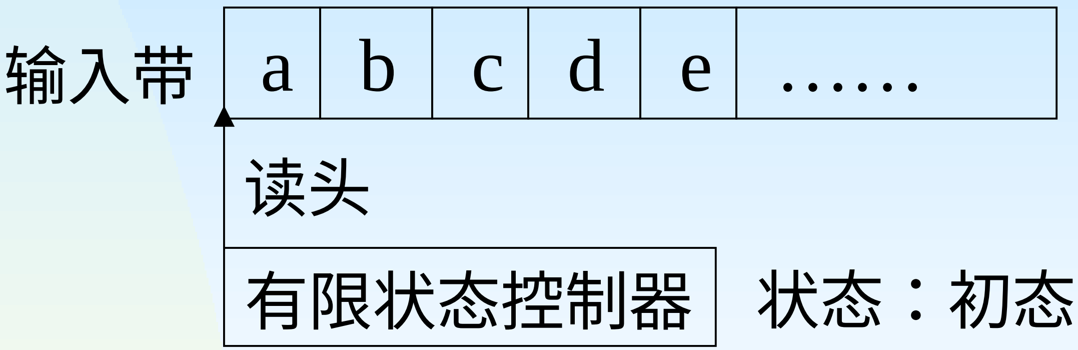


注：状态分为初始状态、中间状态和终止状态。终止状态可以有若干个，而初始状态一般只有一个。

第一节　正规文法和有限自动机

二、有限自动机

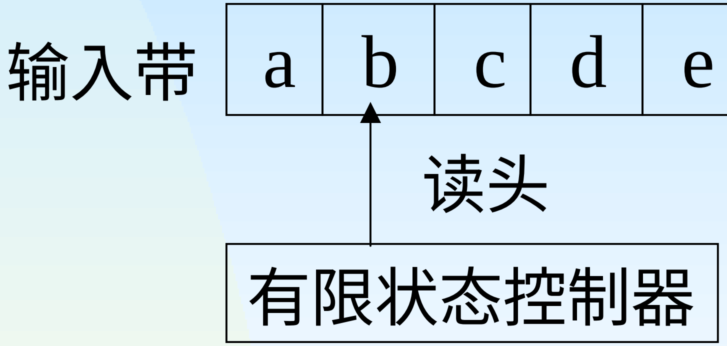
2、有限自动机模型



第一节　正规文法和有限自动机

二、有限自动机

2、有限自动机模型

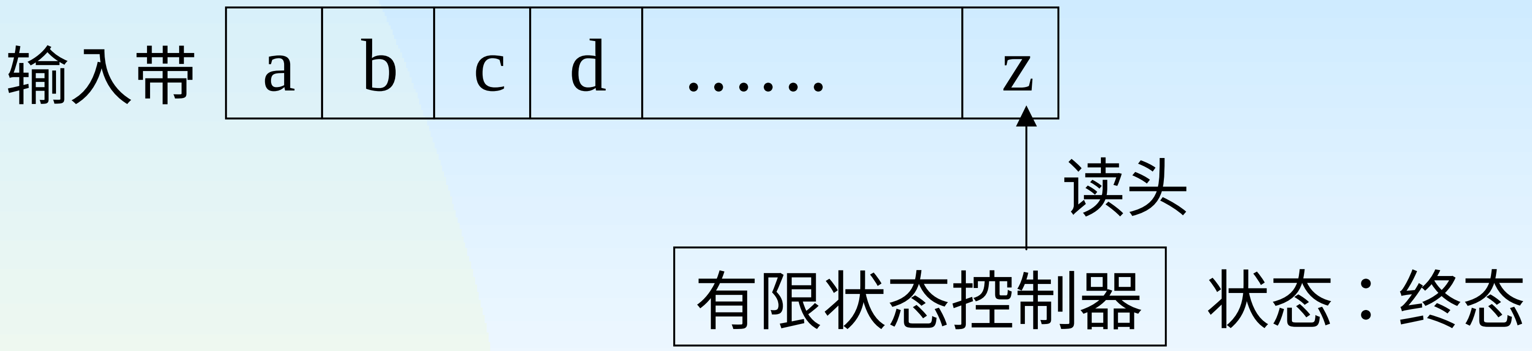


状态：中间状态

第一节　正规文法和有限自动机

二、有限自动机

2、有限自动机模型

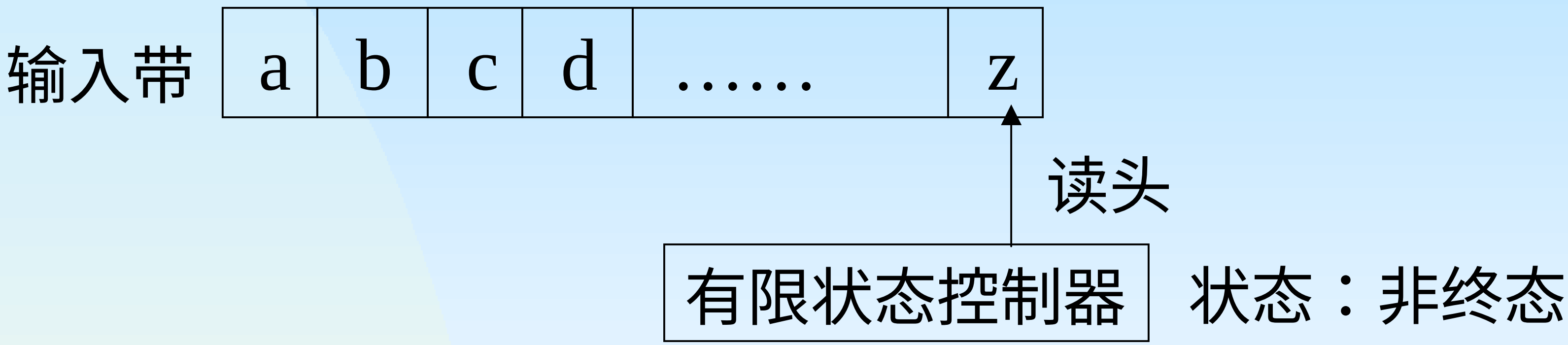


读头全部读完，且此时状态为终止状态，则说明此输入串正确。

第一节　正规文法和有限自动机

二、有限自动机

2、有限自动机模型



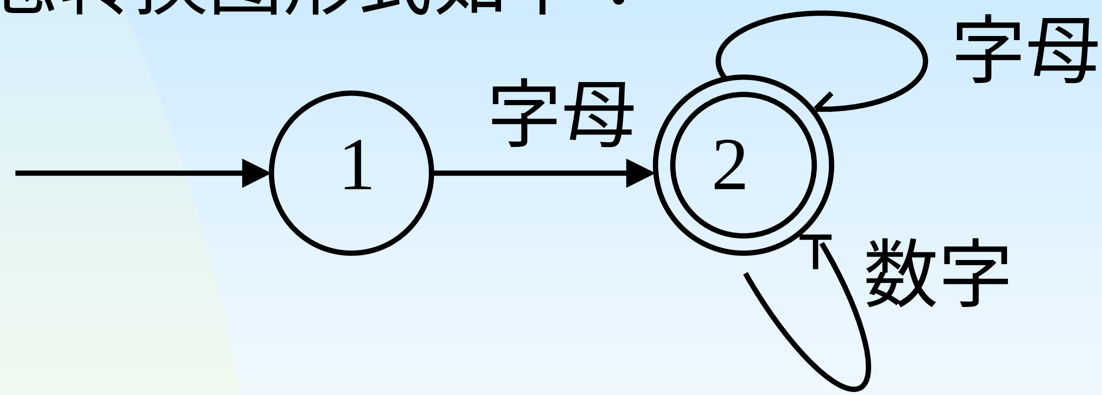
读头全部读完，而此时状态不是终止状态，则说明此输入串错误。

注：可用状态转换图表示状态变换，状态用结点表示，读入符号用边表示。

第一节　正规文法和有限自动机

二、有限自动机

- 例：正规式<标识符>＝<字母>(<字母>|<数字>)* 的状态转换图形式如下：

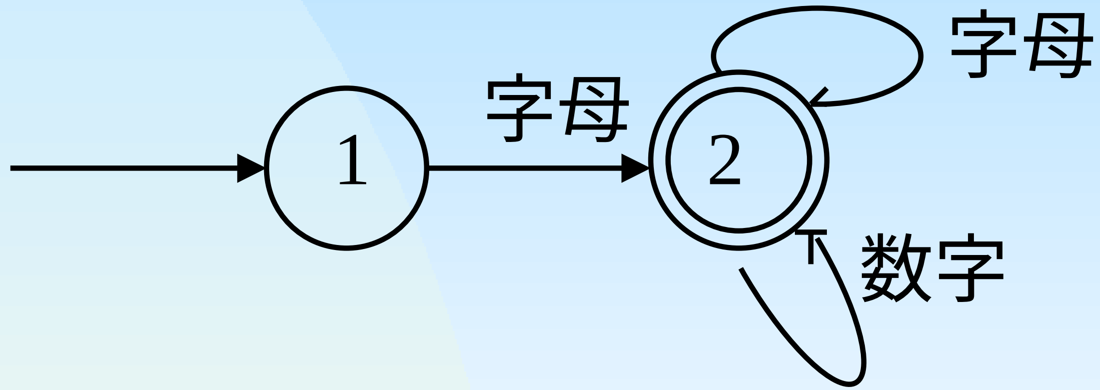


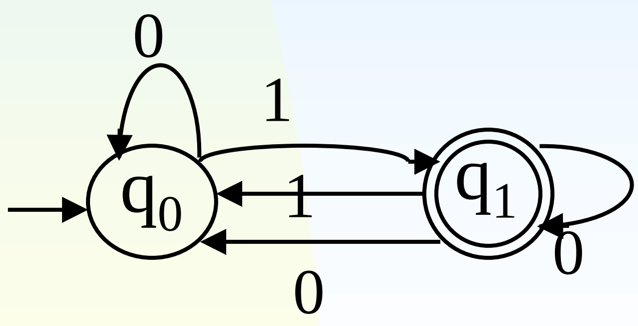
- 程序中标识符xtemp的识别匹配过程为：

$$- \longrightarrow 1 \xrightarrow{x} 2 \xrightarrow{t} 2 \xrightarrow{e} 2 \xrightarrow{m} 2 \xrightarrow{p} 2$$

