正则语言与正则文法

正则表达式

- 正则文法擅长语言的产生，有穷状态自动机擅长语言的识别。
- 正则语言的正则表达式描述在对正则语言的表达上具有特殊的优势，为正则语言的计算机处理提供了方便条件。
  - 简洁、更接近语言的集合表示和语言的计算机表示等。

启示

产生语言$\{a^n b^m c^k | n，m，k \geqslant 1\} \cup$
$\{a^i c^n b x a^m | i \geqslant 0, n \geqslant 1, m \geqslant 2$, x为d和e组成的串$\}$
的正则文法为

$A \to aA|aB|cE$

$B \to bB|bC$

$C \to cC|c$

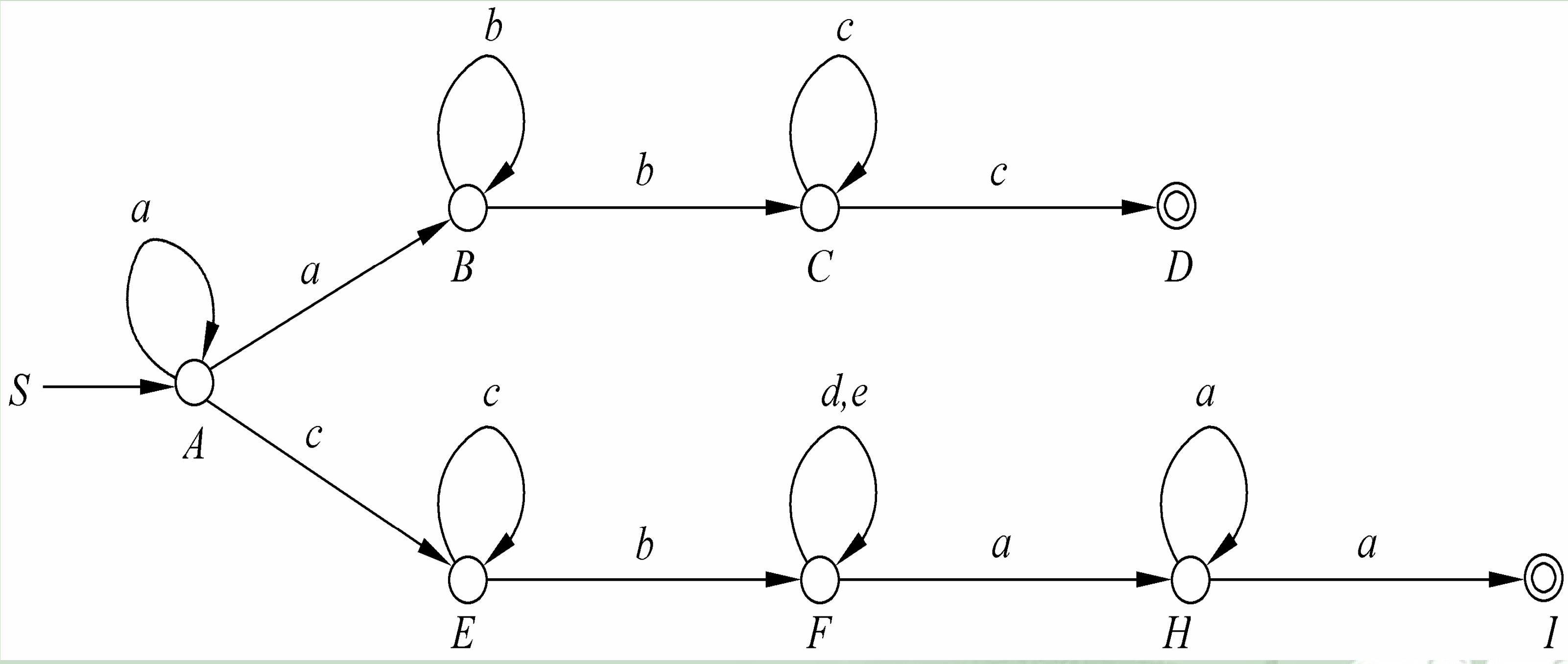
$E \to cE|bF$

$F \to dF|eF|aH$

$H \to aH|a$

启示

- 接受此语言的**NFA M**



启示

- 计算集合 set(q)

$set(A)=\{a^n|n \geqslant 0\}=\{a\}^*$

$set(B)= set(A)\{a\}\{b^n|n \geqslant 0\}$

$=\{a^n a b^m|m，n \geqslant 0\}$

$=\{a\}^*\{a\}\{b\}^*=\{a\}^+\{b\}^*$

$set(C)= set(B)\{b\}\{c\}^*$

$=\{a\}^*\{a\}\{b\}^*\{b\}\{c\}^*=\{a\}^+\{b\}^+\{c\}^*$

$set(D)= set(C) \{c\}=\{a\}^+\{b\}^+\{c\}^*\{c\}$

$=\{a\}^+\{b\}^+\{c\}^+$

启示

$set(E)= set(A)\{c\}\{c\}^*$

$=\{a\}^*\{c\}\{c\}^*=\{a\}^*\{c\}^+$

$set(F)= set(E)\{b\}\{d，e\}^*=\{a\}^*\{c\}^+\{b\}\{d，e\}^*$

$set(H)= set(F)\{a\}\{a\}^*=\{a\}^*\{c\}^+ \{b\}\{d，e\}^*\{a\}\{a\}^*$

$=\{a\}^*\{c\}^+ \{b\}\{d，e\}^*\{a\}^+$

$set(I)= set(H)\{a\}=\{a\}^*\{c\}^+ \{b\}\{d，e\}^*\{a\}^+\{a\}$

$L(M)= set(D)\cup set(I)$

$= \{a\}^+\{b\}^+\{c\}^+\cup\{a\}^*\{c\}^+ \{b\}\{d，e\}^*\{a\}^+\{a\}$

启示

根据集合运算的定义，

{d，e}={d}∪{e}。

从而，

{d，e}*=({d}∪{e})*。

这样可以将L(M)写成如下形式：

$L(M)=\{a\}^+\{b\}^+\{c\}^+\cup\{a\}^*\{c\}^+\{b\}(\{d\}\cup\{e\})^*\{a\}^+\{a\}$

记作：

$a^+b^+c^++a^*c^+b(d+e)^*a^+a=aa^*bb^*cc^*+a^*cc^*b(d+e)^*aaa^*$

RE的形式定义

- **正则表达式(regular expression，RE)**

⑴ *Φ*是Σ上的**RE**，它表示语言 *Φ*;

⑵ ε 是Σ上的**RE**，它表示语言**{** ε **}**;

⑶ 对于∀**a**∈Σ，**a**是Σ上的**RE**，它表示语言**{a}**;

RE的形式定义

(4) 如果**r**和**s**分别是Σ上表示语言**R**和**S**的**RE**，则：

r与**s**的"和" **(r+s)**是Σ上的**RE**，**(r+s)**表达的语言为**R∪S**；

r与**s**的"乘积" **(rs)**是Σ上的**RE**，**(rs)**表达的语言为**RS**；

r的克林闭包**(r^*)**是Σ上的**RE**，**(r^*)**表达的语言为**R^***。

(5) 只有满足(1)、(2)、(3)、(4)的才是Σ上的**RE**。

RE的形式定义

- 例： 设Σ={0，1}

(1) 0，表示语言{0}；

(2) 1，表示语言{1}；

(3) (0+1)，表示语言{0，1}；

(4) (01)，表示语言{01}；

(5) $((0+1)^*)$，表示语言$\{0，1\}^*$；

(6) $((00)((00)^*))$，表示语言$\{00\}\{00\}^*$；

RE的形式定义

⑺ $((((0+1)^*)(0+1))((0+1)^*))$，表示语言$\{0，1\}^+$；

⑻ $((((0+1)^*)000)((0+1)^*))$，表示{0，1}上的至少含有3个连续0的串组成的语言；

⑼ $((((0+1)^*)0)1)$，表示所有以01结尾的0、1字符串组成的语言；

⑽ $(1(((0+1)^*)0))$，表示所有以1开头，并且以0结尾的0、1字符串组成的语言。

RE的形式定义

- 约定

(1) r的正闭包r^+表示r与(r^*)的乘积以及(r^*)与r的乘积：

$$r^+=(r(r^*))=((r^*)r)$$

(2) 闭包运算的优先级最高，乘运算的优先级次之，加运算“+”的优先级最低。所以，在意义明确时，可以省略其中某些括号。

$$((((0+1)^*)000)((0+1)^*))=(0+1)^*000(0+1)^*$$

RE的形式定义

$$((((0+1)^*)(0+1))((0+1)^*))=(0+1)^*(0+1)(0+1)^*$$

(3) 在意义明确时，**RE r**表示的语言记为**L(r)**，也可以直接地记为**r**。

(4) 加、乘、闭包运算均执行左结合规则。

RE的形式定义

- **相等(equivalence)**
  - **r**、**s**是字母表Σ上的一个RE，如果**L(r)=L(s)**，则称**r**与**s**相等，记作**r=s** 。
  - 相等也称为等价。
- **几个基本结论**

  (1) 结合律：**(rs)t=r(st)**

  (r+s)+t=r+(s+t)

  (2) 分配律：**r(s+t)=rs+rt**

  (s+t)r=sr+tr

RE的形式定义

⑶ 交换律：　r+s=s+r。

⑷ 幂等律：　r+r=r。

⑸ r+ Φ=r。

⑹ r ε = ε r=r。

⑺ r Φ= Φr= Φ。

⑻ L(Φ)= Φ。

⑼ L(ε)={ ε }。

⑽ L(a)={a}。

RE的形式定义

(11) $L(rs)=L(r)L(s)$。

(12) $L(r+s)=L(r)\cup L(s)$。

(13) $L(r^*)=(L(r))^*$ 。

(14) $L(\Phi^*)=\{\varepsilon\}$。

(15) $L((r+\varepsilon)^*)=L(r^*)$。

(16) $L((r^*)^*)=L(r^*)$。

(17) $L((r^*s^*)^*)=L((r+s)^*)$。

(18) 如果$L(r)\subseteq L(s)$，则$r+s=s$。

RE的形式定义

⒆ $L(r^n)=(L(r))^n$ 。

⒇ $r^n r^m=r^{n+m}$ 。

一般地， $r+\varepsilon\neq r$，$(rs)^n\neq r^n s^n$，$rs\neq sr$。

- **幂**

r是字母表Σ上的RE，r的n次幂定义为

(1) $r^0=\varepsilon$ 。

(2) $r^n=r^{n-1}r$。

RE的形式定义

- 例：设Σ={0，1}

  00表示语言{00}；

  $(0+1)^*00(0+1)^*$表示所有的至少含两个连续0的0、1串组成的语言；

  $(0+1)^*1(0+1)^9$表示所有的倒数第10个字符为1的串组成的语言；

RE的形式定义

$L((0+1)^*011)$={x|x是以011结尾的0、1串}；

$L(0^+1^+2^+)=\{0^n1^m2^k \mid m，n，k \geqslant 1\}$；

$L(0^*1^*2^*)=\{0^n1^m2^k \mid m，n，k \geqslant 0\}$；

$L(1(0+1)^*1+0(0+1)^*0)$={x|x的开头字符与尾字符相同}。

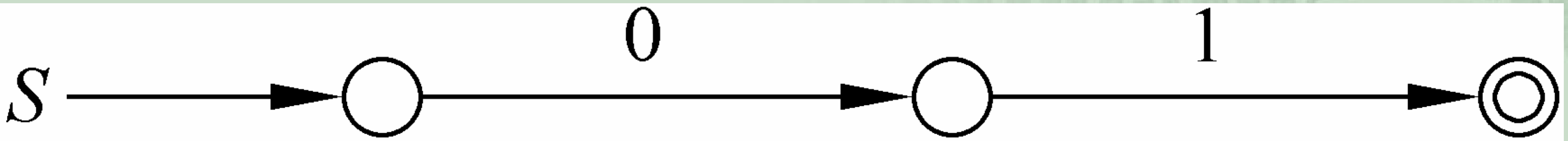
RE与FA等价

- 正则表达式**r**称为与**FA M**等价，如果**L(r)=L(M)**。
- 寻找一种比较“机械”的方法，使得计算机系统能够自动完成**FA**与**RE**之间的转换。

