

计算理论

第一部分 计算模型

[S]

第1章 有限自动机

第3章 图灵机

第4章 可判定性

第7章 计算复杂性

第1章 有限自动机

字符串与语言

字母表: 任意一个有限集. 常用记号 Σ, Γ .

符号: 字母表中的元素

字符串: 字母表中符号组成的**有限序列**

如asdf, 通俗地说即单词

串的**长度** $|\cdot|$, 例: $|abcde|=5$

串的**连接** $*$, 例: $(abc)*(de)=abcde$

串的**反转** R , 例: $(abcde)^R=edcba$

空词: 记为 ϵ , 长度为0

语言: 给定字母表上一些字符串的集合

Σ^* , 语言, 字典序

取字母表 $\Sigma = \{0,1\}$, Σ 上的语言举例:

$A = \{0, 00, 0000\}$, $B = \{0, 00, 01, 000, 001, \dots\}$

- Σ 上所有有限长串记为 Σ^* .
- Σ 上的任一语言都是 Σ^* 的子集.
- Σ^* (字典序): $\epsilon, 0, 1, 00, 01, 10, 11, 000, \dots$
- Σ 上所有无限长串记为 Σ^N .
- Σ 上的语言与 Σ^N 一一对应.

决定性问题与语言一一对应

决定性问题(Dicision Prob): 只需回答是与否的问题

“一数是否是偶数” -----{ 以0结尾的01串 }

“串0,1个数是否相等” -----{ 0,1个数相等的01串 }

“图是否连通” -----{ $\langle G \rangle$ | G 是连通图 }

其中 $\langle G \rangle$ 是图 G 编码成的字符串.

给定有限字母表 Σ ,

- 每个输入是一个串, 任意串都可以是输入串
- 一个决定性问题是满足某性质的串的集合(语言)

确定型有限(穷)自动机的形式定义

定义: **有限自动机**是一个5元组 $(Q, \Sigma, \delta, s, F)$,

1) Q 是有限集, 称为**状态集**;

2) Σ 是有限集, 称为**字母表**;

3) $\delta: Q \times \Sigma \rightarrow Q$ 是**转移函数**;

4) $s \in Q$ 是**起始状态**;

5) $F \subseteq Q$ 是**接受状态集**;

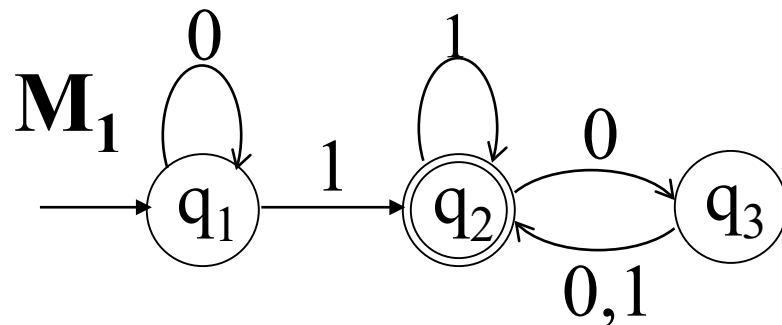
读写头**不能改写**, 且**只能右移**

$Q = \{q_1, q_2, q_3\}$, 状态集

$\Sigma = \{0, 1\}$, 字母表

$s = q_1$, 起始状态

$F = \{q_2\}$ 接受状态集



• **状态图等价于形式定义**

δ	0	1
q_1	q_1	q_2
q_2	q_3	q_2
q_3	q_2	q_2

有限自动机的语言:正则语言

对有限自动机 M , 若 $A = \{ w \in \Sigma^* \mid M \text{ 接受 } w \}$,
即 A 是有限自动机 M 的**语言**, 记为 $L(M)=A$, 也称 M **识别** A .
若存在DFA识别语言 A , 则称 A 是**正则语言**.
称两个有限自动机**等价**若它们语言相同.

