第二章

有限自动机和正规文法

2.1 确定的有限自动机(DFA)
(Determinate Finite Automaton)

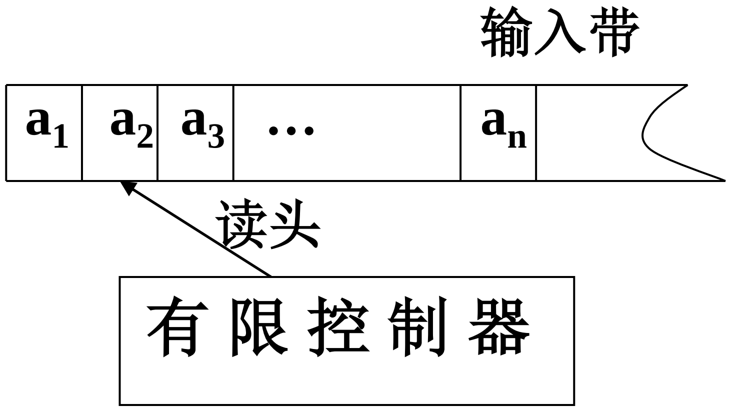
有限自动机是研究自动系统的一种数学模型，它出现于上世纪四十年代。自动机的理论日趋发展，并且与计算机的信息处理密切结合，它不仅用于研究计算机的结构，还用于研究形式语言——语言的识别。

在这里主要是从识别语言这方面来研究自动机。

一.有限自动机(FA)的结构

有限自动机由三部分构成：

1.输入带



输入带可以任意长，带上有若干单元，每个单元内有输入符号。输入带上存放的是被有限自动机识别的符号串。如图所示，

输入带上的符号串为：

$a_1\ a_2\ a_3\ \dots\ a_n$。

2.读头

读头是将输入带上的符号读到有限控制器中，每次读一个单元的符号。

3.有限控制器

有限控制器是有限自动机的核心。

有限自动机有多个状态，有一个开始状态，还有若干个终止状态。

自动机每读带上一个符号，状态可能发生变化，然后读头右移一个单元。

自动机如何从开始状态出发，识别完带上的整个符号串后，要进入某个终止状态，这个过程就是由有限控制器控制的。

二.确定的有限自动机(DFA)的形式描述

定义：确定的有限自动机M写成有序五元组，记作$M=(K,\sum,\delta,q_0,F)$ 其中，

K——有限自动机的状态的有限集合。

$\sum$——输入带上的有穷字母表。

δ——状态转移函数,是 $K\times\sum\rightarrow K$ 的映射。

例如，$\delta(q,a)=p$ (其中$q,p\in K$, $a\in\sum$), 表示在q状态下，读a后，状态改为p ,然后，将读头右移一个单元。

q_0——开始状态 $q_0\in K$

F—— 终止状态集合， $F\subseteq K$

三.确定的有限自动机的表示方法

【例2-1. 1】 给定确定的有限自动机 $M=(K,\sum,\delta,q_0,F)$

$K=\{q_0,q_1,q_2,q_3\}$， $\sum=\{0,1\}$， $F=\{q_0\}$

δ： $\delta(q_0,0)=q_2$ $\delta(q_0,1)=q_1$

$\delta(q_1,0)=q_3$ $\delta(q_1,1)=q_0$

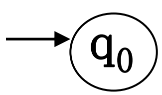
$\delta(q_2,0)=q_0$ $\delta(q_2,1)=q_3$

$\delta(q_3,0)=q_1$ $\delta(q_3,1)=q_2$

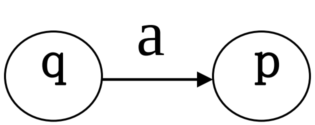
	0	1
q_0	q_2	q_1
q_1	q_3	q_0
q_2	q_0	q_3
q_3	q_1	q_2

状态转移函数δ也可以用函数表表示，如上表所示。

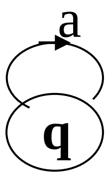
有限自动机还可以用图表示。方法如下：



开始状态q_0
(a)



$\delta(q,a)=p$
(b)

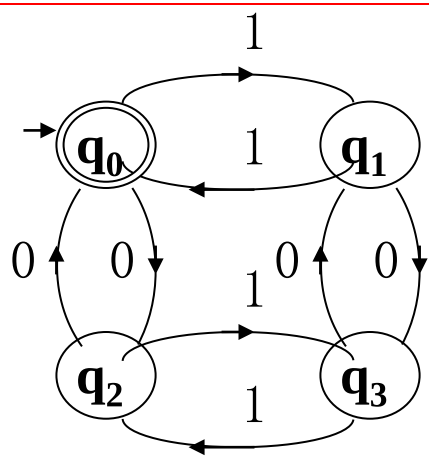


$\delta(q,a)=q$
(c)

$q\in F$
(d)

M的状态转移图:

	0	1
q_0	q_2	q_1
q_1	q_3	q_0
q_2	q_0	q_3
q_3	q_1	q_2



四．状态转移函数δ定义的扩充

原δ： K×∑→K 的映射。

为描述有限自动机M接收的语言，将δ函数扩充成 $\hat{\delta}$

$\hat{\delta}$：K×∑*→K 的映射。对于任何x∈∑*，如果

$\hat{\delta}$(q,x)=p ,表示在状态q下，读符号串x后，到状态p。

一般地表示:

$\hat{\delta}(q,\varepsilon)=q$

$\hat{\delta}(q,xa)=\delta(\hat{\delta}(q,x),a)$ 其中x∈∑*，a∈∑

例如上例中

$$\hat{\delta}(q_0,010)=\delta(\hat{\delta}(q_0,01),0)=\delta(\delta(\hat{\delta}(q_0,0),1),0)$$

$$=\delta(\delta(\delta(\hat{\delta}(q_0,\varepsilon),0),1),0)=\delta(\delta(\delta(q_0,0),1),0)$$

$$=\delta(\delta(q_2,1),0)=\delta(q_3,0)=q_1。$$

可见在确定的有限自动机中，$\hat{\delta}$ 是由δ定义的,为了简单起见，符号δ与 $\hat{\delta}$ 可以不作区分地使用，这样做也不会造成混淆，所以

$\hat{\delta}(q_0,010)=q_1$ 也可以写成 $\delta(q_0,010)=q_1$ 。

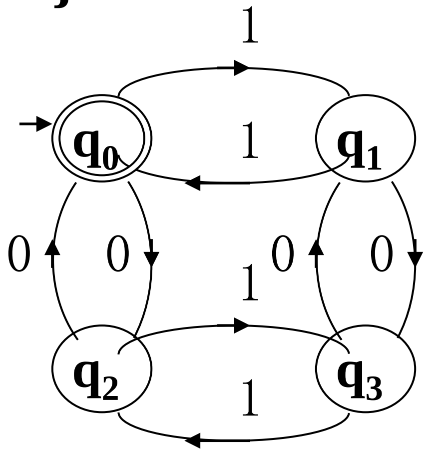
但此时的δ一定理解成 $\hat{\delta}$。

五．确定的有限自动机M接收的语言T(M)

给定确定的有限自动机 **$M=(K,\sum,\delta,q_0,F)$，M接收的语言T(M)为：**

$T(M)=\{w|w\in\sum^* 且\delta(q_0,w)=p 其中p\in F\}$

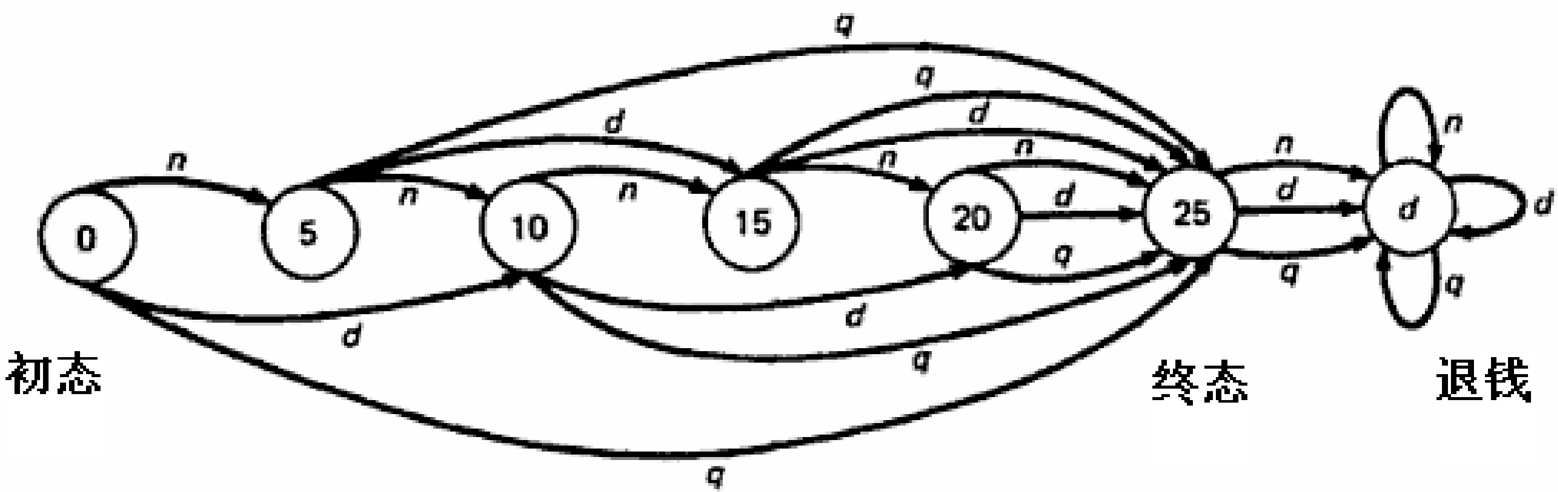
例2-1.1中，T(M)=?



$T(M)=\{w|w\in\{0,1\}^*$ 且w中0和1的个数均为偶数}。

附加一例:糖果售货机

Unlike the previous example, the language accepted by this finite automaton is a finite set. It consists of the following combinations of nickels, dimes, and quarters: {*nnnnn*, *nnnnd*, *nnnnq*, *nnnd*, *nnnq*, *nndn*, *nndd*, *nndq*, *nnq*, *ndnn*, *ndnd*, *ndnq*, *ndd*, *ndq*, *nq*, *dnnn*, *dnnd*, *dnnq*, *dnd*, *dnq*, *ddn*, *ddd*, *ddq*, *dq*, *q*}.



$F = \{25\}$

Figure 3.2

nickel(5美分)，dime（十美分），quarter(25美分）

- 作业题

1. 设计一个有限自动机(FA) M，使得T(M)中的每个句子w同时满足下面三个条件：

   1) $w \in \{a,b,\}^*$；

   2) |w|是3的整数倍；

   3) w以a开头，以b结尾。

2. 设计二个FA M_1和M_2，分别满足

   $T(M_1)=\{0^{2i} \mid i是自然数\}$

   $T(M_2)=\{0^{2i+1} \mid i=0,1,2,3,4,\ldots\}$

2.2 不确定的有限自动机（NFA）
(Non-deterministic Finite Automaton)

DFA是在每个状态下，读一个符号后的下一个状态是唯一确定的，下面讨论的有限自动机是在某个状态下，读一个符号后的下一个状态可能不是唯一确定的，这就是不确定的有限自动机。**例子见LEWIS,P40.**

一.不确定的有限自动机（NFA）的形式定义

定义：不确定的有限自动机**M**，用一个有序五元组表示：

$M=(K,\sum,\delta,q_0,F)$ 其中，

K、∑、q_0、F 的含义同 DFA。

δ—状态转移函数，是 $K\times\sum\rightarrow 2^K$ 的映射。

例. $\delta(q, a)=\{p, s\}$（其中$q\in K$, $\{p, s\}\in 2^K$ $a\in\sum$ ）

设计识别语言L=(ab ∪ aba)*的有限自动机FA。

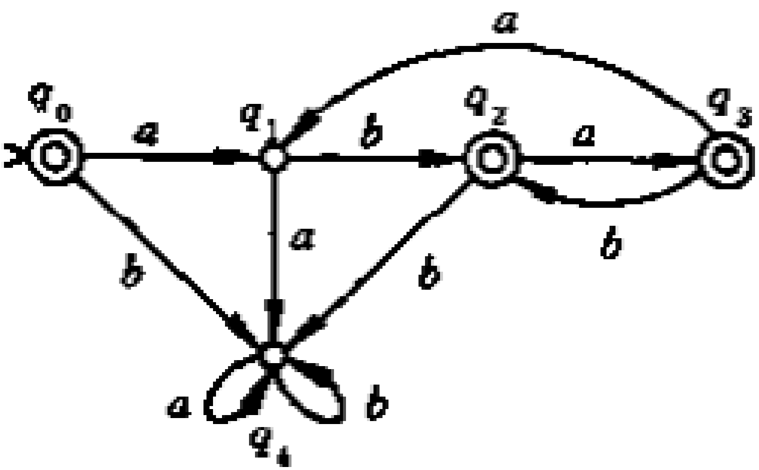


图 2-4

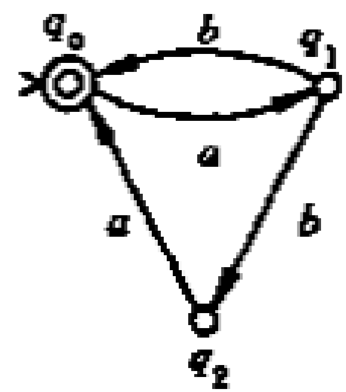


图 2-5

【例2-2.1】给定NFA M，　M=(K,∑,δ,q_0 ,F)

其中，K={q_0,q_1,q_2,q_3,q_4} ∑={0,1}　F={q_2,q_4}

δ：K×∑→2^K 为：

	0	1
q_0	{q_0,q_3}	{q_0,q_1}
q_1	Φ	{q_2}
q_2	{q_2}	{q_2}
q_3	{q_4}	Φ
q_4	{q_4}	{q_4}

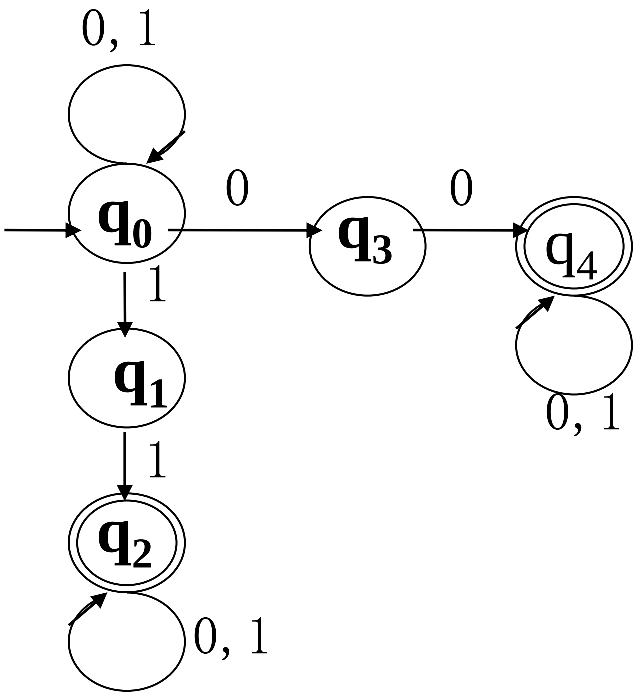


图2-2.1 NFA M状态转移图

(接收含有连续两个0或者连续两个1的字符串构成的语言)

二.状态转移函数δ定义的扩充

原来δ：$K\times\sum\rightarrow2^K$，下面对它进行两次扩充。与确定的有穷自动机相类似，扩充以后的状态转移函数仍然用δ。因为这样做, 在计算时也不会引起错误。

1. 将δ扩充成：δ：$K\times\sum^*\rightarrow2^K$，定义为：$\forall x\in\sum^*$,

$\delta(q,x)=\{p_1,p_2,\ldots,p_n\}$（$\{p_1,p_2,\ldots,p_n\}\in2^K$）

表示在状态q下,读符号串x后,可以达到状态p_i $(1\le i\le n)$。

一般地表示：

$\delta(q,\varepsilon)=\{q\}$　$q\in K$

$\delta(q,xa)=\bigcup_{p\in\delta(q,x)}\delta(p,a)$　其中$q\in K$, $x\in\sum^*$，$a\in\sum$

例2-2.1中 $\delta(q_0,0)=\{q_0,q_3\}$，则

$\delta(q_0,01)=\delta(q_0,1)\cup\delta(q_3,1)=\{q_0,q_1\}\cup\Phi=\{q_0,q_1\}$

为了计算δ(q,xa)方便，

2. 将δ的定义再一次扩充成：δ：$2^K \times \Sigma \rightarrow 2^K$，

对于任何$P=\{p_1,p_2,\ldots,p_n\} \in 2^K$，$a \in \Sigma$

$$\delta(P,a)=\delta(\{p_1,p_2,\ldots,p_n\},a)= \bigcup_{p \in P} \delta(p,a)$$

即δ(P,a)等于P中每个状态分别读a后状态集合的并集。

δ的定义经过扩充之后，计算δ(q,xa)的式子可写成：

$\delta(q,xa)=\delta(\delta(q,x),a)$。

于是对于任何$x \in \Sigma^*, a \in \Sigma$

(1) $\delta(q,\varepsilon)=\{q\} \quad q \in K$

(2) $\delta(q,xa) = \delta(\delta(q,x),a) \quad = \bigcup_{p \in \delta(q,x)} \delta(p,a)$

三. NFA M所接收的语言T(M)

给定NFA $M=(K,\sum,\delta,q_0,F)$, M所接收的语言T(M)为:

$T(M)=\{w|w\in\sum^* 且 \delta(q_0,w)\cap F\neq\Phi\}$

例2-2.1中，

T(M)=?