RE到FA的等价变换

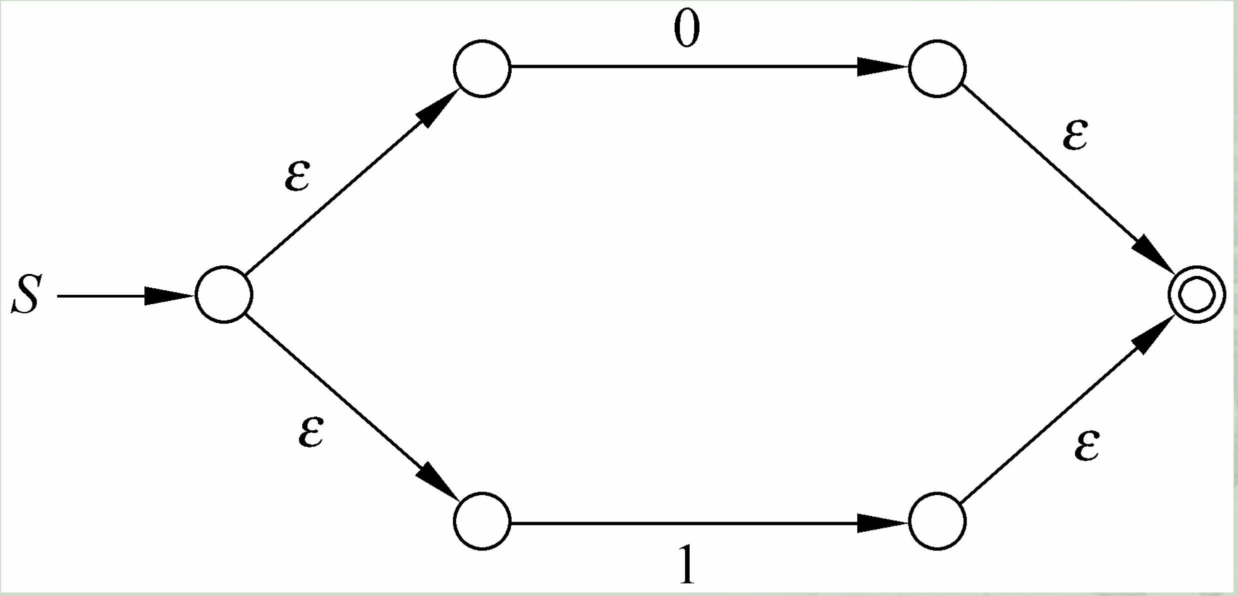
- 0对应的FA

- 01对应的FA



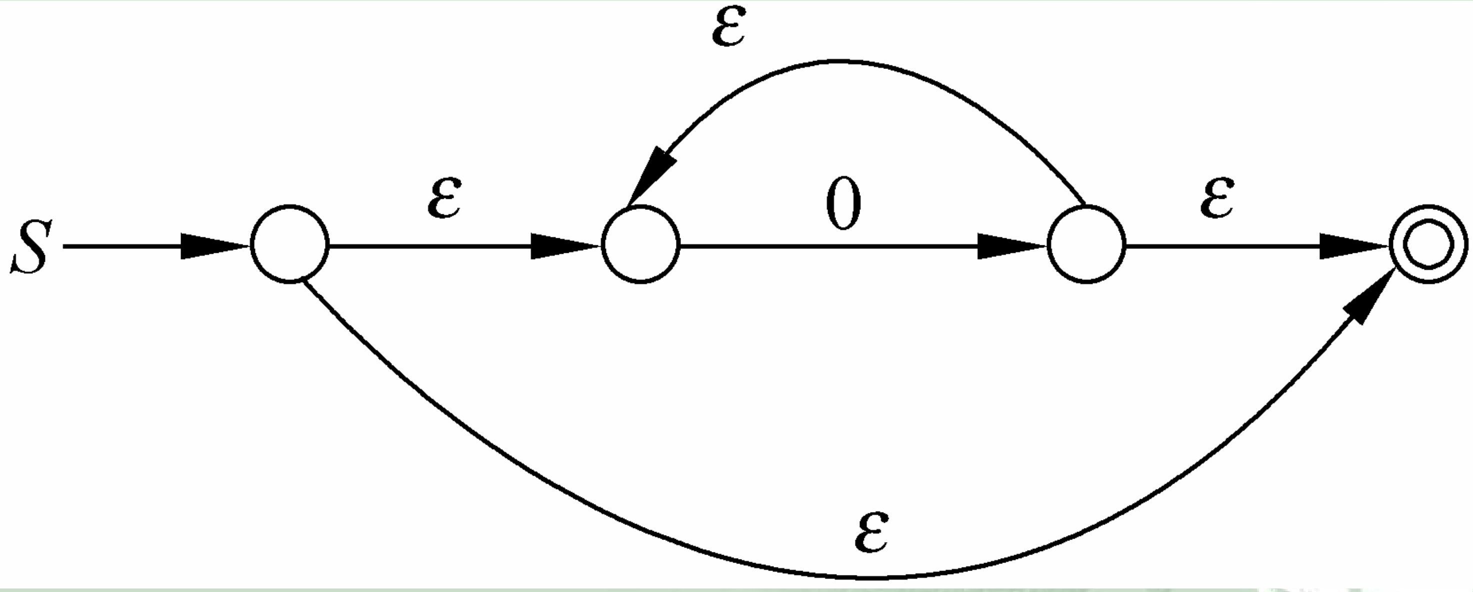
RE到FA的等价变换

- **0+1对应的FA**



RE到FA的等价变换

- 0^*对应的FA



RE到FA的等价变换

定理 RE表示的语言是RL。

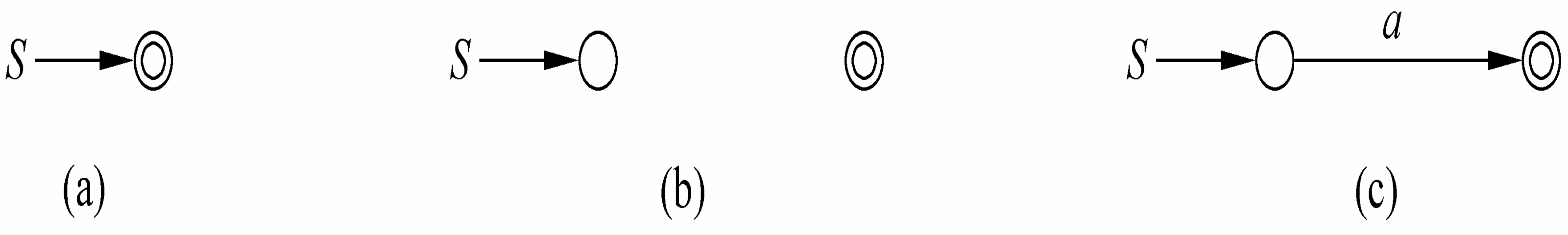
证明：

- 施归纳于正则表达式中所含的运算符的个数**n**，证明对于字母表∑上的任意正则表达式**r**，存在**FA M**，使得**L(M) = L(r)** 。
  - **M**恰有一个终止状态。
  - **M**在终止状态下不作任何移动。

RE到FA的等价变换

n=0

r= ε **r=** Φ **r=a**



RE到FA的等价变换

$n \leq k$

设结论对于**n≤k**时成立，此时有如下**FA**：

$M_1=(Q_1, \Sigma, \delta_1, q_{01}, \{f_1\})$

$M_2=(Q_2, \Sigma, \delta_2, q_{02}, \{f_2\})$

$L(M_1)=L(r_1)$，$L(M_2)=L(r_2)$。

$Q_1 \cap Q_2=\Phi$。

RE到FA的等价变换

$n=k+1$ 时有3种情况：

(1) $r=r_1+r_2$

取q_0，$f \notin Q_1 \cup Q_2$，令

$M=(Q_1 \cup Q_2 \cup \{q_0, f\}, \Sigma, \delta, q_0, \{f\})$

① $\delta(q_0, \varepsilon)=\{q_{01}, q_{02}\}$;

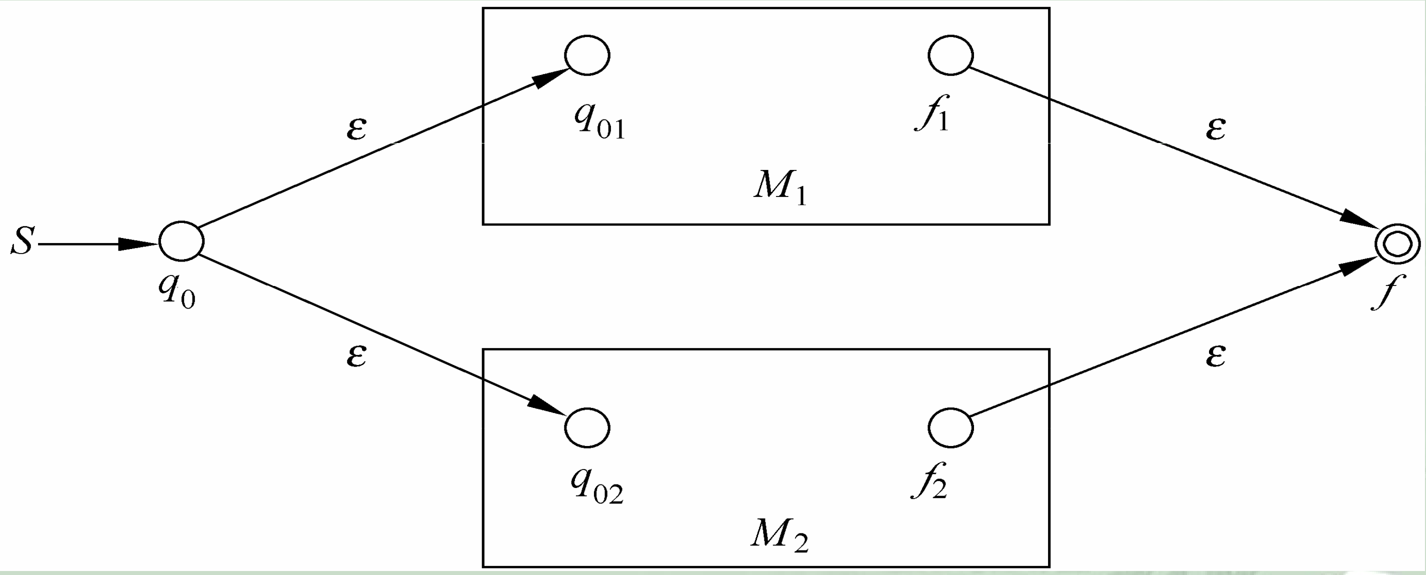
② 对$\forall q \in Q_1$，$a \in \Sigma \cup \{\varepsilon\}$，$\delta(q, a)=\delta_1(q, a)$;

对$\forall q \in Q_2$，$a \in \Sigma \cup \{\varepsilon\}$，$\delta(q, a)=\delta_2(q, a)$;

③ $\delta(f_1, \varepsilon)=\{f\}$;

④ $\delta(f_2, \varepsilon)=\{f\}$。

RE到FA的等价变换



$$r=r_1+r_2$$

RE到FA的等价变换

(2) $r=r_1r_2$

$M=(Q_1 \cup Q_2, \Sigma, \delta, q_{01}, \{f_2\})$

① 对$\forall q \in Q_1-\{f_1\}, a \in \Sigma \cup \{\varepsilon\}$

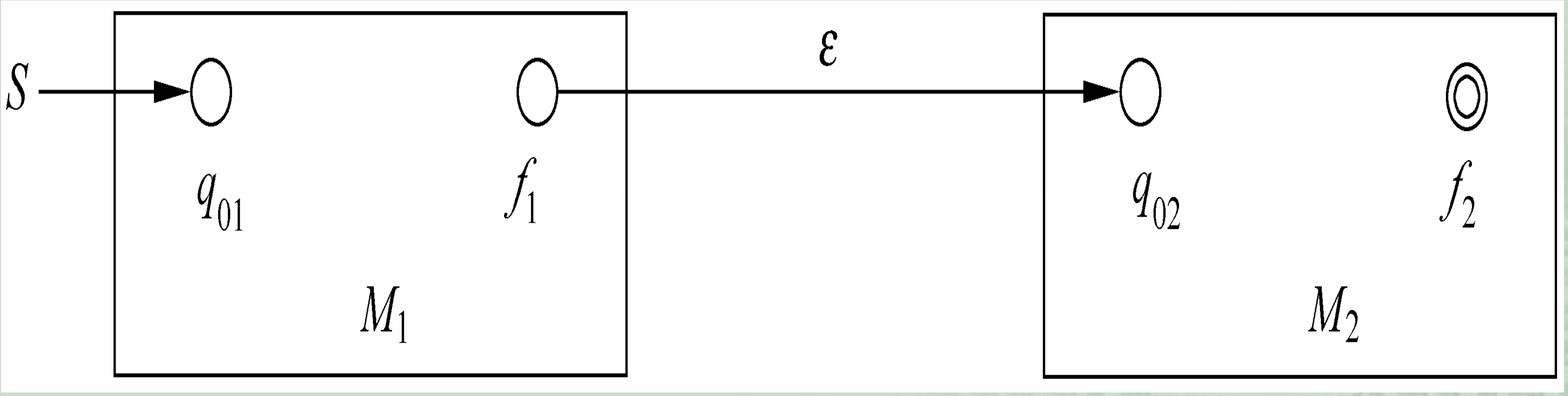
$\delta(q, a)=\delta_1(q, a)$;

② 对$\forall q \in Q_2-\{f_2\}, a \in \Sigma \cup \{\varepsilon\}$

$\delta(q, a)=\delta_2(q, a)$;

③ $\delta(f_1, \varepsilon)=\{q_{02}\}$

RE到FA的等价变换



$$r=r_1r_2$$

RE到FA的等价变换

(3)$r=r_1$*

$M=(Q_1\cup\{q_0, f\}, \Sigma, \delta, q_0, \{f\})$

其中q_0，$f\notin Q_1$，定义δ为

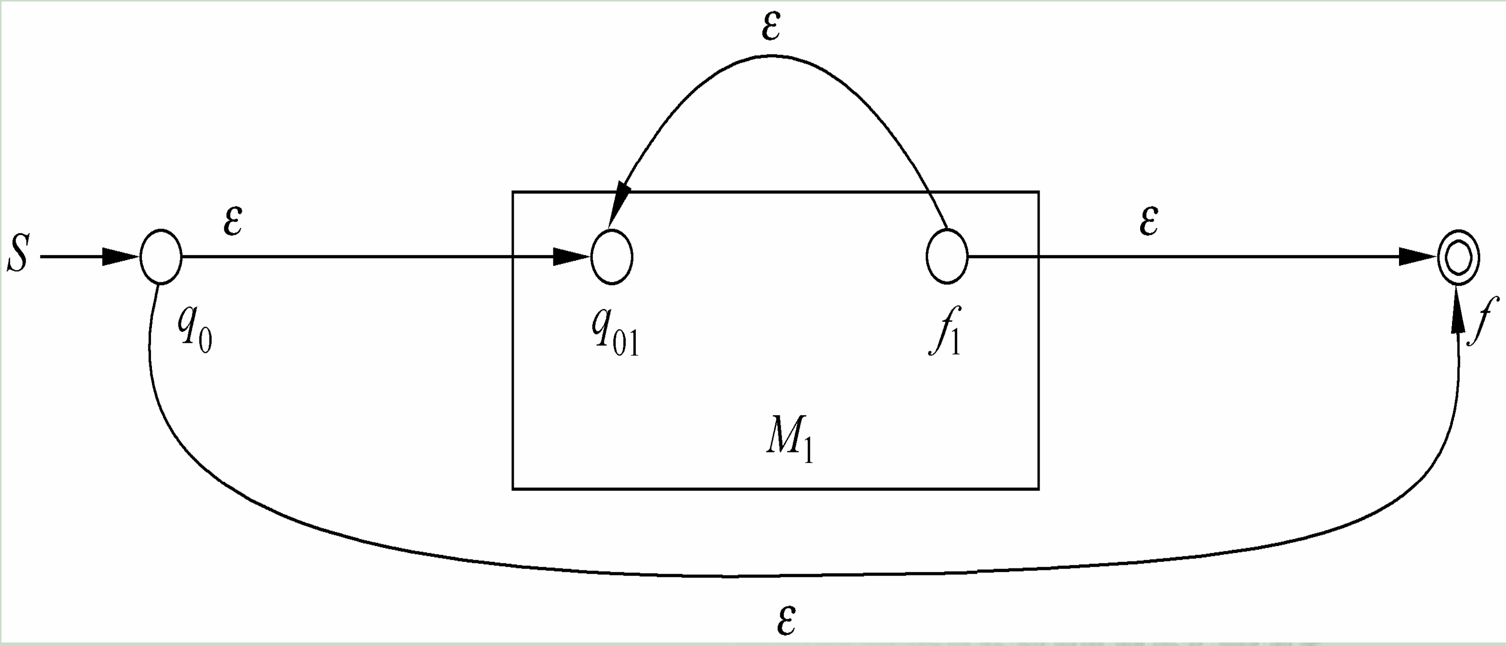
① 对$\forall q\in Q_1-\{f_1\}$，$a\in\Sigma$，

