

正则语言的性质

- 正则语言的泵引理
- 正则语言的封闭性
- 正则语言的判定性质
- 自动机的最小化

正则语言的判定性质

正则语言, 或任何语言, 典型的 3 个判定问题:

- ① 以某种形式化模型描述的语言是否为空? 是否无穷?
- ② 某个特定的串 w 是否属于所描述的语言?
- ③ 以两种方式描述的语言, 是否是相同的? — 语言的等价性

我们想知道, 要回答这类问题的具体算法, 是否存在.

空性, 有穷性和无穷性

定理 14

具有 n 个状态的有穷自动机 M 接受的集合 S :

- ① S 是非空的, 当且仅当 M 接受某个长度小于 n 的串;
- ② S 是无穷的, 当且仅当 M 接受某个长度为 m 的串, $n \leq m < 2n$.

所以, 对于正则语言:

- 存在算法, 判断其是否为空, 只需检查全部长度小于 n 的串;
- 存在算法, 判断其是否无穷, 只需检查全部长度由 n 到 $2n - 1$ 的串.

证明: 设接受正则语言 S 的 DFA 为 M .

① 必要性: 显然成立. 充分性:

- i 如果 S 非空, 设 w 是 M 接受的串中长度最小者之一;
- ii 必然 $|w| < n$, 否则由泵引理 $w = xyz$, 接受 xz 更短.

② 必要性: 由泵引理, 显然成立. 充分性:

- i 如果 S 无穷, 假设没有长度 n 到 $2n - 1$ 之间的串;
- ii 那么取 $w \in L(M)$ 是长度 $\geq 2n$ 中最小者之一;
- iii 由泵引理 $w = xyz$, 且 M 会接受更短的串 xz ;
- iv 于是, 或者 w 不是长度最小的, 或者长度 n 到 $2n - 1$ 之间有被接受的串, 因此假设不成立. \square

正则语言的等价性

定理 15

存在算法, 判定两个有穷自动机是否等价(接受语言相同).

证明:

- ① 设 M_1 和 M_2 是分别接受 L_1 和 L_2 的有穷自动机;
- ② 则 $(L_1 \cap \overline{L_2}) \cup (\overline{L_1} \cap L_2)$ 是正则的, 所以可被某个有穷自动机 M_3 接受;
- ③ 而 M_3 接受某个串, 当且仅当 $L_1 \neq L_2$;
- ④ 由于存在算法判断 $L(M_3)$ 是否为空, 因此得证. \square

正则语言的性质

- 正则语言的泵引理
- 正则语言的封闭性
- 正则语言的判定性质
- 自动机的最小化

状态的等价性

定义

DFA $A = (Q, \Sigma, \delta, q_0, F)$ 中两个状态 p 和 q , 对 $\forall w \in \Sigma^*$:

$$\hat{\delta}(p, w) \in F \Leftrightarrow \hat{\delta}(q, w) \in F,$$

则称这两个状态是等价的, 否则称为可区分的.

- 等价性只要求 $\hat{\delta}(p, w)$ 和 $\hat{\delta}(q, w)$ 同时在或不在 F 中, 而不必是相同状态.

填表算法

递归寻找 DFA 中全部的可区分状态对:

$Q \times Q$

① 如果 $p \in F$ 而 $q \notin F$, 则 $[p, q]$ 是可区分的;

② $\forall a \in \Sigma$, 如果

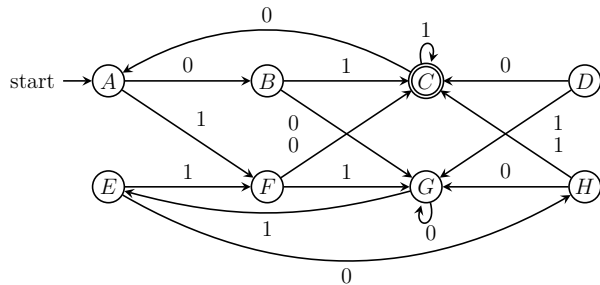
$$[r = \delta(p, a), s = \delta(q, a)]$$

是可区分的, 则 $[p, q]$ 是可区分的.