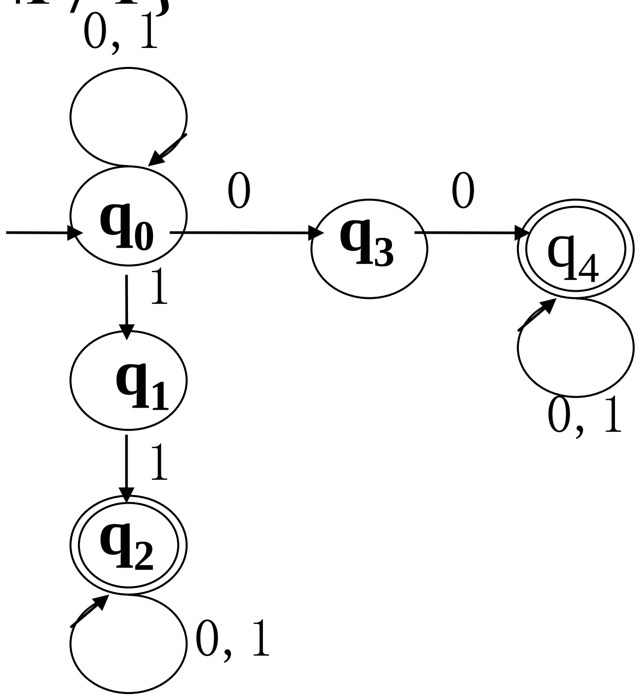


图2-2.1 NFA M状态转移图

$T(M)=\{w|w\in\{0,1\}^*$且w中或者有两个相邻的0或者有两个相邻的1}

四．NFA转换成DFA

任何一个NFA都可以等价地转换成DFA。

定理2-2.1 如果语言L可由一个NFA M所接收，则必存在一个DFA M'，使得T(M')=L。

证明：令NFA $M=(K,\sum,\delta,q_0,F)$，且$T(M)=L$。

构造DFA $M'=(K',\sum,\delta',q_0',F')$, 其中

$K'\subseteq 2^K$，K'中的元素是由K的子集$\{q_1,q_2,\ldots,q_i\}$构成，但是要把子集$\{q_1,q_2,\ldots,q_i\}$作为一个状态看待，因此把此子集写成$[q_1,q_2,\ldots,q_i]$。

$q_0'=[q_0]$,

$F'=\{[q_1,q_2,\ldots,q_i] \mid [q_1,q_2,\ldots,q_i]\in K'$ 且 $\{q_1,q_2,\ldots,q_i\}\cap F\neq\Phi\}$

δ'：$K' \times \sum \rightarrow K'$，对$\forall [q_1,q_2,\ldots,q_i] \in K', \forall a \in \sum$，有

$$\delta'([q_1,q_2,\ldots,q_i],a)=[p_1,p_2,\ldots,p_j]$$

当且仅当

$$\delta(\{q_1,q_2,\ldots,q_i\},a)=\{p_1,p_2,\ldots,p_j\}$$

这说明：计算δ'时，完全取决于NFA中δ

（注：为了更好地理解如何根据NFA M构造DFA M'，可以先看例子，然后再看下面的证明，这样更容易理解证明的过程。）

例. 将例2-2.1中的NFA M等价变换成DFA M'。按照上述定理给定的方法。令$M'=(K',\sum,\delta', q_0',F')$，

其中，K'、F':

因为$K' \subseteq 2^K$，在求K'时，不需要将2^K中的所有元素都列入K',只需要列入从开始状态$[q_0]$可以达到的状态即可,为此可以先求δ'，然后得到K'和F'。

例2-2.1：NFA $M=(K,\sum,\delta,q_0,F)$

其中，$K=\{q_0,q_1,q_2,q_3,q_4\}$

$\sum=\{0,1\}$ $F=\{q_2,q_4\}$

δ:	0	1
q_0	$\{q_0,q_3\}$	$\{q_0,q_1\}$
q_1	Φ	$\{q_2\}$
q_2	$\{q_2\}$	$\{q_2\}$
q_3	$\{q_4\}$	Φ
q_4	$\{q_4\}$	$\{q_4\}$

构造DFA $M'=(K',\sum,\delta',q_0',F')$,

$q_0'=[q_0]$

δ' : $\forall [q_1,q_2,\ldots,q_i]\in K'$, $\forall a\in\sum$,有

$\delta'([p_1,p_2,\ldots,p_j],a)=[q_1,q_2,\ldots,q_n]$

当且仅当

$\delta(\{p_1,p_2,\ldots,p_j\},a)=\{q_1,q_2,\ldots,q_n\}$

从$[q_0]$开始，计算各个可达状态的转移函数。

例如要计算$\delta'([q_0,q_3],0)$ 首先计算$\delta(\{q_0,q_3\},0)$。

$\delta(\{q_0,q_3\},0)=\delta(\{q_0\},0)\cup\delta(\{q_3\},0)=\{q_0,q_3\}\cup\{q_4\}$

$=\{q0,q3,q4\}$，于是

$\delta'([q_0,q_3],0)=[q_0,q_3,q_4]$,其余的依此类推。 最后得下表

δ:	0	1
q_0	$\{q_0,q_3\}$	$\{q_0,q_1\}$
q_1	Φ	$\{q_2\}$
q_2	$\{q_2\}$	$\{q_2\}$
q_3	$\{q_4\}$	Φ
q_4	$\{q_4\}$	$\{q_4\}$

δ’:	0	1
$[q_0]$	$[q_0,q_3]$	$[q_0,q_1]$
$[q_0,q_3]$	$[q_0,q_3,q_4]$	$[q_0,q_1]$
$[q_0,q_1]$	$[q_0,q_3]$	$[q_0,q_1,q_2]$
$[q_0,q_3,q_4]$	$[q_0,q_3,q_4]$	$[q_0,q_1,q_4]$
$[q_0,q_1,q_2]$	$[q_0,q_2,q_3]$	$[q_0,q_1,q_2]$
$[q_0,q_2,q_3]$	$[q_0,q_2,q_3,q_4]$	$[q_0,q_1,q_2]$
$[q_0,q_1,q_4]$	$[q_0,q_3,q_4]$	$[q_0,q_1,q_2,q_4]$
$[q_0,q_2,q_3,q_4]$	$[q_0,q_2,q_3,q_4]$	$[q_0,q_1,q_2,q_4]$
$[q_0,q_1,q_2,q_4]$	$[q_0,q_2,q_3,q_4]$	$[q_0,q_1,q_2,q_4]$

K’=$\{[q_0],[q_0,q_3],[q_0,q_1],[q_0,q_3,q_4],[q_0,q_1,q_2],[q_0,q_2,q_3],$
$[q_0,q_1,q_4],[q_0,q_2,q_3,q_4],[q_0,q_1,q_2,q_4],\}$

F’=$\{[q_0,q_3,q_4],[q_0,q_1,q_2],[q_0,q_2,q_3],[q_0,q_1,q_4],[q_0,q_2,q_3,q_4],$
$[q_0,q_1,q_2,q_4]\}$

M’的图：

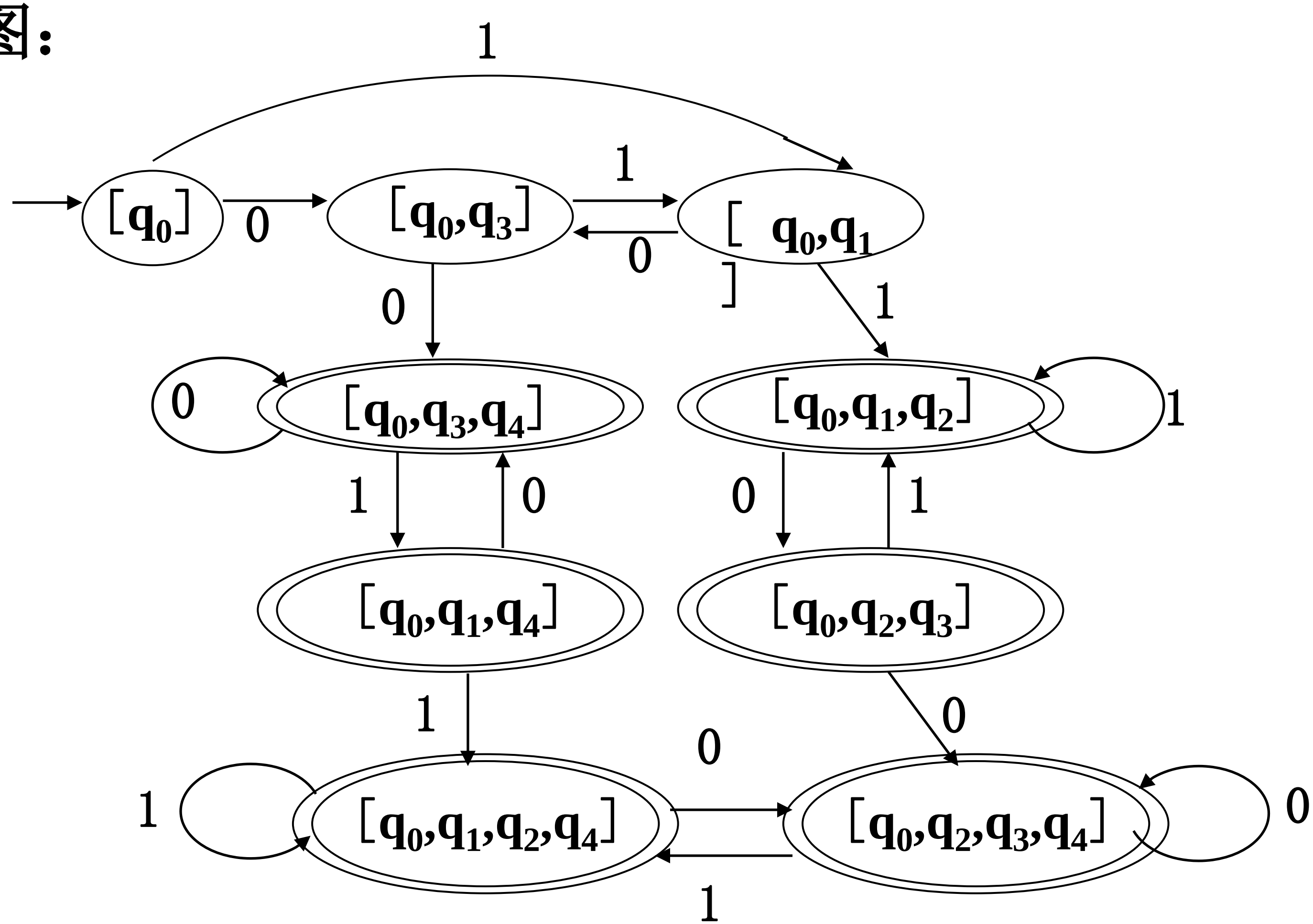


图2-2.2 M’的状态转移图

证明：T(M') = T(M) { goto 274 }

1.先用归纳法证明(对输入符号串|x|归纳)下面命题成立：对于任何$x\in\sum^*$，

$\delta'(q_0',x)=[q_1,q_2,\ldots,q_n]$ 当且仅当 $\delta(q_0,x)=\{q_1,q_2,\ldots,q_n\}$

(1) 当|x|=0，即x=ε时，$\delta'(q_0',\varepsilon)=q_0'=[q_0]$ 当且仅当 $\delta(q_0,\varepsilon)=\{q_0\}$，命题成立。

(2) 假设$|x|\le k$时，命题成立。即

$\delta'(q_0',x)=[p_1,p_2,\ldots,p_j]$ 当且仅当$\delta(q_0,x)=\{p_1,p_2,\ldots,p_j\}$

(3) 当|xa|=k+1 时，$a\in\sum$,有

$\delta'(q_0',xa)=\delta'(\delta'(q_0',x),a)$

$\delta(q_0,xa)=\delta(\delta(q_0,x),a)$

因为|x|=k,由假设(2)得$\delta'(q_0',x)=[p_1,p_2,\ldots,p_j]$ 当且仅当 $\delta(q_0,x)=\{p_1,p_2,\ldots,p_j\}$。故$\delta'(q_0',xa)=\delta'([p_1,p_2,\ldots,p_j],a)$ 当且仅当 $\delta(q_0,xa)=\delta(\{p_1,p_2,\ldots,p_j\},a)$，所以命题成立。

2. 再证明$T(M') = T(M)$

对于任何$x \in \Sigma^*$, 如果$x \in T(M')$，则$\delta'(q_0',x) \in F'$，令 $\delta'(q_0',x)=[p_1,p_2,\ldots,p_j]$，即$[p_1,p_2,\ldots,p_j] \in F'$。

因命题 $\delta'(q_0',x)=[p_1,p_2,\ldots,p_j]$ 当且仅当 $\delta(q_0,x)=\{p_1,p_2,\ldots,p_j\}$ 成立。

由F'定义得 $[p_1,p_2,\ldots,p_j] \in F'$ 当且仅当 $\{p_1,p_2,\ldots,p_j\} \cap F \neq \Phi$，即$\delta(q_0,x) \cap F \neq \Phi$。

于是$x \in T(M)$。

所以$T(M')=T(M)$ 。

定理证明完毕。

从此定理看出:与DFA相比，NFA并没有扩大它所接收语言的范围。

作业题

给定NFA $M_1=(\{p,q,r,s\},\{0,1\},\delta,p,\{s\})$，如下表所示。构造一个DFA M_2，使得 $T(M_1)=T(M_2)$ 。

δ	0	1
p	{p,q}	{p}
q	{r }	{r}
r	{s}	Φ
s	{s}	{s}

2.3 具有ε转移的NFA

（简记成ε-NFA）

前边讨论的NFA中的输入符号只是∑中的符号，在理论上和实际应用中还有另一种NFA，就是输入符号中除了∑中的符号以外，还允许有ε，这就是具有ε转移的NFA(ε-NFA)。例子见LEWIS-P41(PPT下一页).

【例2-3.1】具有ε转移的NFA M(ε-NFA M)，如图2-3.1所示。从图中看出，图中有的有向边上标有ε。

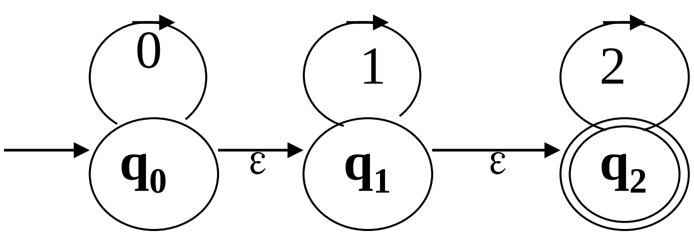


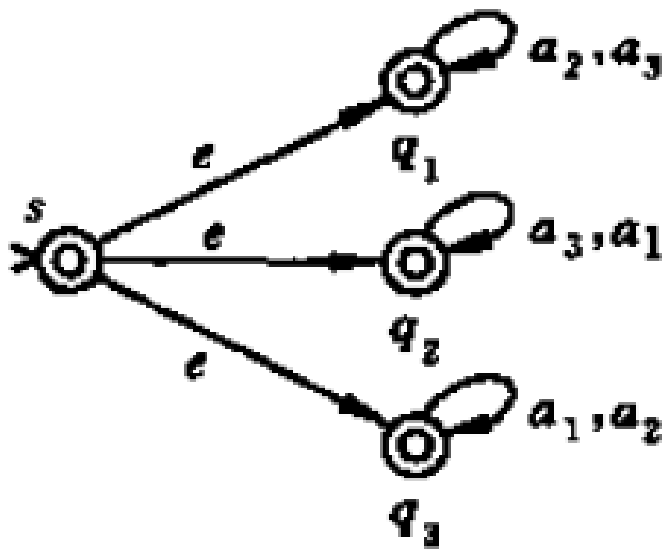
图2-3.1 ε-NFA M图

例 2.2.2 设字母表 $\Sigma=\{a_1,\cdots,a_n\}$,其中 $n\geq 2$。考虑下述语言

$$L=\{w: 有一个符号 a_i\in\Sigma 不出现在 w 中\}$$

即 L 包括 Σ^* 中 Σ 的符号不全出现的所有字符串。例如,如果 $n=3$,则 $e,a_1,a_2,a_1a_1a_3a_1\in L$,而$a_3a_1a_3a_1a_2\notin L$。

设计一台接受这个相当复杂的语言的非确定型有穷自动机相对说来是容易的。这里



DFA、NFA、ε-NFA、正规表达式(RE)之间的等价性

DFA、NFA、ε-NFA、正规表达式(RE)之间的等价链，可以用下面图2-4.1表示。

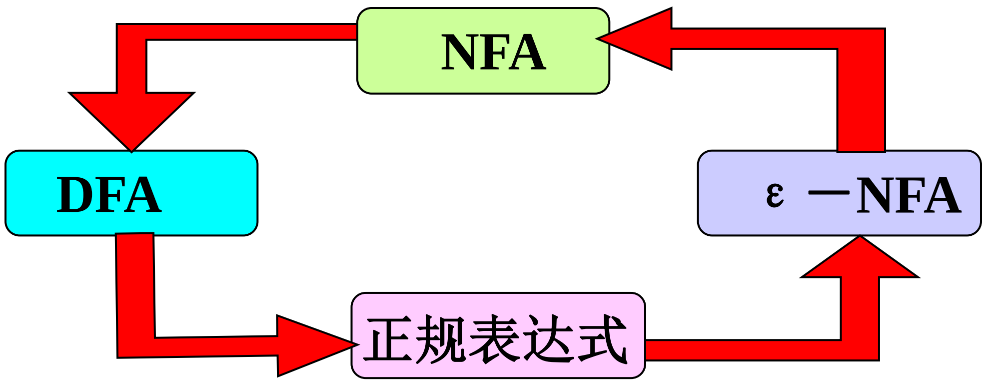


图2-4.1 有限自动机与正规表达式之间的等价转换关系图

一．ε-NFA的形式定义

ε-NFA M=(K,∑,δ,q_0 ,F)，其中K,∑,q_0 ,F的含义同前边NFA, δ: K×(∑∪{ε})→2^K

　例2-3.1中的ε-NFA M 的δ如表2-3.1所示。

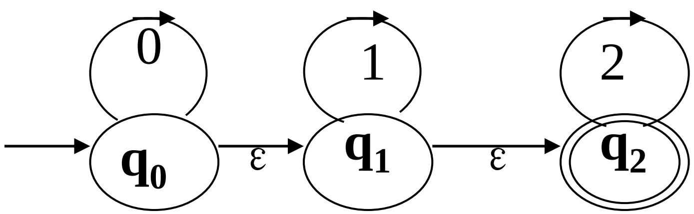


图2-3.1 ε -NFA M图

	0	1	2	ε
q_0	$\{q_0\}$	Φ	Φ	$\{q_1\}$
q_1	Φ	$\{q_1\}$	Φ	$\{q_2\}$
q_2	Φ	Φ	$\{q_2\}$	Φ

表2-3.1 ε -NFA M状态转移表

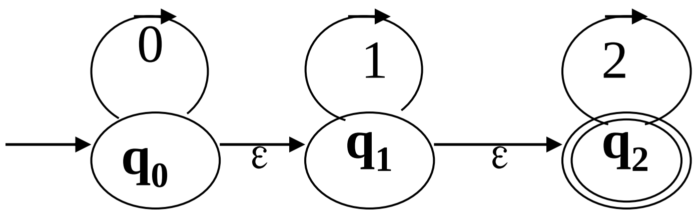
为了讨论ε-NFA 接收的语言需要对δ的定义进行扩充。

二．δ的定义扩充

先扩充成 $\hat{\delta}$：$K\times\Sigma^*\rightarrow 2^K$,定义为,对于$\forall q\in K,\forall w\in\Sigma^*$,

$\hat{\delta}(q,w)=P=\{p|$ 从q出发，沿着标有w的路径达到状态p$\}$

值得注意:在DFA和NFA中,对于扩充后的 $\hat{\delta}$ 与 δ 可以不加区别的使用，因为它们的计算结果相同。但是在这里就必须加以区别,因为它们的计算结果不同了。请看下例:



在ε-NFA中，在例2-3.1中

图2-3.1 ε -NFA M图

$\hat{\delta}(q_0,\varepsilon)=\{q_0,q_1,q_2\}\neq\delta(q_0,\varepsilon)=\{q_1\}$，

因为这里的ε指的是ε路径 ε *，而不是边上标的字符ε。

$\hat{\delta}(q_0,0)=\{q_0,q_1,q_2\}\neq\delta(q_0,0)=\{q_0\}$, 因为这里的0指的是0路径(即，ε *0 ε *, 在0的前后可能有若干个ε)，而不是输入符号0。