有限自动机的设计

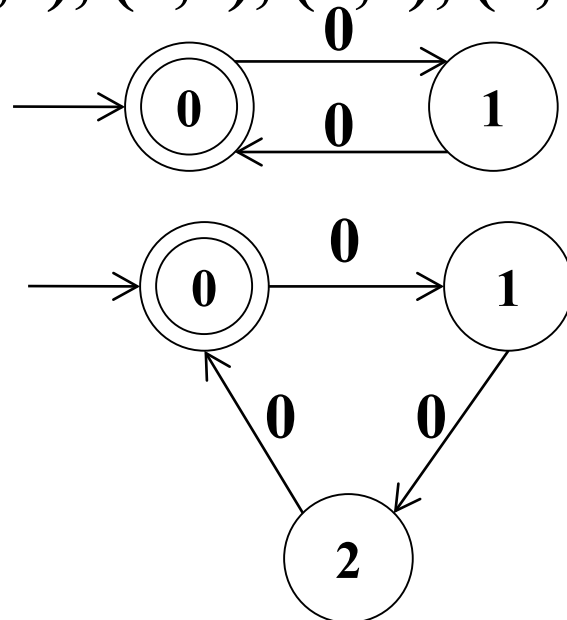
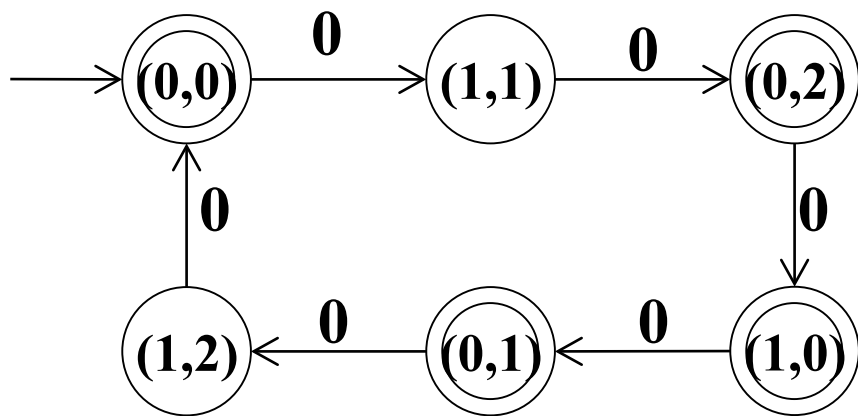
- 自己即自动机
- 寻找需要记录的**关键信息**

有限自动机的设计

$\{ 0^k \mid k \text{ 是 2 或 3 的倍数} \}$

$\Sigma = \{0\}$, 关键信息: $\varepsilon, 0^1, 0^2, 0^3, 0^4, 0^5$,

记为: 0, 1, 2, 3, 4, 5 或 (0,0), (1,1), (0,2), (1,0), (0,1), (1,2)



$\{0^k \mid k \text{ 是 2 或 3 的倍数}\} = \{0^k \mid k \text{ 是 2 倍数}\} \cup \{0^k \mid k \text{ 是 3 的倍数}\}$

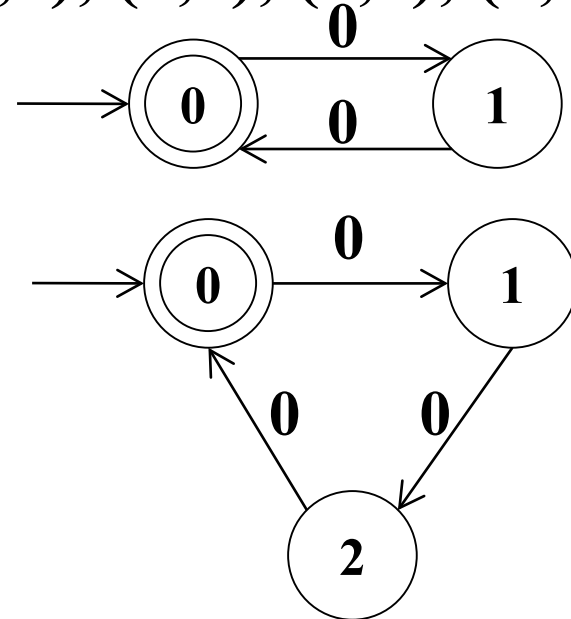
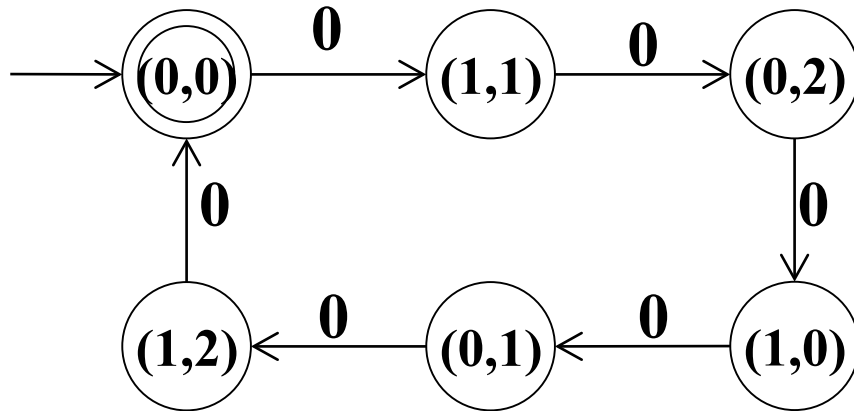
$$F = F_1 \times Q_2 \cup Q_1 \times F_2$$