$\delta(q, a)=\delta_1(q, a)$。

② $\delta(f_1, \varepsilon)=\{q_{01}, f\}$。

③ $\delta(q_0, \varepsilon)=\{q_{01}, f\}$。

RE到FA的等价变换



$$r=r_1^*$$

RE到FA的等价变换

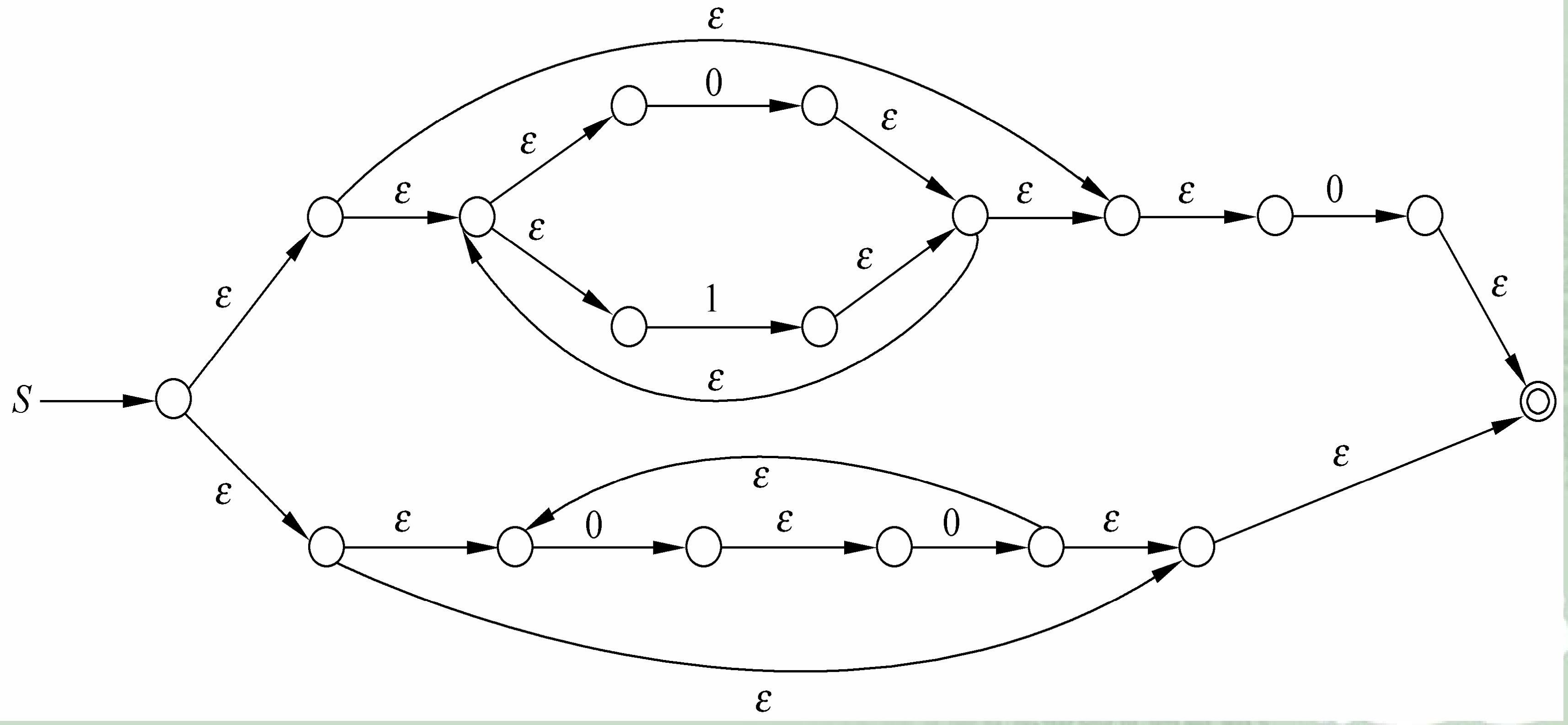
- 按照上述定理证明给出的方法构造一个给定RE的等价**FA**时，该**FA**有可能含有许多的空移动。
- 可以按照自己对给定RE的“理解”以及对**FA**的“理解”“直接地”构造出一个比较“简单”的**FA**。
- 定理证明中所给的方法是机械的。由于“直接地”构造出的**FA**的正确性依赖于构造者的“理解”，所以它的正确性缺乏有力的保证。

RE到FA的等价变换

- **例：构造与 $(0+1)^*0+(00)^*$等价的FA。**

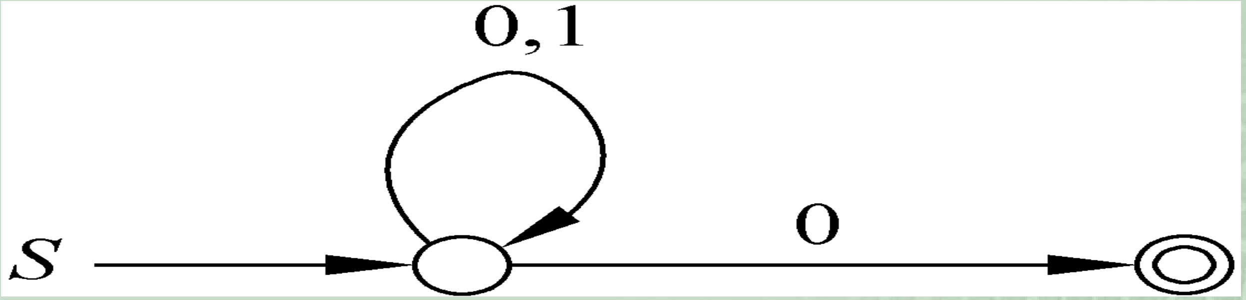
RE到FA的等价变换

- 例：构造与 $(0+1)^*0+(00)^*$等价的FA。



RE到FA的等价变换

- 按照对$(0+1)^*0+(00)^*$的“理解”“直接地”构造出的FA。



RL可以用RE表示

- 计算**DFA**的每个状态对应的集合——字母表的克林闭包的等价分类，是具有启发意义的。这个计算过程难以“机械”地进行。
- 计算$\mathbf{q_1}$到$\mathbf{q_2}$的一类串的集合：$\mathbf{R^k_{ij}}$ 。
- 图上作业法。

RL可以用RE表示

定理 RL可以用RE表示。

设DFA $M=(\{q_1, q_2, \ldots, q_n\}, \Sigma, \delta, q_1, F)$

$R^k_{ij}=\{x \mid \delta(q_i, x)=q_j$ 而且对于x的任意前缀 $y(y \neq x, y \neq \varepsilon)$，如果 $\delta(q_i, y)=q_l$，则 $l \leqslant k\}$。

R^k_{ij}是所有那些将DFA从q_i引导到q_j，并且不经过下标大于k的状态的所有字符串的集合。

RL可以用RE表示

$$R^0_{ij}=\begin{cases}\{a\mid \delta(q_i, a)=q_j\} & \text{如果}i\neq j\\ \{a\mid \delta(q_i, a)=q_j\}\cup\{\varepsilon\} & \text{如果}i=j\end{cases}$$

$$R^k_{ij}= R^{k-1}_{ik}(R^{k-1}_{kk})^* R^{k-1}_{kj} \cup R^{k-1}_{ij}$$

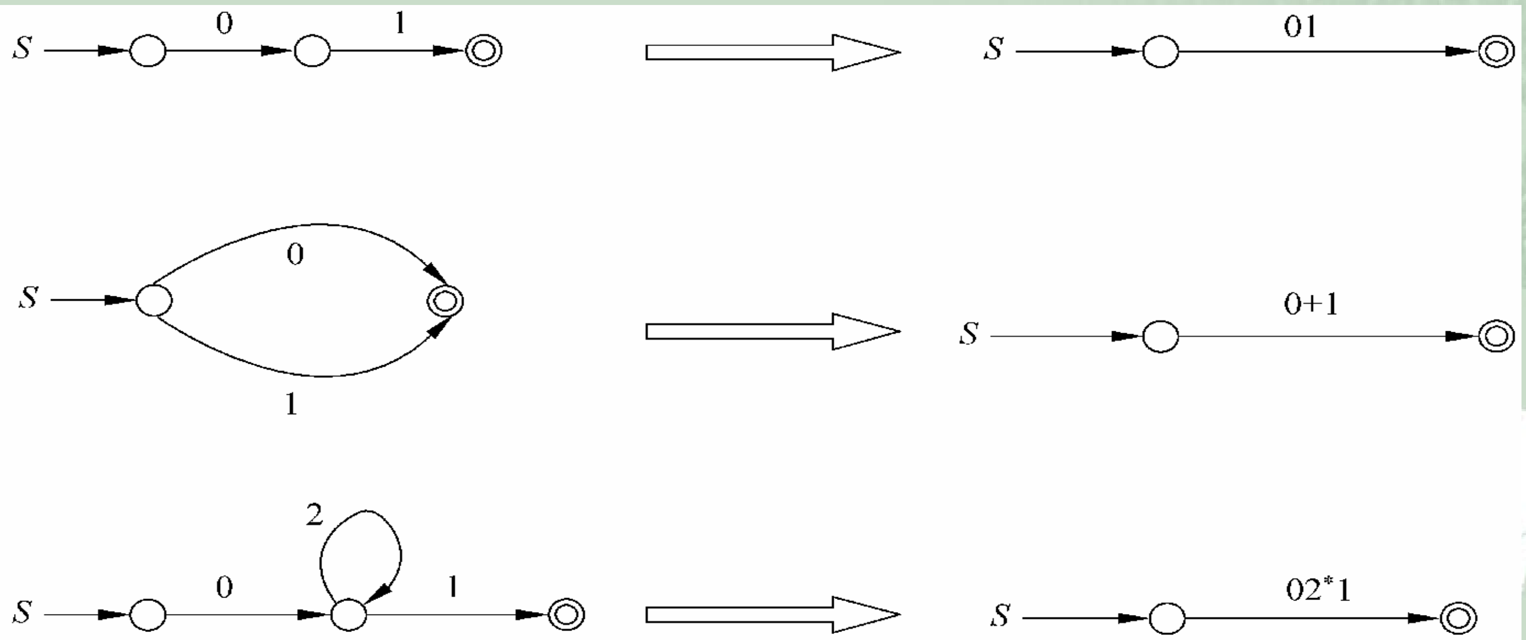
$$L(M)=\bigcup_{q_f\in F} R^n_{1f}$$

RL可以用RE表示

- 图上作业法

  放宽对FA状态转移图中弧标记的限制，允许它是正则表达式。

  启示：



RL可以用RE表示

- 图上作业法操作步骤

（1）预处理：

① 用标记为X和Y的状态将M“括起来”：

在状态转移图中增加标记为X和Y的状态，从标记为X的状态到标记为q_0的状态引一条标记为ε的弧；从标记为q(q∈F)的状态到标记为Y的状态分别引一条标记为ε的弧。

② 去掉所有的不可达状态。

RL可以用RE表示

(2) 对通过步骤(1)处理所得到的状态转移图重复如下操作，直到该图中不再包含除了标记为X和Y外的其他状态，并且这两个状态之间最多只有一条弧。

- 并弧
  - 将从q到p的标记为r_1，r_2，…，r_g并行弧用从q到p的、标记为$r_1+r_2+\ldots+r_g$的弧取代这g个并行弧。

RL可以用RE表示

- 去状态1
  - 如果从q到p有一条标记为r_1的弧，从p到t有一条标记为r_2的弧，不存在从状态p到状态p的弧，将状态p和与之关联的这两条弧去掉，用一条从q到t的标记为r_1r_2的弧代替。
- 去状态2
  - 如果从q到p有一条标记为r_1的弧，从p到t有一条标记为r_2的弧，从状态p到状态p标记为r_3的弧，将状态p和与之关联的这三条弧去掉，用一条从q到t的标记为$r_1r_3{}^*r_2$的弧代替。

RL可以用RE表示

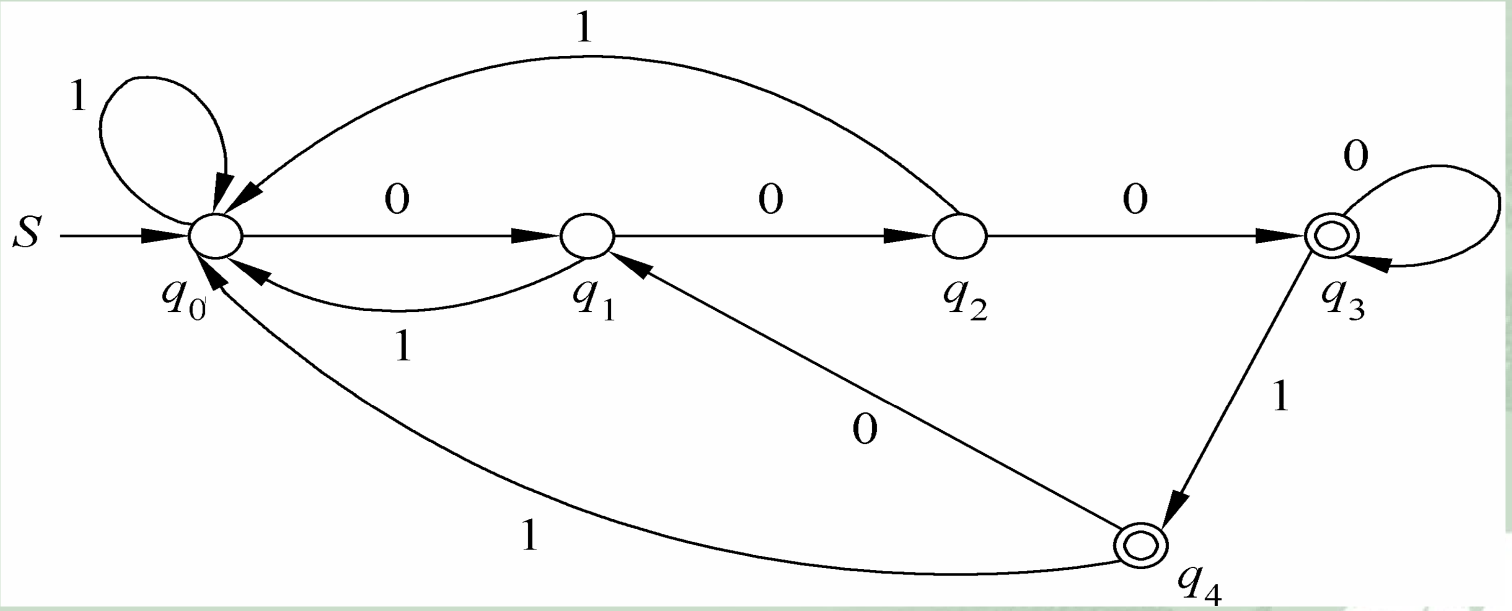
- 去状态3
  - 如果图中只有三个状态，而且不存在从标记为X的状态到达标记为Y的状态的路，则将除标记为X的状态和标记为Y的状态之外的第3个状态及其相关的弧全部删除。