为了写出 $\hat{\delta}$ (q,w)的计算公式，下面再定义(特例)两个概念。

三．ε-CLOSURE(q)　 $q \in K$,

ε-CLOSURE(q)={p|从q出发,沿着ε路径达到状态p}

上例中，ε-CLOSURE(q_0)={q_0,q_1,q_2},

ε-CLOSURE(q_1)={q_1,q_2},

ε-CLOSURE(q_2)={q_2}

四.ε-CLOSURE(P)（其中P是状态的集合）

P是状态的集合，则

$$\varepsilon\text{-CLOSURE}(P)= \bigcup_{q \in P} \varepsilon\text{-CLOSURE}(q)$$

上例中，ε-CLOSURE({q_0,q_1})

=ε-CLOSURE(q_0)∪ε-CLOSURE(q_1)

={q_0,q_1,q_2} ∪ {q_1,q_2}={q_0,q_1,q_2}

五.对δ的定义的再扩充

将δ的定义扩充成δ：$2^K \times \Sigma \rightarrow 2^K$，

对$\forall R \in 2^K$， $\forall a \in \Sigma$，$\delta(R,a)=\bigcup_{r \in R} \delta(r,a)$

再扩充成 $\hat{\delta}$：$2^K \times \Sigma^* \rightarrow 2^K$，

对$\forall R \in 2^K$, $\forall w \in \Sigma^*$, $\hat{\delta}(R,w)=\bigcup_{r \in R} \hat{\delta}(r,w)$

六.$\hat{\delta}$ 的计算公式

基 础：对于$\forall q \in K$, $\hat{\delta}(q,\varepsilon)=\varepsilon\text{-CLOSURE}(q)$

长度加一：对于$\forall q \in K$, $\forall w \in \Sigma^*$, $\forall a \in \Sigma$,

$$\hat{\delta}(q,wa)=\varepsilon\text{-CLOSURE}(\delta(\hat{\delta}(q,w),a))$$

注意，$\hat{\delta}(q,wa) \neq \delta(\hat{\delta}(q,w),a)$，因路径wa,是指$\varepsilon^* w \varepsilon^* a \varepsilon^*$，

所以要在 $\delta(\hat{\delta}(q,w),a)$前边加ε-CLOSURE。

例如,在上例中,计算 $\hat{\delta}(q_0,01)$,按照递归地定义公式(2),要先计算：

$\hat{\delta}(q_0,\varepsilon)=\{q_0,q_1,q_2\}$, 再计算：

$$\hat{\delta}(q_0,0)=\varepsilon\text{-CLOSURE}(\delta(\hat{\delta}(q_0,\varepsilon),0))$$

$$=\varepsilon\text{-CLOSURE}(\delta(\{q_0,q_1,q_2\},0))$$

$$=\varepsilon\text{-CLOSURE}(\delta(q_0,0)\cup\delta(q_1,0)\cup\delta(q_2,0))$$

$$=\varepsilon\text{-CLOSURE}(\{q_0\}\cup\Phi\cup\Phi)$$

$$=\varepsilon\text{-CLOSURE}(\{q_0\})=\{q_0,q_1,q_2\}$$

最后计算

$$\hat{\delta}(q_0,01)=\varepsilon\text{-CLOSURE}(\delta(\hat{\delta}(q_0,0),1)$$

$$=\varepsilon\text{-CLOSURE}(\delta(\{q_0,q_1,q_2\},1))$$

$$=\varepsilon\text{-CLOSURE}(\delta(q_0,1)\cup\delta(q_1,1)\cup\delta(q_2,1))$$

$$=\varepsilon\text{-CLOSURE}(\Phi\cup\{q_1\}\cup\Phi)$$

$$=\varepsilon\text{-CLOSURE}(\{q_1\})$$

$$=\{q_1,q_2\}$$

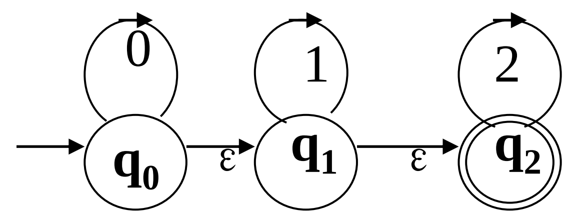


图2-3.1 ε-NFA M图

七.ε-NFA M接收的语言T(M)

给定一个ε-NFA $M=(K,\sum,\delta,q_0,F)$，它接收的语言T(M)为：

注意，ε-NFA识别语言是对字符串操作的。

$$T(M)=\{w|w\in\sum^* 且 \ \hat{\delta}(q_0,w)\cap F\neq\Phi\}$$

上例中 $T(M)=\{0^i1^j2^k|i,j,k\geq 0\}$

八.ε-NFA等价地转换为NFA

一个ε-NFA M可以等价地转换为没有ε转移的NFA。下面介绍定理。

定理2-3.1 如果语言L由一个有ε转移的NFA M所接收，则L可被一个不具有ε转移的NFA M'所接收。

证明： 令一个ε-NFA **M**=(K,∑,δ,q_0,F)，它接收的语言T(M)为L。构造一个不具有ε转移的NFA **M'**

M'=(K, ∑, δ',q_0,F')，其中

$$F' = \begin{cases} F \cup \{q_0\} & \text{如果}\ \varepsilon\text{—CLOUSURE}(q_0) \cap F \neq \Phi \\ F & \text{否则} \end{cases}$$

δ'：对任何q∈K,任何a∈∑, δ'(q,a)= $\hat{\delta}$(q,a)

下面证明T(M')=T(M)。

先用归纳法(任何x∈$\sum^+$，对|x|归纳)证明下面命题成立：

对于任何x∈$\sum^+$,有δ'(q_0,x)= $\hat{\delta}$(q_0,x)，(当讨论x=ε时,不用此结论)

(1) 当|x|=1时，设x=a，a∈∑，根据δ' 的定义，有

δ'(q_0,a)= $\hat{\delta}$(q_0,a)，命题成立。

(2) 假设|x|≤k时,命题成立。即 δ'(q_0,x)= $\hat{\delta}$(q_0,x)。

(本页来自P29,为方便对照)

先扩充成 $\hat{\delta}$：$K\times\Sigma^{*}\rightarrow 2^{K}$,定义为,对于$\forall q\in K,\forall w\in\Sigma^{*}$,

$\hat{\delta}(q,w)=P=\{p|$ 从q出发，沿着标有w的路径达到状态$p\}$

值得注意:在DFA和NFA中,对于扩充后的 $\hat{\delta}$ 与 δ 可以不加区别的使用，因为它们的计算结果相同。但是在这里就必须加以区别,因为它们的计算结果不同了。请看下例:

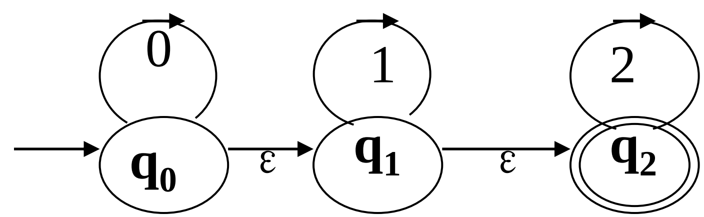


图2-3.1 ε -NFA M图

$\hat{\delta}$	0	1	2
q_0	$\{q_0,q_1,q_2\}$	$\{q_1,q_2\}$	$\{q_2\}$
q_1	Φ	$\{q_1,q_2\}$	$\{q_2\}$
q_2	Φ	Φ	$\{q_2\}$

δ	0	1	2	ε
q_0	$\{q_0\}$	Φ	Φ	$\{q_1\}$
q_1	Φ	$\{q_1\}$	Φ	$\{q_2\}$
q_2	Φ	Φ	$\{q_2\}$	Φ

表2-3.1 ε -NFA M状态转移表

(3). 当$|xa|=k+1$时，令输入符号串xa, $(x\in\sum^{+},|x|=k,a\in\sum)$,

根据归纳假设有$\delta'(q_0,x)=\hat{\delta}(q_0,x)$。

因为$\delta'(q_0,xa)=\delta'(\delta'(q_0,x),a)$,

$$=\bigcup_{r\in\delta'(q_0,x)}\delta'(r,a)$$

$$=\bigcup_{r\in\delta'(q_0,x)}\hat{\delta}(r,a)$$ （根据δ'的定义）

$=\hat{\delta}(\delta'(q_0,x),a)$ (根据$\hat{\delta}$的扩充定义)

$=\hat{\delta}(\hat{\delta}(q_0,x),a)$ (根据假设（2）)

$=\hat{\delta}(q_0,xa)$

综上所述，上述命题成立。

先证T(M)⊆T(M’)

任取x∈T(M)，

若x＝ε，则ε-CLOSURE(q_0)∩F ≠Φ，由F’的构成得

F’=F∪{q_0}，而δ’ (q_0,ε)＝{q_0}, 所以δ’(q_0,ε)∩F’≠Φ，所以

ε∈T(M’)。

若x≠ε，则 $\hat{\delta}(q_0,x)$∩F≠Φ，由已经证明的命题得

δ’(q_0,x)∩F≠Φ，由F’定义得F⊆F’，所以

δ’(q_0,x)∩F’≠Φ，所以 x∈T(M’)。

所以T(M)⊆T(M’)。

再证T(M’) ⊆T(M)

任取x∈T(M’),下面分两种情况

1) 如果x=ε，因M’是无ε的NFA，必有

δ’(q_0,ε)={q_0}⊆F’，所以q_0∈F’，再分两种情况讨论:

(1) 若$q_0 \in F$，又由ε-CLOSURE定义有$q_0 \in$ ε-CLOSURE(q_0),

所以 ε-CLOSURE(q_0)$\cap F \neq \Phi$, 而

ε-CLOSURE(q_0)＝ $\hat{\delta}(q_0,\varepsilon)$，

即$\hat{\delta}(q_0,\varepsilon) \cap F \neq \Phi$，所以$\varepsilon \in T(M)$。

(2) 若$q_0 \notin F$，而$q_0 \in F'$，这说明$F' = F \cup \{q_0\}$，即说明

ε-CLOSURE(q_0)$\cap F \neq \Phi$，即 $\hat{\delta}(q_0, \varepsilon) \cap F \neq \Phi$，所以

$\varepsilon \in T(M)$。

2) 如果$x \neq \varepsilon$，分两种情况讨论

(1). 如果$F' = F$，因为$\delta'(q_0,x) \cap F' \neq \Phi$，由上述命题知

$\delta'(q_0,x) = \hat{\delta}(q_0,x)$，所以必有 $\hat{\delta}(q_0,x) \cap F \neq \Phi$，所以

$x \in T(M)$。

(2) 如果$F'=F\cup\{q_0\}$，又已知$\delta'(q_0,x)\cap F'\neq\Phi$，这又有两种可能：

(a) 若$q_0\notin\delta'(q_0,x)$，则有

$$\begin{aligned}\delta'(q_0,x)\cap F' &= \delta'(q_0,x)\cap(F\cup\{q_0\})\\ &= \delta'((q_0,x)\cap F)\cup(\delta'(q_0,x)\cap\{q_0\})\\ &= (\delta'(q_0,x)\cap F)\cup\Phi\\ &= \delta'(q_0,x)\cap F \neq\Phi\end{aligned}$$

即 $\hat{\delta}(q_0,x)\cap F\neq\Phi$，所以$x\in T(M)$。

(b) 若$q_0\in\delta'(q_0,x)$，

(b) 若$q_0 \in \delta'(q_0,x)$，因为 $\hat{\delta}(q_0,x)= \delta'(q_0,x)$，

所以$q_0 \in \hat{\delta}(q_0,x)$，说明在M中从$q_0$出发沿着x路径可以回到$q_0$，进而也可以达到ε-CLOSURE ($q_0$) 中的各个状态，因而有

$\varepsilon\text{-CLOSURE}(q_0) \subseteq \delta'(q_0,x)$，

又$F'=F\cup\{q_0\}$，由F'定义知

$\varepsilon\text{-CLOSURE}(q_0)\cap F\neq\Phi$，于是

$\delta'(q_0,x)\cap F\neq\Phi$，所以

$\delta'(q_0,x)\cap F'\neq\Phi$，所以$x\in T(M)$。

综上所述，有$T(M')\subseteq T(M)$。

最后得$T(M')=T(M)$。

例.下面将例2-3.1中的ε-NFA M等价变换成NFA M’。

令M’=(K,∑, δ’, q_0,F’)，其中K={q_0,q_1,q_2},

∑={0,1,2},F’={q_0,q_2}, δ’的计算如下：

$\delta'(q_0,0)=\hat{\delta}(q_0,0)=\{q_0,q_1,q_2\}$

$\delta'(q_0,1)=\hat{\delta}(q_0,1)$

$=\varepsilon\text{-CLOSURE}(\delta(\hat{\delta}(q_0,\varepsilon),1))$

$=\varepsilon\text{-CLOSURE}(\delta(\{q_0,q_1,q_2\},1))$

$=\varepsilon\text{-CLOSURE}(\delta(q_0,1)\cup\delta(q_1,1)\cup\delta(q_2,1))$

$=\varepsilon\text{-CLOSURE}(\Phi\cup\{q_1\}\cup\Phi)$

$=\varepsilon\text{-CLOSURE}(\{q_1\})=\{q_1,q_2\}$

其它的依此类推。

最后得δ’的表2-3.2。

M’的图如图2-3.2所示。

δ’:	0	1	2
q_0	$\{q_0,q_1,q_2\}$	$\{q_1,q_2\}$	$\{q_2\}$
q_1	Φ	$\{q_1,q_2\}$	$\{q_2\}$
q_2	Φ	Φ	$\{q_2\}$

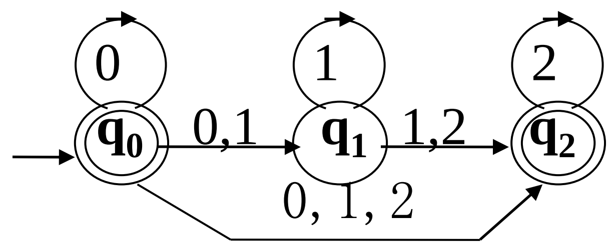


图2-3.2 NFA M’图

(本页来自P29,为方便对照)

先扩充成 $\hat{\delta}$：$K\times\Sigma^*\rightarrow 2^K$,定义为,对于$\forall q\in K,\forall w\in\Sigma^*$,

$\hat{\delta}(q,w)=P=\{p|$ 从q出发，沿着标有w的路径达到状态$p\}$

值得注意:在DFA和NFA中,对于扩充后的 $\hat{\delta}$ 与 δ 可以不加区别的使用，因为它们的计算结果相同。但是在这里就必须加以区别,因为它们的计算结果不同了。请看下例:

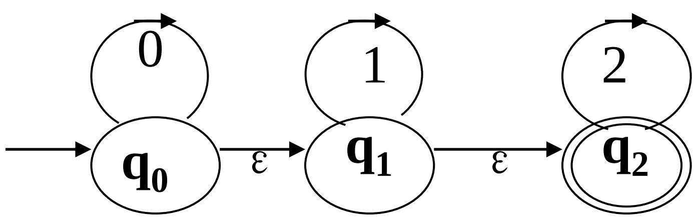
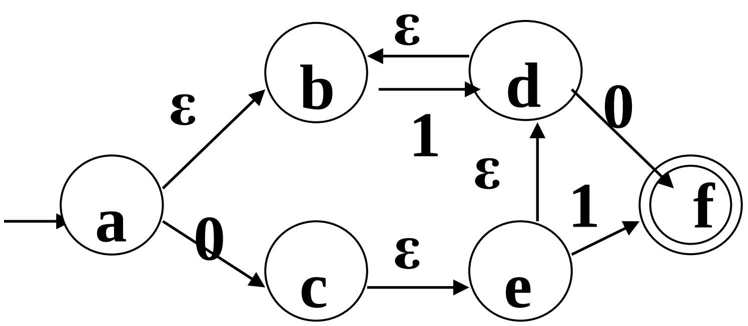


图2-3.1 ε-NFA M图

δ':	0	1	2
q_0	$\{q_0,q_1,q_2\}$	$\{q_1,q_2\}$	$\{q_2\}$
q_1	Φ	$\{q_1,q_2\}$	$\{q_2\}$
q_2	Φ	Φ	$\{q_2\}$

作业题

将下面的ε-NFA M等价变换成NFA M'。



2.4 正规表达式及其与有限自动机之间的等价性

有限自动机所接受的语言，很容易用字母表上的符号串以及符号串上的一些运算符号构成的表达式来表示，即用正规表达式来表示。

一.正规表达式定义

设Σ是有限字母表，Σ上的正规表达式及其所代表的符号串集合递归定义如下：

1.φ是个正规表达式，它代表空集Φ。

2.ε是个正规表达式，它代表集合{ε}。

3. 任何a∈Σ，是个正规表达式，它代表集合{a}。

4.如果r、s分别是代表集合R、S的正规表达式,则 (r+s)、(rs)、(r^*)分别是代表集合(R∪S)、(RS)、(R^*)的正规表达式。

r+s称作r与s的加运算； rs 称作r与s的连接运算；

r^* 称作r的*闭包运算。

设r是个正规表达式，则r所代表的集合记作L(r)。

例如，L(ε)={ε}，L(a)={a}

二．正规表达式运算的优先级

为了书写简单，看起来也简单明了，我们规定了正规表达式运算的优先权，于是正规表达式的最外层的括号以及表达式里的一些括号可以省略。

运算的优先权:从高到低依次是：*、连接、＋。

因此，$(0+(0(1^*)))$可以简记为 $0+01^*$。

例如 设 Σ＝{0,1}， 则

1. 01 表示集合 L(01)＝{0}{1}={01}
2. 0+1 表示集合 L(0+1)= L(0) ∪ L(1)={0}∪{1}={0,1}
3. $(0+1)^*$ 表示集合 $L((0+1)^*)=\{0,1\}^*=\{\varepsilon,0,1,00,01,10,11,000,001,\ldots\}$

4. 0^* 表示集合$L((0)^*)=\{0\}^*=\{\varepsilon,0,00,000,0000,\ldots\}$

5. $1(0+1)^*$ 表示集合

$L(1(0+1)^*)=\{1\}\{0,1\}^*=\{1\}\{\varepsilon,0,1,00,01,10,11,000,001,\ldots\}$

$=\{1,10,11,100,101,110,111,1000,1001,\ldots\}$

三．两个正规表达式相等

设α、β是Σ上的正规表达式，如果L(α)=L(β),则称α与β相等，记作α＝β。

四.重要的等式（正规表达式的运算性质）

α、β、γ是Σ上的正规表达式，则

1. $\alpha+\beta=\beta+\alpha$ (+满足交换律)
2. $\alpha+\alpha=\alpha$ (+满足幂等律)
3. $\alpha+\varphi=\varphi+\alpha=\alpha$ (φ是＋运算的幺元)
4. $(\alpha+\beta)+\gamma=\alpha+(\beta+\gamma)$ (+满足可结合性)
5. $\alpha(\beta+\gamma)=\alpha\beta+\alpha\gamma$ (连接对＋满足分配律)

证明：$L(\alpha(\beta+\gamma))=L(\alpha)L(\beta+\gamma)=L(\alpha)(L(\beta)\cup L(\gamma))=L(\alpha)L(\beta)\cup L(\alpha)L(\gamma)=L(\alpha\beta)\cup L(\alpha\gamma)=L(\alpha\beta+\alpha\gamma)$