第一节 正规文法和有限自动机

二、有限自动机





第一节　正规文法和有限自动机
二、有限自动机

3、确定有限自动机DFA(Deterministic FA)

- (1)定义：确定有限自动机是一个五元式 $M(S,\Sigma,f,s_0,Z)$
  - 其中：S：有限状态集
  - Σ ：有限字母表
  - f：$S\times\Sigma\rightarrow S$上的单值映射，$f(s,a)=s'$
  - s_0：唯一的初态，$s_0\in S$
  - Z：终止状态集，$Z\subseteq S$
- 注：这里确定的含义，就是状态转换关系f是一个函数，即对于给定的当前状态s和当前读入的符号a，有唯一确定的下一状态s'。

第一节 正规文法和有限自动机
二、有限自动机

3、确定有限自动机

(2)状态转换关系表示：

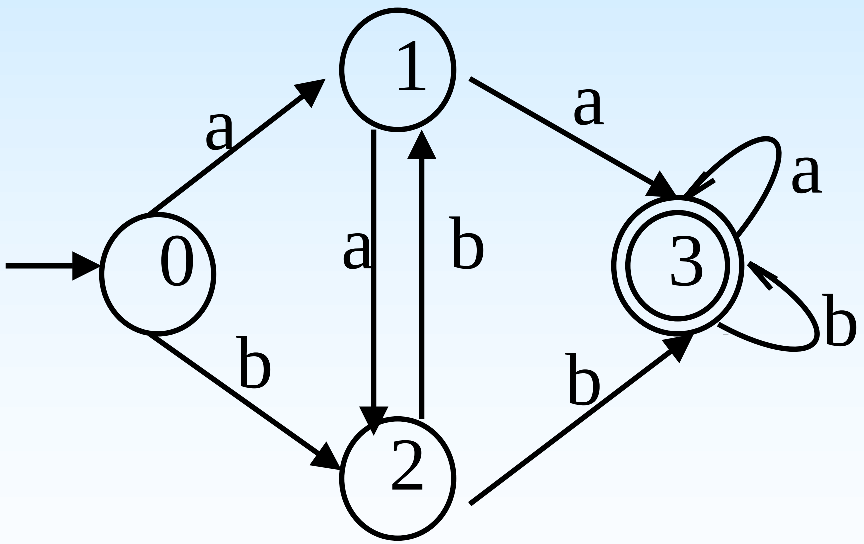
- 状态转换矩阵：DFA的映射关系由一个矩阵来表示。
- 状态转换图：

- 注：1）用矩阵表示转换便于计算机处理，但不直观，而用状态转换图表示比较直观。

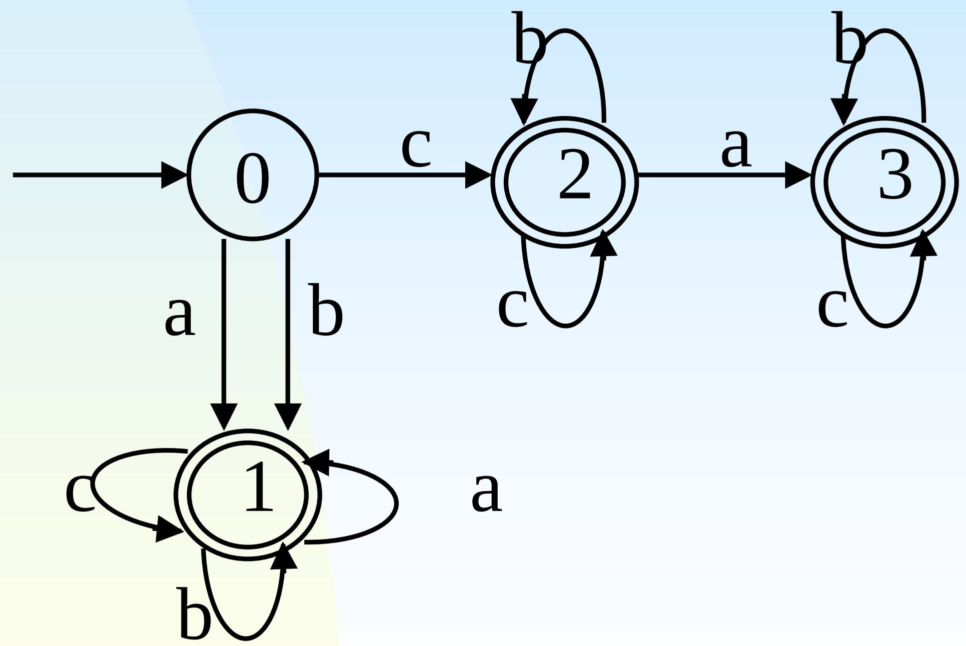
  2）在整个状态转换图中只有一个初始状态结点，用“→”射入的结点表示初始状态。可有若干终止状态(也可能没有)，用双圆圈表示。若初始状态结点同时又是终止状态结点，则表示空串ε可为相应DFA识别。

- 例：DFA M=({0,1,2,3},{a,b},f,0,{3})
  - f: f(0,a)=1 f(0,b)=2 f(1,a)＝3 f(1,b)＝2
  - f(2,a)=1 f(2,b)=3 f(3,a)＝3 f(3,b)＝3
- 状态转换矩阵

输入 / 状态	a	b
0	1	2
1	3	2
2	1	3
3	3	3



- 例：构造一个DFA M，它接受字母表{a,b,c}上，以a或b开始的字符串，或以c开始但所含的a不多于一个的字符串。



故：DFA: M=({0,1,2,3,},{a,b,c},f,0,{1,2,3})

- 其中：f： f(0,a)=1 f(0,b)=1
- f(0,c)=1 f(1,a)=1
- f(1,b)=1 f(1,c)=1
- f(2,a)=3 f(2,b)=2
- f(2,c)=2 f(3,b)=3
- f(3,c)=3

第一节　正规文法和有限自动机

二、有限自动机

3、 确定有限自动机

(3)一步动作

– 每读一个字符，状态就向前进至下一状态；记为：“$\vdash$”

– $\vdash^{K}$ 表示自动机做了K步动作。

– $\vdash^{*}$表示自动机做了0步动作或0步以上动作。

– $\vdash^{+}$表示自动机做了1步动作或1步以上动作。

(4) DFA对字符串的识别

• 定义：串$\alpha\in\Sigma^*$为 DFA $M=(S, \Sigma,f,s_0,Z)$ 所识别，当且仅当$(s_0, \alpha) \vdash^{*} (s,\varepsilon)$,且$s \in Z$

第一节　正规文法和有限自动机

二、有限自动机

3、确定有限自动机

(4) DFA对字符串的识别

- 注：能被DFA　M所接受的字符串的集合，称为自动机M所能识别的语言，记为L(M)。

  不能被自动机接受的字符串有两种情况：

  – 读完输入串，状态不停在终止状态，即： $(s_0,\alpha) \vdash^* (s',\varepsilon)$,且$s' \notin Z$

  – 在读过程中出现不存在的映射，使自动机无法继续动作。

第一节　正规文法和有限自动机

二、有限自动机

4、不确定有限自动机NFA(Non-deterministic FA)

(1)定义：不确定有限自动机是一个五元式

$$M=(S,\Sigma,f,S_0,Z)$$

- 其中：S：有限状态集
- Σ ：有限字母表
- f： $S\times\Sigma\rightarrow 2^S$(S的子集)上的映射
- S_0：非空的初态集，S_0 是S的真子集
- Z：终止状态集，Z是S的子集，可为空集

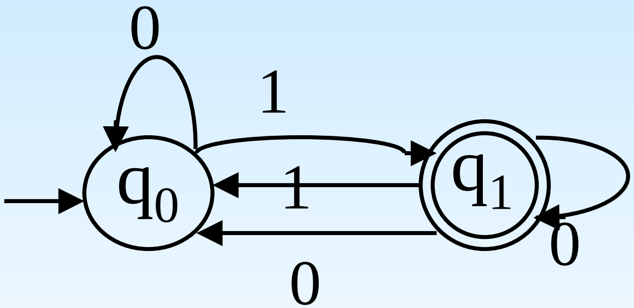
注：1)非确定主要是指后继状态可有多个。

2) DFA是NFA的特例。

例：设NFA M＝$(\{q_0,q_1\},\{0,1\},f,\{q_0\}\{q_1\})$

f映射为

字符 / 状态	0	1
q_0	q_0	q_1
q_1	q_0，q_1	q_0



第一节　正规文法和有限自动机
二、有限自动机

4、不确定有限自动机

(2)两自动机等价：

- 任何两个有限自动机M和M’，若它们识别的语言相同(L(M)=L(M’))，则称M和M’等价。
- 注：存在判定任何两个有限自动机等价性的算法。

5、NFA确定化

(1)定理

对于每个NFA　M，存在一个DFA M’，使得L(M)=L(M’)。即，设L是一NFA接受的正规集，则存在一个DFA接受L。

第一节　正规文法和有限自动机
二、有限自动机

5、NFA确定化

(2)算法

- 由NFA M=(S,Σ,f,S_0,Z)构造一个等价的DFA M'=(Q,Σ,δ,I_0,F)