RL可以用RE表示

(3) 从标记为X的状态到标记为Y的状态的弧的标记为所求的正则表达式。如果此弧不存在，则所求的正则表达式为 Φ。

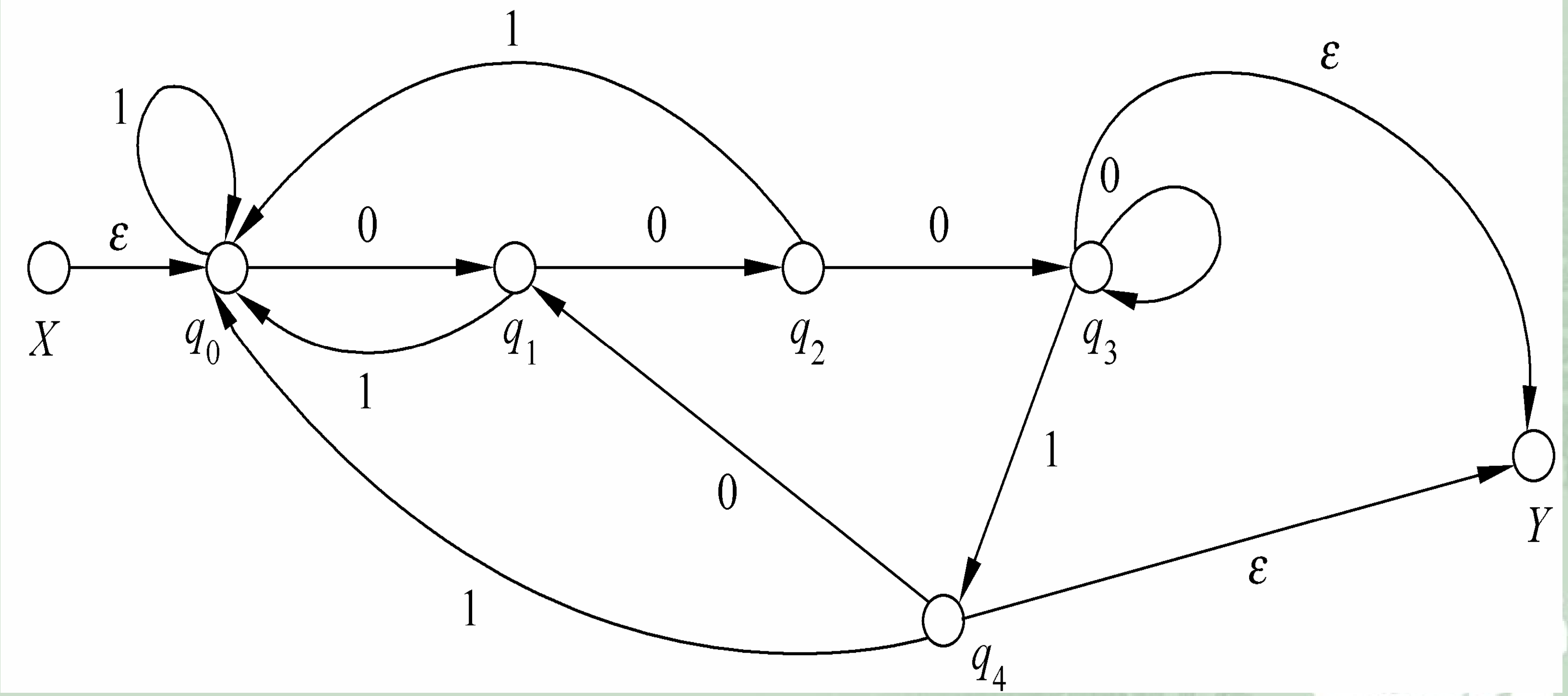
RL可以用RE表示

- 例： 求下图所示的DFA等价的RE 。



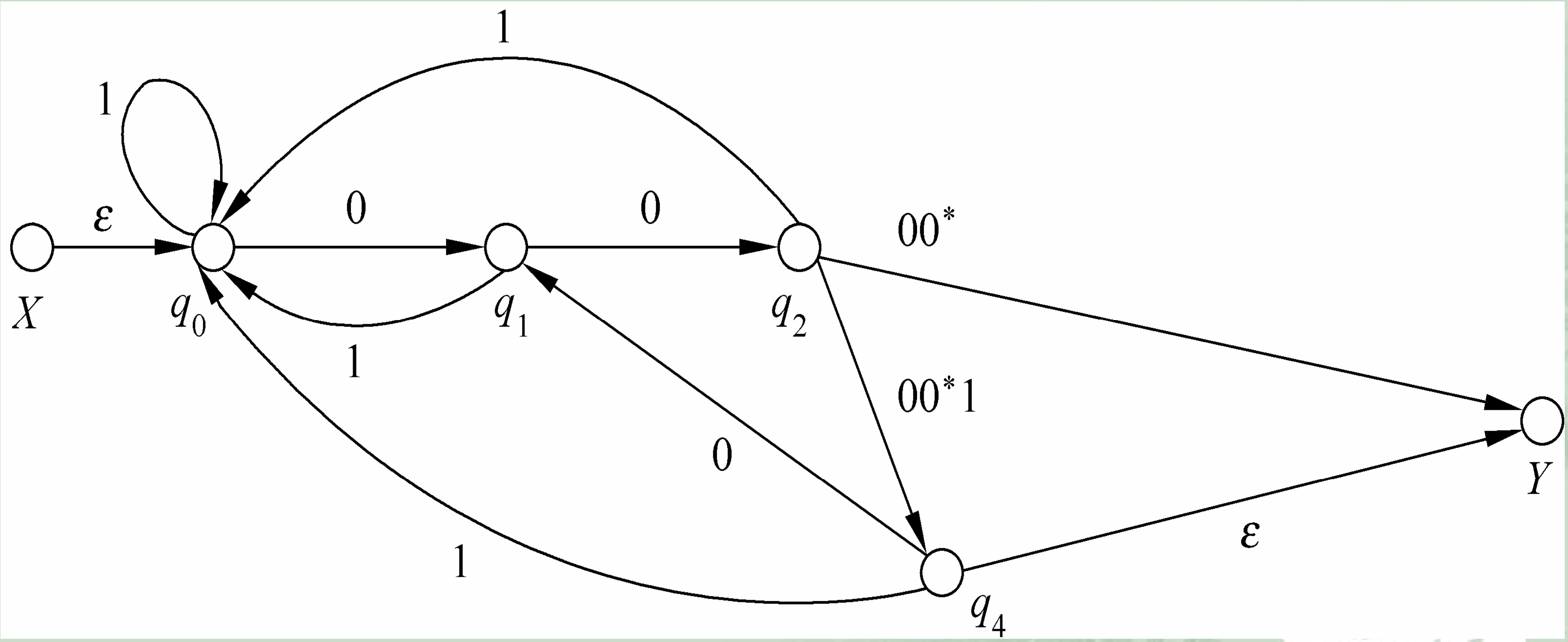
RL可以用RE表示

- 预处理。



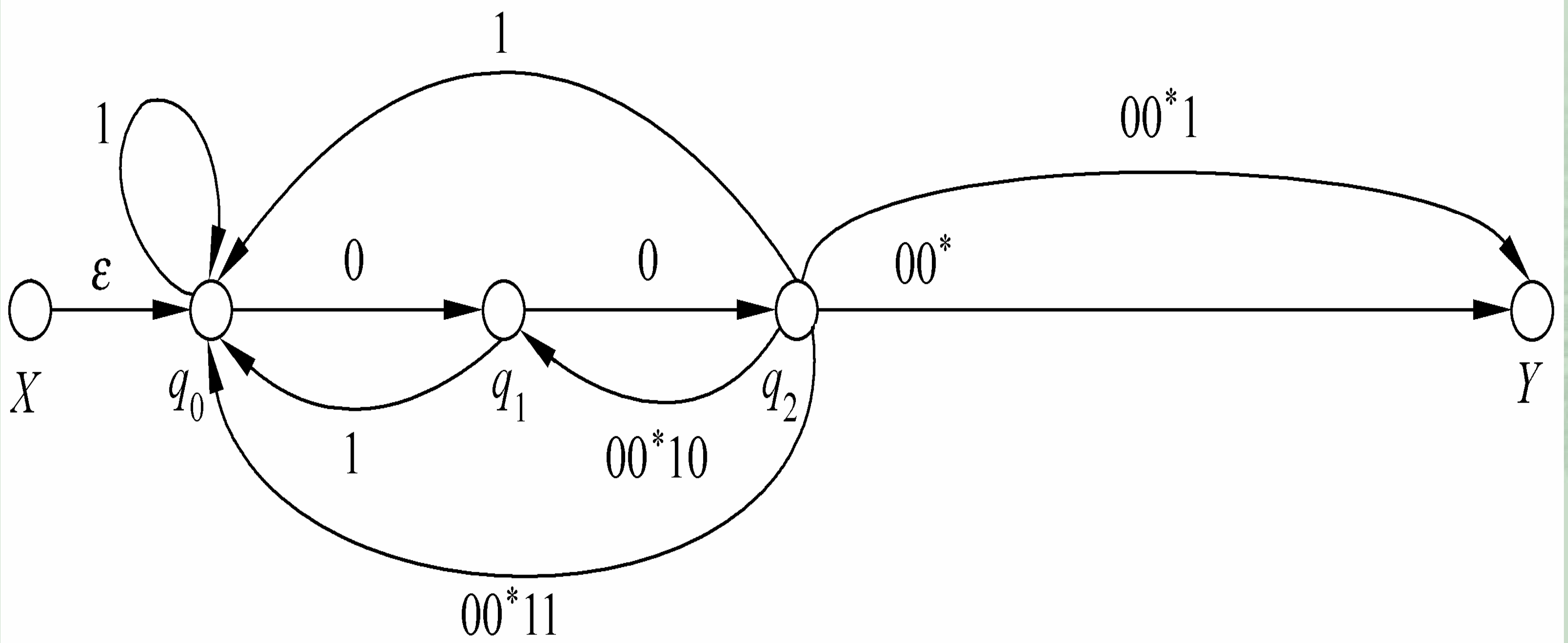
RL可以用RE表示

- 去掉状态q_3。



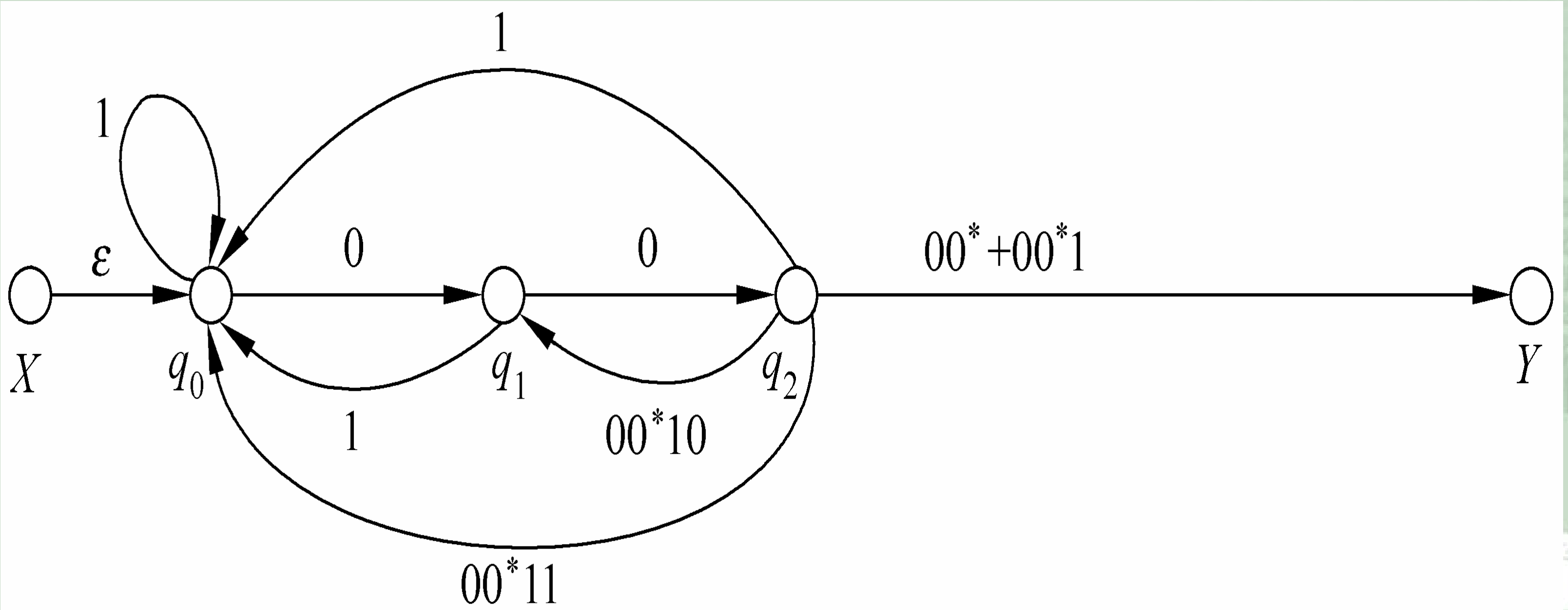
RL可以用RE表示

- 去掉状态q_4。



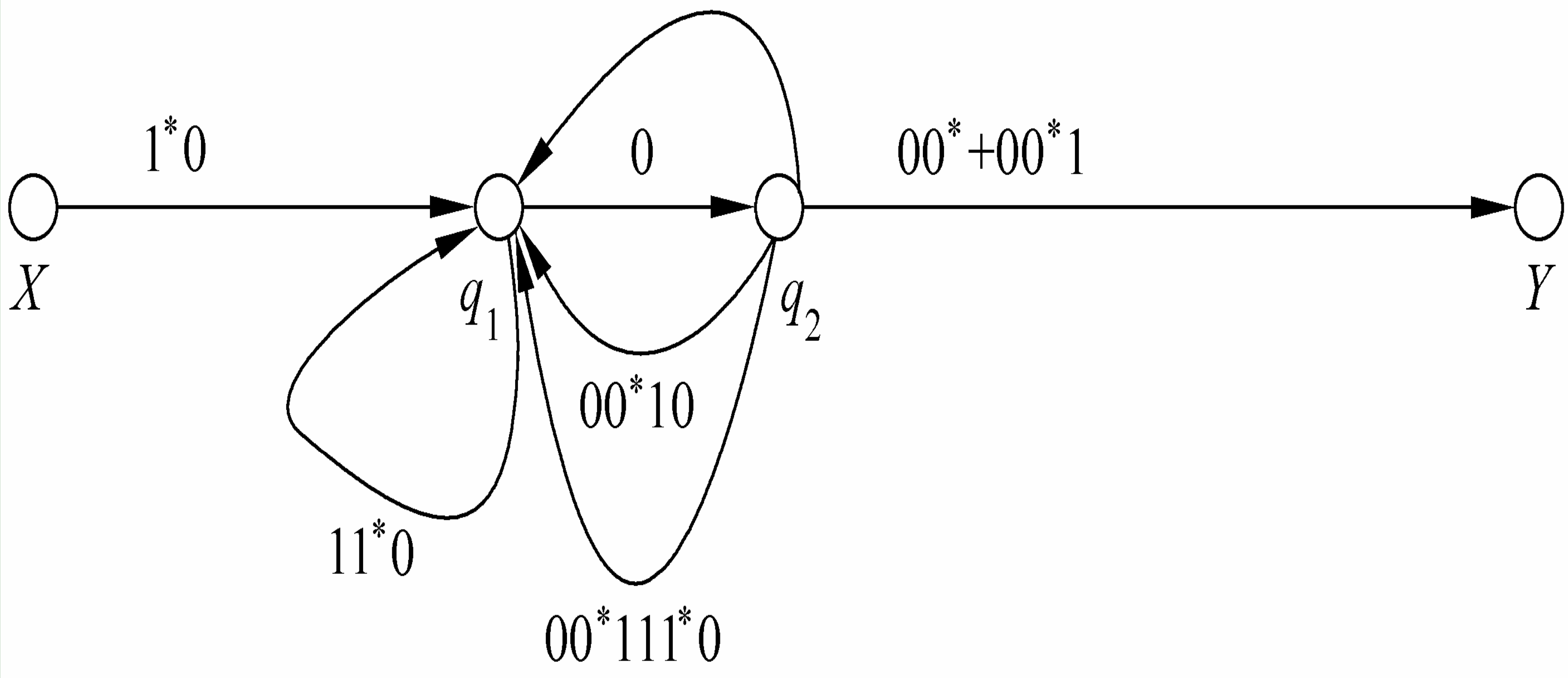
RL可以用RE表示

- 合并从标记为$\mathbf{q_2}$的状态到标记为**Y**的状态的两条并行弧。



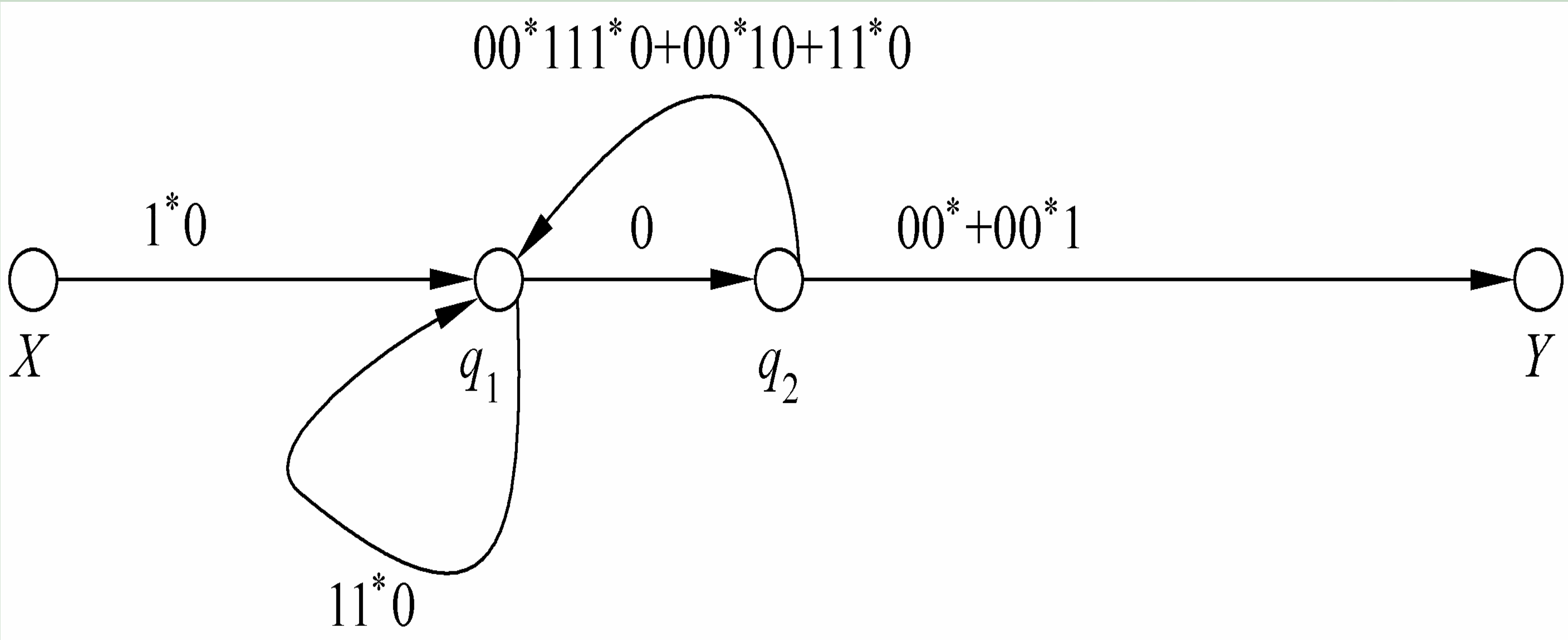
RL可以用RE表示

- 去掉状态q_0。



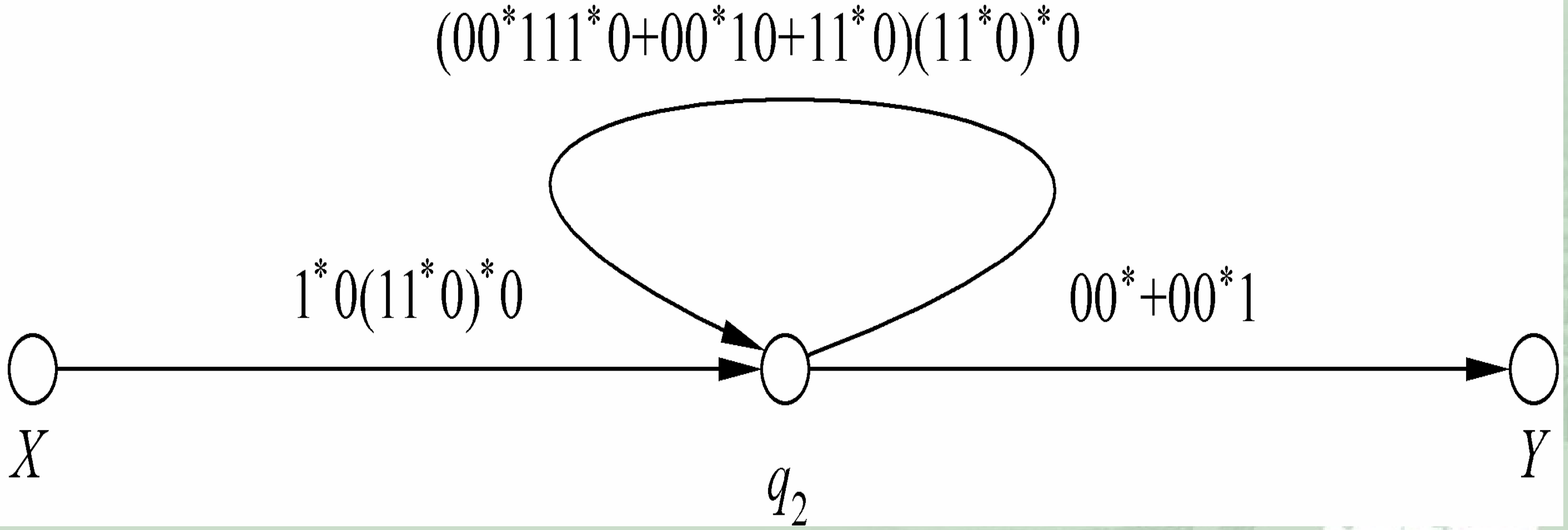
RL可以用RE表示

- 并弧。



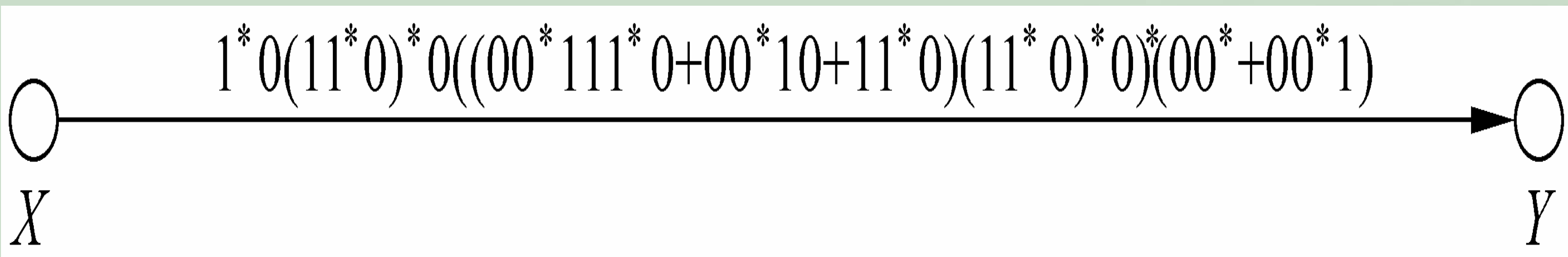
RL可以用RE表示

- 去掉状态q_1。



RL可以用RE表示

- 去掉状态q_2。



$1^*0(11^*0)^*0((00^*111^*0+00^*10+11^*0)(11^*0)^*0)^*(00^*+00^*1)$ 就是所求。

RL可以用RE表示

- 几点值得注意的问题

  (1) 如果去状态的顺序不一样，则得到的RE可能在形式是不一样，但它们都是等价的。

  (2) 当DFA的终止状态都是不可达的时候，状态转移图中必不存在从开始状态到终止状态的路。此时，相应的RE为 Φ。

  (3) 不计算自身到自身的弧，如果状态q的入度为n，出度为m，则将状态q及其相关的弧去掉之后，需要添加n*m条新弧。

  (4) 对操作的步数施归纳，可以证明它的正确性。

正则文法

- 如果对于∀ $\alpha \to \beta \in P$，$\alpha \to \beta$ 均具有形式

  A→w

  A→wB

  其中**A**，**B**∈**V**，**w**∈T^+。则称**G**为**正则文法** **(regular grammar ,RG)**或者正规文法。
- **L(G)** 叫做**正则语言**或者**正规语言** **(regular language ,RL)**。

正则文法

定理 L是RL的充要条件是存在一个文法，该文法产生语言L，并且它的产生式要么是形如：$A \rightarrow a$的产生式，要么是形如$A \rightarrow aB$的产生式。其中A、B为语法变量，a为终极符号。

- 证明：
  - 充分性：设有G′，L(G′)=L，且G′的产生式的形式满足定理要求。这种文法就是RG。所以，G′产生的语言就是RL，故L是RL。

正则文法

- 必要性

构造：用产生式组：

$A \rightarrow a_1A_1$

$A_1 \rightarrow a_2A_2$

…

$A_{n-1} \rightarrow a_n$

代替产生式

$A \rightarrow a_1a_2 \ldots a_n$

正则文法

- 用产生式组

  $A \to a_1 A_1$

  $A_1 \to a_2 A_2$

  …

  $A_{n-1} \to a_n B$

  代替产生式

