

Chapter 4

正则语言的性质

4.1 正则语言

直观来讲, 正则语言是从单独的字符串以并 (*union*), 连接 (*concatenation*) 和重复 (*repetition*) 构建而来的. 我们已经有了两种形式化描述的工具: 有穷自动机和正则表达式. 而实际上, 我们是可以直接给出形式定义的. 即, 如果一个语言 L 是正则的, 那么当且仅当 (递归的) 满足:

1. $L = \emptyset$;
2. L 中仅有一个字符串 (可以是空串);
3. L 是两个正则语言的并;
4. L 是两个正则语言的连接;
5. L 是某个正则语言的克林闭包.

4.2 证明语言的非正则性

“泵引理”是正则语言的一个必要条件: 如果一个语言是正则的, 则一定满足泵引理.

例 1. $L = \{0^m 1^n \mid m, n \geq 0\}$ 是否是正则语言?

例 2. $L = \{0^m 1^n \mid m \geq 2, n \geq 4\}$ 是否是正则语言?

例 3. $L_{01} = \{0^n 1^n \mid n \geq 0\}$ 是否是正则语言?

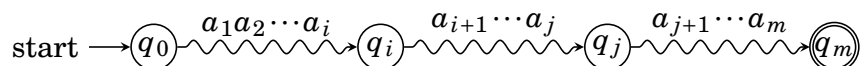
4.2.1 正则语言的泵引理

定理 5 (正则语言的泵引理 (Pumping Lemma)). 如果语言 L 是正则的, 那么存在正整数 N , 它只依赖于 L , 对 $\forall w \in L$, 只要 $|w| \geq N$, 就可以将 w 分为三部分 $w = xyz$ 满足:

1. $y \neq \varepsilon$ ($|y| > 0$);
2. $|xy| \leq N$;
3. $\forall k \geq 0, xy^kz \in L$.

证明:

1. 如果 L 正则, 那么存在有 n 个状态 DFA A 使 $L(A) = L$;
2. 取 $w = a_1 \dots a_m \in L$ ($m \geq n$), 定义 $q_i = \hat{\delta}(q_0, a_1 \dots a_i)$; q_0 是开始状态, 当 A 输入 w 的前 n 个字符时, 经过的状态分别是 q_0, q_1, \dots, q_n 共 $n+1$ 个;



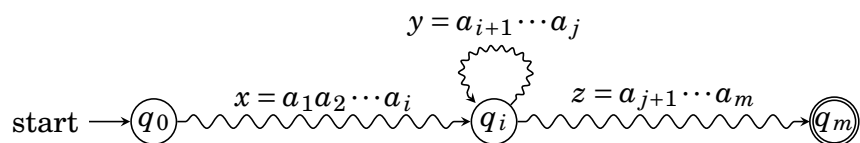
3. 由鸽巢原理, 必有两状态相同 $q_i = q_j$ ($0 \leq i < j \leq n$); 由 q_i 和 q_j 将 w 分为

$$x = a_1 a_2 \dots a_i$$

$$y = a_{i+1} a_{i+2} \dots a_j$$

$$z = a_{j+1} a_{j+2} \dots a_m$$

4. 那么 $w = xyz$ 如图, 且有 $\forall k \geq 0, xy^kz \in L$;



因为如果从 q_i 出发, 输入 y , 会到达 q_j , 而 $q_i = q_j$, 所以当输入 y^k ($k \geq 0$) 时, 始终会回到 q_i . 所以当 DFA A 输入 xy^kz 时, 由 q_0 始终会达到 q_m . 那么, 如果 $xyz \in L(A)$, 一定有 $xy^kz \in L(A)$ 对所有 $k \geq 0$ 成立.

5. 而因为 $i < j$ 所以 $y \neq \varepsilon$ (即 $|y| > 0$), 因为 $j \leq n$ 所以 $|xy| \leq n$. □

任何从开始状态到接受状态的路径, 如果长度超过 n , 一定会经过 $n+1$ 个状态, 必定有一个重复状态, 因此会形成一个循环 (loop); 那么, 这个循环可以被重复多次后, 沿原路径还会到达接收状态. 泵引理中的 N , 是正则语言固有存在的.

泵引理可以用来确定特定语言不在给定语言类 (正则语言) 中. 但是它们不能被用来确定一个语言在给定类中, 因为满足引理是类成员关系的必要条件, 但不是充分条件.

4.2.2 泵引理的应用

续例 3. 证明 $L_{01} = \{0^n 1^n \mid n \geq 0\}$ 不是正则语言.

证明:

1. 假设 L_{01} 是正则的.
2. 那么, 存在 $N \in \mathbb{Z}^+$, 对 $\forall w \in L_{01} (|w| \geq N)$ 满足泵引理.
3. 从 L_{01} 中取 $w = 0^N 1^N$, 显然 $w \in L_{01}$ 且 $|w| = 2N \geq N$.
4. 那么, w 可被分为 $w = xyz$, 且 $|xy| \leq N$ 和 $y \neq \varepsilon$.
5. 因此 y 只能是 0^m 且 $m > 0$.
6. 那么 $xy^2z = 0^{N+m} 1^N \notin L_{01}$, 而由泵引理 $xy^2z \in L_{01}$, 矛盾.
7. 所以假设不成立, L_{01} 不是正则的. □

例 4. 证明 $L_{\text{eq}} = \{w \in \{0, 1\}^* \mid w \text{ 由数量相等的 } 0 \text{ 和 } 1 \text{ 构成}\}$ 不是正则的.

思考题

刚刚已经证明了

$$L_{01} = \{0^n 1^n \mid n \geq 0\}$$

不是正则语言, 那么能否使用

$$\underline{L_{01} \subseteq L_{\text{eq}}}$$

来说明 L_{eq} 也不是正则的呢?

证明:

1. 假设 L_{eq} 是正则的.
2. 那么, 存在 $N \in \mathbb{Z}^+$, 对 $\forall w \in L_{\text{eq}} (|w| \geq N)$ 满足泵引理.
3. 从 L_{eq} 中取 $w = 0^N 1^N$, 显然 $w \in L_{\text{eq}}$ 且 $|w| = 2N \geq N$.
4. 那么, w 可被分为 $w = xyz$, 且 $|xy| \leq N$ 和 $y \neq \varepsilon$.
5. 因此 y 只能是 0^m 且 $m > 0$.
6. 那么 $xy^2z = 0^{N+m} 1^N \notin L_{\text{eq}}$, 而由泵引理 $xy^2z \in L_{\text{eq}}$, 矛盾.
7. 所以假设不成立, L_{eq} 不是正则的. □

例 5. 证明 $L = \{0^i 1^j \mid i > j\}$ 不是正则的.