  1.取$I_0=S_0$,

  2.若状态集Q中有状态$I_i=\{s_0,s_1,\ldots\ldots s_j\}$，$s_k\in S$，$0\le k\le j$;而且M机中有$f(\{s_0,s_1,\ldots\ldots s_j\},a)$

$$= f(s_0,a)\cup f(s_1,a)\cup\ldots\ldots\cup f(s_j,a)=\bigcup_{k=0}^{j} f(s_k,a)$$

$$=\{s_0,s_1,\ldots\ldots s_t\}=I_t,$$

若I_t不在Q中，则将I_t加入Q。

第一节　正规文法和有限自动机
二、有限自动机

5、NFA确定化

(2)算法

3.重复步骤2，直到Q中无新状态加入为止。

4.取终态F={I | I ∈ Q,且I ∩ Z <>Φ}

注：1)上述过程可在有限步内完成，因为M机状态的幂集是有限的；

2)上述过程也可用表格法来描述，其中列是字符集Σ中的字符；行是Q中的各状态，开始仅包含I_0状态，随着算法的执行，Q的状态逐渐增多直止不再增多为止；表格元素就是δ映射函数。

第一节　正规文法和有限自动机

二、有限自动机

5、NFA确定化

(2)算法

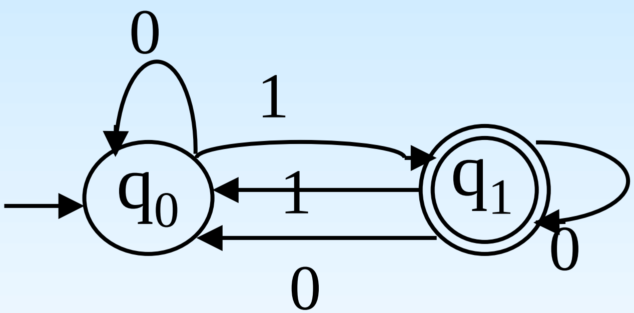
注：3)NFA确定化的实质是以原有状态集上的覆盖片(COVER)作为DFA上的一个状态，将原状态间的转换改为覆盖片间的转换，从而将不确定问题确定化。

4)通常，经确定化后，状态数增加，而且可能出现一些等价状态，这时需要化简。

例：设NFA M＝($\{q_0,q_1\},\{0,1\},f,\{q_0\}\{q_1\}$)

f映射为

字符 / 状态	0	1
q_0	q_0	q_1
q_1	q_0， q_1	q_0



将其确定化。

解：

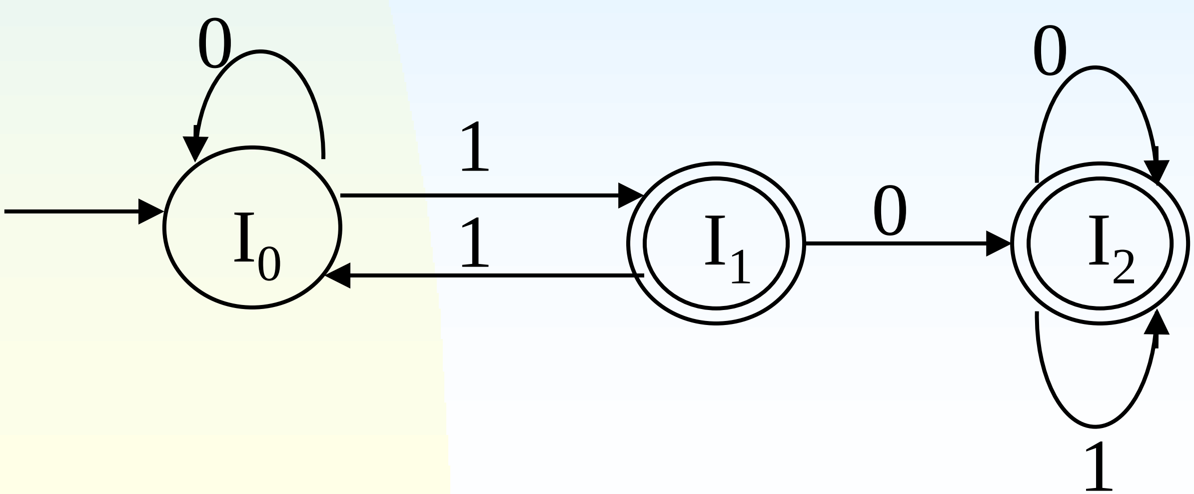
- 1.M'的初态：$I_0=\{q_0\}$。
  则Q中就有了I_0状态。
- 2.由Q中的状态$I_0=\{q_0\}$，查看M机，
  有： $f(\{q_0\},0)=\{q_0\}$、 $f(\{q_0\},1)=\{q_1\}=I_1$
  此时，$Q=\{I_0,I_1\}$
- 3.由Q中的状态$I_1=\{q_1\}$，查看M机，
  有：$f(\{q_1\},0)=\{q_0,q_1\}=I_2$、 $f(\{q_1\},1)=\{q_0\}=I_0$
  此时，$Q=\{I_0,I_1,I_2\}$
- 4.由Q中的状态$I_2=\{q_0,q_1\}$，查看M机，
  有：$f(\{q_0,q_1\},0)=\{q_0,q_1\}$、 $f(\{q_0,q_1\},1)=\{q_0,q_1\}=I_2$
  此时，$Q=\{I_0,I_1,I_2\}$
- 5.$F=\{I_1,I_2\}$

- NFA经过确定化后，变为：
  - DFA　M'=($\{I_0, I_1, I_2\}, \{0,1\}, \Sigma, I_0, \{I_1, I_2\}$)

Σ \ Q	0	1
$I_0=\{q_0\}$	$I_0=\{q_0\}$	$I_1=\{q_1\}$
$I_1=\{q_1\}$	$I_2=\{q_0,q_1\}$	$I_0=\{q_0\}$
$I_2=\{q_0,q_1\}$	$I_2=\{q_0,q_1\}$	$I_2=\{q_0,q_1\}$

δ	0	1
I_0	I_0	I_1
I_1	I_2	I_0
I_2	I_2	I_2

状态转换图如下：



第一节　正规文法和有限自动机

二、有限自动机

6、确定有限自动机的化简（最小化）

(1)化简条件：接受的语言必须相同

(2)化简(最小化)算法基本思想——划分法

- 1.将DFA M 中的状态划分为互不相交的子集，每个子集内部的状态都等价；而在不同子集间的状态均不等价。
- 2.从每个子集中任选一个状态作为代表，消去其它等价状态。
- 3.把那些原来射入其它等价状态的弧改为射入相应的代表状态。

第一节　正规文法和有限自动机
二、有限自动机

6、确定有限自动机的化简

(3) 状态等价：设DFA M中有两个状态s,t，

- s,t 等价：
  - $(s,w) \vdash^* (s_1,\varepsilon)$同时$(t,w) \vdash^* (t_1, \varepsilon)$，$s_1,t_1$都是终态，$w \in V_T^*$，即如果从状态s出发能读出某个字w而停于终态，从t出发也能读出同样的字w而停于终态，则称s,t 等价。
- s,t可区别：
  - 如果s,t不等价，则称为s,t可区别

第一节　正规文法和有限自动机

二、有限自动机

6、确定有限自动机的化简

(4)化简(最小化)算法

1.把状态集S划分为终态集和非终态集：$\Pi_0=\{I_0^1, I_0^2\}$，I_0^1属于非终态集，I_0^2属于终态集。因为终态能识别ε，而非终态不能，所以它们是可区别的；

2.假定经过k次划分后：$\Pi_k=\{I_k^0, I_k^1 \ldots\ldots I_k^m\}$.这m个子集之间可区分，子集内部还是否可以划分？

- 任取一个子集$I_k^i=\{s_1, s_2, \ldots\ldots s_k\}$，若存在某读入字符a，使$f(I_k^i, a)$的结果不是全部包含在$\Pi_k$的某个子集中，则说明$I_k^i$中有不等价的状态，还要进一步划分。
- 对Π_k中所有子集都进行测试，以完成一次划分。