有限自动机的设计

$\{ 0^k \mid k \text{是2和3的倍数} \}$

$\Sigma = \{0\}$, 关键信息: $\varepsilon, 0^1, 0^2, 0^3, 0^4, 0^5$,

记为: 0,1,2,3,4,5 或 (0,0), (1,1), (0,2), (1,0), (0,1), (1,2)



$\{0^k \mid k \text{是2和3的倍数}\} = \{0^k \mid k \text{是2倍数}\} \cap \{0^k \mid k \text{是3的倍数}\}$

$$F = F_1 \times F_2$$

正则语言与正则运算

如果语言A被一DFA识别,则称A是正则语言

定义: 设A和B是两个语言,定义正则运算

并,连接,星号如下:

- 并: $A \cup B = \{x | x \in A \text{ 或 } x \in B\}$
- 连接: $A^\circ B = \{xy | x \in A \text{ 且 } y \in B\}$
- 星号: $A^* = \{x_1 x_2 \dots x_k | k \geq 0 \text{ 且 每个 } x_i \in A\}$

正则语言的并是正则语言

定理: 设A,B都是 Σ 上的正则语言, 则 $A \cup B$ 也是正则语言.

证明: 设 $M_1=(Q_1, \Sigma, \delta_1, s_1, F_1)$ 和 $M_2=(Q_2, \Sigma, \delta_2, s_2, F_2)$ 是DFA,

且 $L(M_1)=A$, $L(M_2)=B$,

令 $Q=Q_1 \times Q_2$, $s=(s_1, s_2)$, $F = F_1 \times Q_2 \cup Q_1 \times F_2$,

$\delta: Q \times \Sigma \rightarrow Q$, $\forall a \in \Sigma, r_1 \in Q_1, r_2 \in Q_2$,

$\delta((r_1, r_2) , a) = (\delta_1(r_1, a), \delta_2(r_2, a))$,

即对 $i=1,2$, 第 i 个分量按 M_i 的转移函数变化.

令 $M=(Q, \Sigma, \delta, s, F)$, 则 $\forall x (x \in L(M) \leftrightarrow x \in A \cup B)$

即 $L(M) = A \cup B$. 证毕

正则语言的交是正则语言

定理: 设A,B都是 Σ 上的正则语言, 则 $A \cap B$ 也是正则语言.

证明: 设 $M_1=(Q_1, \Sigma, \delta_1, s_1, F_1)$ 和 $M_2=(Q_2, \Sigma, \delta_2, s_2, F_2)$ 是DFA,

且 $L(M_1)=A$, $L(M_2)=B$,

令 $Q=Q_1 \times Q_2$, $s=(s_1, s_2)$, $F=F_1 \times F_2$,

$\delta: Q \times \Sigma \rightarrow Q$, $\forall a \in \Sigma, r_1 \in Q_1, r_2 \in Q_2$,

$\delta((r_1, r_2), a) = (\delta_1(r_1, a), \delta_2(r_2, a))$,

即对 $i=1,2$, 第 i 个分量按 M_i 的转移函数变化.

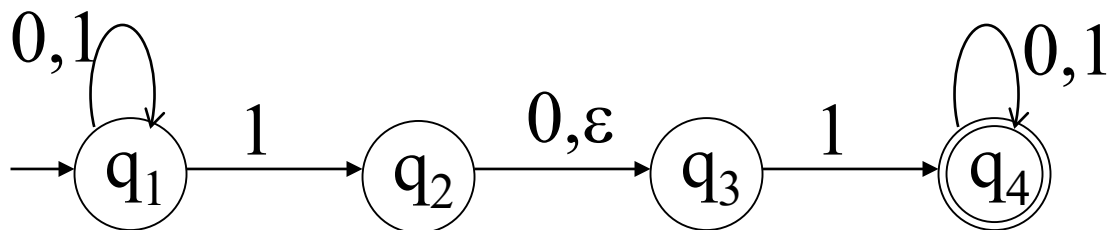
令 $M=(Q, \Sigma, \delta, s, F)$, 则 $\forall x (x \in L(M) \leftrightarrow x \in A \cap B)$