证明:

1. 假设 L 是正则的.

$$\begin{array}{l} 0^{N+1} 1^N \\ 0^N 1^{N-1} \end{array}$$

2. 那么, 存在 $N \in \mathbb{Z}^+$, 对 $\forall w \in L (|w| \geq N)$ 满足泵引理.
3. 从 L 中取 $w = 0^{N+1}1^N$, 则 $w \in L$ 且 $|w| = 2N + 1 \geq N$.
4. 由泵引理, w 可被分为 $w = xyz$, 且 $|xy| \leq N$ 和 $y \neq \varepsilon$.
5. 那么, y 只能是 0^m 且 $m \geq 1$.
6. 那么, $xz = xy^0z = 0^{N+1-m}1^N \notin L$, 因为 $N + 1 - m \leq N$, 而由泵引理 $xy^0z \in L$, 矛盾.
7. 所以假设不成立, L 不是正则的. □

例 6. Prove $L = \{a^3b^nc^{n-3} \mid n \geq 3\}$ is not regular.

$a^{n-3} b^n a^3$

证明:

1. 假设 L 是正则的.
2. 那么, 存在 $N \in \mathbb{Z}^+$, 对 $\forall w \in L (|w| \geq N)$ 满足泵引理.
3. 从 L 中取 $w = a^3b^Nc^{N-3}$, 则 $w \in L$ 且 $|w| = 2N \geq N$.
4. 由泵引理, w 可被分为 $w = xyz$, 且 $|xy| \leq N$ 和 $y \neq \varepsilon$.
5. 那么, 则 y 只可能有 3 种情况 ($m > 0, r > 0, s > 0$):

$u v w x y$

$u x$

- (a) $y = a^m$, 则 $xy^2z = a^{3+m}b^Nc^{N-3} \notin L$;
 - (b) $y = b^m$, 则 $xy^2z = a^3b^{N+m}c^{N-3} \notin L$;
 - (c) $y = a^r b^s$, 则 $xy^2z = a^3b^s a^r b^N c^{N-3} \notin L$.
6. 无论 y 为何种情况, xy^2z 都不可能在 L 中, 与泵引理矛盾.
 7. 所以假设不成立, L 不是正则的. □

例. 证明 $L = \{a^{n!} \mid n > 0\}$ 不是正则的.

... 取 $w = a^{N!}$, 那么 $|y| = m > 0$, $|xy^2z| = N! + m$, 而 $0 < m \leq N < N! \leq N \cdot N!$, 所以 $N! < |xy^2z| = N! + m < N! + N \cdot N! = (N+1)!$, 即 $|xy^2z|$ 在两个阶乘数之间, 不可能是阶乘数, ...

思考题

- $L = \{0^n 1^n \mid 0 \leq n \leq 100\}$ 是否是正则语言?
- 有限的语言, 是否符合泵引理呢? 如 \emptyset , $\{\varepsilon\}$, $\{0, 00\}$ 等.

课堂练习. Prove that $L = \{a^i b^j c^k \mid i + j = k\}$ is not regular with pumping lemma.

4.2.3 泵引理只是必要条件

即“正则 \Rightarrow 泵引理成立”，所以“ \neg 泵引理成立 $\Rightarrow \neg$ 正则”。

- 泵引理只是正则语言的必要条件
- 只能用来证明某个语言不是正则的
- 与正则语言等价的定理 — Myhill-Nerode Theorem

例 7. 语言 L 不是正则的, 但每个串都可以应用泵引理

$$L = \{ca^n b^n \mid n \geq 1\} \cup \{c^k w \mid k \neq 1, w \in \{a, b\}^*\}$$

- 其中 $A = \{ca^n b^n \mid n \geq 1\}$ 部分不是正则的
- 而 $B = \{c^k w \mid k \neq 1, w \in \{a, b\}^*\}$ 部分是正则的
- 而 A 的任何串 $w = ca^i b^i$, 都可应用泵引理, 因为

$$w = (\varepsilon)(c)(a^i b^i)$$

重复字符 c 生成的新串都会落入 B 中

思考题

对任何正则语言 L , 在泵引理中, 与 L 相关联的正整数 N

- 与识别 L 的 DFA 的状态数 n 之间有何关系?
- 与识别 L 的 NFA 的状态数 之间呢?

思考题

语言

$$L = \{0^n x 1^n \mid n \geq 1, x \in \{0, 1\}^*\}$$

是否是正则语言?

4.3 正则语言的封闭性

定义. 正则语言经某些运算后得到的新语言仍保持正则, 称为在这些运算下封闭.

4.3.1 并/连接/闭包

定理 6 (并/连接/闭包的封闭性). 正则语言在并, 连接和闭包运算下保持封闭.

证明: 由正则表达式的定义得证. □

典型运算

4.3.2 补

定理 7 (补运算封闭性). 如果 L 是 Σ 上的正则语言, 那么 $\bar{L} = \Sigma^* - L$ 也是正则的.

证明: 设接受语言 L 的 DFA

$$A = (Q, \Sigma, \delta, q_0, \underbrace{F})$$

即 $L(A) = L$. 构造 DFA

$$B = (Q, \Sigma, \delta, q_0, Q - F)$$

则有 $\bar{L} = L(B)$, 因为 $\forall w \in \Sigma^*$

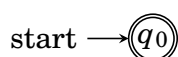
$$w \in \bar{L} \iff \hat{\delta}(q_0, w) \notin F \iff \hat{\delta}(q_0, w) \in Q - F \iff w \in L(B).$$

□

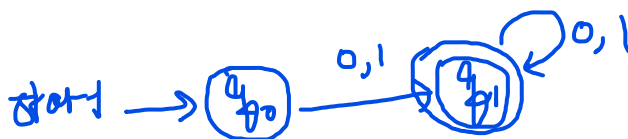
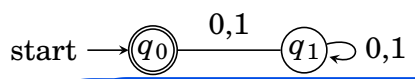
注意

使用这种方法求正则语言的补时, DFA 不能有缺失状态.

