形式语言与自动机期末速通

4. ε-NFA

4.1 ε-NFA的形式定义

[**定义4.1.1**] **带空移动的不确定的有穷状态自动机**(Non-deterministic Finite Automaton with ε-moves, ε-NFA)定义为五元组$M=(Q,\Sigma,\delta,q_0,F)$,其中$Q,\Sigma,q_0,F$的含义同DFA,$\delta:Q\times(\Sigma\bigcup\{\varepsilon\})\rightarrow 2^Q$,其中:

(1)非空转移$\delta(q,a)=\{p_1,\cdots,p_m\}\ \ (q,a)\in Q\times\Sigma$表示$M$在状态$q$读入字符$a$,可选择地将状态变为$p_1,\cdots,p_m$,并将读头向右移动一个带方格,指向读入字符串的下一个字符.

(2)空转移$\delta(q,\varepsilon)=\{p_1,\cdots,p_m\}\ \ (q\in Q)$表示$M$在状态$q$不读入任何字符,可选择地将状态变为$p_1,\cdots,p_m$,也成$M$在状态$q$做的一个**空移动**或$\varepsilon$**移动**.

[**注1**] ε-NFA允许以空串为输入的状态跳转,这些跳转可同时进行,比朴素的NFA更便于构造,更加"智能",但也只能接受RL.

[**定义4.1.2**] 设ε-NFA $M=(Q,\Sigma,\delta,q_0,F)$.

(1)对$q\in Q$,定义ε-NFA的状态的**闭包**$\varepsilon-CLOSURE(q)$为从状态q出发,沿带ε标记的弧所能到达的状态的集合,即$\varepsilon-CLOSURE(q)=\{p\mid\exists\text{从}q\text{到}p\text{的有标记}\varepsilon\text{的路径}\}$.

(2)对状态的集合P,定义ε-NFA的状态集合的**闭包**$\varepsilon-CLOSURE(P)$为从P中所有状态的闭包的并,即$\varepsilon-CLOSURE(P)=\bigcup\limits_{p\in P}\varepsilon-CLOSURE(p)$.

[**注**] 将M中的δ扩充为$\hat{\delta}:Q\times(\Sigma^*\bigcup\{\varepsilon\})\rightarrow 2^Q$.对$\forall q\in Q,w\in\Sigma^*,a\in\Sigma$,定义:

(1)$\hat{\delta}(q,\varepsilon)=\varepsilon-CLOSURE(q)$.

(2)$\hat{\delta}(q,wa)$定义为:从$\delta(q,w)=S$中的状态出发,对所有的$p\in S$,求$\varepsilon-CLOSURE(\delta(p,a))$后取并,

即$\hat{\delta}(q,wa)=\varepsilon-CLOSURE(P)$,其中$P=\{p\mid\exists r\in\hat{\delta}(q,w)\ s.t.\ p\in\delta(r,a)\}=\bigcup\limits_{r\in\hat{\delta}(r,w)}\delta(r,a)$.

(3)$\hat{\delta}(q,w)$是从状态q出发,沿带w标记的路径所能到达的状态的集合.

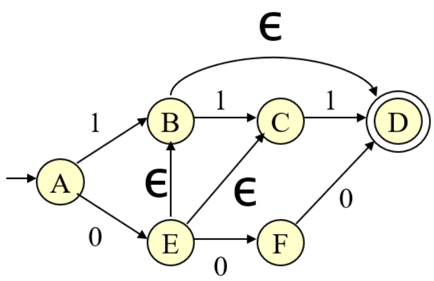
(4)M接受的语言的集合为$\{w\mid w\in\Sigma^*\wedge\hat{\delta}(q_0,w)\text{包含接受状态}\}$,即$L(M)=\{w\mid w\in\Sigma^*\wedge\hat{\delta}(q_0,w)\bigcap F\neq\varnothing\}$.

(5)$\hat{\delta}(q,\varepsilon)=\varepsilon-CLOSURE(q)$.

(6)对$(P,a)\in 2^Q\times\Sigma$,定义$\delta(P,a)=\bigcup\limits_{q\in P}\delta(q,a),\hat{\delta}(P,w)=\bigcup\limits_{q\in P}\hat{\delta}(q,w)$.