6. $(\alpha+\beta)\gamma=\alpha\gamma+\beta\gamma$

7. $\alpha\varepsilon=\varepsilon\alpha=\alpha$ (ε是连接运算的幺元)

8. $\varphi\alpha=\alpha\varphi=\varphi$ (φ是连接运算的零元)

证明：$L(\varphi\alpha)=L(\varphi)L(\alpha)=\Phi L(\alpha)=\Phi=L(\varphi)$ 所以 $\varphi\alpha=\varphi$，

类似可证$\alpha\varphi=\varphi$

9. $(\alpha\beta)\gamma=\alpha(\beta\gamma)$ (连接运算满足可结合性)

10. $\varphi^{*}=\varepsilon$

证明：对任何集合$A^{*}=A^{0}\cup A\cup A^{2}\cup A^{3}\cup A^{4}\cup\ldots$

其中$A^{0}=\{\varepsilon\}$，所以

$L(\varphi^{*})=\Phi^{*}=\Phi^{0}\cup\Phi\cup\Phi^{2}\cup\ldots=\Phi^{0}=\{\varepsilon\}=L(\varepsilon)$

11. $\alpha+\alpha^*=\alpha^*$

12. $(\alpha^*)^*=\alpha^*$

在离散数学中,二元关系R有， $R^*=rt(R)=tr(R)$, $(R^*)^*=R^*$

13. $\alpha\alpha^*=\alpha^*\alpha=\alpha^+$

在离散数学中，二元关系R有，$(R\circ R^*)=R^*\circ R=R^+$

14. $(\varepsilon+\alpha)^*=\alpha^*$

因 $L((\varepsilon+\alpha)^*)=\{\varepsilon,\alpha\}^*=\{\varepsilon,\alpha\}^0\cup\{\varepsilon,\alpha\}^1\cup\{\varepsilon,\alpha\}^2\cup\ldots$

$=\{\varepsilon\}\cup\{\varepsilon,\alpha\}\cup\{\varepsilon,\alpha,\alpha^2\}\cup\{\varepsilon,\alpha,\alpha^2,\alpha^3\}\cup\ldots$

$=\{\varepsilon,\alpha,\alpha^2\ \alpha^3,\ldots\}=\{\alpha\}^*=L(\alpha^*)$

15. $\varepsilon+\alpha^+=\alpha^*$

16. $(\alpha\alpha)^*+\alpha^*=\alpha^*$

因为$\{\alpha\alpha\}^*\subseteq\{\alpha\}^*$

五.正规表达式与有限自动机之间的等价性

下面证明有限自动机所接受的语言正好是一个正规表达式所表示的语言。反之亦然,即任意给定正规表达式r,都可以确定一个FA M，使得T(M)=L(r)。

正规表达式与有限自动机之间的等价关系，可以用下面图2-4.1表示。

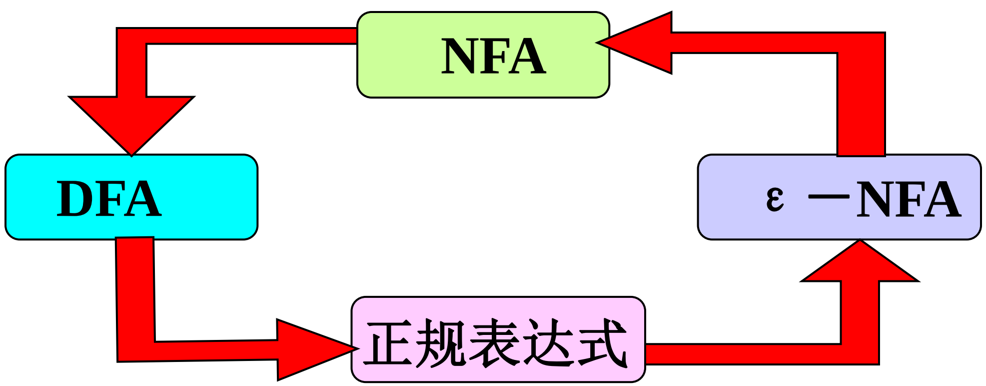
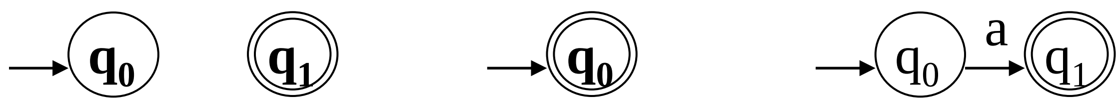


图2-4.1 有限自动机与正规表达式之间的等价转换关系图

定理2-4.1 设r是Σ上的一个正规表达式，则存在一个具有ε转移的NFA M, 使得T(M)=L(r)。

证明：用对r中含有运算的个数归纳证明。

1.如果r中有0个运算，则有三种可能：r=φ,r=ε,r=a　a∈Σ,则有下面三个自动机分别接受它们。



r= ϕ　　　　r= ε　　　　r=a

2.假设r中运算个数少于k个时，结论成立。

3.当r中有k个运算时，因正规表达式中只有三种运算，所以分三种情况讨论：

(1) 如果$r=r_1+r_2$, 则r_1和r_2中运算个数少于k个，由假设得，存在ε—NFA M_1和M_2 ,

使得　$T(M_1)=L(r_1)$　$T(M_2)=L(r_2)$,不妨设

$M_1=(K_1, \Sigma_1, \delta_1, q_1, \{f_1\})$　$M_2=(K_2, \Sigma_2, \delta_2, q_2, \{f_2\})$

设 $K_1 \cap K_2=\Phi$

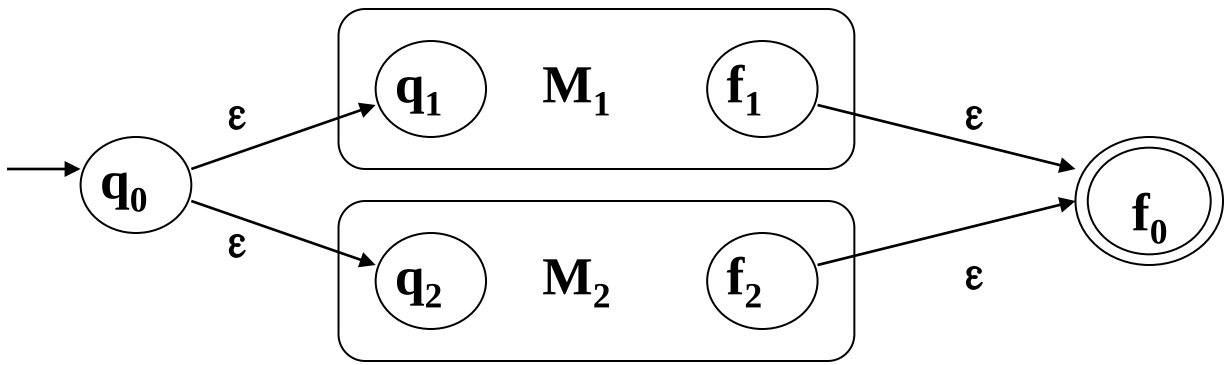
下面构造ε－NFA M

$M=(K_1\cup K_2\cup\{q_0,f_0\},\Sigma_1\cup\Sigma_2,\delta,q_0,\{f_0\})$，

其中$q_0,f_0\notin K_1\cup K_2$，使得

$T(M)=T(M_1)\cup T(M_2)=L(r_1)\cup L(r_2)=L(r_1+r_2)$，

于是M的结构如下图所示：



M的状态转移函数δ为:

① $\delta(q_0,\varepsilon)=\{q_1,q_2\}$

M可以进入M_1和M_2的开始状态。

② 对任何$q\in K_1$，任何$a\in\Sigma_1\cup\{\varepsilon\}$ 有

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

M可以模拟M_1的动作。

③ 对任何$q\in K_2$，任何$a\in\Sigma_2\cup\{\varepsilon\}$ 有

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

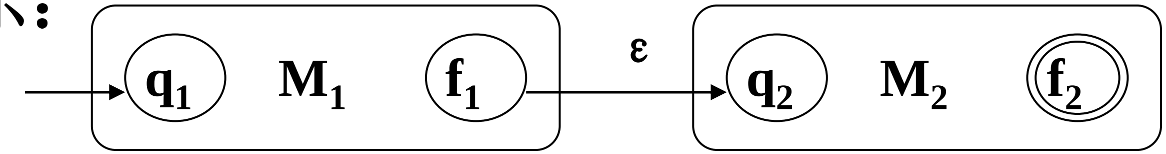
M可以模拟M_2的动作。

④$\delta(f_1,\varepsilon)=\delta(f_2,\varepsilon)=\{f_0\}$

很容易证明$T(M)=T(M_1)\cup T(M_2)$，这里从略。

(2).如果 $r=r_1r_2$，则 r_1和r_2中运算个数少于k个,由假设得，存在ε－NFA M_1和M_2 使得$T(M_1)=L(r_1)$ $T(M_2)=L(r_2)$, M_1和M_2如前所述。

下面构造ε－NFA $M=(K_1\cup K_2,\Sigma_1\cup\Sigma_2,\delta,q_1,\{f_2\})$，使得$T(M)=T(M_1)T(M_2)=L(r_1)L(r_2)=L(r_1r_2)$, 于是M的结构如下图所示：



M的状态转移函数δ为：

① 对任何$q\in K_1$，任何$a\in\Sigma_1\cup\{\varepsilon\}$ 有

$\delta(q,a)=\delta_1(q,a)$ M可以模拟M_1的动作。

② $\delta(f_1,\varepsilon)=\{q_2\}$

③ 对任何$q\in K_2$，任何$a\in\Sigma_2\cup\{\varepsilon\}$ 有

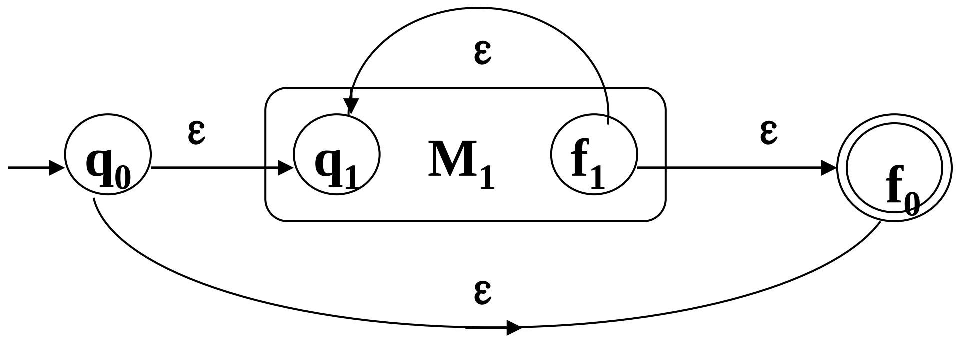
$\delta(q,a)=\delta_2(q,a)$ M可以模拟M_2的动作。

很容易证明$T(M)=T(M_1)T(M_2)$，这里从略。

(3).如果 $r=r_1^*$, 则 r_1中运算个数少于k个，由假设得，存在ε—NFA M_1 使得$T(M_1)=L(r_1)$,M_1 如前所述。

下面构造ε—NFA $M＝(K_1\cup\{q_0,f_0\},\Sigma_1,\delta,q_0,\{f_0\})$，

使得$T(M)=(T(M_1))^*=(L(r_1))^* =L(r_1^*)$, 于是M的结构如下图所示：



M的状态转移函数δ为：

① $\delta(q_0,\varepsilon)=\{q_1,f_0\}$

② 对任何$q\in K_1$, 任何$a\in\Sigma_1\cup\{\varepsilon\}$ 有

$\delta(q,a)=\delta_1(q,a)$　　M可以模拟M_1的动作。

③$\delta(f_1,\varepsilon)=\{q_1,f_0\}$

下面证明$T(M)=(T(M_1))^*$：

任何$w\in T(M)$,

如果$w=\varepsilon$,因为$\varepsilon\in(T(M_1))^0$，而$(T(M_1))^0\subseteq(T(M_1))^*$，所以$\varepsilon\in(T(M_1))^*$，所以$w\in(T(M_1))^*$。

如果$w\neq\varepsilon$，根据M的结构可以看出，一定可以将w写成$w=w_1w_2w_3\ldots w_k$形式，其中$k\geq1$，每个w_i $(1\leq i\leq k)$都使得在M_1中从q_1出发沿着标有w_i的路径达到f_1，识别完w_k后最后经过ε转移达到状态f_0，M接受输入w。

显然上述每个$w_i(1\leq i\leq k)$都属于$T(M_1)$,所以

$w_1w_2w_3\ldots w_k\in(T(M_1))^k$, $(T(M_1))^k\subseteq(T(M_1))^*$，所以

$w_1w_2w_3\ldots w_k\in(T(M_1))^*$，所以$w\in(T(M_1))^*$。

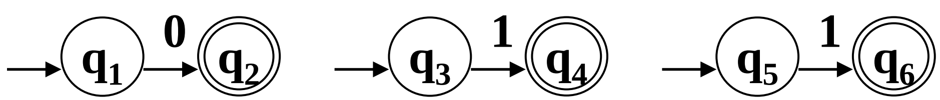
于是得$T(M)\subseteq(T(M_1))^*$

类似可以证明$(T(M_1))^*\subseteq T(M)$。

最后得$T(M)=(T(M_1))^*$。

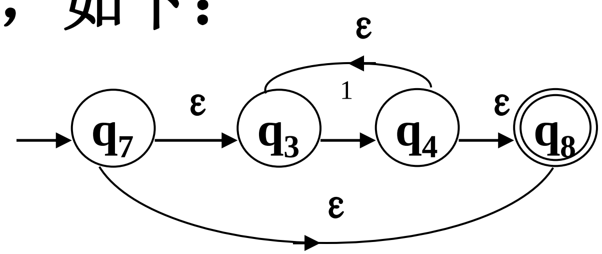
【例2-4.1】 给定正规表达式$r=01^{*}+1$，构造一个ε-NFA M, 使得$T(M)=L(r)$。

解. 1. 先将r 分解，找出r中所含基本符号——0、1、1，然后分别构造出接受这些基本符号的有限自动机，如下图所示。

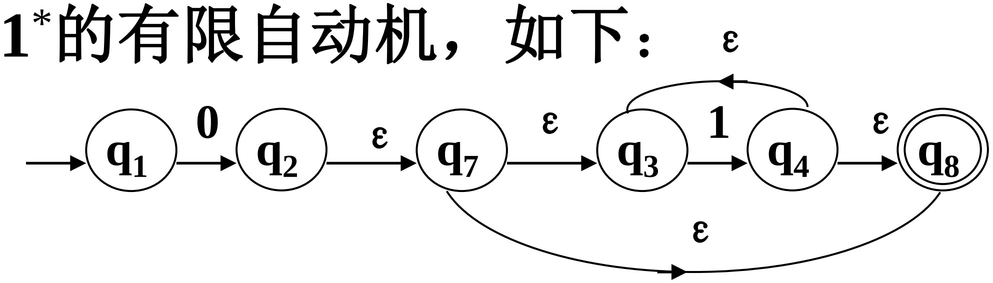


2. 按照运算的优先权，从高到低进行“组装”。

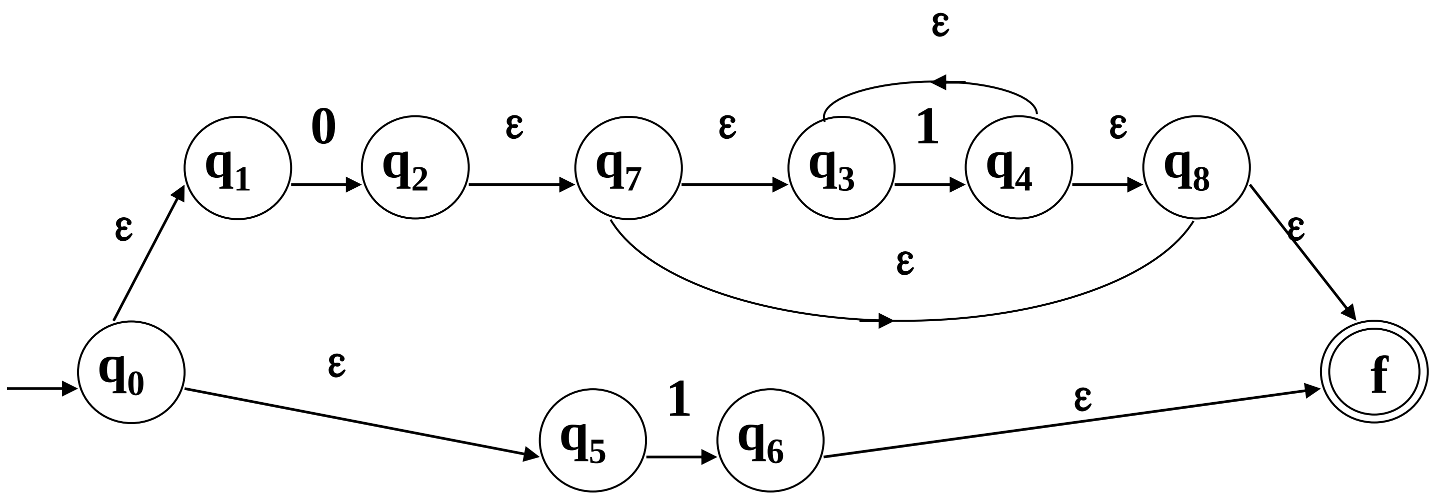
①先组装接受1^{*}的有限自动机，如下：



②再组装接受01^{*}的有限自动机，如下：



③最后组装接受01^*+1的有限自动机M，如下：



此ε一NFA M就是接受正规表达式r的有限自动机。

上面我们讨论了由给定的正规表达式r求接受r的有限自动机，当然可以将ε一NFA M等价转换成NFA, 进而还可以转换成DFA。

下面的问题就是，给定DFA M，如何求出T(M)的正规表达式的问题。(蒋宗礼P385.)