第一节　正规文法和有限自动机
二、有限自动机

6、确定有限自动机的化简

(4)化简(最小化)算法

3.重复步骤2，直到所含的子集数不再增加为止。

4.对每个子集任取一状态为代表。若该子集包含原有的初态，则相应代表状态就是最小化后M的初态；同样，若该子集包含原有的终态，则相应代表状态就是最小化后M的终态。

第一节　正规文法和有限自动机

二、有限自动机

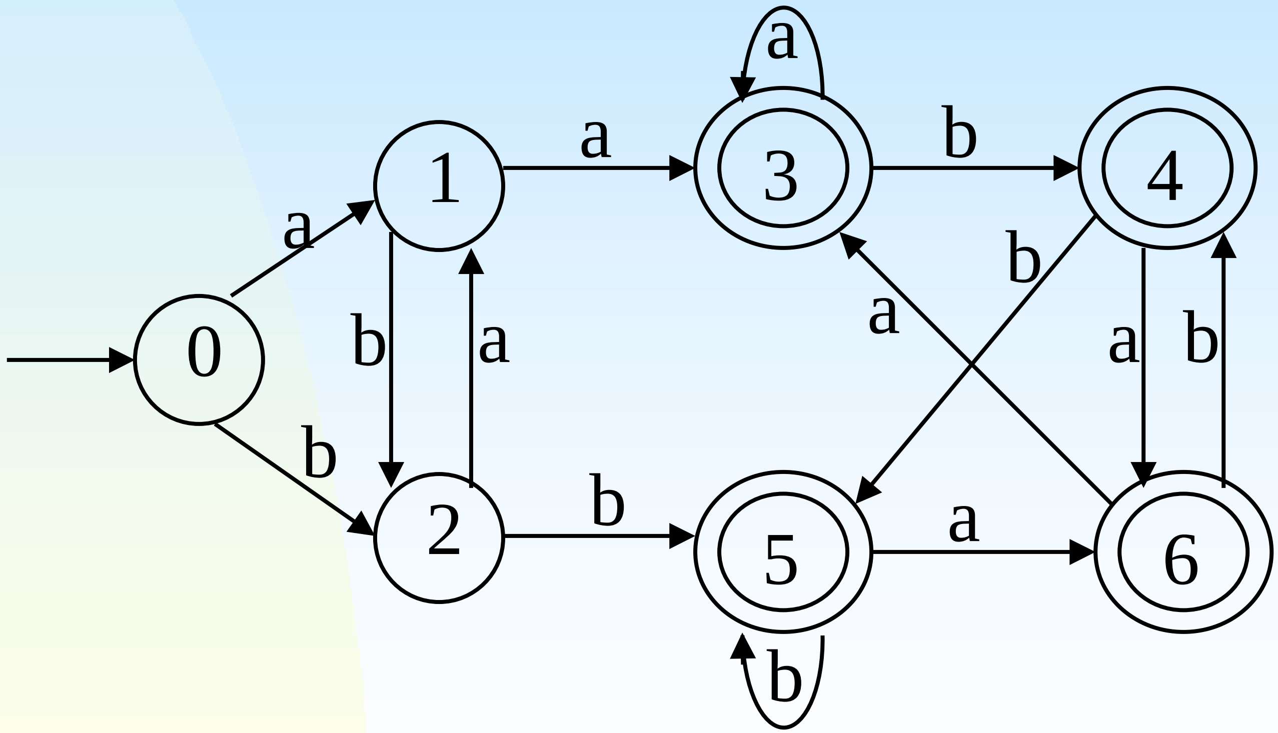
6、确定有限自动机的化简

(4)化简(最小化)算法

注：1）当一个自动机没有任何多余的状态，并且它的状态中没有两个是互相等价的时，我们说这个有限自动机是化简了的。

2）可以通过消除多余状态，合并等价状态而转化成一个最小化的与之等价的有限自动机。

• 例：设有一DFA 的状态转换图如下，试化简之。



解：

- 1.把M的状态分为两组：终态集，非终态集
  - Π_0＝{{0,1,2},{3,4,5,6}}
- 2.1考察非终态集：

  f({0,1,2},a)={1,3} 不属于Π_0的任何一个子集，所以{0，1，2}要分开
  - 得到：Π_1`＝{{1}，{0,2}，{3,4,5,6}}

  再看：f({0,2},a)={1}属于Π_1'的子集
  - f({0,2},b)={2,5}不属于Π_1'的任何子集，所以{0,2}要分开
  - 得到：Π_1``＝{{1}，{0}，{2}，{3,4,5,6}}

解：（续）

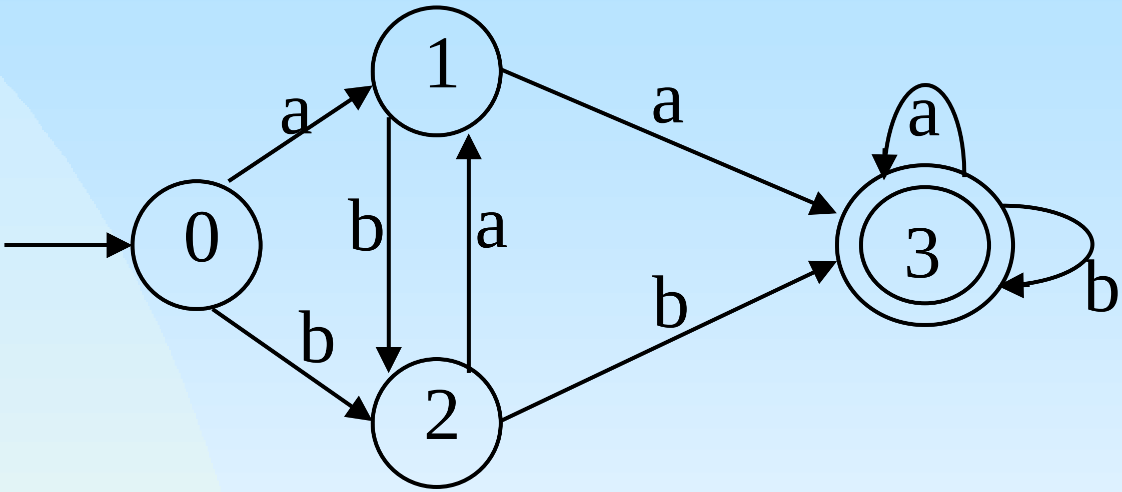
2.2考察终态集：

- $f(\{3,4,5,6\},a)=\{3,6\}$包含于Π_1“的子集$\{3,4,5,6\}$
- $f(\{3,4,5,6\},b)=\{4,5\}$包含于Π_1“的子集$\{3,4,5,6\}$
- 所以$\{3,4,5,6\}$不可再划分

3.整个划分为4个组：

- $\Pi_2=\{\{1\},\{0\},\{2\},\{3,4,5,6\}\}$

4.令状态3代表{3,4,5,6}，把原来到达状态4，5，6的弧都导入3，并删除4，5，6。得：



即为化简了的DFA

第一节　正规文法和有限自动机
三、正规式与有限自动机之间的关系

1、关系定理

定理：Σ上的NFA　M所能识别的语言L(M)可以用Σ上的正规式来表示。即：对Σ上的NFA M，可构造一个正规式α，使得L(α)=L(M)

定理：Σ上任何正规式α，存在DFA　M使得L(M)=L(α)。即：由 正规式α可以构造一个DFA M，使得L(M)＝L(α)

第一节　正规文法和有限自动机

三、正规式与有限自动机之间的关系

2、有限自动机M向正规式α的转换

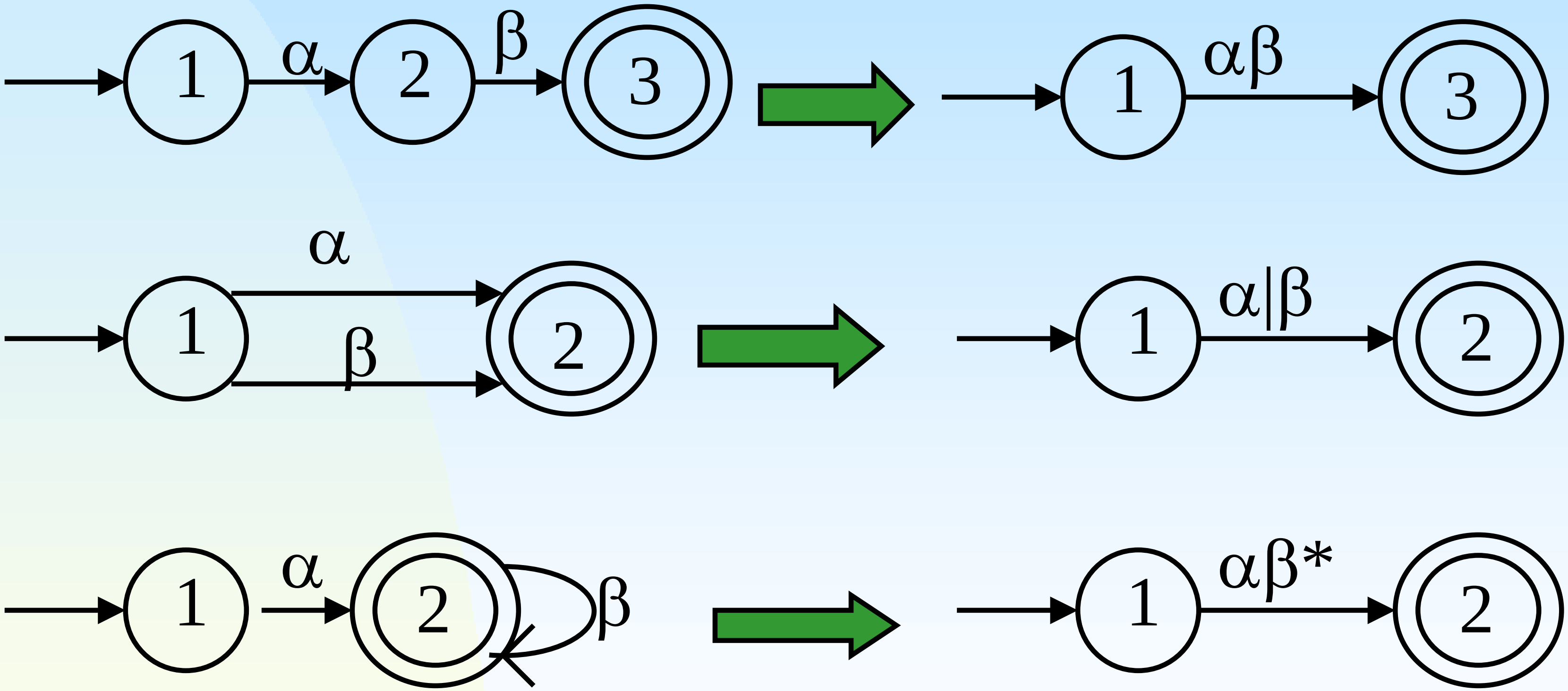
1)把状态转换图的概念拓广，令每条弧上都可以用一个正规式作标记。

2)在M的转换图上加两个结点：x,y。从 x用ε弧连接到M的所有初态结点；从M的终态结点用ε 弧连接到y。这个新的NFA为M’，且L(M)=L(M’)