(7)在ε-NFA中,对$a\in\Sigma,q\in Q$,一般$\hat{\delta}(q,a)\neq\delta(q,a)$,故需严格区分.

[**例4.1.1**] 对下图所示的ε-NFA,有:



(1)$\varepsilon-CLOSURE(A)=\{A\},\varepsilon-CLOSURE(E)=\{B,C,D,E\}.$

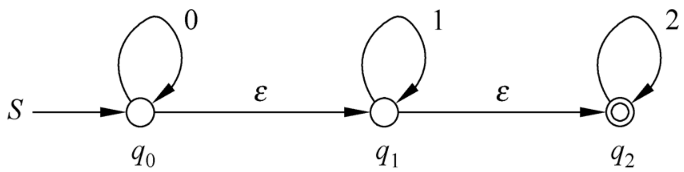
(2)$\varepsilon-CLOSURE(\{A,E\})=\{A,B,C,D,E\}.$

(3)$\hat{\delta}(A,\varepsilon)=\varepsilon-CLOSURE(A)=\{A\}.$

(4)$\hat{\delta}(A,0)=\varepsilon-CLOSURE(E)=\{B,C,D,E\}.$

(5)$\hat{\delta}(A,01)=\varepsilon-CLOSURE(C,D)=\{C,D\}.$

[例4.1.2] 下图是一个接受语言$\{0^n n 1^m 2^k\mid n,m,k\geq 0\}$的$\varepsilon-$NFA:



其δ和$\hat{\delta}$为:

状态	δ				$\hat{\delta}$			
	ε	**0**	**1**	**2**	ε	**0**	**1**	**2**
q_0	$\{q_1\}$	$\{q_0\}$	Φ	Φ	$\{q_0,q_1,q_2\}$	$\{q_0,q_1,q_2\}$	$\{q_1,q_2\}$	$\{q_2\}$
q_1	$\{q_2\}$	Φ	$\{q_1\}$	Φ	$\{q_1,q_2\}$	Φ	$\{q_1,q_2\}$	$\{q_2\}$
q_2	Φ	Φ	Φ	$\{q_2\}$	$\{q_2\}$	Φ	Φ	$\{q_2\}$

4.2 NFA与ε-NFA的等价性

[定理4.2.1] NFA与ε-NFA等价.

[证] 显然NFA都是不包含空转移的ε-NFA,故只需对任一$\varepsilon-$NFA,构造一个与之等价的NFA.

注意到$\hat{\delta}(q,wa)=\varepsilon-CLOSURE\left(\bigcup\limits_{r\in P}\delta(r,a)\right)$,故只需消除空转移即可.

对ε-NFA $M_1=(Q,\Sigma,\delta_1,q_0,F)$,取NFA $M_2=(Q,\Sigma,\delta_2,q_0,F_2)$,

其中$F_2=\begin{cases}F\bigcup\{q_0\},F\bigcap\varepsilon-CLOSURE(q_0)\neq\varnothing\\F,otherwise\end{cases},\delta_2(q,a)=\hat{\delta}_1(q,a)$.

4.3 识别RL的FA的构造

4.3.1 识别右线性文法的FA

[**DFA处理句子的特性**] DFA $M=(Q,\Sigma,\delta,q_0,F)$处理句子$a_1\cdots a_n$时,

(1)M按句子$a_1\cdots a_n$中字符出现的顺序,从q_0状态开始,依次处理字符$a_1,\cdots,a_n$,每处理一个字符进入一个状态,最后停在某一终止状态.

(2)M每次处理恰一个字符,其中第$i\ \ (1\leq i\leq n)$步处理输入字符a_i.这对应于使用转移式$\delta(q,a)=p$,即在q状态完成对字符a的处理,再由p状态继续处理后续字符.

①状态转移:

$A_0\Rightarrow a_1A_1$ *使用产生式$A_0\rightarrow a_1A_1$.

$\Rightarrow a_1a_2A_2$ *使用产生式$A_1\rightarrow a_2A_2$.