例 8. 若 $\Sigma = \{0, 1\}$, $L = \{\epsilon\}$ 的 DFA 如图, 请给出 \bar{L} 的 DFA.



应使用完整的 DFA 去求补:



思考题

如何求正则表达式的补?

例 9. 证明 $L_{\text{neq}} = \{w \mid w \text{ 由数量不相等的 } 0 \text{ 和 } 1 \text{ 构成}\}$ 不是正则的.

证明:

- 由泵引理不易直接证明 L_{neq} 不是正则的;

- 因为无论如何取 w , 将其分为 $w = xyz$ 时, 都不易产生 L_{neq} 之外的串;
- 而证明 L_{eq} 非正则很容易;
- 由补运算的封闭性, 所以 $L_{\text{neq}} = \overline{L_{\text{eq}}}$ 也不是正则的. □

4.3.3 交

定理 8. [习题 4.2.15] 若 DFA A_L , A_M 和 A 的定义如下

$$\begin{aligned} A_L &= (Q_L, \Sigma, \delta_L, q_L, F_L) \\ A_M &= (Q_M, \Sigma, \delta_M, q_M, F_M) \\ A &= (\underline{Q_L \times Q_M}, \Sigma, \underline{\delta}, \underline{(q_L, q_M)}, \underline{F_L \times F_M}) \end{aligned}$$

其中

$$\begin{aligned} \delta &: (Q_L \times Q_M) \times \Sigma \rightarrow Q_L \times Q_M \\ \delta((p, q), a) &= (\delta_L(p, a), \delta_M(q, a)). \end{aligned}$$

则对任意 $w \in \Sigma^*$,

$$\hat{\delta}((q_L, q_M), w) = (\hat{\delta}_L(q_L, w), \hat{\delta}_M(q_M, w)).$$

证明: 对 w 的结构归纳.

归纳基础: 当 $w = \varepsilon$ 时

$$\begin{aligned} \hat{\delta}((q_L, q_M), \varepsilon) &= (q_L, q_M) && \hat{\delta} \text{ 的定义} \\ &= (\hat{\delta}_L(q_L, \varepsilon), \hat{\delta}_M(q_M, \varepsilon)) && \text{同理} \end{aligned}$$

归纳递推: 当 $w = xa$ 时

$$\begin{aligned} \hat{\delta}((q_L, q_M), xa) &= \delta(\hat{\delta}((q_L, q_M), x), a) && \hat{\delta} \text{ 的定义} \\ &= \delta((\hat{\delta}_L(q_L, x), \hat{\delta}_M(q_M, x)), a) && \text{归纳假设} \\ &= (\delta_L(\hat{\delta}_L(q_L, x), a), \delta_M(\hat{\delta}_M(q_M, x), a)) && \delta \text{ 的构造} \\ &= (\hat{\delta}_L(q_L, xa), \hat{\delta}_M(q_M, xa)) && \hat{\delta} \text{ 的定义} \end{aligned}$$

□

定理 9 (交运算封闭性). 如果 L 和 M 是正则语言, 那么 $L \cap M$ 也是正则语言.

证明 1: 由 $L \cap M = \overline{\overline{L} \cup \overline{M}}$ 得证. (De Morgan's laws, 笛摩根定理) □

证明 2: 由定理 8 构造识别 $L \cap M$ 的 DFA A , 则 $\forall w \in \Sigma^*$,

$$\begin{aligned} w \in L \cap M &\iff \hat{\delta}_L(q_L, w) \in F_L \wedge \hat{\delta}_M(q_M, w) \in F_M \\ &\iff (\hat{\delta}_L(q_L, w), \hat{\delta}_M(q_M, w)) \in F_L \times F_M \\ &\iff \hat{\delta}((q_L, q_M), w) \in F_L \times F_M \\ &\iff w \in \mathbf{L}(A). \end{aligned}$$

因此 $\mathbf{L}(A) = L \cap M$, 所以 $L \cap M$ 也是正则的. □

思考题

有语言 M, N 和 L , 满足

$$M \cap N = L$$

和 2 求 1 ?

1. 若 M 和 N 都是正则语言, 由定理 9, 则 L 是一定正则语言.

2. 若 M 和 N 都不是正则语言, 则 L 一定不是正则语言吗? ?

3. 若 M 和 L 都是正则语言, 则 N 一定是正则语言吗? ?

4. 若 M 正则而 L 非正则, 则 N 一定非正则吗? 对 - 否 not

5. 若 M 非正则而 L 正则, 则 N 呢? ?

例 10: 如果已知语言

$$L_{01} = \{0^n 1^n \mid n \geq 0\}$$

不是正则的, 请用封闭性证明语言

$$L_{\text{eq}} = \{w \in \{0, 1\}^* \mid w \text{ 由数量相等的 } 0 \text{ 和 } 1 \text{ 构成}\}$$

也不是正则的.

证明:

1. 首先, 因为 $0^* 1^*$ 是正则语言;

2. 而 $L_{01} = \mathbf{L}(0^* 1^*) \cap L_{\text{eq}};$

3. 如果 L_{eq} 是正则的, L_{01} 必然也是正则的;

M N L
 leg off L_{01}
 is is not
 矛盾

4. 因为已知 L_{01} 不是正则的, 所以 L_{eq} 一定不是正则的. □

思考题

为什么又能用 L_{eq} 的子集 L_{01} 是非正则的, 来证明 L_{eq} 是非正则的呢?

特 殊

例 11. 如果 L_1 和 L_2 都不是正则的, 那么 $L_1 \cap L_2$ 一定不是正则的吗? 否

4.3.4 差

定理 10 (差运算封闭性). 如果 L 和 M 都是正则语言, 那么 $L - M$ 也是正则的.