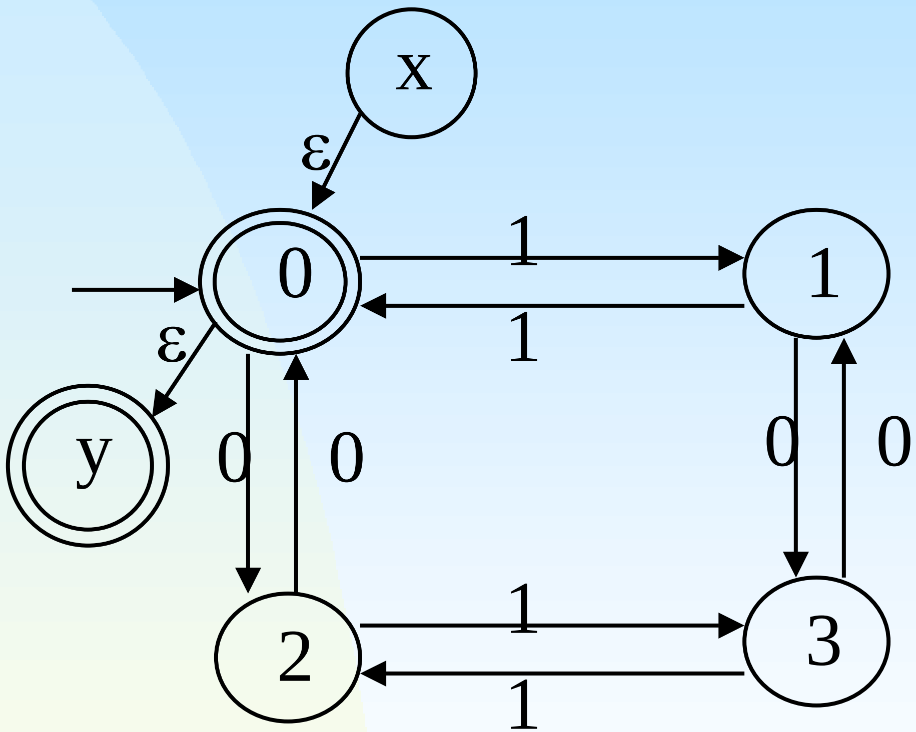
3)通过引入的3条有限自动机替换规则逐步消去M’中的所有结点，直到只剩下x和y为止。这样，在x至y的弧线上的标记就是Σ上的正规式，也就是M接受的正规式。

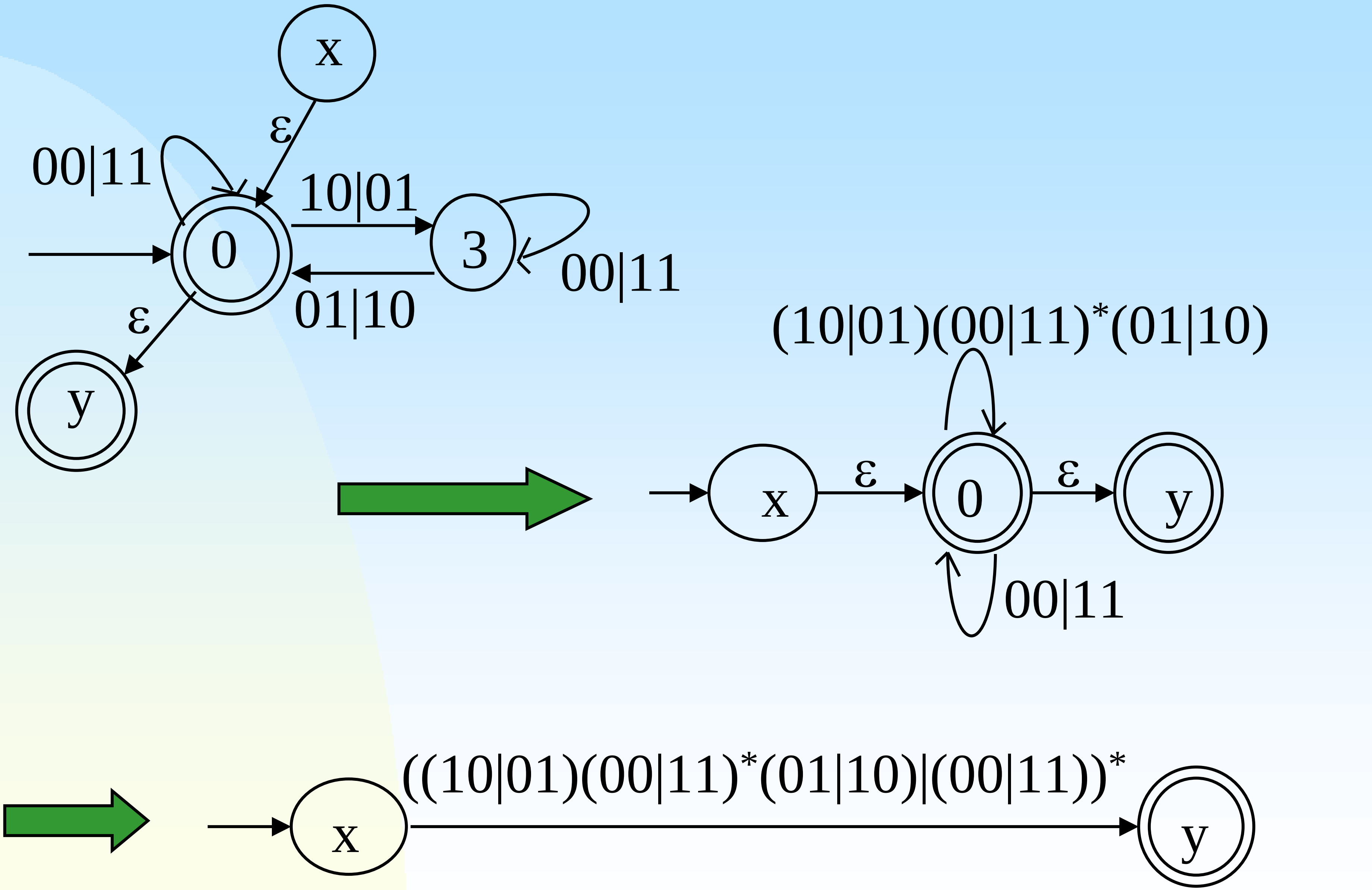
注：在消除结点过程中，逐步用正规式来标记弧。

有限自动机替换规则



• 例：将下面的DFA M所接受的语言表示为正规式



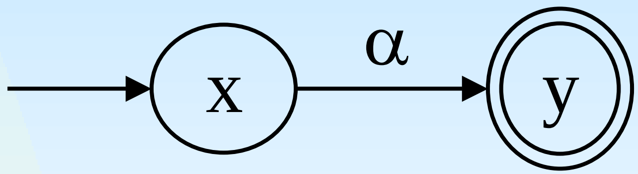

x
ε
00|11
10|01
0
3
00|11
01|10
ε
y
(10|01)(00|11)*(01|10)
x
ε
0
ε
y
00|11
((10|01)(00|11)*(01|10)|(00|11))*
x
y