$\cdots$

$\Rightarrow a_1\cdots a_{n-1}A_{n-1}$ *使用产生式$A_{n-2}\rightarrow a_{n-1}A_{n-1}$.

$\Rightarrow a_1\cdots a_n$ *使用产生式$A_{n-1}\rightarrow a_n$.

②即时描述:

$q_0a_1\cdots a_n$

$\vdash a_1q_1a_2\cdots a_{n-1}a_n$ *使用转移函数$\delta(q_0,a_1)=q_1$.

$\cdots$

$\vdash a_1\cdots a_nq_n$ *使用转移函数$\delta(q_{n-1},a_n)=q_n$,其中q_n为终止状态.

③让上述的A_i与$q_i\ \ (0\leq i\leq n)$对应即得RL的推导与DFA相互模拟的方式.

(3)$\delta(q,a)=p\in F$且a是输入串的最后一个字符时,M完成对该输入串的处理.

[**定理4.3.1**] FA接受的语言是RL.

[证]

(1)构造:让RG派生对应的DFA的移动.

设DFA $M=(Q,\Sigma,\delta,q_0,F)$.

取右线性文法$G=(Q,\Sigma,P,q_0)$,其中$P=\{(q\rightarrow qp)\mid\delta(q,a)=p\}\bigcup\{(q\rightarrow a)\mid\delta(q,a)=p\in F\}$.

(2)证明$L(G)=L(M)\backslash\{\varepsilon\}$.

对$a_1\cdots a_n\in\Sigma^+$,

$q_0\Rightarrow^+a_1\cdots a_n$

$\Leftrightarrow q_0\rightarrow a_1q_1,q_1\rightarrow a_2q_2,\cdots,q_{n-2}\rightarrow a_{n-1}q_{n-1},q_{n-1}\rightarrow a_n\in P$.

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

$\Leftrightarrow\delta(q_0,a_1\cdots a_n)=q_n\in F$.

$\Leftrightarrow a_1 \cdots a_n \in L(M)$.

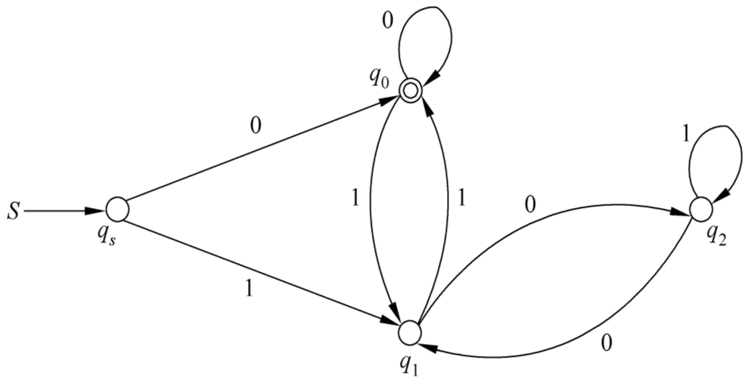
(3)空串的处理.

①若$q_0 \notin F$,则$\varepsilon \notin L(M)$,此时$L(G) = L(M)$.

②若$q_0 \in F$,则存在RG $G'\ s.t.\ L(G') = L(G) \bigcup \{\varepsilon\} = L(M)$.

综上,对$\forall$DFA M,$\exists$RG $G'\ s.t.\ L(G) = L(M)$.

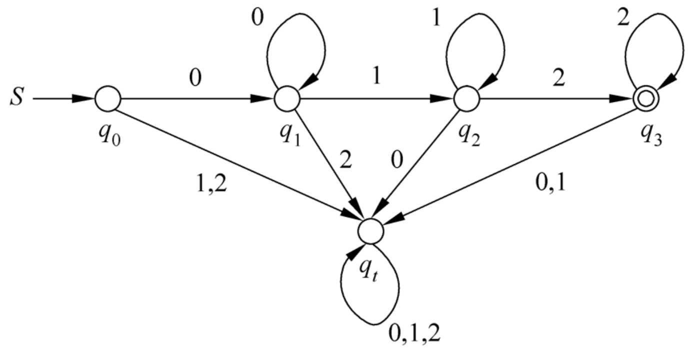
[**例4.3.1**]



与上图所示的DFA等价的RG为:

$q_s \to 0|0q_0|1q_1, q_0 = 0|0q_0|1q_1, q_1 = 0q_2|1|1q_0, q_2 \to 0q_1|1q_2$.