定理2-4.2 如果语言L可由一个**DFA M**接受，则L可用一个正规表达式表示。

证明：设L由一个DFA $M=(K,\Sigma,\delta,q_1,F)$接受，不妨设$K=\{q_1,q_2,\ldots,q_n\}$。求T(M)所对应的正规表达式r，即使得L(r)=T(M)。

因为T(M)是由Σ上这样字符串w构成的集合，这些字符串w都使得M从开始状态q_1出发沿着标有w的路径，途中经过K中某些状态，而最后达到F中的某个状态。为此我们定义符号串集合 R_{ij}^{k} 。

R_{ij}^{k}：是由Σ上这样字符串x构成的集合，这些x都使得M从状态q_i出发沿着标有x的路径，最后到达q_j，而途中经过所有状态的下标都不大于k。

R_{ij}^{n} 就表示使得M从状态q_i出发，最后到达q_j的所有符号串构成的集合。于是 $T(M)=\bigcup_{q_i\in F} R_{1i}^{n}$

问题是如何计算R_{ij}^{k}，下面就给出它的递归计算公式：

$$R_{ij}^{0}=\begin{cases}\{a|\delta(q_i,a)=q_j\} & i\neq j\\ \{a|\delta(q_i,a)=q_j\}\cup\{\varepsilon\} & i=j\end{cases}$$

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

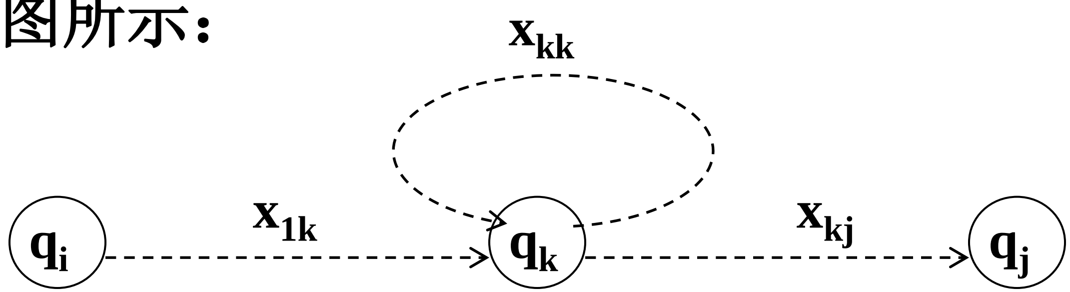
其中R_{ij}^{0}表示从状态$\mathbf{q_i}$出发，不经过任何状态直接到达$\mathbf{q_j}$的所有符号串构成的集合。因此当**i≠j**时，就是**M**的有向图上从$\mathbf{q_i}$到$\mathbf{q_j}$的有向边上标的符号构成的集合。当**i=j**时，除了有向环上标的符号外，还包括**ε**。

R_{ij}^{k} 由两部分组成：

一部分是R_{ij}^{k-1}，R_{ij}^{k-1} 中的符号串**x**可使得**M**从状态$\mathbf{q_i}$出发沿着标有**x**的路径，最后到达$\mathbf{q_j}$,而途中经过所有状态的下标都不大于**k-1**, 而不经过状态$\mathbf{q_k}$。

另一部分是 $R_{ik}^{k-1}(R_{kk}^{k-1})^*R_{kj}^{k-1}$，其中的符号串**x**可使得**M**从状态$\mathbf{q_i}$出发沿着标有**x**的路径，最后到达$\mathbf{q_j}$,而途中必经过状态$\mathbf{q_k}$。这里**x**可以由三个子串的连接组成，例如，

$\mathbf{x_{ik}} \in R_{ik}^{k-1}$,$\mathbf{x_{kk}} \in R_{kk}^{k-1}$, $\mathbf{x_{kj}} \in R_{kj}^{k-1}$, 则 $\mathbf{x=x_{ik}(x_{kk})^*x_{kj}}$，识别**x**的路线如下图所示：



显然R_{ij}^k 可以用正规表达式 r_{ij}^k 表示，它的计算公式如下：

$$r_{ij}^0 = \begin{cases} a_1 + a_2 + \ldots + a_m & i \neq j \\ a_1 + a_2 + \ldots + a_m + \varepsilon & i = j \end{cases} \quad \delta(q_i, a_k) = q_j \quad (1 \leq k \leq m)$$

$$r_{ij}^k = r_{ik}^{k-1}(r_{kk}^{k-1})^* r_{kj}^{k-1} + r_{ij}^{k-1}$$

下面通过一个例子说明如何应用上面公式求**DFA M**的**T(M)**的正规表达式**r.**

【例2-4.1】 给定**DFA M**如图**2-4.2**所示。求**T(M)**的正规表达式**r.**

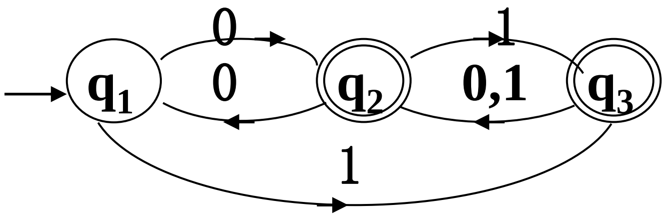


图2-4.2

解 **1. k=0**

$$r_{11}^{0}=\varepsilon \quad r_{12}^{0}=0 \quad r_{13}^{0}=1$$

$$r_{21}^{0}=0 \quad r_{22}^{0}=\varepsilon \quad r_{23}^{0}=1 \qquad r_{31}^{0}=\phi \quad r_{32}^{0}=0+1 \quad r_{33}^{0}=\varepsilon$$

2. K=1:

$$r_{11}^{1} = r_{11}^{0}(r_{11}^{0})^{*}r_{11}^{0} + r_{11}^{0} = (r_{11}^{0})^{+}r_{11}^{0} + r_{11}^{0} = ((r_{11}^{0})^{+} + \varepsilon)r_{11}^{0} = (r_{11}^{0})^{*}r_{11}^{0} = (\varepsilon)^{*}\varepsilon = \varepsilon$$

$$r_{12}^{1} = r_{11}^{0}(r_{11}^{0})^{*}r_{12}^{0} + r_{12}^{0} = (r_{11}^{0})^{+}r_{12}^{0} + r_{12}^{0} = ((r_{11}^{0})^{+} + \varepsilon)r_{12}^{0} = (r_{11}^{0})^{*}r_{12}^{0} = (\varepsilon)^{*}0 = 0$$

$$r_{13}^{1} = r_{11}^{0}(r_{11}^{0})^{*}r_{13}^{0} + r_{13}^{0} = (r_{11}^{0})^{+}r_{13}^{0} + r_{13}^{0} = ((r_{11}^{0})^{+} + \varepsilon)r_{13}^{0} = (r_{11}^{0})^{*}r_{13}^{0} = (\varepsilon)^{*}1 = 1$$

$$r_{21}^{1} = r_{21}^{0}(r_{11}^{0})^{*}r_{11}^{0} + r_{21}^{0} = r_{21}^{0}(r_{11}^{0})^{+} + r_{21}^{0} = r_{21}^{0}((r_{11}^{0})^{+} + \varepsilon) = r_{21}^{0}(r_{11}^{0})^{*} = 0(\varepsilon)^{*} = 0$$

$$r_{22}^{1} = r_{21}^{0}(r_{11}^{0})^{*}r_{12}^{0} + r_{22}^{0} = 0(\varepsilon)^{*}0 + \varepsilon = 0\varepsilon 0 + \varepsilon = 00 + \varepsilon = \varepsilon + 00$$

$$r_{23}^{1} = r_{21}^{0}(r_{11}^{0})^{*}r_{13}^{0} + r_{23}^{0} = 0(\varepsilon)^{*}1 + 1 = 0\varepsilon 1 + 1 = 01 + 1 = (0 + \varepsilon)1 = (\varepsilon + 0)1$$

$$r_{31}^{1} = r_{31}^{0}(r_{11}^{0})^{*}r_{11}^{0} + r_{31}^{0} = r_{31}^{0}(r_{11}^{0})^{+} + r_{31}^{0} = r_{31}^{0}((r_{11}^{0})^{+} + \varepsilon) = r_{31}^{0}(r_{11}^{0})^{*} = \varphi(\varepsilon)^{*} = \varphi$$

$$r_{32}^{1} = r_{31}^{0}(r_{11}^{0})^{*}r_{12}^{0} + r_{32}^{0} = \varphi(\varepsilon)^{*}0 + (0 + 1) = \varphi + (0 + 1) = 0 + 1$$

$$r_{33}^{1} = r_{31}^{0}(r_{11}^{0})^{*}r_{13}^{0} + r_{33}^{0} = \varphi(\varepsilon)^{*}1 + \varepsilon = \varphi + \varepsilon = \varepsilon$$

3.K=2

因为 $T(M)=R_{12}^3 \cup R_{13}^3$ 所以 $r=r_{12}^3+r_{13}^3$ 而

$$r_{12}^3=r_{13}^2(r_{33}^2)*r_{32}^2+r_{12}^2$$

$r_{13}^3=r_{13}^2(r_{33}^2)*r_{33}^2+r_{13}^2$ 于是我们只求与之有关的：

$$r_{12}^2=r_{12}^1(r_{22}^1)*r_{22}^1+r_{12}^1=r_{12}^1(r_{22}^1)^+ +r_{12}^1=r_{12}^1((r_{22}^1)^+ +\varepsilon)=r_{12}^1(r_{22}^1)*$$

$$=0(\varepsilon+00)*=0(00)*$$

$$r_{13}^2=r_{12}^1(r_{22}^1)*r_{23}^1+r_{13}^1=0(\varepsilon+00)*(\varepsilon+0)1+1=0(00)*(\varepsilon+0)1+1$$

因为$0(00)*(\varepsilon+0)$表示集合：

$$\{0\}\{\varepsilon,00,0000,000000,\ldots\}\{\varepsilon,0\}=\{0,000,00000,\ldots\}(\{\varepsilon\}\cup\{o\})$$

$$=\{0,000,00000,\ldots\}\cup\{00,0000,000000,\ldots\}$$

$=\{0,00,000,0000,00000,\ldots\}=0^+$ 于是上式

$$r_{13}^2=0^+1+1=(0^+ +\varepsilon)1=0*1$$

$$r_{32}^2=r_{32}^1(r_{22}^1)*r_{22}^1+r_{32}^1=r_{32}^1(r_{22}^1)^+ +r_{32}^1=r_{32}^1((r_{22}^1)^+ +\varepsilon)=r_{32}^1(r_{22}^1)*$$

$$=(0+1)(\varepsilon+00)*=(0+1)(00)*$$

$$r_{33}^2=r_{32}^1(r_{22}^1)*r_{23}^1+r_{33}^1=(0+1)(\varepsilon+00)*(\varepsilon+0)1+\varepsilon$$

$$=(0+1)(00)*(\varepsilon+0)1+\varepsilon=(0+1)0*1+\varepsilon=\varepsilon+(0+1)0*1$$

4.K=3

$$r_{12}^{3}=r_{13}^{2}(r_{33}^{2})^{*}r_{32}^{2}+r_{12}^{2}=0^{*}1(\varepsilon+(0+1)0^{*}1)^{*}(0+1)(00)^{*}+0(00)^{*}$$
$$=0^{*}1((0+1)0^{*}1)^{*}(0+1)(00)^{*}+0(00)^{*}$$

$$r_{13}^{3}=r_{13}^{2}(r_{33}^{2})^{*}r_{33}^{2}+r_{13}^{2}=r_{13}^{2}(r_{33}^{2})^{+}+r_{13}^{2}=r_{13}^{2}(r_{33}^{2})^{*}$$
$$=0^{*}1(\varepsilon+(0+1)0^{*}1)^{*}=0^{*}1((0+1)0^{*}1)^{*}$$

5. T(M)的正规表达式r为:

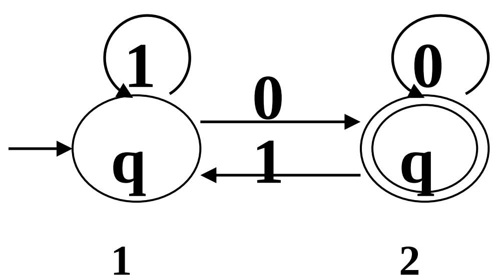
$$r=r_{12}^{3}+r_{13}^{3}=0^{*}1((0+1)0^{*}1)^{*}(0+1)(00)^{*}+0(00)^{*}+0^{*}1((0+1)0^{*}1)^{*}$$
$$=0^{*}1((0+1)0^{*}1)^{*}(\varepsilon+(0+1)(00)^{*})+0(00)^{*}$$

• 作业题

1. 化简正规表达式
$a(\varepsilon+aa)^*(\varepsilon+a)b+b+\varphi(ab^*+b)^*$

2. 构造一个FA M，使得T(M)的正规表达式为$01+((0+1)^*1)^*$。

3. 给定FA M如下图所示，求它所接收的语言T(M)的正规表达式。



2-5 有限自动机的简化

对于给定的一个有限自动机M，有时可以在保证接受语言T(M)不变的情况下，进行简化，即求一个含有更少状态的有限自动机M’，使得T(M’)=T(M)。

例如下面有限自动机M就可以简化成M’，它们接受的语言都是1^*,如图2-5.1所示。

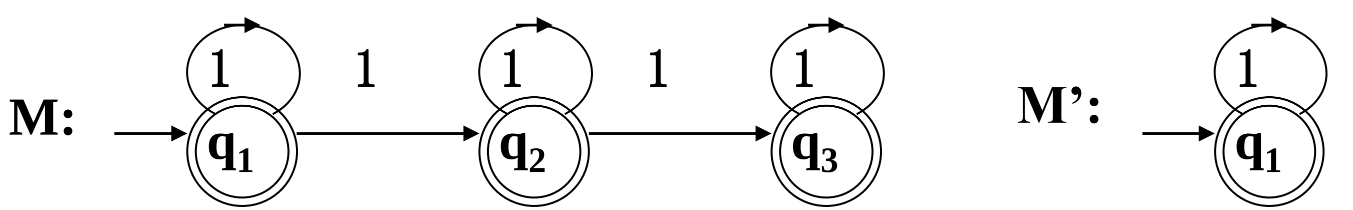


图2-5.1

给定一个DFA M ,如何进行简化。主要的思路是先定义自动机状态集合K上的一个等价关系≡，然后用≡将K进行划分得到商集K/≡，再构造状态更少的有限自动机。

一.定义K上等价关系≡

给定**DFA** $M=(K,\sum,\delta,q_0,F)$，

$\forall p,q \in K$,

$p \equiv q \Leftrightarrow$ 对$\forall\ x \in \sum^*$, 有

$$\delta(p,x) \in F \leftrightarrow \delta(q,x) \in F$$

如果$p \equiv q$ 也称p与q是不可区分的。

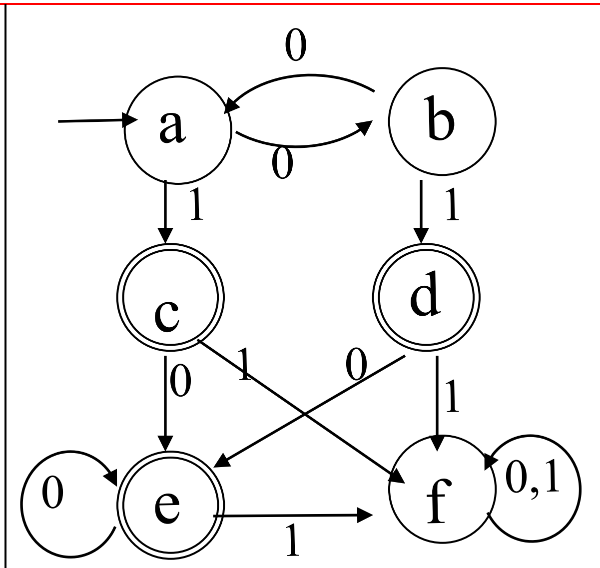


图2-5.2

二.商集K/≡

【例2-5.1】 给定DFA $M=(K,\sum,\delta,q_0,F)$，$K=\{a,b,c,d,e,f\}$, $\sum=\{0,1\}$,$F=\{c,d,e\}$，M的δ如图2-5.2所示，显然T(M)的正规表达式为 0^*10^*。

为了求商集K/≡, 需要对所有$x \in \sum^*$,进行考察,才能判断哪些状态之间有等价关系≡，进而得到商集K/≡,但是由于$\sum^*$是无限集合，这里

$\sum^*=\{\varepsilon,0,1,00,01,10,11,000,001,010,\ldots\}$

无法考察$\sum^*$中的每个x，为方便，将$\sum^*$分成三个子集：

A＝{x| x∈$\sum^*$且x中无1}

B＝{x| x∈$\sum^*$且x中只有一个1}

C＝{x| x∈$\sum^*$且x中至少有两个1}

$\sum^*$＝A∪B∪C, {A,B,C}是$\sum^*$的一个划分。

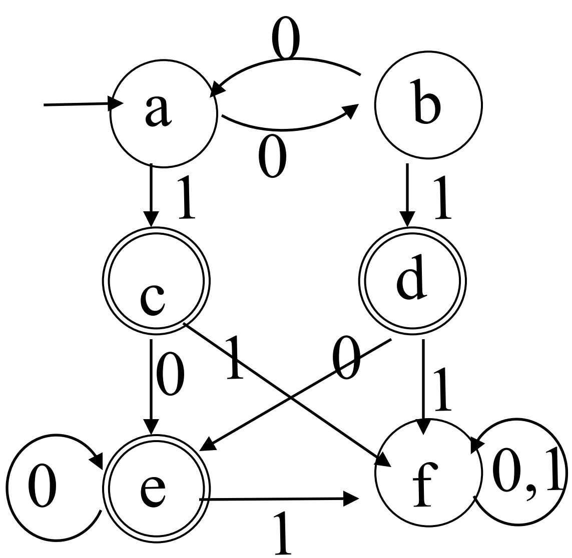


图2-5.2

$\delta(q,x)$ q \ x	x∈A	x∈B	x∈C
a	**∉F**	**∈F**	**∉F**
b	**∉F**	**∈F**	**∉F**
c	**∈F**	**∉ F**	**∉F**
d	**∈F**	**∉ F**	**∉F**
e	**∈F**	**∉ F**	**∉F**
f	**∉F**	**∉F**	**∉F**

a≡b, c≡d,c≡e,d≡e, f≡f,于是得商集：K/≡{{a,b},{c,d,e},{f}}

从此例子看出，由于一般来说$\sum^*$是无限集合，所以按照$\equiv$的定义判断哪些状态之间有等价关系$\equiv$，是个很难的事情，甚至无法实现。

下面我们换一种思维，是否可以判断哪些状态之间无等价关系$\equiv$，即考虑$\equiv$的逆关系$\not\equiv$，从而求得商集。

三．$\equiv$的逆关系$\not\equiv$

对于任何$p,q\in K$，

$$p\equiv q \Leftrightarrow \forall x(x\in\sum^*\rightarrow(\delta(p,x)\in F\leftrightarrow\delta(q,x)\in F)),$$

对于此式两边同时取否定得：

$$p\not\equiv q \Leftrightarrow \exists x(x\in\sum^*\wedge\neg(\delta(p,x)\in F\leftrightarrow\delta(q,x)\in F))$$

$$\Leftrightarrow\exists x(x\in\sum^*\wedge$$

$$((\delta(p,x)\in F\wedge\delta(q,x)\notin F)\vee(\delta(p,x)\notin F\wedge\delta(q,x)\in F)))$$

$\Leftrightarrow\exists x(x\in\sum^*$，使得$\delta(p,x)$与$\delta(q,x)$恰有一个在F中)