第一节　正规文法和有限自动机

三、正规式与有限自动机之间的关系

3、正规式α向确定有限自动机M的转换

1)由正规式α 构造一个如下仅有两个结点x,y的状态图。

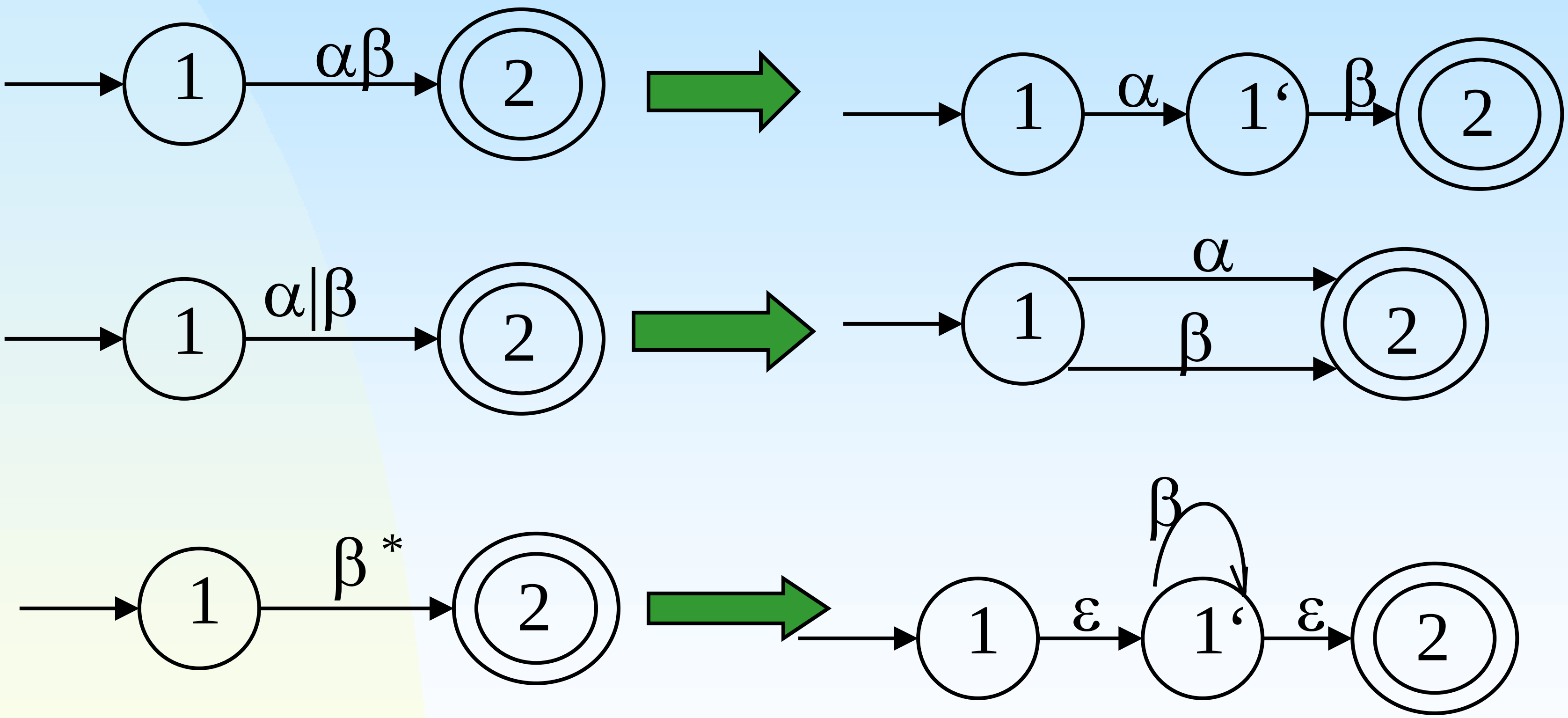


2)按所引入的3条正规式分裂规则分裂α 。

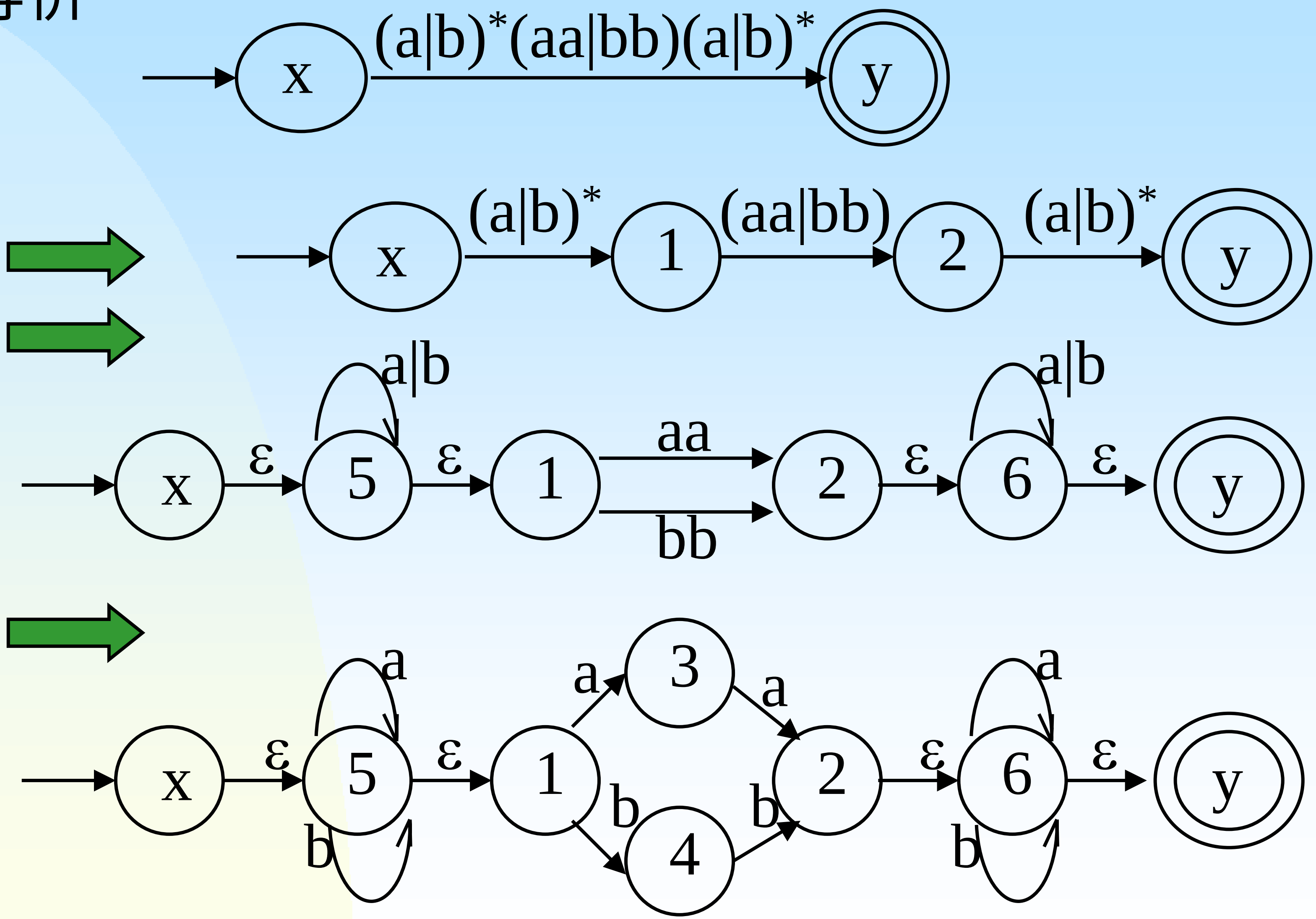
3)重复步骤2直到每个弧上的标记是Σ上的一个字符或ε为止。

4)将所得的NFA M(因为包含ε弧)进行确定化就得到DFA。

正规式分裂规则



例：根据正规式$(a|b)^*(aa|bb)(a|b)^*$，构造DFA M，使之等价



第一节　正规文法和有限自动机

三、正规式与有限自动机之间的关系

3、正规式α向确定有限自动机M的转换

注：这里将NFA M进行确定化与前面所讲的子集法确定化是一回事。不过，这里的NFA M中含有ε弧，所以在求覆盖片时应考虑ε弧。方法是求ε闭包(ε-closure)，将此闭包(状态子集)作为DFA的一个状态使用，而将NFA上的状态间转换变为闭包间的转换，使得不确定的自动机确定化。

第一节　正规文法和有限自动机

三、正规式与有限自动机之间的关系

4、对含有ε弧的NFA进行确定化

(1) ε闭包

是可以从某状态或某些状态通过ε弧所能到达的所有状态的集合。

状态集合I的ε闭包(ε-closure(I))形式定义如下：

(a)若s ∈ I，则s ∈ε-closure(I)

(b)若s ∈ I，那么从s出发经过任意段的ε弧所能达到的任意状态s’都属于ε-closure(I)

第一节　正规文法和有限自动机
三、正规式与有限自动机之间的关系

4、对含有ε弧的NFA进行确定化

(2) 闭包间转换

- 设ε-closure(I)＝{q_0,q_1,……q_n}，当读入字母表中字母a时，它转换到另一闭包ε -closure(J)。
- ε -closure(J)的组成
  - $J=f(q_0,a) \cup f(q_1,a) \cup \ldots\ldots \cup f(q_n,a)$
  - 对得到的J按ε闭包的定义求ε-closure(J)

第一节　正规文法和有限自动机

三、正规式与有限自动机之间的关系

4、对含有ε弧的NFA进行确定化

(3)对含有ε弧的NFA进行确定化方法

设由NFA $M=(S,\Sigma,f,S_0,Z)$构造一个等价的DFA $M'=(Q,\Sigma,\delta,I_0,F)$

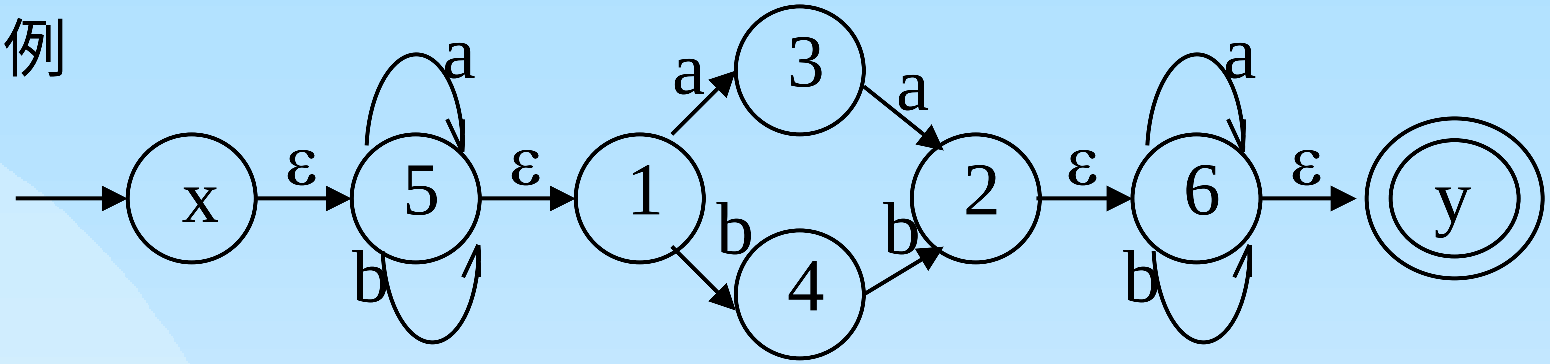
(a) $I_0 = \varepsilon\text{-closure}(S_0)$, $I_0 \in Q$