[**例4.3.2**]



与上图所示的DFA等价的RG为:

$q_0 \to 0q_1|1q_t|2q_t, q_1 \to 0q_1|1q_2|2q_t, q_2 \to 0q_t|1q_2|2q_3|2, q_3 \to 0q_t|1q_t|2q_3|2, q_t \to 0q_t|1q_t|2q_t$.

[**定理4.3.2**] RL可被FA接受.

[**证**]

(1)构造:用FA模拟RG的派生.

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

取FA $M = (V \bigcup \{Z\}, T, \delta, S, \{Z\})$,其中$Z \notin V$.

对$a \in T, A \in V$,定义$\delta(A, a) = \begin{cases} \{B \mid (A \to aB) \in P \bigcup \{Z\}\}, (A \to a) \in P \\ \{B \mid (A \to aB) \in P\}, (A \to a) \notin P \end{cases}$.

用$B \in \delta(A, a)$对应于产生式$A \to aB$,用$Z \in \delta(A, a)$对应于产生式$A \to a$.

(2)证明$L(M)=L(G)$.

对$a_1\cdots a_n\in T^+$,

$a_1\cdots a_n\in L(G)$

$\Leftrightarrow S\Rightarrow^+ a_1\cdots a_n$.

$\Leftrightarrow S\Rightarrow a_1A_1\Rightarrow a_1a_2A_2\Rightarrow\cdots\Rightarrow a_1\cdots a_{n-1}A_{n-1}\Rightarrow a_1\cdots a_n$.

$\Leftrightarrow S\to a_1A_1,A_1\to a_2A_2,\cdots,A_{n-2}\to a_{n-1}A_{n-1},A_{n-1}\to a_n\in P$.

$\Leftrightarrow A_1\in\delta(S,a_1),A_2\in\delta(A_1,a_2),\cdots,A_{n-1}\in\delta(A_{n-2},a_{n-1}),Z\in\delta(A_{n-1},a_n)$.

$\Leftrightarrow Z\in\delta(S,a_1\cdots a_n)$,.

$\Leftrightarrow a_1\cdots a_n\in L(M)$.

(3)空串的处理同**定理4.3.1**.

[**定理4.3.3**] FA与RL等价.

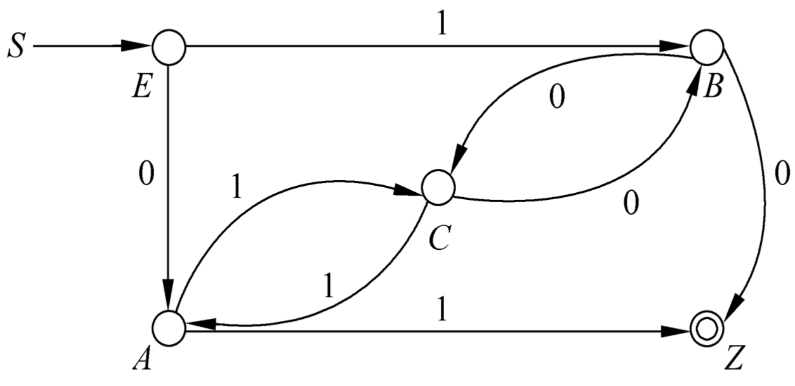
[**例4.3.3**] 构造与下列右线性文法等价的FA.

(1)$G_1:E\to 0A|1B,A\to 1|1C,B\to 0|0C,C\to 0B|1A$.

将产生式转化为转移函数:

①产生式$E\to 0A$对应转移函数$\delta(E,0)=\{A\}$,同理有$\delta(E,1)=\{B\},\delta(C,0)=\{B\},\delta(C,1)=\{A\}$.

②产生式$A\to 1|1C$对应转移函数$\delta(A,1)=\{Z,C\}$,同理有$\delta(B,0)=\{Z,C\}$.