如果$p\not\equiv q$, 称p与q是可区分的。判断$p\not\equiv q$是比较容易的。

4．判断可区分状态对的算法

引理2-1 设$M=(K,\sum,\delta,q_0,F)$是DFA，则状态对(p,q)是可区分的(即$p\neq q$)，当且仅当在下面算法中(p,q)格写上×。

begin

1．for $p\in F, q\in K-F$, do 给(p,q)格写×；

2．for $F\times F$或$(K-F)\times(K-F)$中每个状态对(p,q) ($p\neq q$)，do

3．if $\exists a\in\sum$，使得格$(\delta(p,a),\delta(q,a))$内已经写上×，then

  begin

4． 给(p,q)格写×；

5． 如果刚刚写上×的格内有先前写入的状态对，此状态对的格同时也写入×。反复执行5, 直到写入×的格内没有先前写入的状态对为止；

  end

 else /* 格$(\delta(p,a),\delta(q,a))$内无× */

6．for 每个$a\in\sum$ do /* 建立关联链表 */

7．If $\delta(p,a)\neq\delta(q,a)$ 且$(p,q)\neq(\delta(p,a),\delta(q,a))$ then 把(p,q)写入格$(\delta(p,a),\delta(q,a))$内，除非$\delta(p,a)=\delta(q,a)$。

end

为了了解此算法的思想，我们先看一看对例2-6应用此算法的结果，再证明此引理。

执行此算法的结果用一个表表示，实际上，执行此算法的过程就是向这个表内写入“×”的过程。

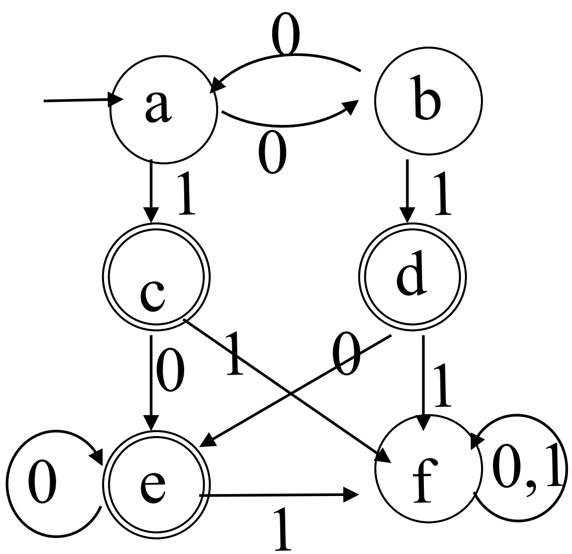


图2-5.2

b					
c	×	×			
d	×	×	**(a,b)**		
e	×	×			
f	×	×	×	×	×
	a	b	c	d	e

最后得**a≡b，c≡d，c≡e，d≡e，f≡f**，于是**K/≡{{a,b},{c,d,e},{f}}**，这与前面得到的结果是一致的。

下面就是对例2-5.1执行此算法的过程：

1．因为a,b,f∈K-F, c,d,e∈F,所以下面格写×：

(a,c) ,(a,d), (a,e), (b,c), (b,d) ,(b,e) ,(f,c), (f,d) ,(f,e)。

因为这些状态对可以用ε区分。

2．下面考察F×F或(K-F)×(K-F)中状态对(p,q)

(其中p≠q)：

考察(a,b)：因δ(a,0)=b，δ(b,0)=a，

这说明a与b用0是区分不开的，

又δ(a,1)=c，δ(b,1)=d，

因(c,d)格内无×，

说明a与b是否可区分，

取决于c与d，

所以根据算法的第7步，

将(a,b)写入(c,d)格内。

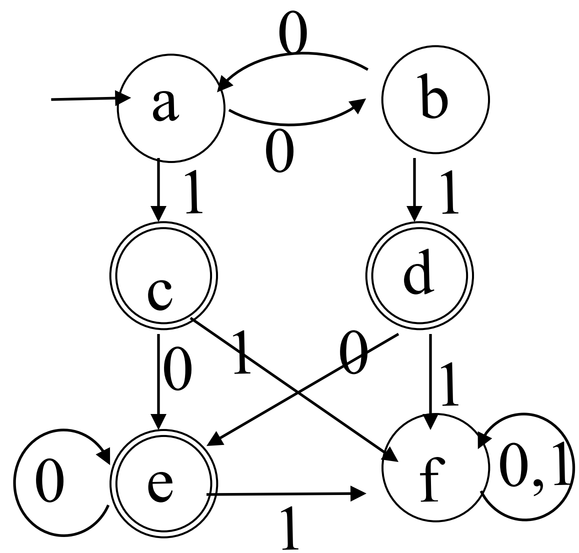


图2-5.2

考察**(a,f)**:因为δ(a,0)=b,δ(f,0)=f， 而(b,f)格无×；
再看δ(a,1)=c，δ(f,1)=f， 而(c,f)格内已写×，所以根据 算法的第3,4步,(a,f)格也写入×。

考察**(b,f)**：δ(b,0)=a δ(f,0)=f，
而(a,f)格内已写×，
根据算法的第3，4步，
(b,f)格写入×。

考察**(c,d)**：δ(c,0)=e δ(d,0)=e，
δ(c,1)=f , δ(d,1)=f，
可见(c,d)不可区分，故c≡d。

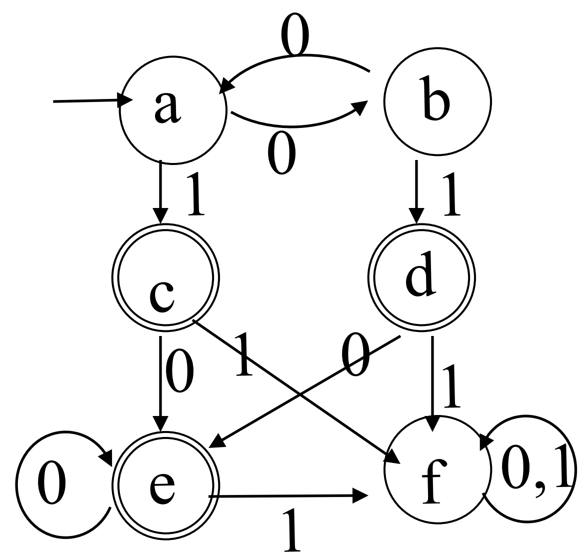


图2-5.2

考察(c,e)：$\delta(c,0)=e$ $\delta(e,0)=e$，

$\delta(c,1)=f$,$\delta(e,1)=f$，

可见(c,e)不可区分，故$c\equiv e$。

考察(d,e)：$\delta(d,0)=e$ $\delta(e,0)=e$，

$\delta(d,1)=f$ $\delta(e,1)=f$，

可见(d,e)不可区分，故$d\equiv e$。

此外，由于(c,d)不可区分，

所以(a,b)也是不可区分的，

故有$a\equiv b$ 。

最后得$a\equiv b$，$c\equiv d$，$c\equiv e$，$d\equiv e$，$f\equiv f$，

于是

$K/\equiv\{\{a,b\},\{c,d,e\},\{f\}\}$，

这与前面得到的结果是一致的。

b					
c	×	×			
d	×	×	(a,b)		
e	×	×			
f	×		×	×	×
	a	b	c	d	e

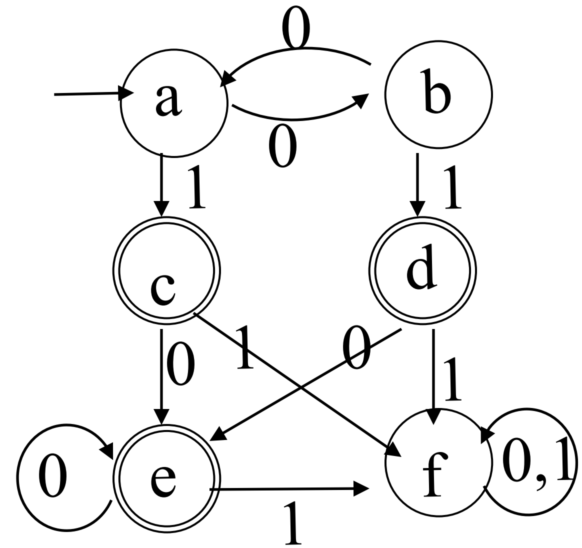


图2-5.2

引理证明：必要性，(设(p,q)可区分，证出(p,q)格内必写入×。)
充分性，(设(p,q)格写入×，通过对表中×的个数归纳证明(p,q)一定可区分。) 证明过程从略。

五．构造简化的有限自动机

定理2-5.1 给定DFA $M=(K,\sum,\delta,q_0,F)$，可根据引理2-1中的算法构造出除去不可达状态的具有更少状态的DFA M’，使得T(M’)=T(M)。

证明：先对M用引理2-1中的算法求出K/≡。再构造M’：

$M'=(K',\sum,\delta',[q_0], F')$，其中

$K'=\{[q] \mid [q]\in K/\equiv$ 且在M中q是从q_0可达的状态$\}$

$F'=\{[q] \mid q\in F\}$

δ'：对任何$[q]\in K'$，任何$a\in\sum$,

$\delta'([q],a)=[\delta(q,a)]$

下面我们还是先针对例2-5.1中的M，按照此定理的方法，求M’。

应用引理2-1后已经求得

K/≡={{a,b},{c,d,e},{f}}={[a],[c],[f]}，

其中 [a]={a,b}，[c]={c,d,e}，[f]={f}。

K’={[a],[c],[f]}，[q_0]=[a] ， F’={[c]}

M’=({[a],[c],[f]},{0,1},δ’,[a],{[c]})

δ’([a],0)=[δ(a,0)]=[b]=[a]

δ’([a],1)=[δ(a,1)]=[c]

δ’([c],0)=[δ(c,0)]=[e]=[c]

δ’([c],1)=[δ(c,1)]=[f]

δ’([f],0)=[δ(f,0)]=[f]

δ’([f],1)=[δ(f,1)]=[f]

M’的图，如图2-5.2所示。

显然，M’接受的语言也是0^*10^*。

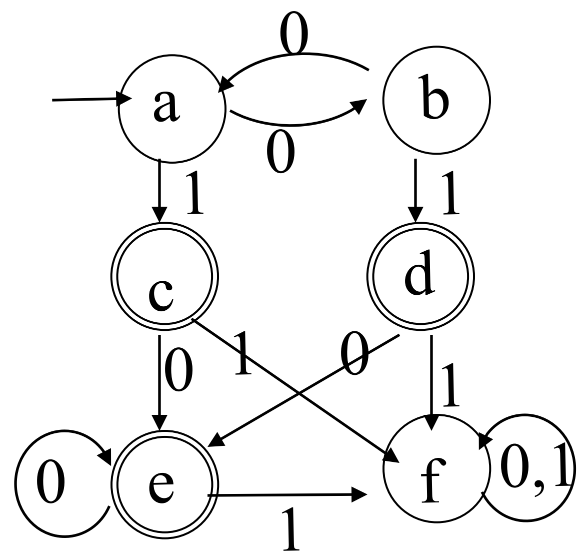


图2-5.2 M

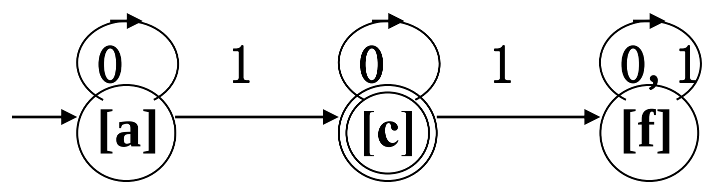


图2-5.2 M’

定理的证明——简要说明：

首先，证明δ'定义的一致性。

即对任何p,q∈K，如果p≡q，即[p]=[q],则对任何a∈∑，必有δ(p,a) ≡δ(q,a), 进而有[δ(p,a)]=[δ(q,a)]。

这样说明，一个等价类中，任哪一个元素作为此等价类的代表元素是无关紧要的。

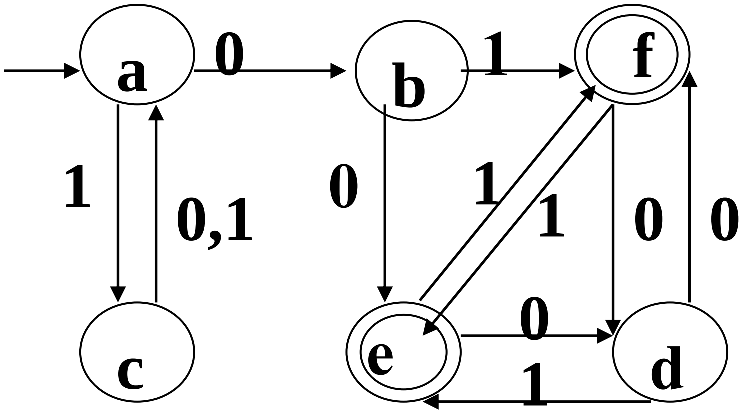
其次，用对|w|归纳证明：对任何 w∈Σ*，有

$$\delta'([q_0],w)=[\delta(q_0,w)]$$ 。

最后，证明T(M')=T(M)。

作业题

将下面有限自动机简化(要求有简化过程)。



2.6 正规集对运算的封闭性

所谓正规集，就是有限自动机所接收的语言，或者是正规表达式表示的语言，或者是正规文法产生的语言(后面要介绍正规文法与有限自动机之间的等价性)。

这一节将介绍正规集对并、交、补、乘积、*闭包及逆转运算的封闭性。

定理2-6.1 正规集对并、乘积、* 闭包运算是封闭的。即设L_1和L_2都是正规集,则$L_1 \cup L_2$、L_1L_2及L_1^*也是正规集。

证明： 用正规表达式证明。令r_1、r_2,分别是表示L_1、L_2的正规表达式。则r_1+r_2、r_1r_2 及r_1^*分别是表示$L_1 \cup L_2$、L_1L_2及L_1^*的正规表达式。所以$L_1 \cup L_2$、L_1L_2及L_1^*也是正规集。

定理2-6.2 正规集对补运算封闭。

即设L是正规集，且$L\subseteq\sum_1^*$，又$\sum_1\subseteq\sum_2$，$\sum_1$与$\sum_2$都是字母表，则$\sum_2^*-L$也是正规集。

分析： 为了便于理解证明过程，先用一个例子说明证明思想。

假设有一个有限自动机

$M=(K,\sum_1,\delta,q_0,F)$，

且$T(M)=L$，$\sum_1\subseteq\sum_2$，

- L中的所有符号串都使得M从q_0出发最后达到F中的某个状态而被M接收。

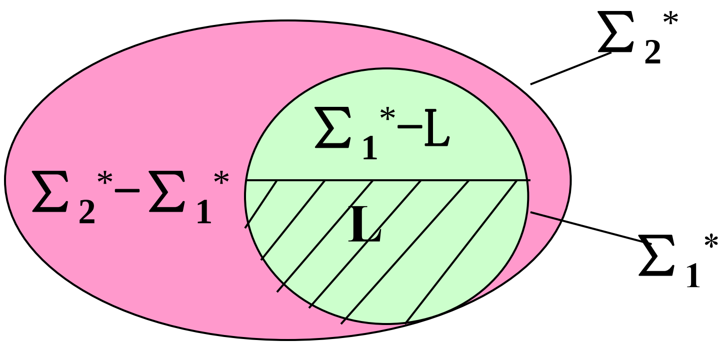


图2-6.1

- $\sum_1^*-L$中的所有符号串都使得M从q_0出发最后达到K－F中的某个状态，或者无可达状态，而不被M接收。

如图2-6.1所示。

构造M’，使得 $T(M')=\Sigma_2^*—L$。为此让M’如此工作：

1. M的终止状态集合F中的状态成为M’的非终止状态，从而不接收L中的符号串。

2. M’的终止状态集合 $F'=(K-F)\cup\{d\}$ 终止状态d称之为收容状态，

3. $\Sigma_1^*—L$中的所有符号串。使M’从开始状态从q_0出发最后达到K－F 中的状态，或者d状态而被M’接收。

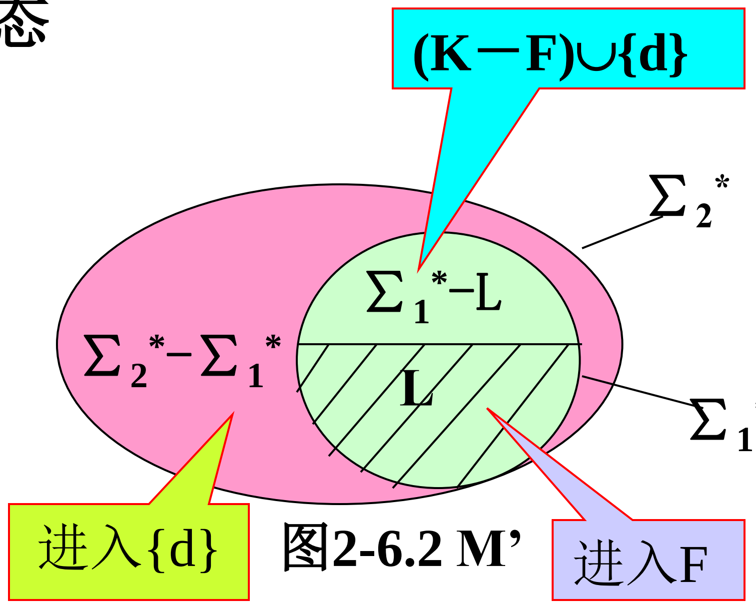


图2-6.2 M’

4.而$\Sigma_2^*—\Sigma_1^*$中的所有符号串都使M’从开始状态从q_0出发最后达到d状态而被M’接收。

*.这样就使得M能接收的符号 串M’都不接收，而M不接收符号串可被M’接收。

例如，给定接收0^*10^*的有限自动机M（$\sum_1=\{0,1\}$）如图2-6.3所示，T(M)=L，$\sum_2=\{0,1,2\}$,求一个有限自动机M’，使得$T(M')=\sum_2{}^*—L$。

按照上面思路构造M’如图2-6.4所示。

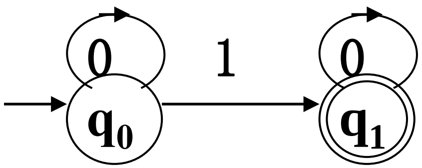


图2-6.3 M

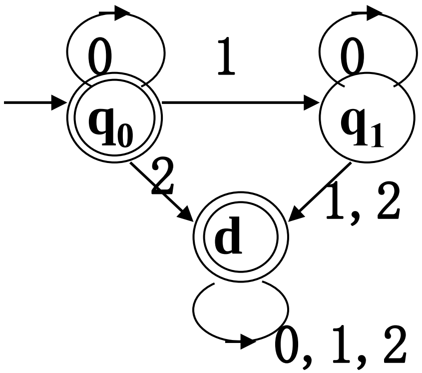


图2-6.4 M’

下面继续证明定理。

证明：令DFA $M=(K,\sum_1,\delta,q_0,F)$，且$T(M)=L,\sum_1\subseteq\sum_2$，$d\notin K$。现在构造一个有限自动机M', 使得$T(M')=\sum_2^*-L$，令$M'=(K\cup\{d\},\sum_2,\delta',q_0,(K-F)\cup\{d\})$,δ'定义如下：

1. 对任何$q\in K$，任何$a\in\sum_1$，如果$\delta(q,a)$有定义，则 $\delta'(q,a)=\delta(q,a)$；
2. 对任何$q\in K$，任何$a\in\sum_1$，如果$\delta(q,a)$无定义，则 $\delta'(q,a)=d$；
3. 对任何$q\in K$，任何$a\in\sum_2-\sum_1$，有$\delta'(q,a)=d$；
4. 对任何$a\in\sum_2$，有$\delta'(d,a)=d$。

于是任何$w\in T(M')$，当且仅当$\delta'(q_0,w)\in(K-F)\cup\{d\}$。因此$\delta(q_0,w)\notin F$，即$w\notin T(M)$，所以$w\in\sum_2^*—L$。所以$T(M')=\sum_2^*—L$。