证明: $L - M = L \cap \overline{M}$. □

例 12. [习题 4.2.6 a)] 证明正则语言在以下运算下封闭

$$\min(L) = \{w \mid w \text{ is in } L, \text{ but no proper prefix of } w \text{ is in } L\}$$

证明 1:

设接受 L 的 DFA $A = (Q, \Sigma, \delta, q_0, F)$, 构造接受 $\min(L)$ 的 DFA $B = (Q, \Sigma, \delta', q_0, F)$, 其中 δ' 如下, 往证 $L(B) = \min(L)$.

$$\delta'(q, a) = \begin{cases} \delta(q, a) & \text{if } q \notin F \\ \emptyset & \text{if } q \in F \end{cases}$$

1. $\forall w \in L(B)$, 存在状态转移序列 $q_0, q_1, \dots, q_{n-1}, q_n \in F$ 使 B 接受 w , 其中 $q_i \notin F (0 \leq i \leq n-1)$. $\therefore w \in \min(L)$.
2. $\forall w \in \min(L) \subseteq L$, A 接受 w 的状态序列如果为 q_0, q_1, \dots, q_n , 则显然 $q_i \notin F (0 \leq i \leq n-1)$ 且 $q_n \in F$, 否则 w 会有 L 可接受的前缀. $\therefore w \in L(B)$. □

证明 2:

由封闭性

$$\min(L) = L - L\Sigma^+$$

得证. □

4.3.5 反转

定义. 字符串 $w = a_1a_2\dots a_n$ 的反转 (Reverse), 记为 w^R , 定义为

$$w^R = a_na_{n-1}\dots a_1.$$

定义. 语言 L 的反转, 记为 L^R , 定义为

$$L^R = \{w^R \in \Sigma^* \mid w \in L\}.$$

定理 11 (反转的封闭性). 如果 L 是正则语言, 那么 L^R 也是正则的.

两种证明方法:

$\begin{matrix} R \\ \leftarrow \end{matrix}$

- 对正则表达式 E 的结构归纳, 往证

$$\mathbf{L}(E^R) = (\mathbf{L}(E))^R.$$

- 构造识别 L 的 NFA $A = (Q, \Sigma, \delta_A, q_0, F)$, 将其转换为识别 L^R 的 NFA

$$B = (Q \cup \{q_s\}, \Sigma, \delta_B, q_s, \{q_0\})$$

1. 将 A 的边调转方向;
2. 将 A 的初始状态 q_0 , 改为唯一的接受状态;
3. 新增初始状态 q_s , 且令 $\delta_B(q_s, \epsilon) = F$. 1 no f

例 13. 语言 L 及其反转 L^R 分别为

$$L = \{w \in \{0, 1\}^* \mid w \text{ ends in } 01.\}$$

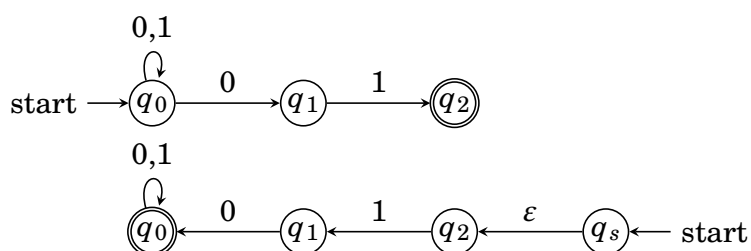
$$L^R = \{w \in \{0, 1\}^* \mid w \text{ starts with } 10.\}$$

正则表达式分别为

$$L = (0 + 1)^* 01$$

$$L^R = 10(0 + 1)^*.$$

自动机分别为



证明: 往证如果有正则表达式 E , 则存在正则表达式 E^R 使

$$\mathbf{L}(E^R) = (\mathbf{L}(E))^R.$$

归纳基础:

1. 当 $E = \emptyset$ 时, 有 $\emptyset^R = \emptyset$;
2. 当 $E = \epsilon$ 时, 有 $\epsilon^R = \epsilon$;
3. $\forall a \in \Sigma$, 当 $E = a$ 时, 有 $a^R = a$;

都满足 $\mathbf{L}(E^R) = (\mathbf{L}(E))^R$, 因此命题成立.

归纳递推:

1. 当 $E = E_1 + E_2$ 时, 有 $(E_1 + E_2)^R = E_1^R + E_2^R$

$$\begin{aligned} & (\mathbf{L}(E_1 + E_2))^R \\ &= (\mathbf{L}(E_1) \cup \mathbf{L}(E_2))^R \\ &= \{w^R \mid w \in \mathbf{L}(E_1) \text{ or } w \in \mathbf{L}(E_2)\} \\ &= (\mathbf{L}(E_1))^R \cup (\mathbf{L}(E_2))^R \\ &= \mathbf{L}(E_1^R) \cup \mathbf{L}(E_2^R) \\ &= \mathbf{L}(E_1^R + E_2^R) \end{aligned}$$

正则表达式的加
语言的反转
同上
归纳假设
正则表达式的加

2. 当 $E = E_1 E_2$ 时, 有 $(E_1 E_2)^R = E_2^R E_1^R$

$$\begin{aligned} & (\mathbf{L}(E_1 E_2))^R = (\mathbf{L}(E_1) \mathbf{L}(E_2))^R \\ &= \{w_1 w_2 \mid w_1 \in \mathbf{L}(E_1), w_2 \in \mathbf{L}(E_2)\}^R \\ &= \{(w_1 w_2)^R \mid w_1 \in \mathbf{L}(E_1), w_2 \in \mathbf{L}(E_2)\} \\ &= \{w_2^R w_1^R \mid w_1 \in \mathbf{L}(E_1), w_2 \in \mathbf{L}(E_2)\} \\ &= \{w_2^R \mid w_2 \in \mathbf{L}(E_2)\} \{w_1^R \mid w_1 \in \mathbf{L}(E_1)\} \\ &= (\mathbf{L}(E_2))^R (\mathbf{L}(E_1))^R \\ &= \mathbf{L}(E_2^R) \mathbf{L}(E_1^R) = \mathbf{L}(E_2^R E_1^R) \end{aligned}$$

正则表达式的连接
语言的连接
语言的反转
字符串的反转
语言的连接
语言的反转
正则表达式的连接

3. 当 $E = E_1^*$ 时, 有 $(E_1^*)^R = (E_1^R)^*$

$$\begin{aligned} & (\mathbf{L}(E_1^*))^R \\ &= \{w_1 w_2 \dots w_n \mid n \geq 0, w_i \in \mathbf{L}(E_1)\}^R \\ &= \{(w_1 w_2 \dots w_n)^R \mid n \geq 0, w_i \in \mathbf{L}(E_1)\} \end{aligned}$$