(2)$G_2:S\to a|aA,A\to a|aA|cA|bB,B\to a|b|c|aB|bB|cB$.

产生式组	➔	迁移函数
S→a|aA	➔	**δ(S，a)={A,Z}**
A→a|aB	➔	**δ(A，a)={B,Z}**
A→cA	➔	**δ(A，c)={A}**
A→bB	➔	**δ(A，b)={B}**
B→a|aB	➔	**δ(B，a)={B,Z}**
B→b|bB	➔	**δ(B，b)={B,Z}**
B→c|cB	➔	**δ(B，c)={B,Z}**

开始状态：**S**，接受状态：**{Z}**

4.3.2 识别左线性文法的FA

[构造识别左线性文法的FA的思路]

(1)按推导来说,句子$a_1\cdots a_n$中的字符被推导出的先后顺序与它们在句子中出现的顺序相反;按归约来说,它们被归约成的语法变量的顺序与它们在句子中的出现顺序相同,故归约的过程与FA处理句子字符的过程顺序一致.

(2)按推导来说,在推导中使用形如$A\rightarrow a$的产生式可将句型变为句子,且a是该句子的首字符;按归约来说,对句子的首字符,按形如$A\rightarrow a$的产生式规约,即将字符a归约为变量A.引入一个开始状态Z,对形如$A\rightarrow a$的产生式,定义$A\in\delta(Z,a)$.同理对形如$A\rightarrow Ba$的产生式,FA在B状态读入字符a,转移到A状态,按归约来说,FA在B状态时已将当前句子中处理过的前缀归约为B,读入a时将Ba归约为A.

(3)按归约来说,若一个句子是文法G的语言中的句子,则它最终会被归约为G的开始符号,故G的开始符号对应的状态即FA的终止状态.注意解决文法的开始符号只有一个,但DFA的终止状态可有多个的问题.

[例4.3.4] 构造与下列左线性文法等价的FA.

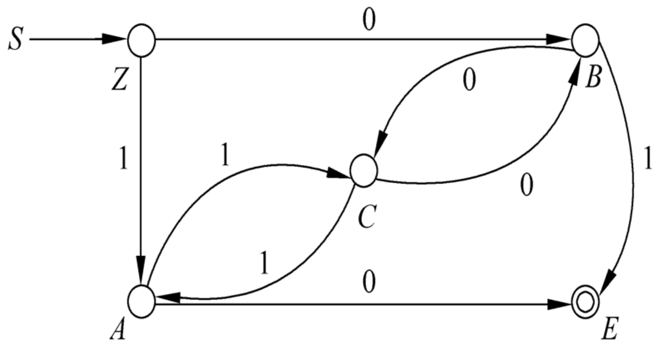
(1)$G_1:E\rightarrow A0|B1,A\rightarrow 1|C1,B\rightarrow 0|C0,C\rightarrow B0|A1.$

将产生式转化为转移函数:

①产生式$E\rightarrow A0|B1$对应转移函数$\delta(A,0)=\{E\},\delta(B,1)=\{E\}$.

同理有$\delta(A,1)=\{C\},\delta(C,0)=\{B\},\delta(B,0)=\{C\},\delta(A,1)=\{C\}$.

②产生式$A\rightarrow 1$对应转移函数$\delta(Z,1)=\{A\}$.同理有$\delta(Z,0)=\{B\}$.



(2)$G_2:S\rightarrow a|Aa,A\rightarrow a|Aa|Ac|Bb,B\rightarrow a|b|c|Ba|Bb|Bc.$

$M=(\{S,A,B,Z\},\{a,b,c\},\delta,Z,\{S\})$,其中:

$\delta(Z,a)=\{S,A,B\},\delta(Z,b)=\{B\},\delta(Z,c)=\{B\},\delta(A,a)=\{S,A\},$

$\delta(A,c)=\{A\},\delta(B,a)=\{B\},\delta(B,b)=\{A,B\},\delta(B,c)=\{B\}.$

[定理4.3.4] 左线性文法与FA等价.