当$\sum_2=\sum_1$时，$\sum_1^*—L=\overline{L}$ 所以如果L是正规集，则L的补集也是正规集。即正规集对补运算是封闭的。

例如，Lewis-P48

定理2-6.2’ 正规集对补运算封闭。

证明：设DFA $M=(K,\Sigma_1,\delta,q_0,F)$，则补语言$\overline{L}=\Sigma^*-T(M)$

被$\overline{M}=(K,\Sigma_1,\delta,q_0, K-F)$接受，也就是说，$\overline{M}$和M完全一样，只是恰好交换它们的终结状态和非终结状态。

回看前面例子（你看出问题了吗？）

定理2-6.3 正规集对交运算封闭。如果L_1和L_2都是正规集，则$L_1 \cap L_2$也是正规集。

证明：设L_1和L_2都是正规集，因为$L_1 \cap L_2 =$ ______ ，且根据正规集对并、补运算封闭，所以$L_1 \cap L_2$也是正规集。

下面介绍**语言的“逆转”运算R：**

L是一个语言，则L的逆转记作L^R。

$L^R = \{w^R | w \in L\}$。

w^R是符号串w的逆转，

例如w=0100，则$w^R=(0100)^R=0010$。

设L是正规集，则L的逆转L^R也是正规集。

分析：举例说明证明的思路。

给定有限自动机M如图2-6.5所示，它所接收的语言

T(M)=L的正规表达式是10^*，

L^R是0^*1，接收的有限自动机为M’如图2-6.6所示。

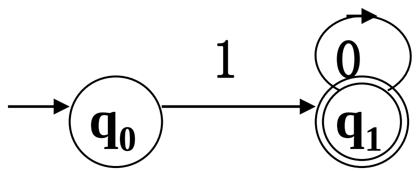


图2-6.5 M

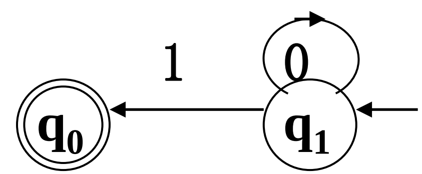
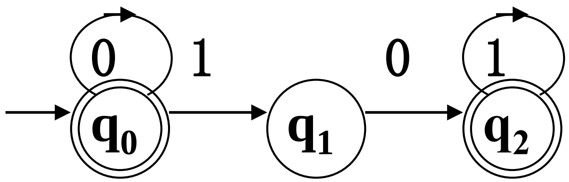


图2-6.6 M’

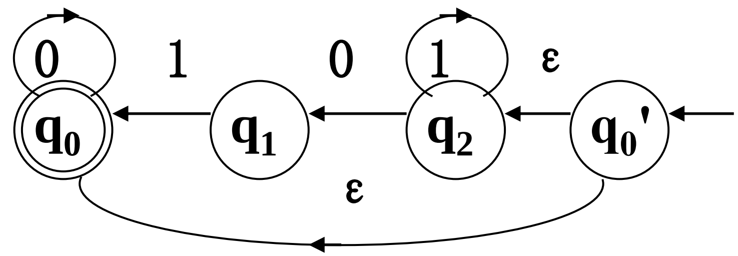
M’的构造方法是：将M的开始状态变成终止状态，将终止状态变成开始状态（因为M只有一个终止状态），再将所有有向边的方向颠倒即可（环的方向不必改变）。

如果M有两个终止状态，如何处理？



M:

图2-6.7



M’:

若如果M有多个终止状态，可以增加一个新的开始状态，构造M’如图2-6.7所示。

定理2-6.4 正规集在“逆转”运算下是封闭的。

证明：令DFA $M=(K,\sum,\delta,q_0,F)$，且$T(M)=L$，L是正规集。构造接收L^R的ε-NFA M’，

$M'=(K\cup\{q_0'\},\sum_1,\delta',q_0',\{q_0\})$，δ’构造如下：

(1) $\delta'(q_0',\varepsilon)=F$;

(2) 对任何$q\in K$，任何$a\in\sum$，如果$\delta(q,a)=p$，则$q\in\delta'(p,a)$。(即将M的有向图中有向边的方向颠倒)

很容易证明$T(M')=(T(M))^R=L^R$ 。

2.7　3型文法与有限自动机之间的等价性

通过证明3型文法与有限自动机之间的相互转换来说明：
　3型文法所产生的语言，也是正规集。

正规文法，它分成右线性和左线性两类。

右线性：产生式形式为：$A \to xB$, $C \to y$。

左线性：产生式形式为：$A \to Bx$, $C \to y$,

　A,B,C是非终极符，x,y是终极符串。

【例2-7.1】

$G_1=(\{S,A\},\{0,1\},P,S)$：$S \to 0A$，$A \to 10A$，$A \to \varepsilon$

$G_2=(\{S,A\},\{0,1\},P,S)$：$S \to A$，$A \to A10$，$A \to 0$

G_1就是右线性文法，G_2是左线性文法。容易看出它们产生的语言是相同的，都是$0(10)^*$。

1. 右线性和左线性相互转换

定理2-7.1 给定右线性文法G，当且仅当存在左线性文法G’，使得L(G)=L(G’)。

证明： 设文法$G=(V_N,V_T,P,S)$是右线性文法，不妨设$V_N=\{S,A_1,A_2,\ldots,A_n\}$，且S不出现在P中产生式的右侧（如果出现，就引进新的开始变元S_0，再增加产生式$S_0\rightarrow S$）。

下面构造左线性文法$G'=(V_N,V_T,P',S)$，

其中P’构成如下：对任何$A_i,A_j\in V_N$，任何$\alpha\in V_T^*$，

(1) $S\rightarrow\alpha$, 当且仅当 $S\rightarrow\alpha\in P$；

(2) $A_i\rightarrow\alpha$, 当且仅当 $S\rightarrow\alpha A_i\in P$；

(3) $A_i\rightarrow A_j\alpha$, 当且仅当 $A_j\rightarrow\alpha A_i\in P$；

(4) $S\rightarrow A_j\alpha$, 当且仅当 $A_j\rightarrow\alpha\in P$。

下面证明L(G’)=L(G)。

先证明$L(G')\subseteq L(G)$，任取$w\in L(G')$, 则G'有派生$S\Rightarrow^{*}w$，下面分两种情况讨论：

a) 如果此派生是一步完成的，即$S\Rightarrow w$，则有$S\rightarrow w\in P'$，由P'构成可知$S\rightarrow w\in P$，所以在G中也有派生$S\Rightarrow w$，所以$w\in L(G)$。

b) 如果此派生是多于一步完成的，不妨设此派生为：

$S\Rightarrow A_{i1}\alpha_1\Rightarrow A_{i2}\alpha_2\alpha_1\Rightarrow A_{i3}\alpha_3\alpha_2\alpha_1\Rightarrow\ldots\Rightarrow A_{im-1}\alpha_{m-1}\alpha_{m-2}\ldots\alpha_2\alpha_1$

$\Rightarrow\alpha_m\alpha_{m-1}\ldots\alpha_2\alpha_1=w$，则说明P'中有产生式：

$S\rightarrow A_{i1}\alpha_1$，$A_{i1}\rightarrow A_{i2}\alpha_2$，$A_{i2}\rightarrow A_{i3}\alpha_3$，…，$A_{im-2}\rightarrow A_{im-1}\alpha_{m-1}$，

$A_{im-1}\rightarrow\alpha_m$。

由P'构成可知在P中有相应产生式：

$A_{i1}\rightarrow\alpha_1$，$A_{i2}\rightarrow\alpha_2A_{i1}$，$A_{i3}\rightarrow\alpha_3A_{i2}$，…，$A_{im-1}\rightarrow\alpha_{m-1}A_{im-2}$，

$S\rightarrow\alpha_m A_{im-1}$。

于是G中有派生：

$S \Rightarrow \alpha_m A_{im-1} \Rightarrow \alpha_m \alpha_{m-1} A_{im-2} \Rightarrow \ldots \Rightarrow \alpha_m \alpha_{m-1} \ldots \alpha_2 A_{i1}$
$\Rightarrow \alpha_m \alpha_{m-1} \ldots \alpha_2 \alpha_1 = w$ 。

所以$w \in L(G)$，所以$L(G') \subseteq L(G)$。

用类似的方法可以证明$L(G) \subseteq L(G')$，

最后得 $L(G') = L(G)$。

根据此定理，以后我们讲正规文法时，它是右线性文法还是左线性文法是无关紧要的。

下面为了讨论正规文法与有限自动机之间的等价性，先介绍简单右线性文法。

2. 简单右线性文法

定义：设G是个右线性文法，如果它的所有产生式都具有如下形式：$A \rightarrow aB$, $C \rightarrow b$, $D \rightarrow \varepsilon$, 其中A,B,C,D是变元，a,b是终极符，则称G是个简单右线性文法。

任何一个右线性文法，都可以很容易地等价变换为简单右线性文法。

例如，上面文法G_1：$S \rightarrow 0A$, $A \rightarrow 10A$, $A \rightarrow \varepsilon$，其中 $A \rightarrow 10A$，可以变换为：$A \rightarrow 1B$, $B \rightarrow 0A$。

最后得到与之等价的简单右线性文法:

$G_1' = (\{S,A,B\},\{0,1\},P',S)$，其中P'为：

$S \rightarrow 0A$, $A \rightarrow 1B$, $B \rightarrow 0A$, $A \rightarrow \varepsilon$，

3．根据正规文法，求与之等价的有限自动机

定理2-7.2设G是个正规文法，则可以构造有限自动机 M，
使之T(M)=L(G)。

证明：假设G已经是简单的右线性文法，否则，要先将G等价变换为简单的右线性文法，

令$G=(V_N, V_T, P, S)$，设$E \notin V_N$。

构造一个ε-NFA $M=(K,\sum,\delta,q_0,F)$，

其中$K=V_N \cup \{E\}$，$\sum=V_T$，$q_0=S$，$F=\{E\}$，即

$M=(V_N \cup \{E\}, V_T, \delta, S, \{E\})$，其中δ构成如下：

对任何$A,B,C \in V_N$，任何$a,b \in V_T$，则

（1）$B \in \delta(A,a)$ 当且仅当 $A \to aB \in P$

（2）$E \in \delta(B,b)$ 当且仅当 $B \to b \in P$

（3）$E \in \delta(C,\varepsilon)$ 当且仅当 $C \to \varepsilon \in P$

例如，对于上面给定的简单右线性文法G_1’：

$S \to 0A,\ A \to 1B,\ B \to 0A,\ A \to \varepsilon$，

按照此方法，构成与G_1’等价的有限自动机M如图2-7.1所示。可见M接收的语言也是$0(10)^*$。

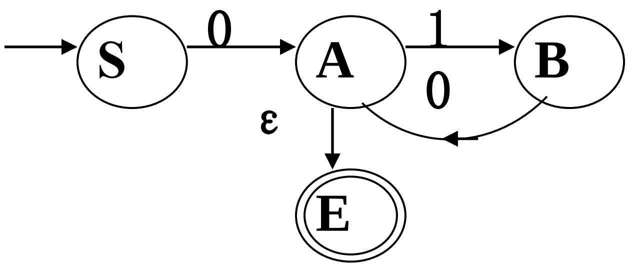


图2-7.1 M

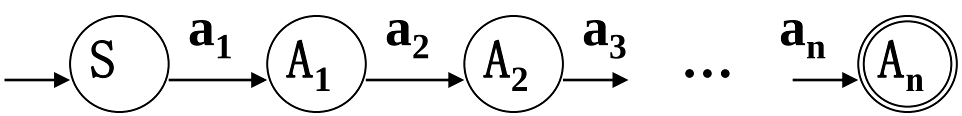
下面继续证明T(M)=L(G)。

证明$T(M) \subseteq L(G)$，任取$w \in T(M)$，

下面分两种情况讨论：

a).如果$|w| \le 1$，则$w = a$（$a \in \sum$，即$a \in V_T$）或者$w = \varepsilon$，这说明$E \in \delta(S,a)$或者$E \in \delta(S,\varepsilon)$，由δ的构成可知$S \to a \in P$或者$S \to \varepsilon \in P$，于是有$a \in L(G), \varepsilon \in L(G)$，即$w \in L(G)$。

b).如果$|w|\geq 2$，令$w=a_1a_2\ldots a_n$，$a_i \in V_T \cup \{\varepsilon\}$ $(i=1,2,\ldots,n)$，不妨设M在识别w时的路径如图2-7.2所示。



其中, A_n=E

图2-7.2

于是G中有产生式：$S\rightarrow a_1A_1$，$A_1\rightarrow a_2A_2,\ldots, A_{n-1}\rightarrow a_n$，所以G中有派生：

$S\Rightarrow a_1A_1\Rightarrow a_1a_2A_2\Rightarrow\ldots\Rightarrow a_1a_2\ldots a_{n-1}A_n\Rightarrow a_1a_2\ldots a_{n-1}a_n=w$，所以，$w\in L(G)$。

所以有$T(M)\subseteq L(G)$。

类似可以证明$L(G)\subseteq T(M)$，最后可得$T(M)=L(G)$。

4．根据有限自动机，求与之等价的正规文法

定理2-7.3 给定确定有限自动机$M=(K,\sum,\delta,q_0,F)$，则存在正规文法G，使得$L(G)=T(M)$。

证明：构造右线性文法$G=(K,\sum,P,q_0)$，其中

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

显然有下面命题成立：

对任何$x\in V_T^*$,$\delta(q,x)=p$,当且仅当 G中有派生 $q\Rightarrow^* xp$。

(此命题的证明与定理2-7.2证明b)的过程类似，这里从略)

下面证明$T(M)=L(G)$

先证$T(M)\subseteq L(G)$，任取$w\in T(M)$，

如果$w=\varepsilon$，则说明有$\delta(q_0,\varepsilon)\in F$。由P的构成得P中有产生式$q_0\rightarrow\varepsilon$，所以$\varepsilon\in L(G)$;

如果$w \neq \varepsilon$，不妨是$w=xa$且有

$\delta(q_0,xa)=\delta(\delta(q_0,x),a)=\delta(p,a)\in F$，

其中$\delta(q_0,x)=p$，再根据前命题得$q_0 \Rightarrow^* xp$ 。

又由$\delta(p,a)\in F$， 得P中有产生式 $p\to a$，于是G中有派生：

$q_0 \Rightarrow^* xp \Rightarrow xa=w$，所以$w\in L(G)$。

反之，类似可证$L(G)\subseteq T(M)$，最后可得$T(M)=L(G)$。

由有限自动机M，求一个与之等价的左线性文法G时，有两种方法：

1.可以先按照上述方法，先求一个与之等价的右线性文法后，再将此文法等价地转换为左线性文法。

2.另外一种方法，在介绍此方法之前，先介绍一个文法"逆转"的实例，然后再介绍有关定理。

例如，上面给定右线性文法$G_1=(\{S,A\},\{0,1\},P,S)$，

P：$S\rightarrow 0A$，$A\rightarrow 10A$，$A\rightarrow \varepsilon$。

显然G_1所产生的语言为：$0(10)^*$。

构造文法$G_1'=(\{S,A\},\{0,1\},P',S)$，

P’：$S\rightarrow A0$, $A\rightarrow A01$, $A\rightarrow \varepsilon$。

其中，$P'=\{A\rightarrow \alpha^R | A\rightarrow \alpha\in P$，其中$\alpha^R$是$\alpha$的逆转$\}$。

显然G_1'产生的语言是$(01)^*0$，其刚好是$0(10)^*$的逆转。

定理2-7.4 设$G=(V_N,V_T,P,S)$是右线性文法，则构造一个左线性文法$G'=(V_N,V_T,P',S)$，

其中 $P'=\{A\rightarrow \alpha^R | A\rightarrow \alpha\in P$，其中$\alpha^R$是$\alpha$的逆转$\}$，

则$L(G')=(L(G))^R$。

证明：先用归纳法(对推导步数归纳)证明如下结论：对任何$\alpha \in (V_N \cup V_T)^*$，在G中有派生$S \Rightarrow^* \alpha$，当且仅当 在G'中有派生 $S \Rightarrow^* \alpha^R$。

（1）如果G中一步派生出α，即有$S \Rightarrow \alpha$，则说明$S \rightarrow \alpha \in P$，根据P'的构成可知，当且仅当 P'中有产生式$S \rightarrow \alpha^R$，于是G'中有派生$S \Rightarrow \alpha^R$，结论成立。

（2）假设G中有派生 $S \Rightarrow^* \alpha$，是少于k步完成的，当且仅当 在G'中有派生$S \Rightarrow^* \alpha^R$。

（3）当$S \Rightarrow^* \alpha_1 B \Rightarrow \alpha_1 \alpha_2 = \alpha$是由k步完成时，则$S \Rightarrow^* \alpha_1 B$是由k-1步完成的，由假设（2）得，当且仅当在G'中有派生 $S \Rightarrow^* (\alpha_1 B)^R = B\alpha_1^R$，又在第k步派生 $\alpha_1 B \Rightarrow \alpha_1 \alpha_2$中，实际上是应用产生式$B \rightarrow \alpha_2$，由P'构成得，当且仅当P'中有产生式$B \rightarrow \alpha_2^R$，于是G'中有派生：

$S\Rightarrow^*(\alpha_1B)^R=B\alpha_1^R\Rightarrow\alpha_2^R\alpha_1^R=(\alpha_1\alpha_2)^R=(\alpha)^R$。

所以结论成立。

于是有任何 $w\in L(G)$，即 $S\Rightarrow^* w$，当且仅当 在G'中有派生 $S\Rightarrow^* w^R$，$w^R\in L(G')$，所以 $L(G')=(L(G))^R$。

【例2-7.2】给定DFA M如图2-7.3所示，分别求一个右线性文法和一个左线性文法使得产生的语言都是T(M)。

解：先求右线性文法G，令

G=({S,A,B,C,},{0,1},P,S),其中

P：　$S\rightarrow 0A|1C|0|1$

$A\rightarrow 1B|0C|0$

$B\rightarrow 0A|1C|0|1$

$C\rightarrow 0C|1C|0|1$

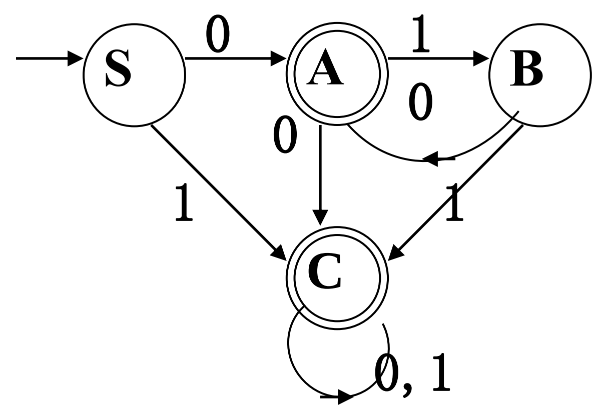


图2-7.3　M:

求左线性文法，有两种方法：

一种方法是：根据右线性文法G求左线性文法G'：

按照定理2-7.1中给定的方法，得

由P中S→0|1，　　得P'中有S→0|1；

由P中S→0A|1C，得P'中有A→0,C→1；

由P中A→1B|0C，得P'中有B→A1, C→A0；

由P中A→0，　　　得P'中有S→A0；

由P中B→0A|1C，得P'中有A→B0, C→B1；

由P中B→0|1，　　得P'中有S→B0|B1；

由P中C→0C|1C，得P'中有C→C0, C→C1；

由P中C→0|1，　　得P'中有S→C0|C1。

最后得P'：S→A0|B0|B1|C0|C1|0|1；

A→B0|0；

B→A1；

C→A0|B1|C0|C1|1。

另一种方法是：分成如下三步完成。

a）先将M逆转成ε-NFA M’，如图2-7.4所示。显然T(M’)=(T(M))^R。

b）再根据M’求一个与之等价的右线性文法G：

G=({D,S,A,B,C},{0,1},P,D)