  $A \to a_1 a_2 \ldots a_n B$

正则文法

- 证明L(G′)=L(G) 。

  施归纳于推导的步数，证明一个更一般的结论：对于$\forall A\in V$，$A \Rightarrow_G^+ x \Leftrightarrow A \Rightarrow_{G'}^+ x$。因为$S\in V$，所以结论自然对S成立。

正则文法

几点注意事项：

- 为了证明一个特殊的结论，可以通过证明一个更为一般的结论来完成。这从表面上好像是增加了我们要证明的内容，但实际上它会使我们能够更好地使用归纳假设，以便顺利地获得我们所需要的结论。

正则文法

- 施归纳于推导的步数是证明不少问题的较为有效的途径。有时我们还会对字符串的长度施归纳。

本证明的主要部分含两个方面，首先是构造，然后对构造的正确性进行证明。这种等价性证明的思路是非常重要的。

线性文法

- **线性文法(liner grammar)**
  - 设G=(V，T，P，S)，如果对于∀ $\alpha \rightarrow \beta \in P$，$\alpha \rightarrow \beta$ 均具有如下形式：
  - **A→w**
  - **A→wBx**
  - 其中**A**，**B**∈**V**，**w**，**x**∈**T***，则称**G**为线性文法。
- **线性语言(liner language)**
  - **L(G)**叫做线性语言

右线性文法

- 右线性文法(right liner grammar)
  - 设G=(V，T，P，S)，如果对于∀α→β∈P，α→β均具有如下形式：
  - A→w
  - A→wB
  - 其中A，B∈V，w，x∈T*，则称G为右线性文法。
- 右线性语言(right liner language)
  - L(G)叫做右线性语言。

左线性文法

- 左线性文法(left liner grammar)
  - 设G=(V，T，P，S)，如果对于$\forall \alpha \to \beta \in P$，$\alpha \to \beta$均具有如下形式：
  - $A \to w$
  - $A \to Bw$
  - 其中A，$B \in V$，w，$x \in T^*$，则称G为左线性文法。
- 左线性语言(left liner language)
  - L(G)叫做左线性语言。

左线性文法

定理 **L**是一个左线性语言的充要条件是存在文法**G**，**G**中的产生式要么是形如：**A→a**的产生式，要么是形如**A→Ba**的产生式，使得**L(G)=L**。其中**A**、**B**为语法变量，**a**为终极符号。

左线性文法与右线性文法

定理 左线性文法与右线性文法等价。

- 按照前面定理的证明经验，要想证明本定理，需要完成如下工作：
  - 对任意右线性文法G，我们能够构造出对应的左线性文法G′，使得L(G′)=L(G);
  - 对任意左线性文法G，我们能够构造出对应的右线性文法G′，使得L(G′)=L(G)。

左线性文法与右线性文法

- **例：**语言{**0123456**}的左线性文法和右线性文法的构造。
- 右线性文法

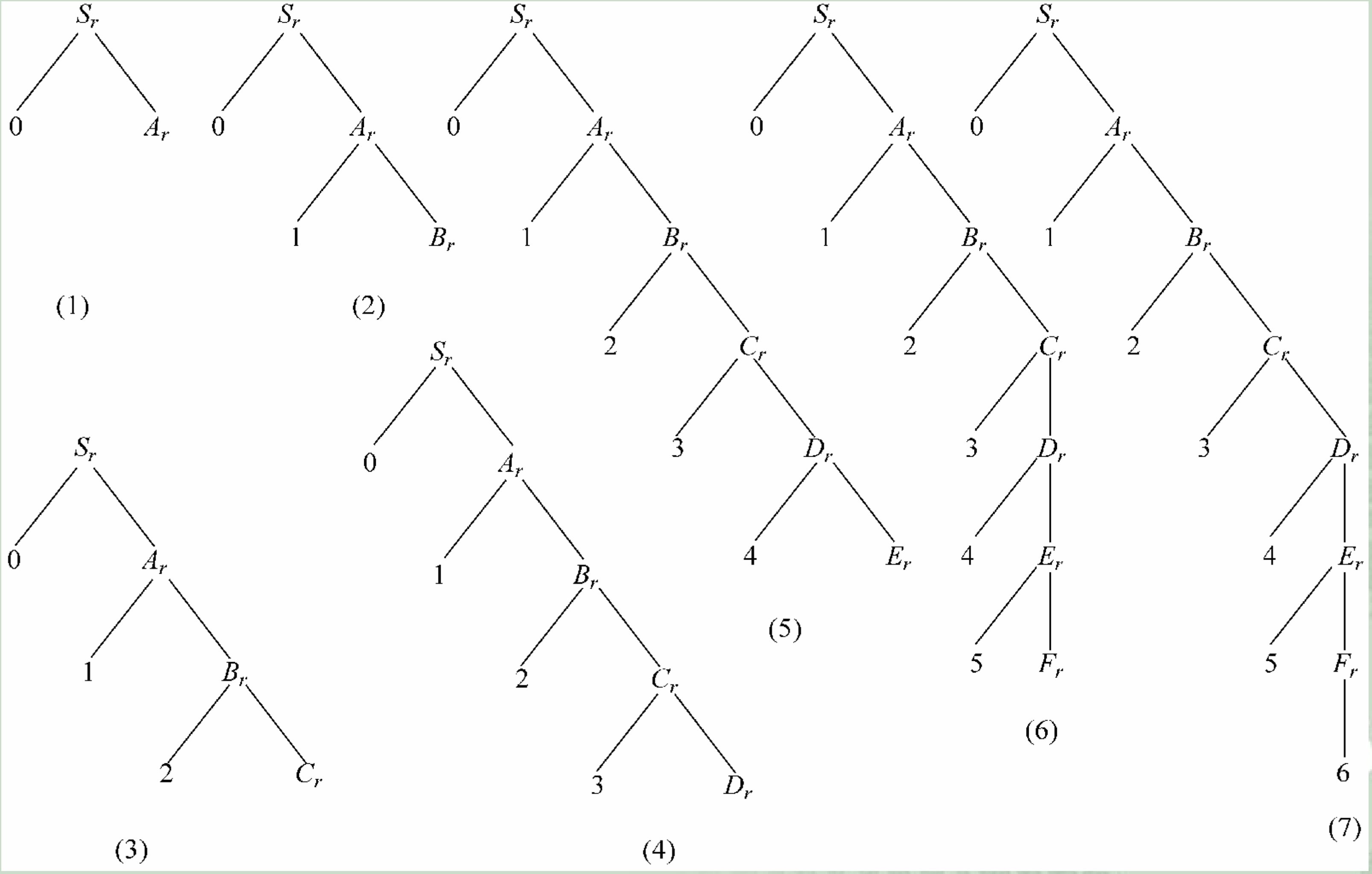
$$
\begin{aligned}
G_r\text{：}\ & S_r \to 0A_r \\
& A_r \to 1B_r \\
& B_r \to 2C_r \\
& C_r \to 3D_r \\
& D_r \to 4E_r \\
& E_r \to 5F_r \\
& F_r \to 6
\end{aligned}
$$