正则表达式的闭包
语言的反转

$$\begin{aligned}
&= \{w_n^R w_{n-1}^R \dots w_1^R \mid n \geq 0, w_i \in \mathbf{L}(E_1)\} && \text{字符串的反转} \\
&= \{w_n^R w_{n-1}^R \dots w_1^R \mid n \geq 0, w_i^R \in \mathbf{L}(E_1^R)\} && \text{归纳假设} \\
&= \{w_1 w_2 \dots w_n \mid n \geq 0, w_i \in \mathbf{L}(E_1^R)\} && \text{变量重命名} \\
&= \mathbf{L}((E_1^R)^*) && \text{正则表达式的闭包}
\end{aligned}$$

都满足 $(\mathbf{L}(E))^R = \mathbf{L}(E^R)$, 因此命题成立, 所以 L^R 也是正则语言. \square

4.3.6 同态与逆同态

同态 (Homomorphism)

定义. 若 Σ 和 Γ 是两个字母表, 同态定义为函数 $h: \Sigma \rightarrow \Gamma^*$

$$\forall a \in \Sigma, h(a) \in \Gamma^*.$$

扩展 h 的定义到字符串,

$$(1) h(\epsilon) = \epsilon$$

$$(2) h(xa) = h(x)h(a)$$

再扩展 h 到语言, 对 $\forall L \subseteq \Sigma^*$,

$$h(L) = \{h(w) \mid w \in L\}.$$

例 14. 若由 $\Sigma = \{0, 1\}$ 到 $\Gamma = \{a, b\}$ 的同态函数 h 为

$$h(0) = ab, \quad h(1) = \epsilon.$$

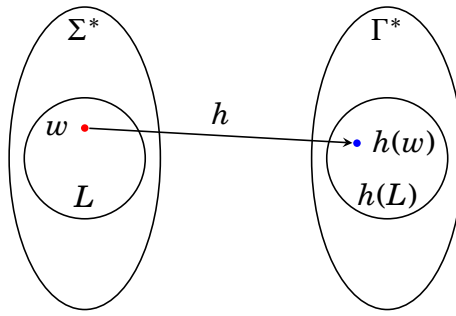
则 Σ 上的字符串 0011, 在 h 的作用下

$$\begin{aligned}
h(0011) &= h(\epsilon)h(0)h(0)h(1)h(1) \\
&= \epsilon \cdot ab \cdot ab \cdot \epsilon \cdot \epsilon \\
&= abab.
\end{aligned}$$

语言 $L = \mathbf{1^*0 + 0^*1}$, 在 h 的作用下, $h(L)$ 为:

$$\begin{aligned}
h(\mathbf{1^*0 + 0^*1}) &= (h(\mathbf{1})^* h(\mathbf{0}) + (h(\mathbf{0})^* h(\mathbf{1})) \\
&= (\epsilon)^*(ab) + (ab)^*(\epsilon) \\
&= (ab)^*
\end{aligned}$$

定理 12 (同态的封闭性). 若 L 是字母表 Σ 上的正则语言, h 是 Σ 上的同态, 则 $h(L)$ 也是正则的.



- 若 L 的正则表达式为 E , 即 $L = \mathbf{L}(E)$, 按如下规则构造表达式 $h(E)$

$$h(\emptyset) = \emptyset$$

$$h(\mathbf{r} + \mathbf{s}) = h(\mathbf{r}) + h(\mathbf{s})$$

$$h(\epsilon) = \epsilon$$

$$h(\mathbf{rs}) = h(\mathbf{r})h(\mathbf{s})$$

$$\forall a \in \Sigma, h(\mathbf{a}) = h(a)$$

$$h(\mathbf{r}^*) = (h(\mathbf{r}))^*$$

- 往证 $\mathbf{L}(h(E)) = h(\mathbf{L}(E))$, 而 $h(E)$ 显然也是正则表达式, 因此 $h(L)$ 正则

证明: 对 E 的结构归纳, 往证 $\mathbf{L}(h(E)) = h(\mathbf{L}(E))$.

归纳基础:

- 当 $E = \epsilon$ 时

$$h(\mathbf{L}(\epsilon)) = h(\{\epsilon\}) = \{\epsilon\} = \mathbf{L}(\epsilon) = \mathbf{L}(h(\epsilon))$$

- 当 $E = \emptyset$ 时

$$h(\mathbf{L}(\emptyset)) = h(\emptyset) = \emptyset = \mathbf{L}(\emptyset) = \mathbf{L}(h(\emptyset))$$

- $\forall a \in \Sigma$, 当 $E = \mathbf{a}$ 时

$$h(\mathbf{L}(\mathbf{a})) = h(\{a\}) = \{h(a)\} = \mathbf{L}(h(a)) = \mathbf{L}(h(\mathbf{a}))$$

所以命题成立.

归纳递推: 假设对正则表达式 F, G 分别有

$$\mathbf{L}(h(F)) = h(\mathbf{L}(F)), \quad \mathbf{L}(h(G)) = h(\mathbf{L}(G))$$

- 当 $E = F + G$ 时:

$$h(\mathbf{L}(F + G)) = h(\mathbf{L}(F) \cup \mathbf{L}(G))$$

正则表达式的加

$$= h(\mathbf{L}(F)) \cup h(\mathbf{L}(G))$$

h 作用在每个集合的串上

$$= \mathbf{L}(h(F)) \cup \mathbf{L}(h(G))$$

归纳假设

$$= \mathbf{L}(h(F) + h(G))$$

正则表达式的加

$$= \mathbf{L}(h(F + G))$$

$h(F + G)$ 的定义

- 当 $E = FG$ 时:

$$\begin{aligned}
 h(\mathbf{L}(E)) &= h(\mathbf{L}(F)\mathbf{L}(G)) && \text{正则表达式的连接} \\
 &= h(\mathbf{L}(F))h(\mathbf{L}(G)) && \heartsuit \\
 &= \mathbf{L}(h(F))\mathbf{L}(h(G)) && \text{归纳假设} \\
 &= \mathbf{L}(h(F)h(G)) && \text{正则表达式的连接} \\
 &= \mathbf{L}(h(FG)) && h(FG) \text{ 的定义} \\
 &= \mathbf{L}(h(E))
 \end{aligned}$$