即 $L(M) = A \cap B$. 证毕

证明特点: 构造性证明

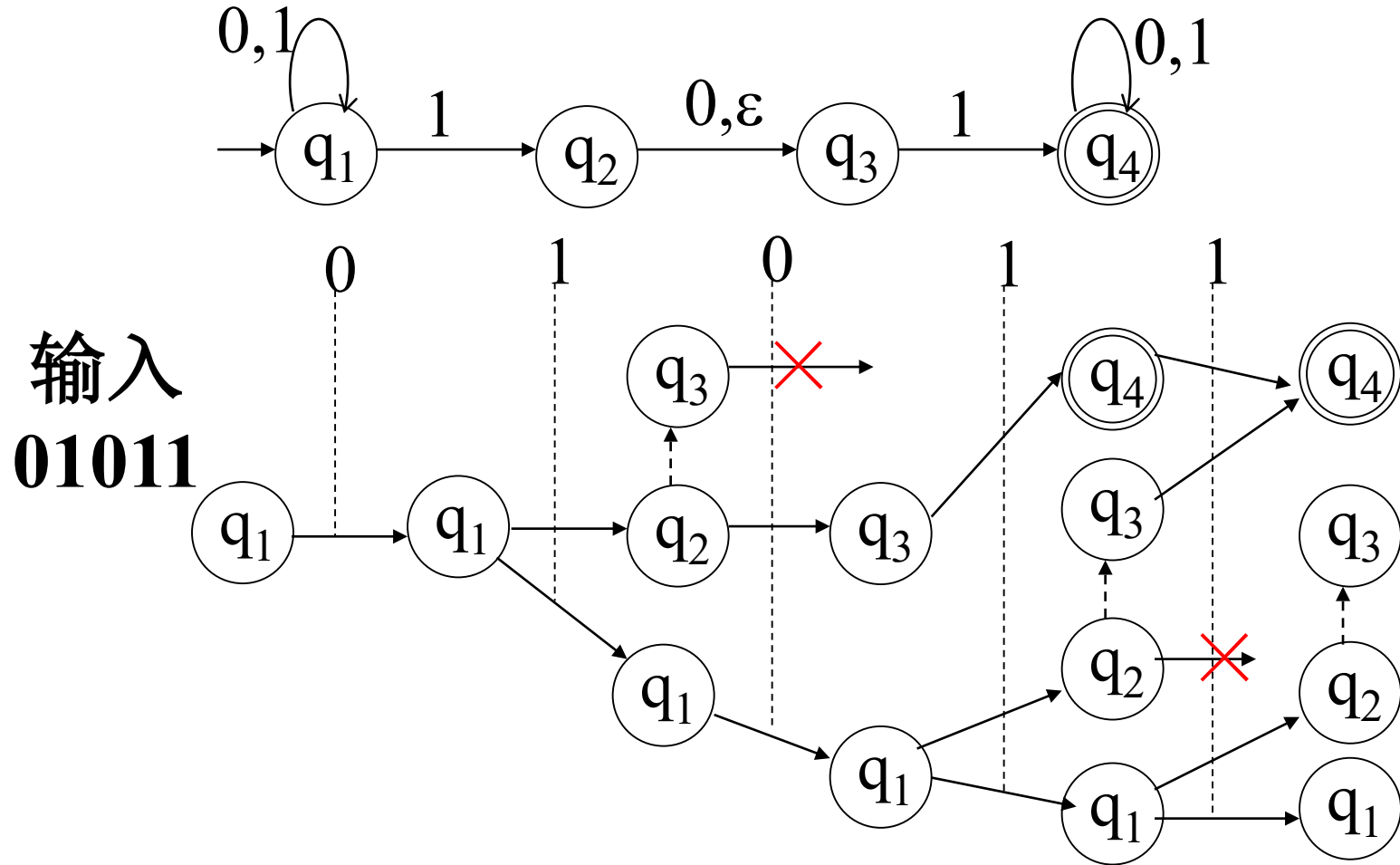
非确定型机器

现在引入**非确定型**有限自动机(NFA)

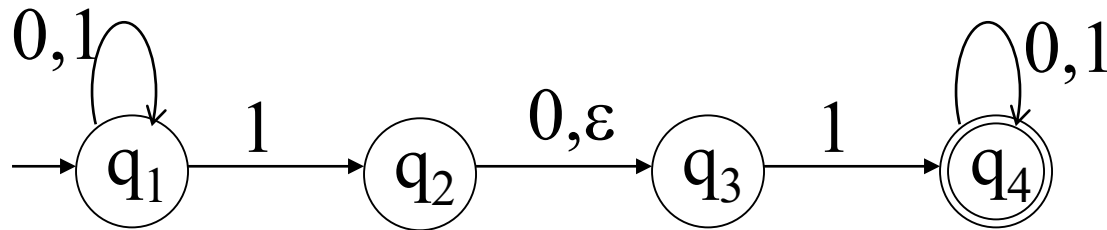


- 每步可以**0**至多种方式进入下一步
- 转移箭头上的符号可以是空串 ϵ ,
表示不读任何输入就可以转移过去

非确定型计算



NFA的形式定义



定义: **NFA**是一个5元组 $(Q, \Sigma, \delta, s, F)$,

1) Q 是状态集;