左线性文法与右线性文法

- 0123456在文法G_r中的推导

$S_r \Rightarrow 0A_r$	使用产生式$S_r \rightarrow 0A_r$
$\Rightarrow 01B_r$	使用产生式$A_r \rightarrow 1B_r$
$\Rightarrow 012C_r$	使用产生式$B_r \rightarrow 2C_r$
$\Rightarrow 0123D_r$	使用产生式$C_r \rightarrow 3D_r$
$\Rightarrow 01234E_r$	使用产生式$D_r \rightarrow 4E_r$
$\Rightarrow 012345F_r$	使用产生式$E_r \rightarrow 5F_r$
$\Rightarrow 0123456$	使用产生式$F_r \rightarrow 6$

左线性文法与右线性文法



左线性文法与右线性文法

- 左线性文法

G_l： $S_l \to A_l 6$

$A_l \to B_l 5$

$B_l \to C_l 4$

$C_l \to D_l 3$

$D_l \to E_l 2$

$E_l \to F_l 1$

$F_l \to 0$

左线性文法与右线性文法

- **0123456在文法G_l中的推导**

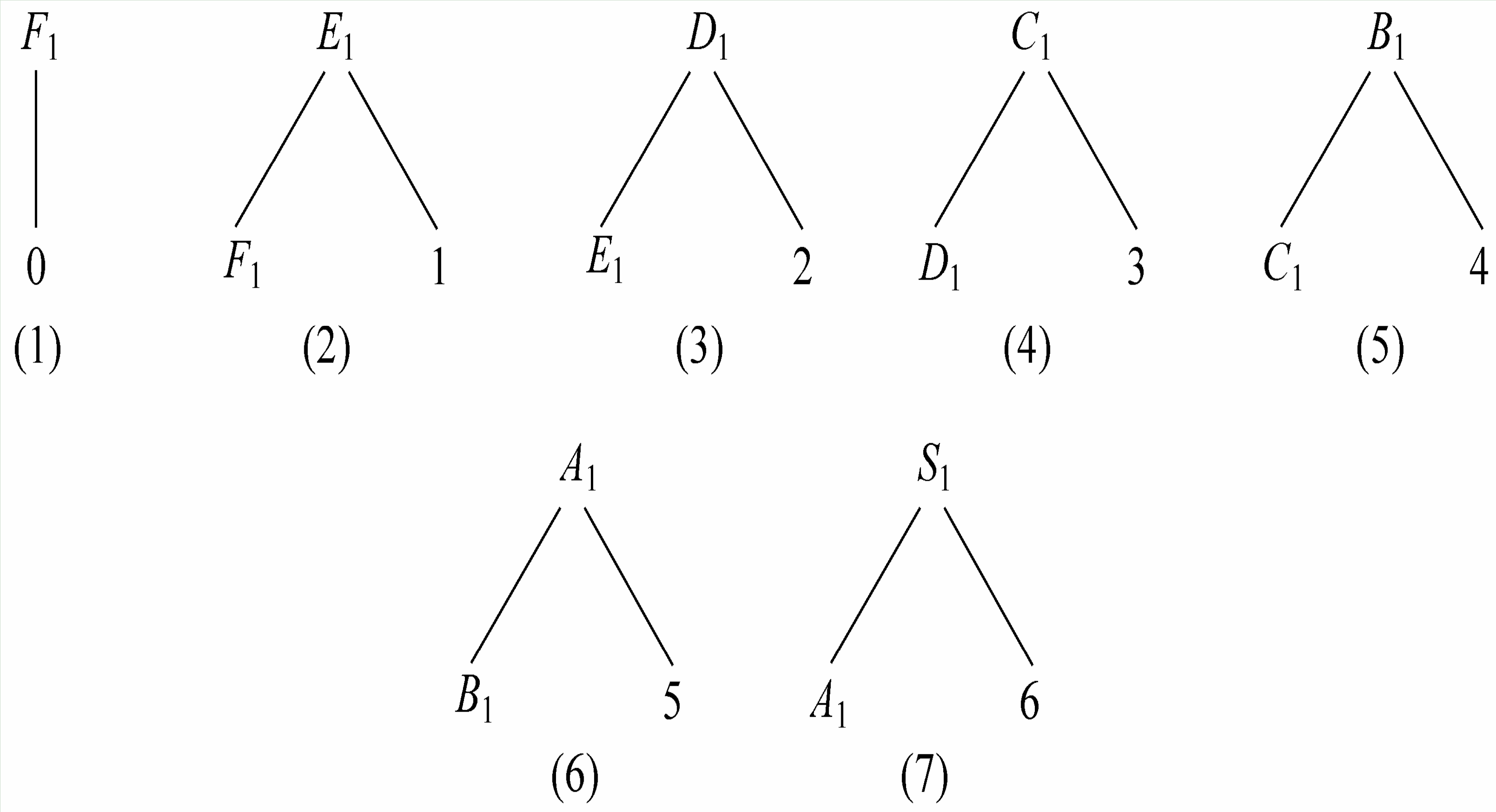
$S_l \Rightarrow A_l6$	使用产生式$S_l \rightarrow A_l6$
$\Rightarrow B_l56$	使用产生式$A_l \rightarrow B_l5$
$\Rightarrow C_l456$	使用产生式$B_l \rightarrow C_l4$
$\Rightarrow D_l3456$	使用产生式$C_l \rightarrow D_l3$
$\Rightarrow E_l23456$	使用产生式$D_l \rightarrow E_l2$
$\Rightarrow F_l123456$	使用产生式$E_l \rightarrow F_l1$
$\Rightarrow 0123456$	使用产生式$F_l \rightarrow 0$

左线性文法与右线性文法

- **0123456被归约成文法G_l的开始符号S**

$\underline{\mathbf{0}}\mathbf{123456}$	
$\Leftarrow \underline{\mathbf{F_l1}}\mathbf{234456}$	使用产生式$F_l \rightarrow 0$
$\Leftarrow \underline{\mathbf{E_l2}}\mathbf{3456}$	使用产生式$E_l \rightarrow F_l1$
$\Leftarrow \underline{\mathbf{D_l3}}\mathbf{456}$	使用产生式$D_l \rightarrow E_l2$
$\Leftarrow \underline{\mathbf{C_l4}}\mathbf{56}$	使用产生式$C_l \rightarrow D_l3$
$\Leftarrow \underline{\mathbf{B_l5}}\mathbf{6}$	使用产生式$B_l \rightarrow C_l4$
$\Leftarrow \underline{\mathbf{A_l6}}$	使用产生式$A_l \rightarrow B_l5$
$\Leftarrow S_l$	使用产生式$S_l \rightarrow A_l6$

左线性文法与右线性文法



左线性文法与右线性文法

定理 左线性文法的产生式与右线性文法的产生式混用所得到的文法不是**RG**。

证明：设有文法

G_{15}： $S \to 0A$

$A \to S1|1$

不难看出，$L(G_{15})=\{0^n1^n|n \geqslant 1\}$。我们构造不出RG G，使得$L(G)= L(G_{15})=\{0^n1^n|n \geqslant 1\}$。因为$L(G_{15})=\{0^n1^n|n \geqslant 1\}$不是RL。所以，$G_{15}$不是RG。有关该语言不是RL的证明我们将留到研究RL的性质时去完成。

FA与右线性文法

- 正则语言L有一个满足前面定理的正则文法G=(Q，T，P，S)。L中的一个句子$a_1a_2…a_n$在G中推导的特性：

(1)从S开始，除$a_1a_2…a_n$外，其它每个句型中有且仅有一个语法变量，而且此语法变量总是句型的尾字符。因此，句型中的终极符号是根据它被推导出来的先后顺序a_1、a_2、…、a_n依次排列的。

FA与右线性文法

(2) 每步推导产生且仅能产生一个终极符号：第**i**步产生终极符号$\mathbf{a_i}$。

(3) 使用形如**A→ aB**的产生式的推导，相当于是变量**A**产生出**aB**，而**B**接下去实现后续字符的产生。

(4)使用形如**A→ a**的产生式的推导，相当于是变量**A**产生出**a**后，整个推导结束。

FA与右线性文法

- **DFA M=(Q，**∑，δ，**q_0，F)**，处理句子**$a_1a_2\ldots a_n$**的特性。

(1) **M**按照句子**$a_1a_2\ldots a_n$**中字符的出现顺序，从开始状态**q_0**开始，依次处理字符**a_1**、**a_2**、**…**、**a_n**，在这个处理过程中，每处理一个字符进入一个状态，最后停止在某个终止状态。

FA与右线性文法

(2) 它每次处理且仅处理一个字符：第**i**步处理输入字符$\mathbf{a_i}$。

(3) 对应于使用δ(**q**，**a**)=**p**的状态转移函数的处理，相当于是在状态**q**完成对**a**的处理，然后由**p**接下去实现对后续字符的处理。

(4) 当δ(**q**，**a**)=**p**∈**F**，且**a**是输入串的最后一个字符时，**M**完成对此输入串的处理。

FA与右线性文法

$A_0 \Rightarrow a_1A_1$　　对应产生式$A_0 \rightarrow a_1A_1$

$\Rightarrow a_1a_2A_2$　　对应产生式$A_1 \rightarrow a_2A_2$

$\cdots$

$\Rightarrow a_1a_2\ldots a_{n-1}A_{n-1}$　　对应产生式$A_{n-2} \rightarrow a_{n-1}A_{n-1}$

$\Rightarrow a_1a_2\ldots a_{n-1}a_n$　　对应产生式$A_{n-1} \rightarrow a_n$

FA与右线性文法

$q_0\ a_1a_2\ldots a_{n-1}a_n$

$\vdash a_1q_1\ a_2\ldots a_{n-1}a_n$ 对应 $\delta(q_0，a_1)=q_1$

$\vdash a_1a_2q_2\ldots a_{n-1}a_n$ 对应 $\delta(q_1，a_2)=q_2$

……

$\vdash a_1a_2\ldots a_{n-1}q_{n-1}a_n$ 对应 $\delta(q_{n-2}，a_{n-1})=q_{n-1}$

$\vdash a_1a_2\ldots a_{n-1}a_nq_n$ 对应 $\delta(q_{n-1}，a_n)=q_n$

FA与右线性文法

- 其中q_n为M的终止状态。考虑根据a_1、a_2、…、a_n，让A_0与q_0对应、A_1与q_1对应、A_2与q_2对应、…、A_{n-2}与q_{n-2}对应、A_{n-1}与q_{n-1}对应。这样，就有希望得到正则文法推导与DFA的互相模拟的方式。

FA与右线性文法

定理 **FA**接受的语言是正则语言。

证明：

(1) 构造。

基本思想是让**RG**的推导对应**DFA**的移动。

设**DFA** $M=(Q,\ \Sigma,\ \delta,\ q_0,\ F)$，

取右线性文法 $G=(Q,\ \Sigma,\ P,\ q_0)$，

$P=\{q\rightarrow ap \mid \delta(q,\ a)=p\}\cup\{q\rightarrow a \mid \delta(q,\ a)=p\in F\}$

FA与右线性文法

（2）证明 $L(G)=L(M)-\{\varepsilon\}$。

对于 $a_1a_2\ldots a_{n-1}a_n \in \Sigma^+$,

$q_0 \Rightarrow^+ a_1a_2\ldots a_{n-1}a_n$

$\Leftrightarrow q_0 \rightarrow a_1q_1$， $q_1 \rightarrow a_2q_2$， …,

$q_{n-2} \rightarrow a_{n-1}q_{n-1}$， $q_{n-1} \rightarrow a_n \in P$

$\Leftrightarrow \delta(q_0, a_1)=q_1$, $\delta(q_1, a_2)=q_2$, …, $\delta(q_{n-2}, a_{n-1})=q_{n-1}$， $\delta(q_{n-1}, a_n)=q_n$， 且 $q_n \in F$

$\Leftrightarrow \delta(q_0, a_1a_2\ldots a_{n-1}a_n)= q_n \in F$

$\Leftrightarrow a_1a_2\ldots a_{n-1}a_n \in L(M)$

FA与右线性文法

（3）关于ε句子。

如果$q_0 \notin F$，则ε $\notin L(M)$，$L(G)=L(M)$。

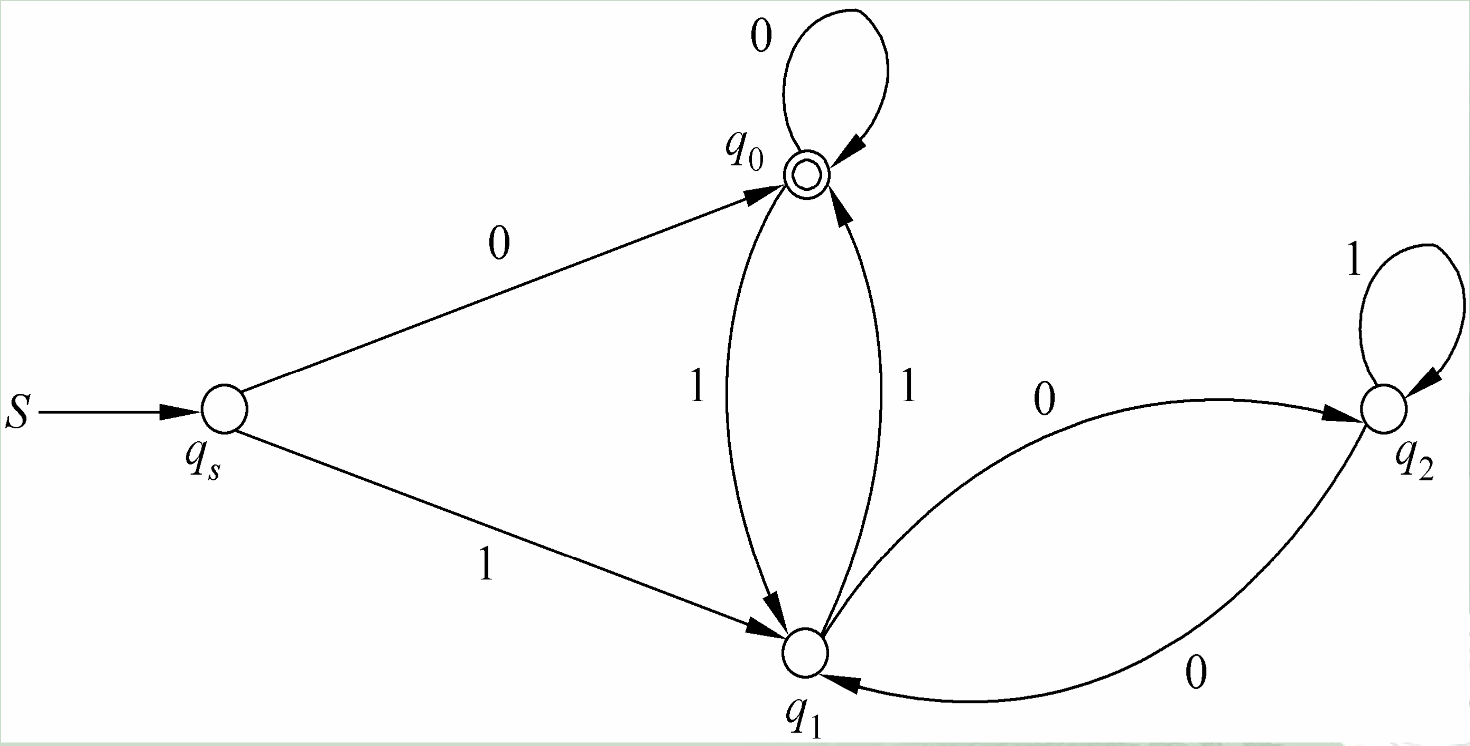
如果$q_0 \in F$，则存在正则文法G'，使得$L(G')=L(G) \cup \{\varepsilon\}=L(M)$。