(b)若状态集Q中有状态$I_i=\{s_0,s_1,\ldots\ldots s_j\}$, $s_k \in S$, $0 \le k \le j$;且有$I_t= \varepsilon\text{-closure}(f(I_i,a))$,若$I_t$不在Q中，则将$I_t$加入Q。

(c)重复步骤2，直到Q中无新状态加入。

(d)取终态$F=\{I \mid I \in Q$,且$I \cap Z <> \Phi\}$

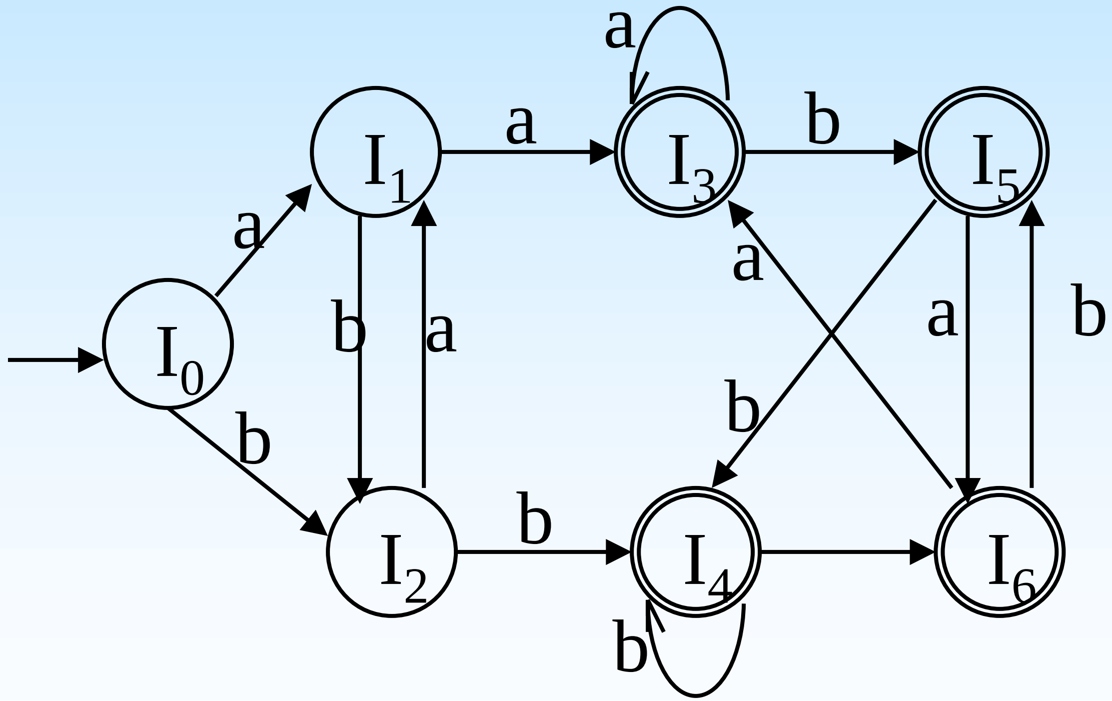
例



I	a	b
I_0={x,5,1}	I_1={5,3,1}	I_2={5,4,1}
I_1={5,3,1}	I_3={5,3,2,1,6,y}	I_2={5,4,1}
I_2={5,4,1}	I_1={5,3,1}	I_4={5,4,1,2,6,y}
I_3={5,3,2,1,6,y}	I_3={5,3,2,1,6,y}	I_5={5,1,4,6,y}
I_4={5,4,1,2,6,y}	I_6={5,3,1,6,y}	I_4={5,4,1,2,6,y}
I_5={5,1,4,6,y}	I_6={5,3,1,6,y}	I_4={5,4,1,2,6,y}
I_6={5,3,1,6,y}	I_3={5,3,2,1,6,y}	I_5={5,1,4,6,y}

I	a	b
I_0	I_1	I_2
I_1	I_3	I_2
I_2	I_1	I_4
I_3	I_3	I_5
I_4	I_6	I_4
I_5	I_6	I_4
I_6	I_3	I_5

DFA为：



第一节　正规文法和有限自动机

四、正规文法和有限自动机

1、关系定理

- 设$G=(V_N,V_T,P,S)$是正规文法，则存在一个有限自动机$M=(Q,\Sigma,f,q_0,Z)$使得$L(G)=L(M)$。
- 注：1)正规文法分为右线性文法和左线性文法。但对一个正规文法，不能既是左线性，又是右线性。

  2)对每个有限自动机 M，都存在一个右线性正规文法G_R和左线性正规文法G_L，使得$L(M)=L(G_R)=L(G_L)$

四、正规文法和有限自动机

2、右线性文法转换为等价自动机

- 设有右线性文法：$G=(V_N,V_T,P,S)$，将其转换为自动机$M=(Q, \Sigma,f,q_0,Z)$。转换步骤如下：
- 1)将V_N中的每个非终结符视为状态符号，并增加一个新的终结状态符号T，即令$Q=V_N \cup \{T\}$;同时，令$\Sigma = V_T$，$q_0 = S$;若P中含有$S \to \varepsilon$，则令$Z=\{S,T\}$，否则令 $Z=\{T\}$；

第一节　正规文法和有限自动机

四、正规文法和有限自动机

2、右线性文法转换为有限自动机

2)P中的产生式用如下映射f来代替。

a)对于P中每一条形如 $A_1 \to aA_2$的产生式，在M中设为 $f(A_1,a)= A_2$.

b)对于P中每一条形如 $A_1 \to a$的产生式，在M中设为 $f(A_1,a)= T$.

c)对Σ上的所有a，取 $f(T,a)=\Phi$,即在终态下有限自动机无动作。

例：有文法$G=(\{S,A,B\},\{a,b,c\},P,S)$，其中产生式P：

- $S \rightarrow aS \mid aB$
- $B \rightarrow bB \mid bA$
- $A \rightarrow cA \mid c$

构造与之等价的FA。

解：构造自动机$M=(Q,\Sigma,f,q_0,Z)$

1)增加一个新的终结状态符号T，$Q=\{S,B,A,T\}$

$\Sigma = \{a,b,c\}$， $q_0 = S$， $Z=\{T\}$

2)　f：

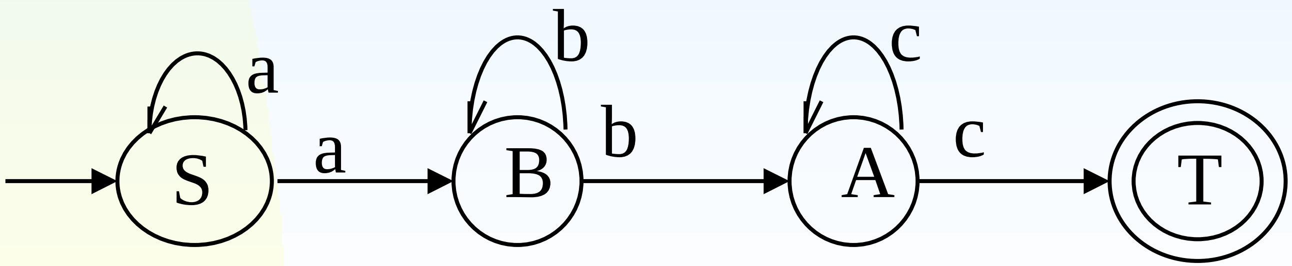
$f(S,a)=S$　$f(S,a)=B$

$f(B,b)=B$　$f(B,b)=A$

$f(A,c)=A$　$f(A,c)=T$

显然，这是一个NFA。

其状态转换图为：



第一节　正规文法和有限自动机

四、正规文法和有限自动机

3、有限自动机向右线性文法的转换

- 设有限自动机$M=(S,\Sigma,f,s_0,Z)$,右线性文法$Rg=(V_N,V_T,P,s_0)$,

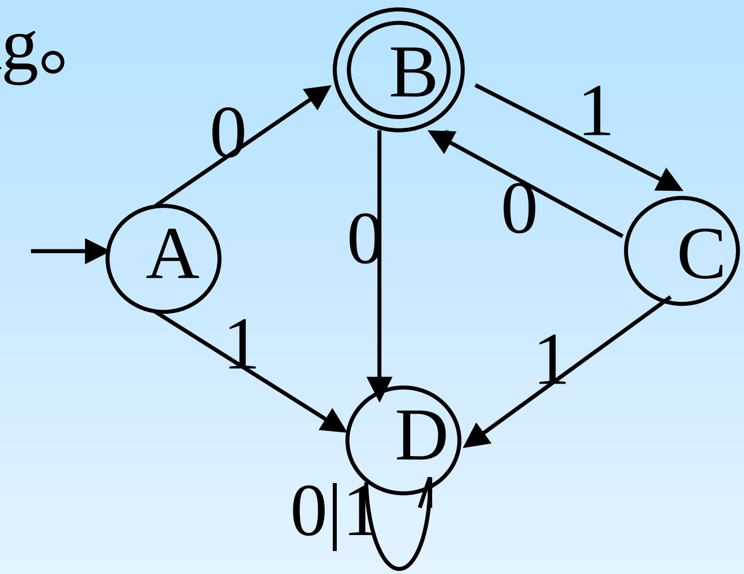
1)若$s_0 \notin Z$，则P是由以下规则定义的产生式集合：

a)若M中有映射$f(A_i,a)=A_j$,则P中有$A_i \to aA_j$;

b)若$A_j \in Z$,则P中增加产生式$A_i \to a$，即$A_i \to a|aA_j$;