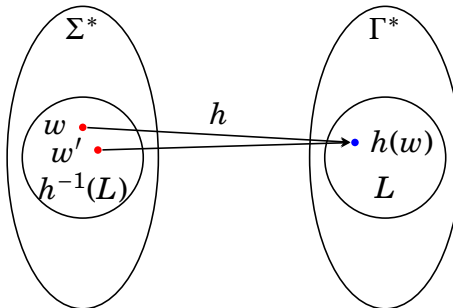
$$\heartsuit: h(a_1 \cdots a_n b_1 \cdots b_m) = h(a_1) \cdots h(b_m) = h(a_1 \cdots a_n)h(b_1 \cdots b_m)$$

- 当 $E = F^*$ 时: 略 (提示: $\forall w \in \mathbf{L}(F^*)$ 可看作 $w = w_1 w_2 \cdots w_n$, 其中 $w_i \in \mathbf{L}(F)$.) □

逆同态 (Inverse homomorphism)

定义. 若 h 是字母表 Σ 到 Γ 的同态, 且 L 是 Γ 上的语言, 那么使 $h(w) \in L$ 的 w ($w \in \Sigma^*$) 的集合, 称为语言 L 的 h 逆, 记为 $h^{-1}(L)$, 即

$$h^{-1}(L) = \{w \in \Sigma^* \mid h(w) \in L\}.$$



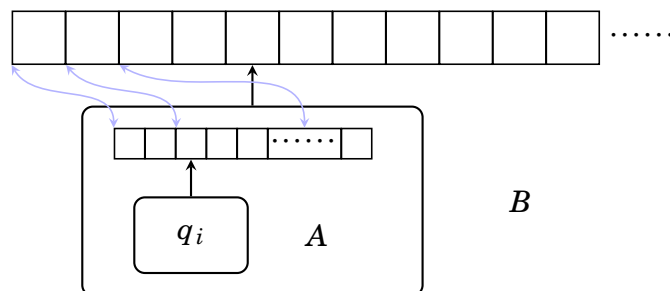
定理 13 (逆同态的封闭性). 如果 h 是字母表 Σ 到 Γ 的同态, L 是 Γ 上的正则语言, 那么 $h^{-1}(L)$ 也是正则语言.

证明: 由 L 的 DFA $A = (Q, \Gamma, \delta, q_0, F)$, 构造识别 $h^{-1}(L)$ 的 DFA

$$B = (Q, \Sigma, \delta', q_0, F),$$

其中

$$\delta'(q, a) = \hat{\delta}(q, h(a)).$$



为证明 $\mathbf{L}(B) = h^{-1}(L)$, 先证明 $\hat{\delta}'(q, w) = \hat{\delta}(q, h(w))$.

对 $|w|$ 归纳, 往证 $\hat{\delta}'(q, w) = \hat{\delta}(q, h(w))$.

1. 归纳基础: 若 $w = \varepsilon$

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

2. 归纳递推: 若 $w = xa$

$$\begin{aligned} \hat{\delta}'(q, xa) &= \delta'(\hat{\delta}'(q, x), a) && \hat{\delta}' \text{ 定义} \\ &= \delta'(\hat{\delta}(q, h(x)), a) && \text{归纳假设} \\ &= \hat{\delta}(\hat{\delta}(q, h(x)), h(a)) && \delta' \text{ 构造} \\ &= \hat{\delta}(q, h(x)h(a)) && \text{DFA 节例 5} \\ &= \hat{\delta}(q, h(xa)). \end{aligned}$$

所以 $\forall w \in \Sigma^*, \hat{\delta}'(q_0, w) = \hat{\delta}(q_0, h(w)) \in F$, 即 w 被 B 接受当且仅当 $h(w)$ 被 A 接受, B 是识别 $h^{-1}(L)$ 的 DFA, 因此 $h^{-1}(L)$ 是正则的. \square

例 15. Prove that $L = \{0^n 1^{2n} \mid n \geq 0\}$ is a language not regular.

证明: 设同态 $h: \{0, 1\} \rightarrow \{0, 1\}^*$ 为

$$h(0) = 0,$$

$$h(1) = 11,$$

那么

$$h^{-1}(L) = \{0^n 1^n \mid n \geq 0\} = L_{01},$$

我们已知 L_{01} 非正则, 由封闭性, L 不是正则的. \square

例 16. 若语言 $L = (00 + 1)^*$, 同态 $h: \{a, b\} \rightarrow \{0, 1\}^*$ 为

$$h(a) = 01, \quad h(b) = 10,$$

请证明 $h^{-1}(L) = (ba)^*$.

证明: 往证 $h(w) \in L \iff w = (ba)^n$.

(\Leftarrow) 若 $w = (ba)^n$, 而 $h(ba) = 1001$, 因此 $h(w) = (1001)^n \in L$.

(\Rightarrow) 若 $h(w) \in L$, 假设 $w \notin (ba)^*$, 则只能有四种情况:

1. w 以 a 开头, 则 $h(w)$ 以 01 开头, 显然 $h(w) \notin (00 + 1)^*$;
2. w 以 b 结尾, 则 $h(w)$ 以 10 结尾, 显然 $h(w) \notin (00 + 1)^*$;

认识语言
 $b^n a^n$
 $(ba)^*$

3. w 有连续的 a , 即 $w = xaa y$, 则 $h(w) = z1010v$, 显然 $h(w) \notin (00+1)^*$;
 4. w 有连续的 b , 即 $w = xbb y$, 则 $h(w) = z0101v$, 显然 $h(w) \notin (00+1)^*$;

因此 w 只能是 $(ba)^n, n \geq 0$ 的形式. □

例 17. [习题 4.2.6 b)] 证明正则语言在以下运算下封闭

$$\max(L) = \{w \mid w \text{ is in } L \text{ and for no } x \text{ other than } \varepsilon \text{ is } wx \text{ in } L\}.$$

证明 1:

由接受 L 的 DFA $M = (Q, \Sigma, \delta, q_0, F)$, 构造接受 $\max(L)$ 的 DFA $M' = (Q, \Sigma, \delta, q_0, F')$, 其中

$$F' = F - \{q \mid q \in F, \exists x \in \Sigma^+, \hat{\delta}(q, x) \in F\}$$

则 $L(M') = \max(L)$. □

证明 2: 利用封闭性.