其中P中生成式为：

D→A|C

A→0S|0B|0

B→1A

C→0A|1B|1S|1C|0C|1

由于没有变元S的产生式，所以其中的A→0S，C→1S可去掉。

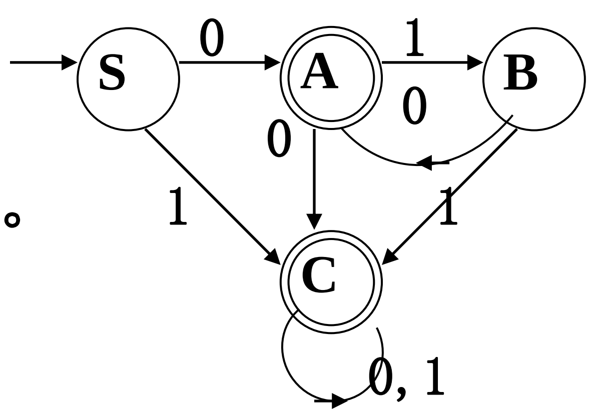


图2-7.3 M:

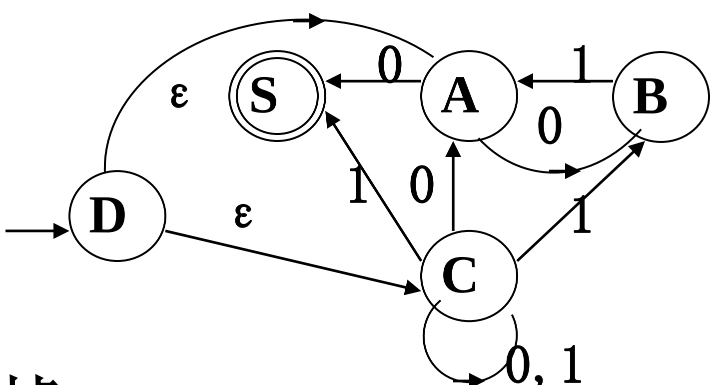


图2-7.4 M’

最后得P: $D \to A|C$

$A \to 0B|0$

$B \to 1A$

$C \to 0A|1B|1C|0C|1$

显然L(G)=T(M’)。

c）将右线性文法G逆转成左线性文法G’，使得$L(G')=(L((G))^R$。

令G’=({D,S,A,B,C},{0,1},P’,D)

其中P’为： $D \to A|C$

$A \to B0|0$

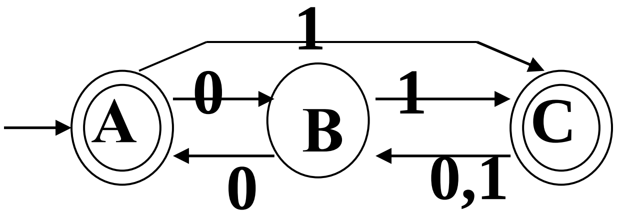
$B \to A1$

$C \to A0|B1|C1|C0|1$

显然$L(G')=(L((G))^R=(T(M'))^R=((T(M))^R)^R=T(M)$。

作业题

1．给定DFA M如图所示。求一个左线性文法G，使得 L(G)=T(M)。



2．首先构造一个右线性文法G，使得

$L(G)=\{a^i b^j|i,j\geq 0\}\cup\{c^k|k\geq 0\}$

再构造一个有限自动机M，使得T(M)=L(G)。

3．给定右线性文法G=({S,B,C,D},{0,1}, P, S)，其中P：

$S\rightarrow B \mid C$，　　$B\rightarrow 0B \mid 1B \mid 011$,

$C\rightarrow 0D \mid 1C \mid \varepsilon$,　$D\rightarrow 0C \mid 1D$

试求一个FA M，使得 T(M)=L(G)。

2.8 正规集的判定问题

这一节将讨论如下四个判定问题：

☆ 某个语言不是正规集；

☆ 给定的正规集是否为空集；

☆ 给定正规集不是空集时，那么它是有穷集合还是无穷集合；

☆ 以及判定两个自动机是否等价。

在判定这些问题时，都用到一个引理——正规集的泵作用引理。

下面先介绍这个引理。

1．正规集的泵作用(pumping)引理

给定一个确定的有限自动机$M=(K,\sum,\delta,q_0,F)$，设$|K|=n$，考虑一个输入符号串$z\in\sum^*,|z|\geq n$,不妨设

$$z=a_1a_2\ldots a_{n-1}a_n,$$

如果$z\in T(M)$，设M识别z的路径如图2-8.1所示。(其中$q_n\in F$)。

$$\longrightarrow q_0 \xrightarrow{a_1} q_1 \xrightarrow{a_2} q_2 \xrightarrow{a_3} \cdots \xrightarrow{a_n} q_n$$

图2-8.1

从此图看出，图中有n+1状态，n+1＞n，而M中只有n个不同状态，根据鸽洞原理, 上面图中至少有两个状态相同，不妨设$q_j=q_k$,(k≥j $0\leq j<k\leq n$)，于是上面图就变成如图2-8.2所示的图。

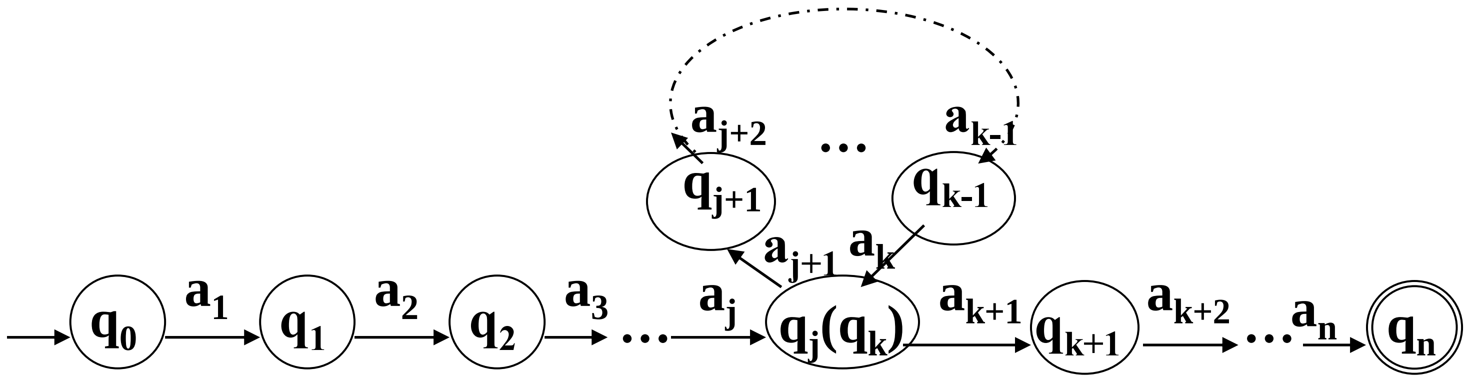


图2-8.2

此图可以简化如图2-8.3所示。

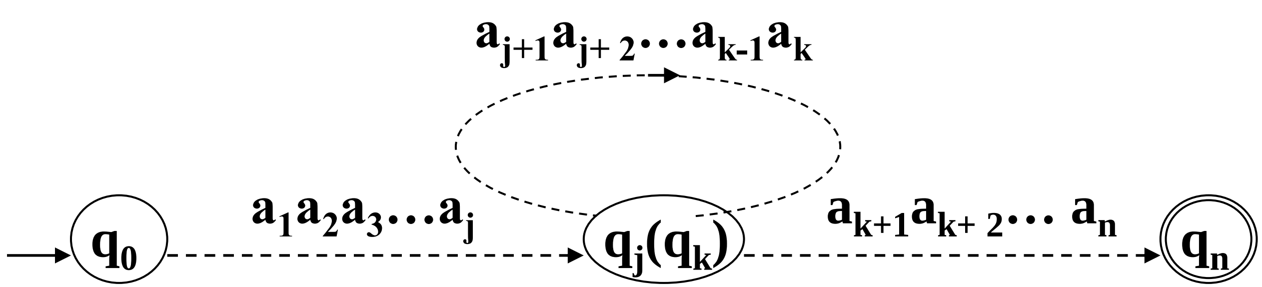


图2-8.3

可将z写成$z=(a_1a_2a_3\dots a_j)(a_{j+1}a_{j+2}\dots a_{k-1}a_k)(a_{k+1}a_{k+2}\dots a_n)$，

因为$z\in T(M)$，所以对任何整数$i\geq 0$，有$(a_1a_2a_3\ldots a_j)(a_{j+1}a_{j+2}\ldots a_{k-1}a_k)^i(a_{k+1}a_{k+2}\ldots a_n)\in T(M)$。

如果设$u=a_1a_2a_3\ldots a_j$，$v=a_{j+1}a_{j+2}\ldots a_{k-1}a_k$，$w=a_{k+1}a_{k+2}\ldots a_n$，于是$z=uvw$，$|v|\geq 1$，$|uv|\leq n$,且对任何$i\geq 0$，有$uv^iw\in T(M)$，其中$v^i$就是从$q_j$到$q_k$的循环i次。这就引出正规集的泵作用引理。

引理2-8.1（正规集的泵作用引理）：设L是正规集，则必存在一个常数n，使得任何$z\in L$，只要$|z|\geq n$，就可以将z写成$z=uvw$，其中$|uv|\leq n$，$|v|\geq 1$，且对任何整数$i\geq 0$，有$uv^iw\in L$。

证明：因为L是个正规集，必存在有限自动机M，使得$T(M)=L$，设 $M=(K,\sum,\delta,q_0,F)$，设$|K|=n$，对任何$z\in L$，只要$|z|\geq n$，如上面所述，就可以将z写成$z=uvw$，其中$|uv|\leq n$，$|v|\geq 1$，且对任何整数$i\geq 0$，有$uv^iw\in L$。

2．判断某个语言不是正规集

【例2-8.1】证明语言 L={ 0^{i^2} |i是自然数}不是正规集。

证明：(用反证法)

（1）假设L是个正规集。

（2）设n是L满足正规集泵作用引理常数。

（3）设z＝0^{n^2} ，|z|=n^2>n，显然z∈L，于是根据正规集的泵作用引理，可将z写成：z=uvw，其中|uv|≤n，|v|≥1，且对任何整数i≥0，有uv^iw∈L。

（4）适当地选取i，找矛盾。从L的定义看出，只有0的个数是个完全平方数，才是L中的句子。所以就设法选取一个i，使得|uv^iw|不是完全平方数。

取i=2，因为n^2＜|uv^2w|≤n^2+n＜$(n+1)^2$，可见|uv^2w|不是完全平方数，所以uv^2w∉L，这与泵作用引理矛盾。所以L不是正规集。

2．判断某个语言不是正规集

【例2-8.2】证明语言 $L=\{0^m1^m|m\geq1\}$不是正规集。

证明：(用反证法)

（1）假设L是个正规集。

（2）设n是L满足正规集泵作用引理常数。

（3）设$z= 0^n1^n$，$|z|=2n>n$，显然$z\in L$，于是根据正规集的泵作用引理，可将z写成：$z=uvw$，其中$|uv|\leq n$，$|v|\geq1$，且对任何整数$i\geq0$，有$uv^iw\in L$。

（4）简单考虑，设$|uv|=n$，$|v|=0^k$，则$|u|= 0^{n-k}$,而$w=1^n$

|----n------|-----n---|

$$z = \underbrace{0\ldots0}_{u}\underbrace{0\ldots0}_{v}\underbrace{1\ldots\ldots\ldots1}_{w}$$

依泵引理，$uv^iw\in L$ 对任何$i\geq0$成立。则当$i=0$时，有$uw\in L$。但依语言L的定义，$uw= 0^{n-k}1^n \notin L$ 这与泵引理矛盾。所以L不是正规集。

3．正规集的有穷性判定

定理2-8.1 具有n个状态的有限自动机M接收的语言T(M)：

（1）是非空集， 当且仅当 M接收一个长度小于n的句子。

（2）是无穷集合，当且仅当 M接收一个长度大于或等于n而小于2n的句子。

证明：（1）充分性显然成立。即如果M接收一个长度小于n的句子，则$T(M)\neq\Phi$。

下面证明必要性：已知$T(M)\neq\Phi$，假设T(M)中没有一个长度小于n的句子。又令$z\in T(M)$,且z是T(M)中长度最短的句子。假设z的长度不小于n，即$|z|\geq n$，根据正规集的泵作用引理得，可将z写成z=uvw，其中$|uv|\leq n$,，$|v|\geq 1$，且对任何整数$i\geq 0$，有$uv^iw\in T(M)$。取i=0，得$uv^0w=uw\in L$，$|uw|<|z|$，这与z是T(M)中长度最短的句子矛盾。所以T(M)中必有一个长度小于n的句子。

（2）充分性，已知有$z \in T(M)$，且$n \le |z| < 2n$。根据正规集的泵作用引理，可将z写成：$z=uvw$，其中$|uv| \le n$，$|v| \ge 1$，且对任何整数$i \ge 0$，有$uv^iw \in T(M)$。因为i的取值有无穷多个$(i=1,2,3,4,\ldots)$，因而，这样的句子有无穷多个，所以T(M)是无限集合。

必要性，已知T(M)是无限集合，假设T(M)中没有一个长度大于或等于n而小于2n的句子。因T(M)是无限集合，则必存在$z \in T(M)$，且$|z| \ge n$，令z是长度大于n的句子中长度最短的句子。假设|z|不小于2n，根据正规集泵作用引理得，可将z写成$z=uvw$，其中$|uv| \le n$，$|v| \ge 1$，且对任何整数$i \ge 0$有$uv^iw \in T(M)$。取$i=0$，得$uv^0w = uw \in T(M)$，则

a）要么$n \le |uw| < 2n$。

b）要么z 不是T(M)中长度大于n的句子中最短的句子，产生矛盾。

所以T(M)中必有一个长度大于或等于n而小于2n的句子。必要性成立。

所以T(M)是无穷集合，当且仅当 M接收一个长度大于或等于n而小于2n的句子。

按照上面定理，可以说，存在算法用以判定具有n个状态的有限自动机M接收的语言T(M)是否为非空集、有穷集或者为无穷集合。

4．判定两个自动机等价

定理2-8.2 存在算法，用以判定两个自动机是否等价(即它们接收的语言是否相等)。

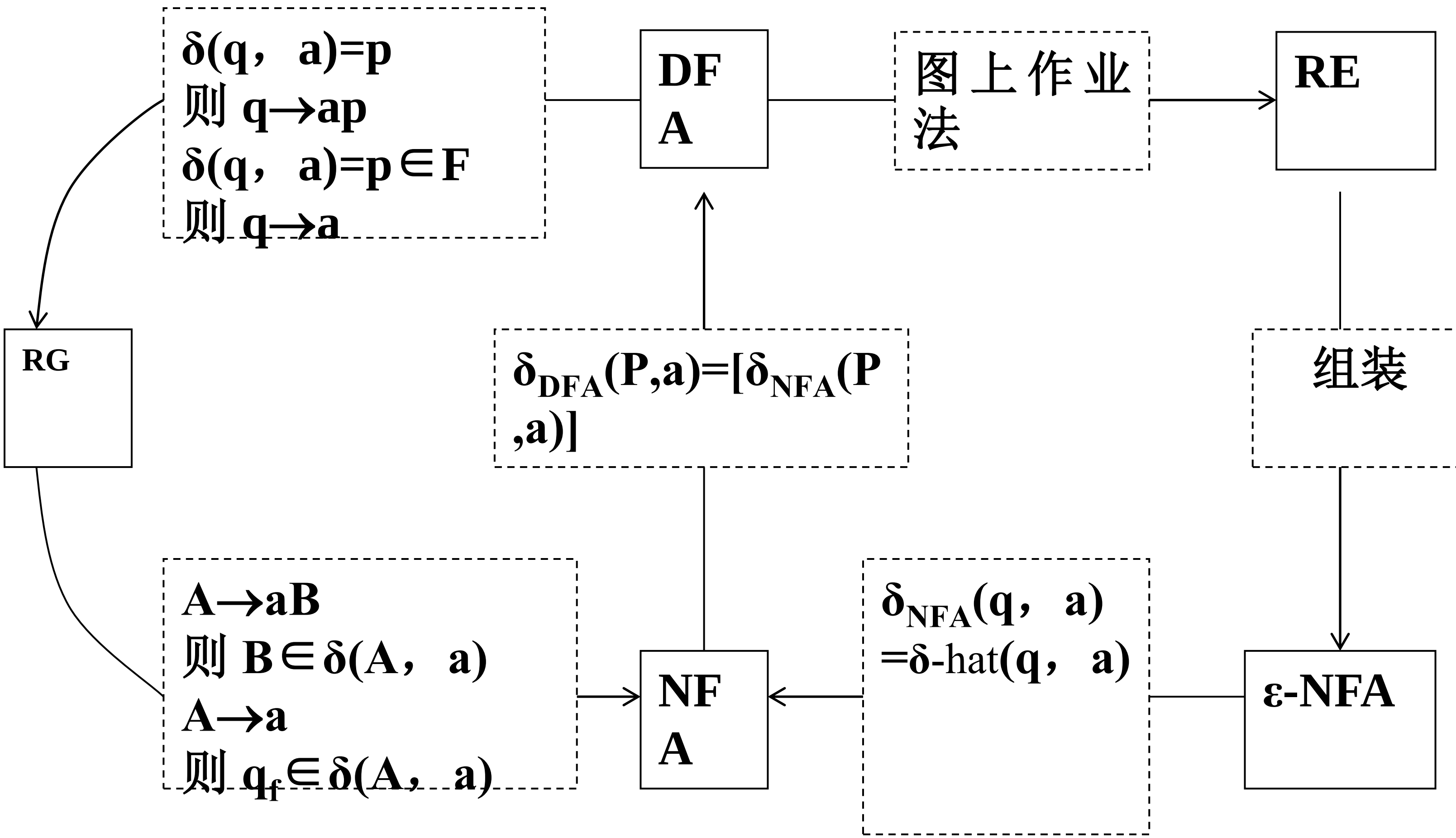
证明：令有限自动机M_1与M_2接收语言分别是L_1和L_2。根据正规集对交、并、补运算封闭得,可以由一个有限自动机M接收。即

$$T(M)=(\overline{L_1}\cap L_2)\cup(L_1\cap\overline{L_2})$$

而T(M)=Φ，当且仅当 $L_1=L_2$。根据定理2-8.1得存在算法用以判定具有n个状态的有限自动机M接收的语言T(M)是否为非空集。即存在算法用以判定是否有$L_1=L_2$。

- 作业题
- 证明L＝$\{a^i | i$是个素数$\}$不是正规集。

4.4 正则语言等价模型的总结



- 鸽洞原理:

  若有n个笼子和n+1只鸽子，所有的鸽子都被关在鸽笼里，那么至少有一个笼子有至少2只鸽子。

back