2)若$s_0 \in Z$,除了上述映射所构成产生式之外，还有映射$f(s_0,\varepsilon)=s_0$,此时需要在P中增加产生式：$s_0' \to \varepsilon\ |s_0$，以$s_0'$代替$s_0$作为开始符号。

例：写出DFA M=({A,B,C,D},{0,1},f,A,{B})相应的右线性文法Rg。



解：Rg=({A,B,C,D},{0,1},P,A)

$A \rightarrow 0B \mid 1D \mid 0$

$B \rightarrow 1C \mid 0D$

$C \rightarrow 0B \mid 1D \mid 0$

$D \rightarrow 0D \mid 1D$

$L(Rg)=L(M)=0(10)^{*}$

第一节　正规文法和有限自动机

四、正规文法和有限自动机

4、左线性文法转换为有限自动机

- 设有左线性文法：$G=(V_N,V_T,P,S)$，有限自动机 $M=(Q,\Sigma,f,q_0,Z)$，将G转换为M的转换步骤如下：

1)令$Q=V_N\cup\{q_0\}$, q_0是M中新增的初态；$\Sigma=V_T$ ；S对应于M中的Z；若P中含有$S\rightarrow\varepsilon$，则令$Z=\{S, q_0\}$，否则，令$Z=\{S\}$；

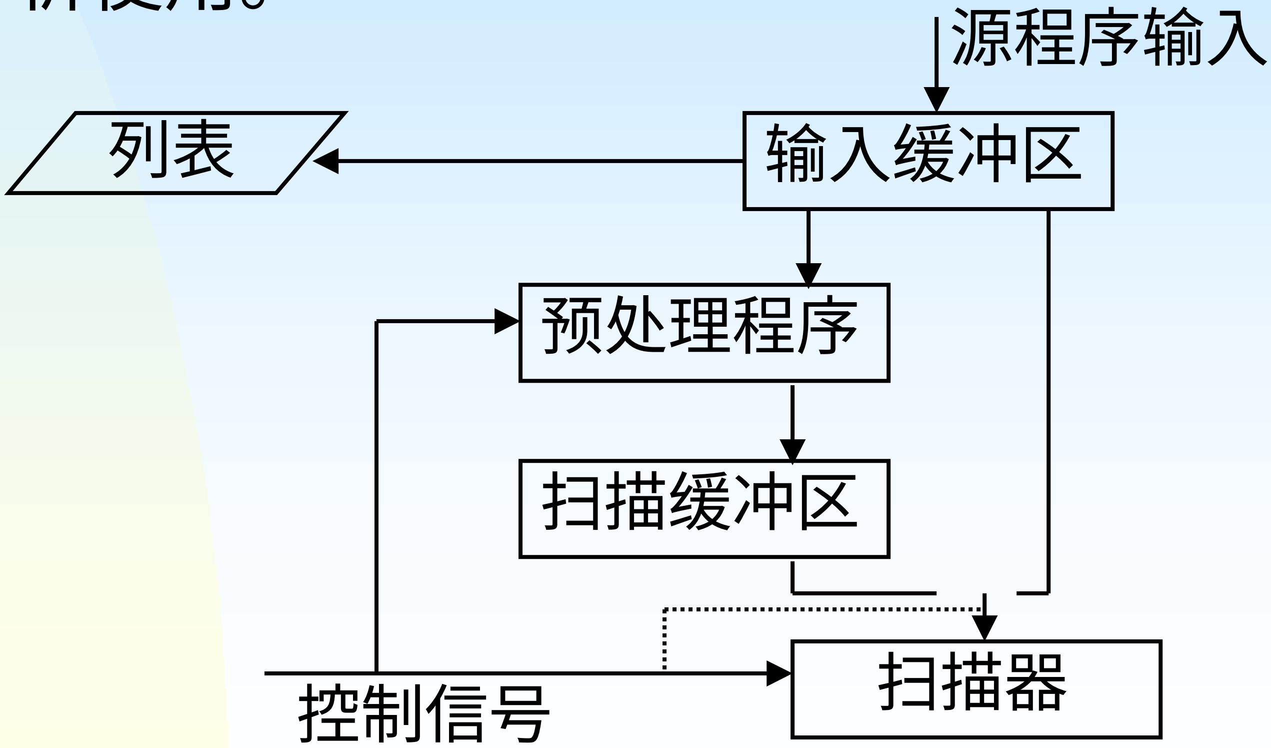
2)P中的产生式用如下映射 f 来代替:

- a)对于P中每一条形如 $A_1\rightarrow A_2a$的产生式，在M中设映射式 $f(A_2,a)= A_1$.
- b)对于P中每一条形如 $A_1\rightarrow a$的产生式，在M中设映射式$f(q_0,a)= A_1$.

第二节　词法分析程序

一、任务

- 从左至右扫描源程序的字符串，按照词法规则识别出一个个正确的单词，并转换为相应的二元式（类号，内码）形式，交给语法分析使用。



第二节　词法分析程序
二、预处理与超前搜索

- 1、预处理原因：

1)源程序中包含注解部分，还有无用的空格、跳格、回车换行等编辑字符，它们与词法分析无关。

2)一行语句结束应配上一个特殊字符说明。

3)有些语言要识别标号区，区分标号语句，找出续行符连接成完整语句等。

4)输出源程序清单以便复核。

第二节　词法分析程序

二、预处理与超前搜索

2、预处理子程序任务：

1)从输入缓冲区中读取源程序，预处理后送入扫描缓冲区。此时，扫描缓冲区中的字符都是有效字符。

2)词法分析程序这时可以再对扫描缓冲区进行扫描。

第二节　词法分析程序

二、预处理与超前搜索

3、超前搜索

注：一般高级语言不必超前搜索，但有些对关键字不加保护的语言，单词间没有明确界符，要在上下文环境中识别单词，这时需要超前搜索。

- 例如：FORTRAN中对“IF”的使用
  - IF (5 .EQ. M) GOTO 50
  - IF=100
  - IF(100)=5

第二节　词法分析程序
三、扫描器的输出格式

1、单词分类（以C语言为例）

- 基本字（关键字、保留字）
- 标识符：变量名、数组名、函数名、过程名……
- 常量
- 运算符
- 界符：.　,　;　(　)　:等。有时把运算符也当作界符。

第二节　词法分析程序
三、扫描器的输出格式

2、扫描器的输出格式

- 使用二元式：(类号，内码)，每个单词对应一个二元式。 其中类号用整数表示，类号既可区分单词种类，又可便于程序处理。类号考虑原则是：

1)每个基本字占有一个类号，内码缺省；

2)各种标识符统一为一类，由内码来区分不同的标识符名。通常将各标识符的符号表入口地址作为其内码。

3)对于常量，以常量的数据类型区分不同类号，对每一类设置相应常量表。各常量在其常量表中的入口地址作为其内码。

4)对于界符，通常一个符号一个类号，内码缺省。

第二节　词法分析程序

四、扫描器的设计

设计方法：

1.写出该语言的词法规则。

2.把词法规则转换为相应的状态转换图。

3.把各转换图的初态连在一起，构成识别该语言的自动机。

4.设计扫描器

– 把扫描器作为语法分析的一个过程，当语法分析需要一个单词时，就调用扫描器。
– 扫描器从初态出发，当识别一个单词后便进入终态，送出二元式。

注意：可用状态矩阵代替状态图，以便于计算机处理。

第三节　词法分析程序的自动生成

- 词法分析程序=状态转换图+控制程序
- 控制程序很简单，关键是构造状态转换矩阵及其相应的语义动作。可根据单词的正规式及其相应的语义动作自动产生词法分析程序。

第三节　词法分析程序的自动生成

一、LEX语言

用来描述词法分析程序的一组单词的正规式及其相应的语义动作，称为LEX语言。

一个LEX源程序主要包括两部分：正规式的辅助定义和识别规则。识别规则又分为正规式和相应语义动作两个部分。

控制程序的基本原则是：最长子串匹配原则和优先原则。

二、LEX编译程序的构造

小结

1、正规文法、正规集与正规式的概念和关系；

2、如何由正规文法得到正规式；

3、NFA的确定化；

4、DFA的最小化；

5、对含有ε弧的NFA进行确定化；

6、正规文法、正规式和自动机之间的相互转换。