如果 $\Sigma = \{a, b\}$, 设 $\Gamma = \{a, \hat{a}, b, \hat{b}\}$, 定义同态 $h (\Gamma \rightarrow \Sigma^*)$ 和 $g (\Gamma \rightarrow \Sigma^*)$:

$h(a) = a$	$h(\hat{a}) = a$	$g(a) = a$	$g(\hat{a}) = \varepsilon$
$h(b) = b$	$h(\hat{b}) = b$	$g(b) = b$	$g(\hat{b}) = \varepsilon$

那么

$$\max(L) = L - g(h^{-1}(L) \cap (a+b)^*(\hat{a}+\hat{b})^+).$$

□

注意: $\min(L)$ 和 $\max(L)$ 不同, 设 $L = 10^+$, 则 $\min(L) = \{10\}$ 而 $\max(L) = \emptyset$.

例 18. For a language L , define $\text{head}(L)$ to be the set of all prefixes of strings in L . Prove that if L is regular, so is $\text{head}(L)$.

证明. 设 L 是 Σ 上的正则语言且 $\Sigma = \{a, b\}$, $\Gamma = \{a, b, \hat{a}, \hat{b}\}$. 定义同态 $h: \Gamma \rightarrow \Sigma^*$ 和 $g: \Gamma \rightarrow \Sigma^*$ 分别为:

$h(a) = a$	$h(\hat{a}) = a$	$g(a) = a$	$g(\hat{a}) = \varepsilon$
$h(b) = b$	$h(\hat{b}) = b$	$g(b) = b$	$g(\hat{b}) = \varepsilon$

则因为 $(a+b)^*(\hat{a}+\hat{b})^*$ 是 Γ 上的正则语言, 所以

$$\text{head}(L) = g((a+b)^*(\hat{a}+\hat{b})^* \cap h^{-1}(L))$$

是 Σ 上的正则语言, 因此 $\text{head}(L)$ 是正则的. □

例如, 若字符串 $aab \in L$, 则

$$\begin{aligned} h^{-1}(\{aab\}) &= \{aab, a\hat{a}\hat{b}, a\hat{a}b, a\hat{a}\hat{b}, \hat{a}ab, \hat{a}a\hat{b}, \hat{a}\hat{a}b, \hat{a}\hat{a}\hat{b}\}, \\ (\mathbf{a} + \mathbf{b})^*(\hat{\mathbf{a}} + \hat{\mathbf{b}})^* \cap h^{-1}(\{aab\}) &= \{aab, a\hat{a}\hat{b}, a\hat{a}b, \hat{a}\hat{a}\hat{b}\}, \\ g((\mathbf{a} + \mathbf{b})^*(\hat{\mathbf{a}} + \hat{\mathbf{b}})^* \cap h^{-1}(\{aab\})) &= \{aab, aa, a, \varepsilon\}. \end{aligned}$$

4.4 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.

4.4.1 空性, 有穷性和无穷性

正则语言的空, 有穷和无穷 (Emptiness, finiteness and infiniteness), 可以通过定理14来判定.

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 A .

1. 必要性: 显然成立. 充分性:

i 如果 S 非空, 设 w 是 A 接受的串中长度最小者之一;

n

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 $w \in L(A)$ 是长度 $\geq 2n$ 中最小者之一;

iii 由泵引理 $w = xyz$, 且 A 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

4.4.2 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

利用定理 14

4.5 自动机的最小化

4.5.1 DFA 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, 而不必相同.

4.5.2 填表算法与 DFA 最小化

填表算法

递归寻找 DFA 中全部的可区分状态对:

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

2. $\exists a \in \Sigma$, 如果

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

是可区分的, 则 $[p, q]$ 是可区分的.

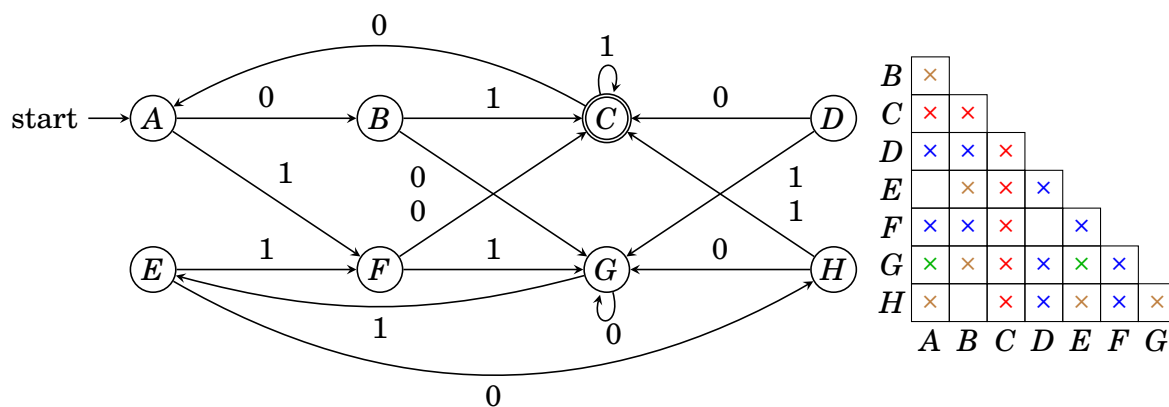
定理 16. 如果填表算法不能区分两个状态, 则这两个状态是等价的.

Algorithm 1 MinimizeDFA($Q, \Sigma, \delta, q_0, F$)

```

1: for all  $(p, q) \in Q \times Q$  do
2:   if  $(p \in F \text{ and } q \notin F) \text{ or } (p \notin F \text{ and } q \in F)$  then
3:      $T[p, q] \leftarrow \star$ 
4: repeat
5:    $done \leftarrow \text{True}$ 
6:   for all  $(p, q) \in Q \times Q$  do
7:     if  $T[p, q] \neq \star$  then
8:       for all  $a \in \Sigma$  do
9:         if  $T[\delta(p, a), \delta(q, a)] = \star$  then
10:           $T[p, q] \leftarrow \star$ 
11:         $done \leftarrow \text{False}$ 
12: until  $done$ 
13: return  $T$ 
  
```

例 19. 用填表算法找到如图 DFA 中全部可区分状态对.