4.4 FA的变形

4.4.1 双向有穷状态自动机

[定义4.4.1] **确定的双向有穷状态自动机**(Two-way Deterministic Finite Automaton, 2DFA)定义为五元组$M=(Q,\Sigma,\delta,q_0,F)$,其中$Q,\Sigma,q_0,F$的含义同DFA,$\delta:Q\times\Sigma\rightarrow Q\times\{L,R,S\}$,其中:

(1)$\delta(q,a)=\{p,L\}$表示M在状态q读入字符a后转移到状态p,并将读头向左移动一个带方格,指向输入字符串的前一个字符.

(2)$\delta(q,a)=\{p,R\}$表示M在状态q读入字符a后转移到状态p,并将读头向右移动一个带方格,指向输入字符串的后一个字符.

(3)$\delta(q,a)=\{p,S\}$表示M在状态q读入字符a后转移到状态p,读头保持原位.

[**定义4.4.2**] 对2DFA $M=(Q,\Sigma,\delta,q_0,F)$,定义其接受的语言$L(M)=\{x\mid (q_0x\vdash^* xp)\wedge(p\in F)\}$.

[**定理4.4.1**] 2DFA与FA等价.

[**定义4.4.2**] **不确定的双向有穷状态自动机**(Two-way Non-deterministic Finite Automaton, 2NFA)定义为五元组$M=(Q,\Sigma,\delta,q_0,F)$,其中$Q,\Sigma,q_0,F$的含义同DFA,$\delta:Q\times\Sigma\rightarrow 2^{Q\times\{L,R,S\}}$,其中$\delta(q,a)=\{\{p_1,D_1\},\cdots,\{p_m,D_m\}\}$表示$M$在$q$状态读入字符$a$,可选择地将状态变为$p_i\ \ (1\leq i\leq m)$,同时按$D_i$实现对读头的移动,其中$D_i\in\{L,R,S\}$的含义同2DFA.

[**定理4.4.2**] 2NFA与FA等价.

4.4.2 带输出的FA

[**定义4.4.3**] **Moore机**定义为六元组$M=(Q,\Sigma,\Delta,\delta,\lambda,q_0)$,其中$Q,\Sigma,q_0,\delta$的含义同DFA,$\Delta$为**输出字母表**,$\lambda:Q\rightarrow\Delta$为**输出函数**.

(1)对$\forall q\in Q,\forall a\in\Sigma,\lambda(q)=a$表示$M$在状态$q$时输出字符$a$.

(2)对$\forall a_1\cdots a_n\in\Sigma^*$,若$\delta(q_0,a_1)=q_1,\delta(q_1,a_2)=a_2,\cdots,\delta(q_{n-2},a_{n-1})=q_{n-1},\delta(q_{n-1},a_n)=q_n$,则$M$的输出为$\lambda(q_0)\lambda(q_1)\cdots\lambda(q_{n-1})$.

[**注**] Moore机处理字符串时每经过一个状态就输出一个字符,即输出字符与状态一一对应.

[**定义4.4.4**] **Mealy机**定义为六元组$M=(Q,\Sigma,\Delta,\delta,\lambda,q_0)$,其中$Q,\Sigma,q_0,\delta$的含义同DFA,$\Delta$输出字母表,$\lambda:Q\times\Sigma\rightarrow\Delta$为输出函数.

(1)对$\forall q\in Q,\forall a\in\Sigma$,$\lambda(q,a)=d$表示$M$在状态$q$读入字符$a$时输出$d$.

(2)对$\forall a_1\cdots a_n\in\Sigma^*$,$M$的输出串为$\lambda(q_0,a_1)\lambda(\delta(q_0,a_1),a_2)\cdots\lambda((\cdots\delta(\delta(q_0,a_1),a_2)\cdots),a_n)$.

设$\delta(q_0,a_1)=q_1,\delta(q_1,a_2)=q_2,\cdots,\delta(q_{n-1},a_n)=q_n$,

则M的输出可表示为$\lambda(q_0,a_1)\lambda(q_1,a_2)\cdots\lambda(q_{n-1},a_n)$.

[**注**] Mealy机处理字符串时每一个移动谁出一个字符,即输出字符与移动一一对应.

[**定理4.4.3**] Moore机和Mealy机等价.