综上所述，对于任意DFA M，存在正则文法G，使得$L(G)=L(M)$。

定理得证。

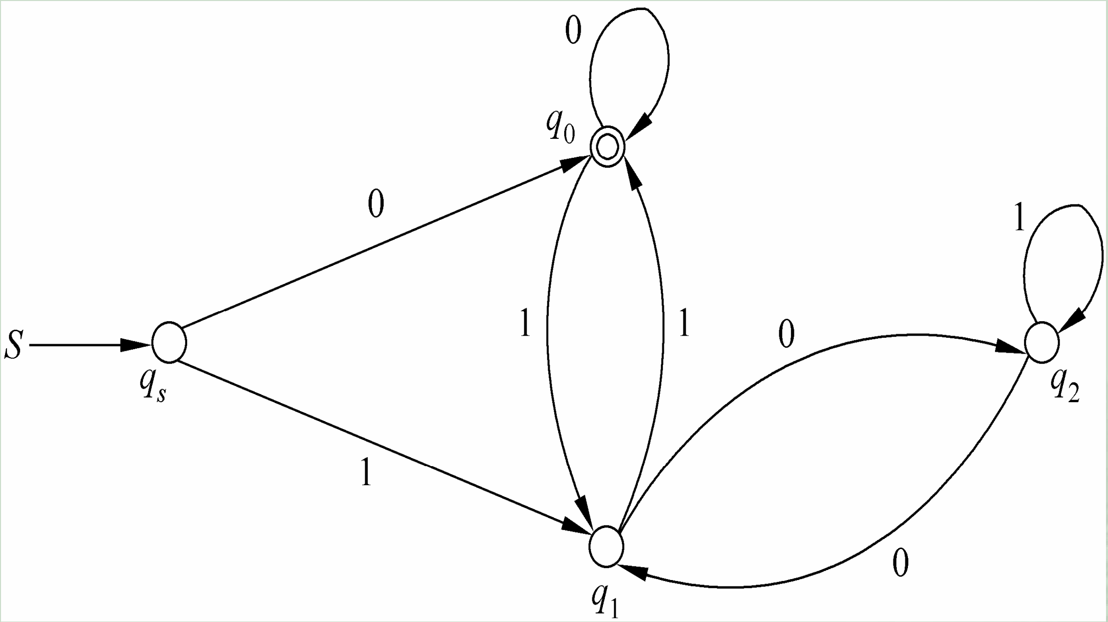
FA与右线性文法

- 例：与下图所给DFA等价的正则文法



FA与右线性文法

- **例：**与下图所给**DFA**等价的正则文法



$$q_s \rightarrow 0|0q_0|1q_1$$

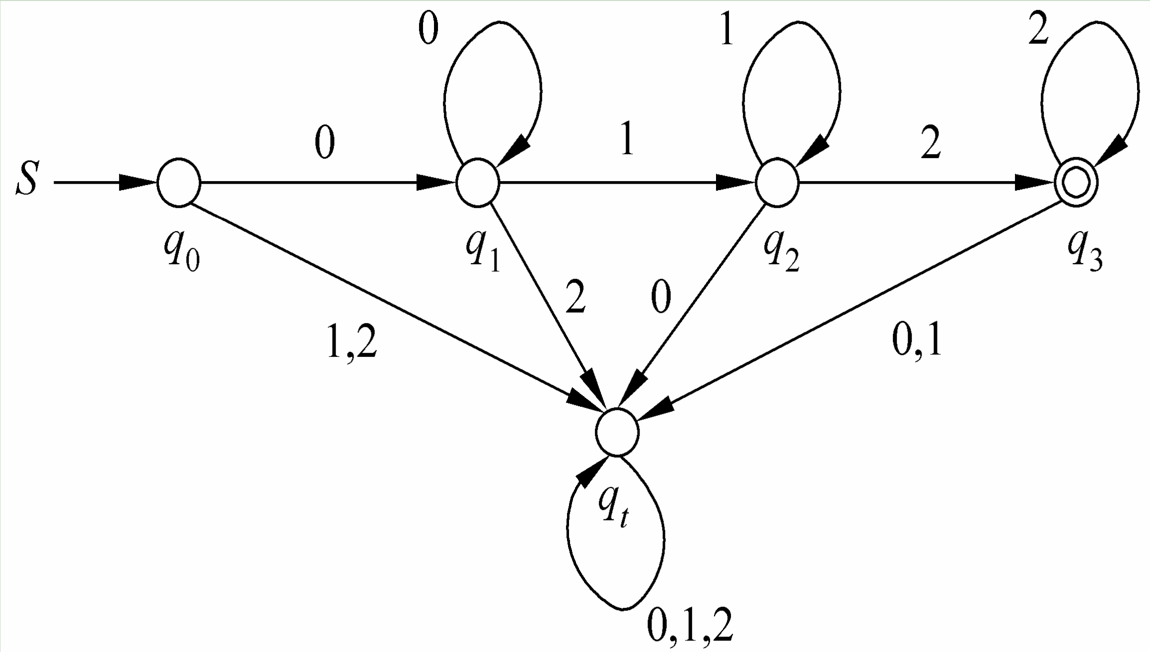
$$q_0 \rightarrow 0|0q_0|1q_1$$

$$q_1 \rightarrow 0q_2|1|1q_0$$

$$q_2 \rightarrow 0q_1|1q_2$$

FA与右线性文法

- 与下图所给的**DFA**等价的正则文法



$q_0 \to 0q_1|1q_t|2q_t$

$q_1 \to 0q_1|1q_2|2q_t$

$q_2 \to 0q_t|1q_2|2q_3|2$

$q_3 \to 0q_t|1q_t|2q_3|2$

$q_t \to 0q_t|1q_t|2q_t$

FA与右线性文法

定理：正则语言可以由**FA**接受。

证明：

（**1**）构造。

基本思想：让**FA**模拟**RG**的推导。

设$G=(V，T，P，S)$，且$\varepsilon \notin L(G)$，

取FA $M=(V\cup\{Z\}，T，\delta，S，\{Z\})$，$Z\notin V$。

FA与右线性文法

- 对∀(a，A)∈T×V

$$\delta(A, a)=\begin{cases}\{B \mid A\rightarrow aB\in P\}\cup\{Z\} & \text{如果} A\rightarrow a\in P \\ \{B \mid A\rightarrow aB\in P\} & \text{如果} A\rightarrow a\notin P\end{cases}$$

用B∈δ(A，a)与产生式A→aB对应

用Z∈δ(A，a)与产生式A→a对应。

FA与右线性文法

（2）证明L(M)=L(G)

对于$a_1a_2\ldots a_{n-1}a_n \in T^+$,

$a_1a_2\ldots a_{n-1}a_n \in L(G) \Leftrightarrow S \Rightarrow^+ a_1a_2\ldots a_{n-1}a_n$

$\Leftrightarrow S \Rightarrow a_1A_1 \Rightarrow a_1a_2A_2 \Rightarrow \ldots$

$\Rightarrow a_1a_2\ldots a_{n-1}A_{n-1} \Rightarrow a_1a_2\ldots a_{n-1}a_n$

$\Leftrightarrow S \rightarrow a_1A_1$，$A_1 \rightarrow a_2A_2$，…，

$A_{n-2} \rightarrow a_{n-1}A_{n-1}$，$A_{n-1} \rightarrow a_n \in P$

FA与右线性文法

$\Leftrightarrow A_1 \in \delta(S，a_1)，A_2 \in \delta(A_1，a_2)，\ldots，$

$A_{n-1} \in \delta(A_{n-2}，a_{n-1})，Z \in \delta(A_{n-1}，a_n)$

$\Leftrightarrow Z \in \delta(S，a_1a_2\ldots a_{n-1}a_n)$

$\Leftrightarrow a_1a_2\ldots a_{n-1}a_n \in L(M)$

对于 ε，按照上文处理。

FA与右线性文法

- **例：构造与所给正则文法等价的FA：**

G_1： $E \to 0A|1B$

$A \to 1|1C$

$B \to 0|0C$

$C \to 0B|1A$

FA与右线性文法

$\delta(E, 0)=\{A\}$ 对应 $E \to 0A$

$\delta(E, 1)=\{B\}$ 对应 $E \to 1B$

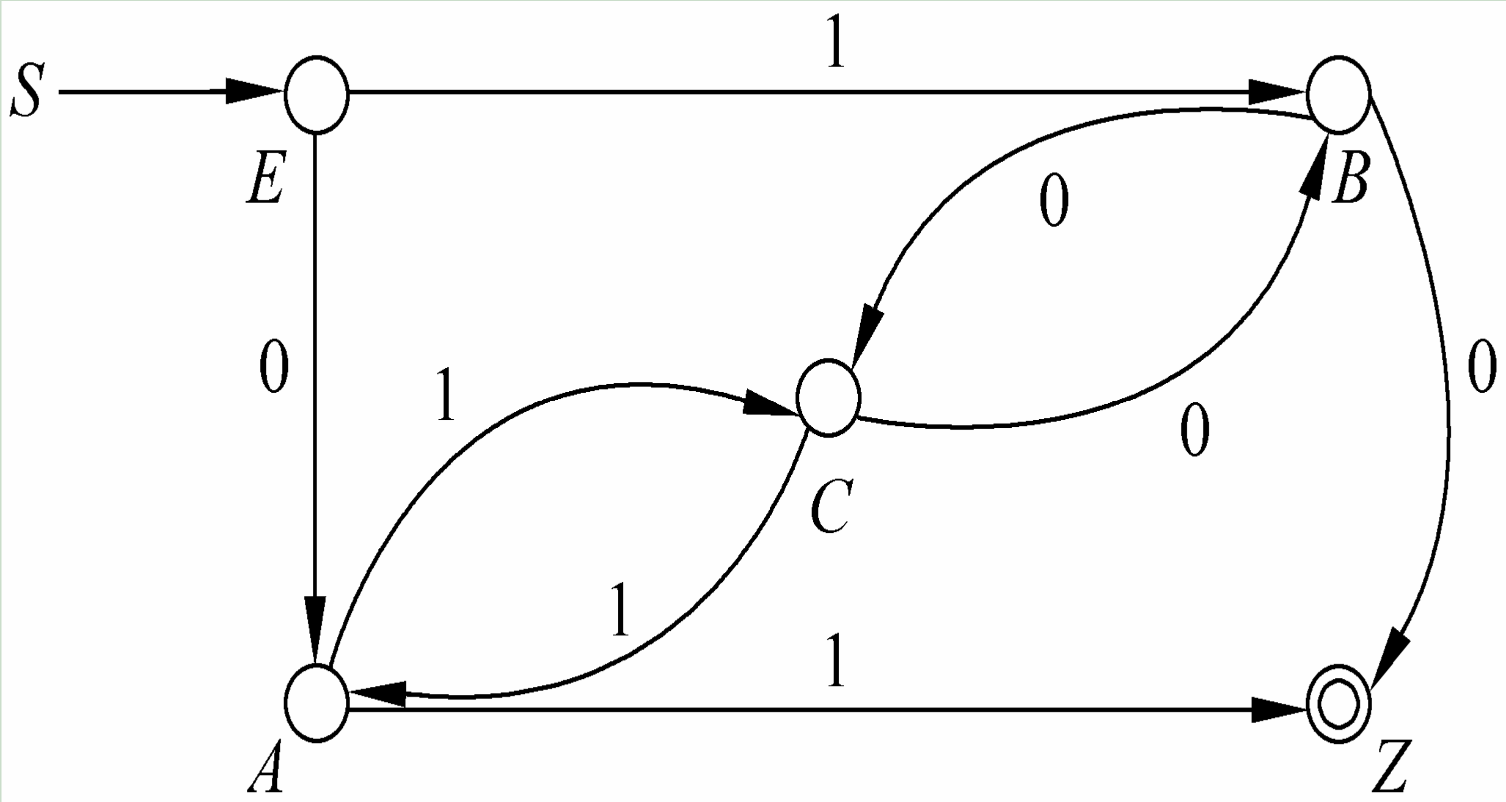
$\delta(A, 1)=\{Z,C\}$ 对应 $A \to 1|1C$

$\delta(B, 0)=\{Z,C\}$ 对应 $B \to 0|0C$

$\delta(C, 0)=\{B\}$ 对应 $C \to 0B$

$\delta(C, 1)=\{A\}$ 对应 $C \to 1A$

FA与右线性文法



FA与右线性文法

推论： FA与正则文法等价。

FA与左线性文法

- 对于左线性文法，按照推导来说，句子**$a_1a_2\ldots a_{n-1}a_n$**中的字符被推导出的先后顺序正好与它们在句子中出现的顺序相反；而按照归约来看，它们被归约成语法变量的顺序则正好与它们在句子中出现的顺序相同：**a_1**，**a_2**，…，**a_{n-1}**，**a_n**。可见，归约过程与**FA**处理句子字符的顺序是一致的，所以可考虑依照“归约”来研究**FA**的构造。

FA与左线性文法

- 对于形如A→a的产生式：在推导中，一旦使用了这样的产生式，句型就变成了句子，而且a是该句子的第一个字符；按“归约”理解，对句子的第1个字符，根据形如A→a的产生式进行归约。对应到FA中，FA从开始状态出发，读到句子的第一个字符a，应将它“归约”为A。我们如果考虑用语法变量对应FA的状态，那么，此时我们需要引入一个开始状态，比如Z。这样，对应形如A→a的产生式，可以定义A∈δ(Z，a)。

FA与左线性文法

- 按照上面的分析，对应于形如**A→Ba**的产生式：**FA**应该在状态**B**读入**a**时，将状态转换到**A**。也可以理解为：在状态**B**，**FA**已经将当前句子的、处理过的前缀“归约”成了**B**，在此时它读入**a**时，要将**Ba**归约成**A**，因此，它进入状态**A**。

FA与左线性文法

- 按照“归约”的说法，如果一个句子是文法G产生的语言的合法句子，它最终应该被归约成文法G的开始符号。所以，G的开始符号对应的状态就是相应的FA的终止状态。
- 如何解决开始符号只有一个，而DFA的终止状态可以有多个的问题。