1. 直接标记终态和非终态之间的状态对:

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

2. 标记所有经过字符 0 到达终态和非终态的状态对:

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

3. 标记所有经过字符 1 到达终态和非终态的状态对:

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

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, 因为字符 0 到可区分的 $[B, G]$;

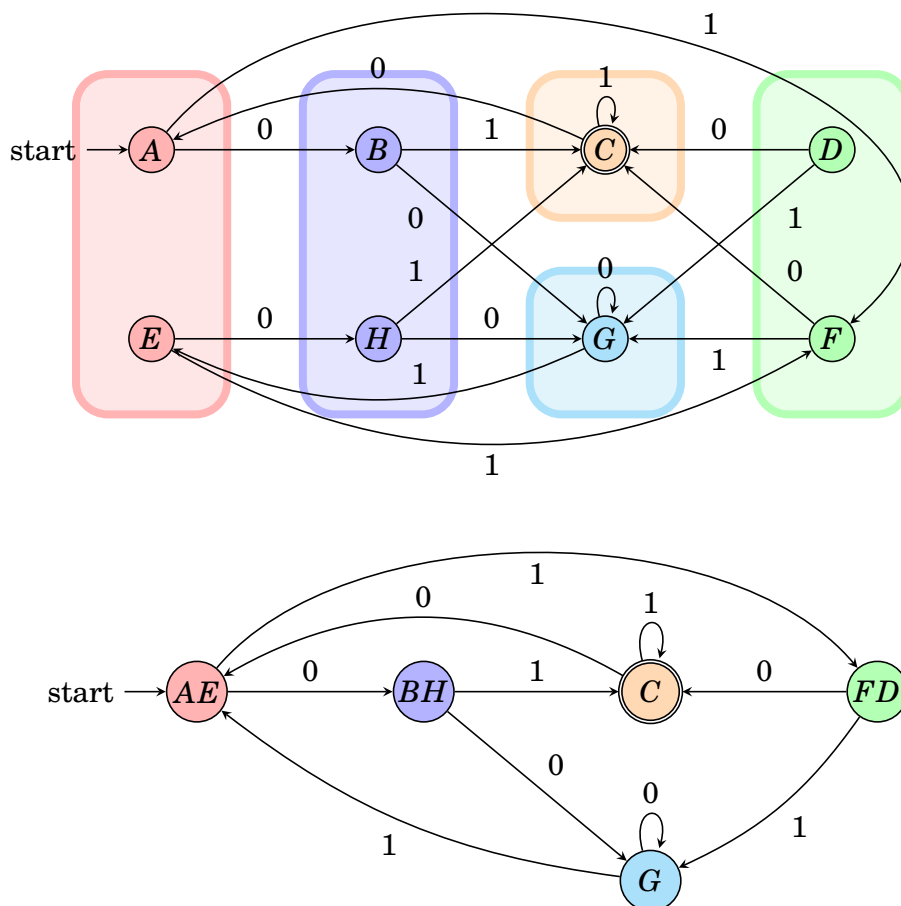
× $[E, G]$ 是可区分的, 因为字符 1 到可区分的 $[E, F]$.

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. 根据填表算法取得的 DFA A 状态间的等价性, 将状态集进行划分, 得到不同的块; 利用块构造新的 DFA B , B 的开始状态为包含 A 初始状态的块, B 的接受状态为包含 A 的接收状态的块, 转移函数为块之间的转移; 则 B 是 A 的最小化 DFA.

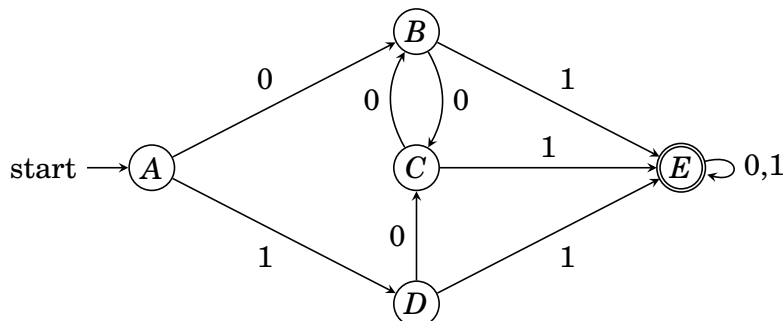
续例 19. 构造其最小化的 DFA.



思考题

NFA 能否最小化?

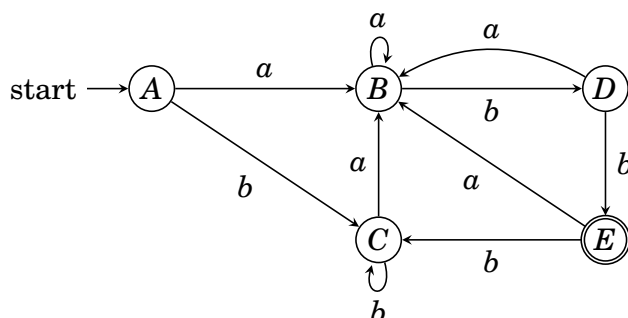
课堂练习. Minimize the given DFA.



4.6 练习题

1. [Exercise 4.1.2] Prove that the following are not regular languages.
 - d) The set of strings of 0's and 1's whose length is a perfect square.
 - e) The set of strings of 0's and 1's that are of the form ww , that is some string repeated.
2. [Exercise 4.2.2] If L is a language, and a is a symbol, then L/a , the quotient of L and a , is the set of strings w such that wa is in L . For example, if $L = \{a, aab, baa\}$, then $L/a = \{\epsilon, ba\}$. Prove that if L is regular, so is L/a . Hint: Start with a DFA for L and consider the set of accepting states.
3. [Exercise 4.2.6] Show that the regular languages are closed under the following operations:
 - (a) $\min(L) = \{w | w \text{ is in } L, \text{ but no proper prefix of } w \text{ is in } L\}$.
 - (b) $\max(L) = \{w \mid w \text{ is in } L \text{ and for no } x \text{ other than } \epsilon \text{ is } wx \text{ in } L\}$
 - (c) $\text{init}(L) = \{w \mid \text{for some } x, wx \text{ is in } L\}$

4. Minimize the given DFA.



chunyu@hit.edu.cn
<http://iilab.net/chunyu>