定理 16

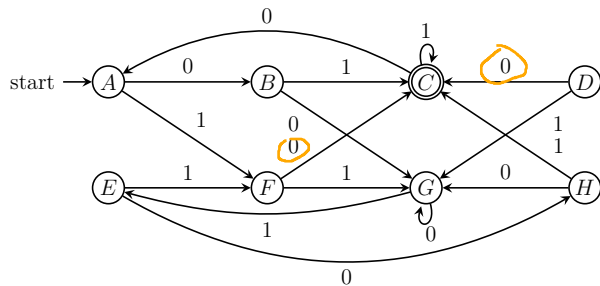
如果填表算法不能区分两个状态, 则这两个状态是等价的.

例 16. 用填表算法找到如图 DFA 中全部可区分状态对.



B							
C							
D							
E							
F							
G							
H							
	A	B	C	D	E	F	G

例 16. 用填表算法找到如图 DFA 中全部可区分状态对.



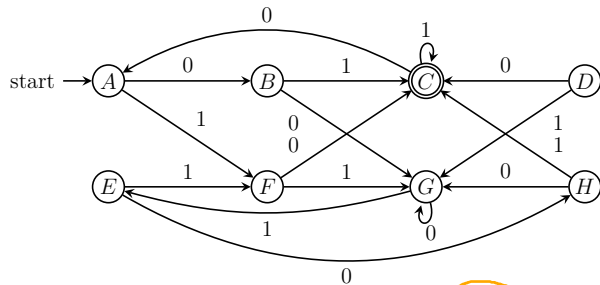
B							
C	×	×					
D			×				
E			×				
F			×				
G			×				
H			×				
	A	B	C	D	E	F	G

① 直接标记终态和非终态之间的状态对:

$$\{C\} \times \{A, B, D, E, F, G, H\}.$$

$$\forall w \in \Sigma^*, w = 0$$

例 16. 用填表算法找到如图 DFA 中全部可区分状态对.

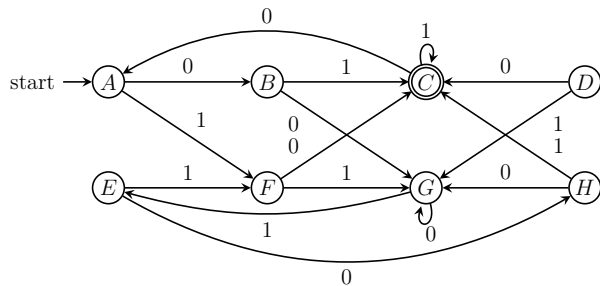


B							
C	×	×					
D	×	×	×				
E			×	×			
F	×	×	×		×		
G			×	×		×	
H			×	×		×	
	A	B	C	D	E	F	G

② 标记所有经过字符 0 到达终态和非终态的状态对:

$\{D, F\} \times \{A, B, C, E, G, H\}$. 可区分

例 16. 用填表算法找到如图 DFA 中全部可区分状态对.

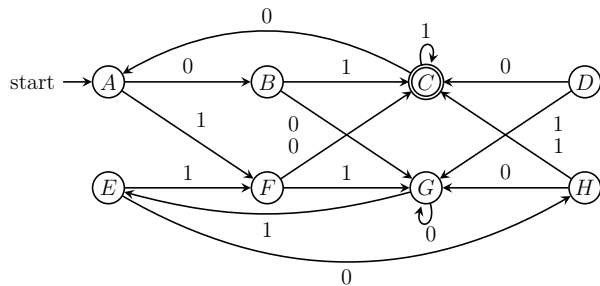


B	×						
C	×	×					
D	×	×	×				
E		×	×	×			
F	×	×	×		×		
G		×	×	×		×	
H	×		×	×	×	×	×
	A	B	C	D	E	F	G

③ 标记所有经过字符 1 到达终态和非终态的状态对:

$$\{B, H\} \times \{A, C, D, E, F, G\}.$$

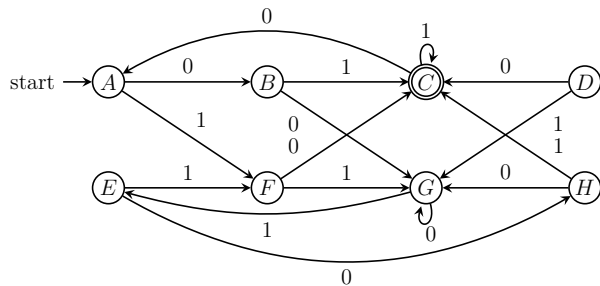
例 16. 用填表算法找到如图 DFA 中全部可区分状态对.



B	×						
C	×	×					
D	×	×	×				
E		×	×	×			
F	×	×	×		×		
G		×	×	×		×	
H	×		×	×	×	×	×
	A	B	C	D	E	F	G

④ 此时还有 $[A,E]$, $[A,G]$, $[B,H]$, $[D,F]$, $[E,G]$ 未标记, 只需逐个检查.

例 16. 用填表算法找到如图 DFA 中全部可区分状态对.



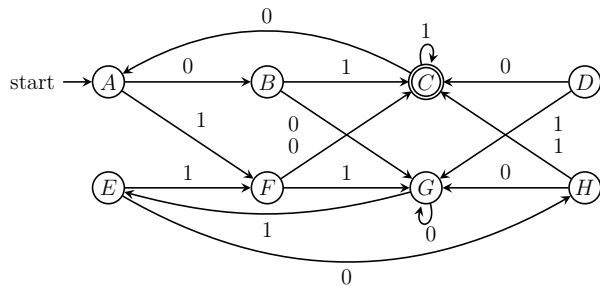
B	×						
C	×	×					
D	×	×	×				
E		×	×	×			
F	×	×	×		×		
G	×	×	×	×	×	×	
H	×		×	×	×	×	×
	A	B	C	D	E	F	G

④ 此时还有 $[A,E]$, $[A,G]$, $[B,H]$, $[D,F]$, $[E,G]$ 未标记, 只需逐个检查.

× $[A,G]$ 是可区分的, 因为经串 01 到可区分的 $[C,E]$;

× $[E,G]$ 是可区分的, 因为经串 10 到可区分的 $[C,H]$

例 16. 用填表算法找到如图 DFA 中全部可区分状态对.



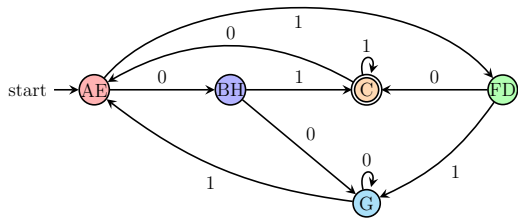
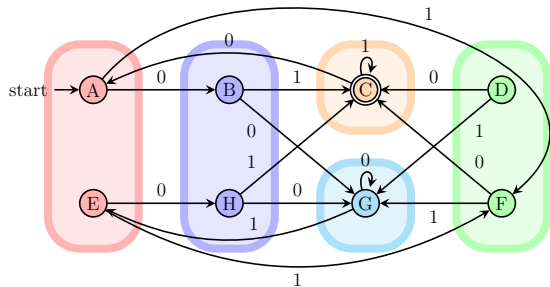
B	×						
C	×	×					
D	×	×	×				
E		×	×	×			
F	×	×	×		×		
G	×	×	×	×	×	×	
H	×		×	×	×	×	×
	A	B	C	D	E	F	G

- ⑤ 而 $[A,E]$, $[B,H]$ 和 $[D,F]$ 在经过很短的字符串后, 都会到达相同状态, 因此都是等价的.

DFA 最小化

根据等价状态, 将状态集划分成块, 构造等价的最小化 DFA.

续例 16. 构造其最小化的 DFA.



思考题

NFA 能否最小化?