FA与左线性文法

例如：对文法

G_2： $E \to A0|B1$

$A \to 1|C1$

$B \to 0|C0$

$C \to B0|A1$

对应：

$\delta(A，0)=\{E\}$

$\delta(B，1)=\{E\}$

$\delta(Z，1)=\{A\}$

$\delta(C，1)=\{A\}$

$\delta(Z，0)=\{B\}$

$\delta(C，0)=\{B\}$

$\delta(B，0)=\{C\}$

$\delta(A，1)=\{C\}$

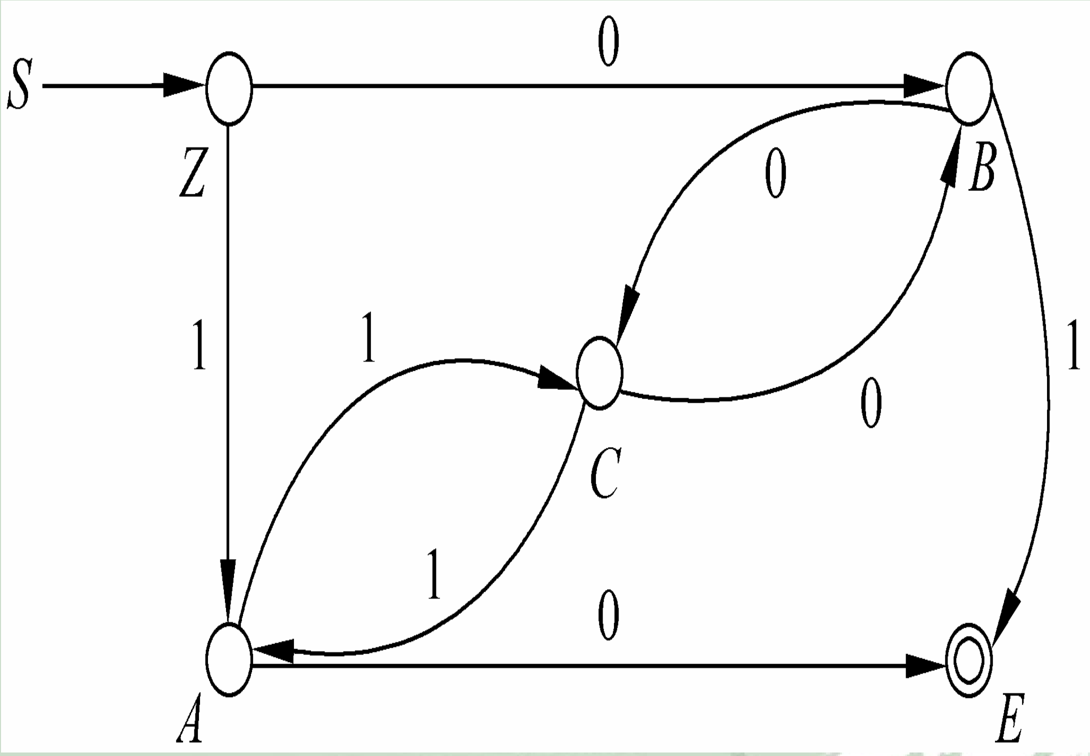
FA与左线性文法

G_2： $E \rightarrow A0|B1$

$A \rightarrow 1|C1$

$B \rightarrow 0|C0$

$C \rightarrow B0|A1$



FA与左线性文法

- **DFA (**的状态转移图**)**作如下“预处理”：

(1) 删除**DFA**的陷阱状态**(**包括与之相关的弧**)**；

(2) 在图中加一个识别状态；

(3) “复制”一条原来到达终止状态的弧，使它从原来的起点出发，到达新添加的识别状态。

FA与左线性文法

(1) 如果 δ (**A，a)=B**，则有产生式**B→Aa**；

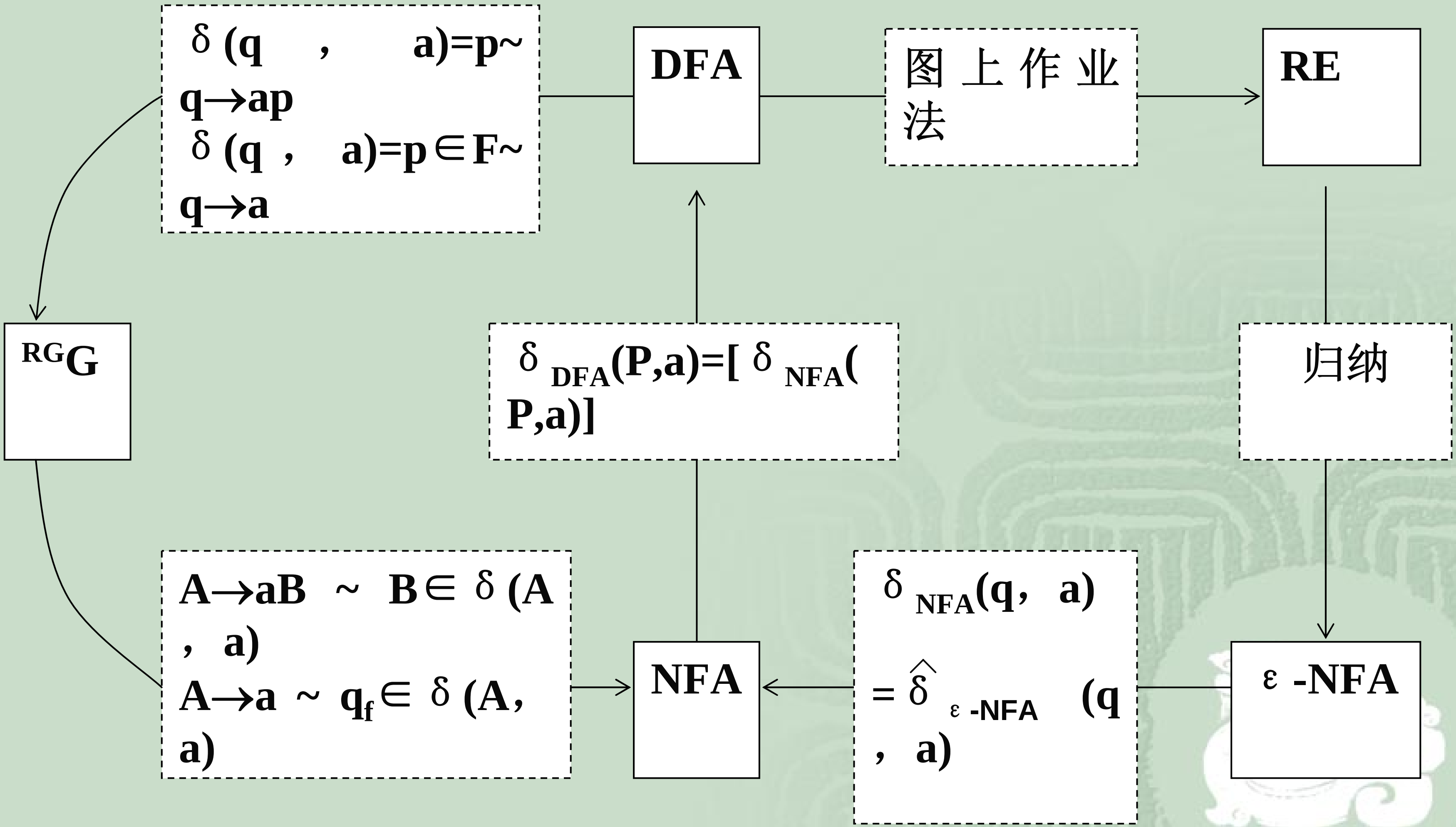
(2) 如果 δ (**A，a)=B**，且**A**是开始状态，则有产生式**B→a**。

定理 左线性文法与**FA**等价。

正则语言等价模型的总结

推论　正则表达式与FA、正则文法等价，是正则语言的表示模型。

正则语言等价模型的总结



习题

1 写出表示下列语言的正则表达式。

(1) $\{0, 1\}^+$。

(2) $\{x \mid x \in \{0, 1\}^+$且x中不含形如00的子串$\}$。

(3) $\{x \mid x \in \{0, 1\}^+$且x中含形如110的子串$\}$。

(4) $\{x \mid x \in \{0, 1\}^+$且x的第十个字符是1$\}$。

(5) $\{x \mid x \in \{0, 1\}^+$且x以1开头以0结尾$\}$。

(6) $\{x \mid x \in \{0, 1\}^+$且x中至少含有两个1$\}$。

习题

2 理解如下正则表达式，说明它们表示的语言。

(1) $(00+11)^+$。

(2) $(1+01+001)^*(\varepsilon+0+00)$ 。

(3) $((0+1)(0+1))^*+((0+1)(0+1)(0+1))^*$ 。

(4) $((0+1)(0+1))^*((0+1)(0+1)(0+1))^*$ 。

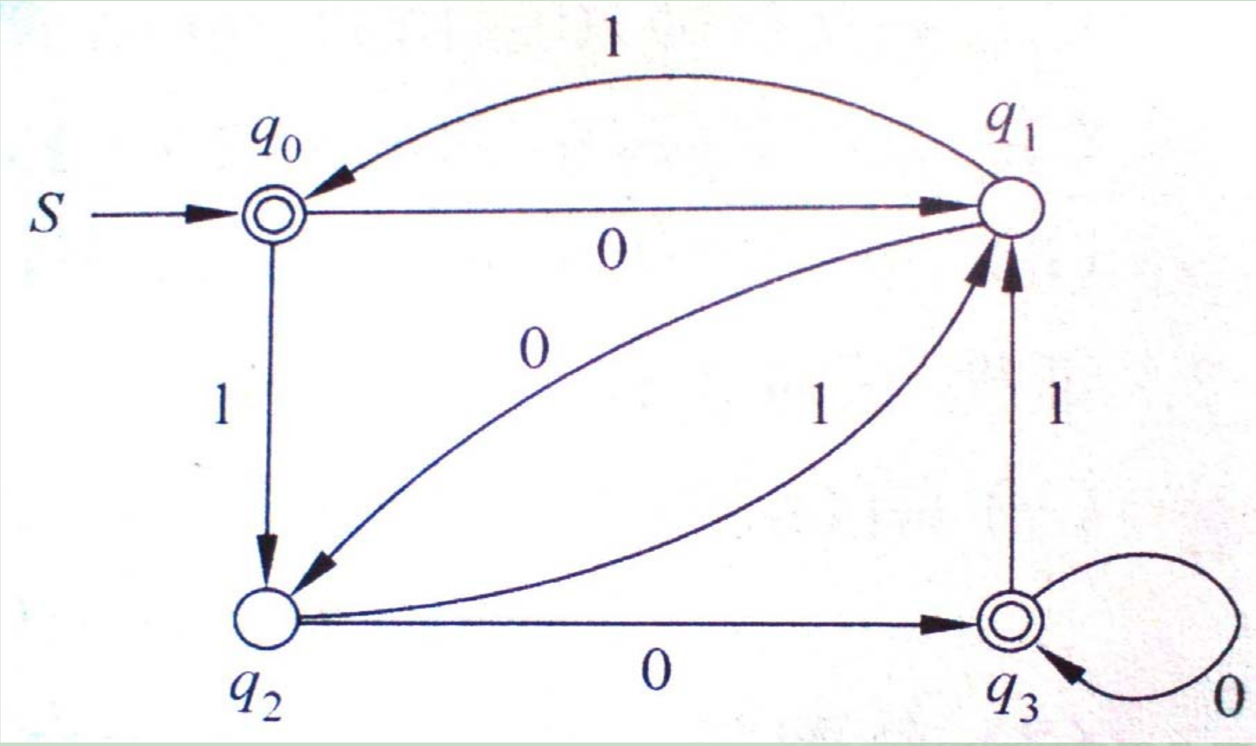
3 构造下列正则表达式的等价FA。

(1) $(1+01+001)^*(\varepsilon+0+00)$。

(2) $((0+1)(0+1))^*+((0+1)(0+1)(0+1))^*$ 。

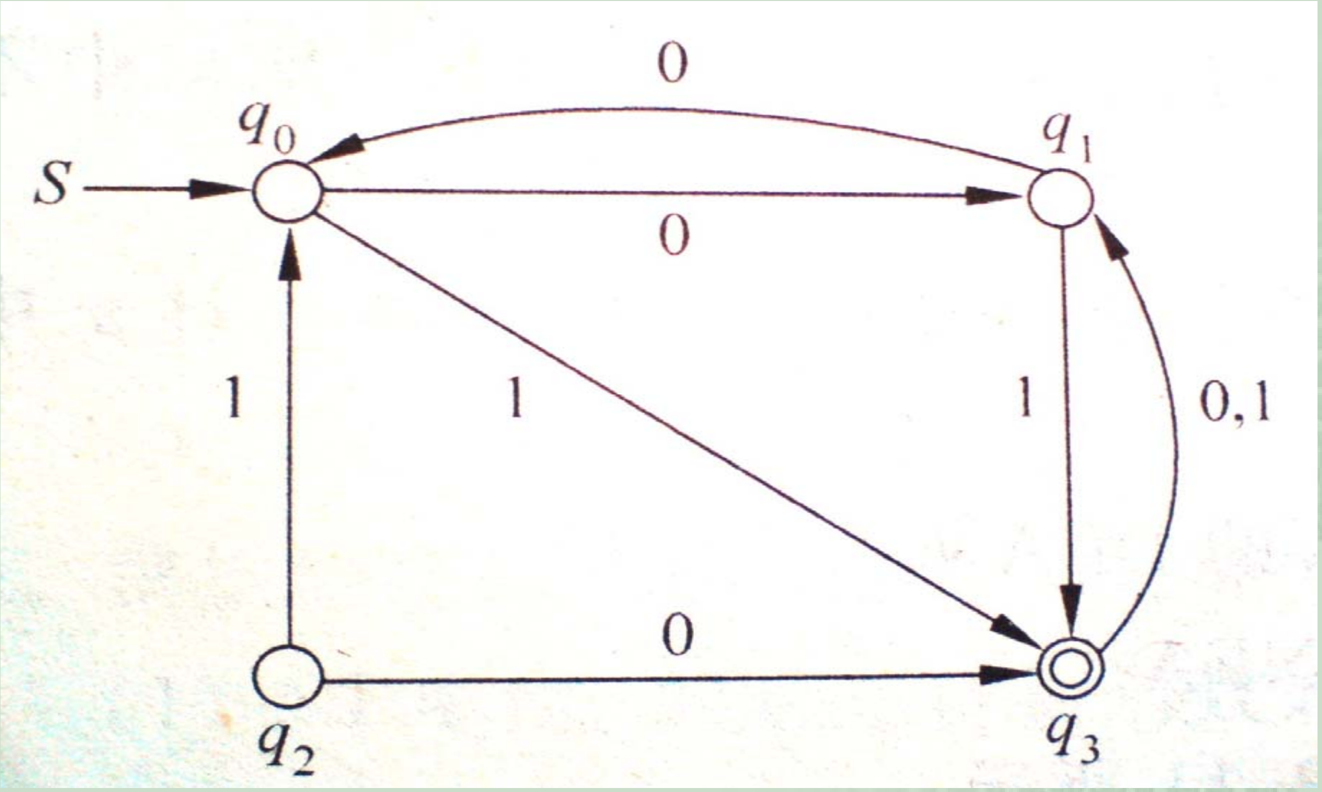
习题

4 构造等价于下图所示DFA的正则表达式。



习题

5 构造下图所示DFA对应的右线性文法和左线性文法。



习题

6 根据下列文法构造相应FA。

(1)G_1： $S \to a|aA$

$A \to a|aA|cA|bB$

$B \to a|b|c|aB|bB|cB$

(2)G_2： $S \to a|Aa$

$A \to a|Aa|Ac|Bb$

$B \to a|b|c|Ba|Bb|Bc$