2) Σ 是字母表;

3) $\delta: Q \times \Sigma_\epsilon \rightarrow P(Q)$ 是转移函数;

4) $s \in Q$ 是起始状态;

5) $F \subseteq Q$ 是接受状态集;

其中 $\Sigma_\epsilon = \Sigma \cup \{\epsilon\}$

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

$$\delta(q_2, \epsilon) = \{q_3\}$$

$$\delta(q_2, 1) = \emptyset$$

$$\delta(q_1, \epsilon) = \emptyset$$

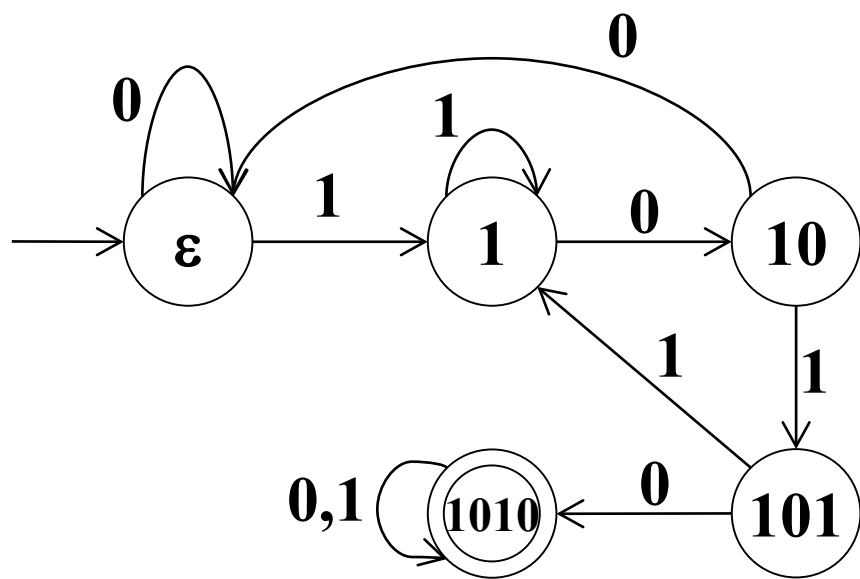
NFA的设计

- 自己即自动机
- 寻找需要记录的关键信息

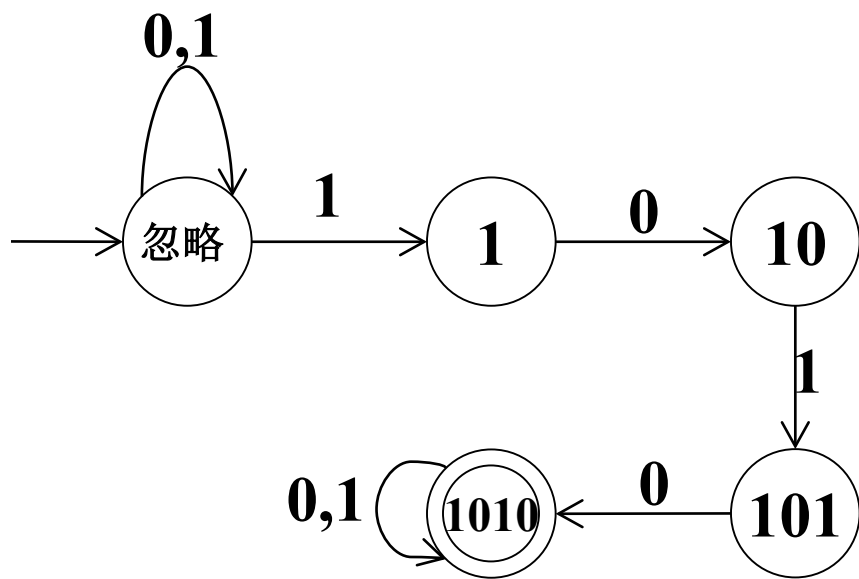
NFA的设计

$\{ w \in \{0,1\}^* \mid w \text{ 含有子串 } 1010 \}$

$\Sigma = \{0,1\}$, 关键信息: 忽略(ϵ), 1, 10, 101, 1010

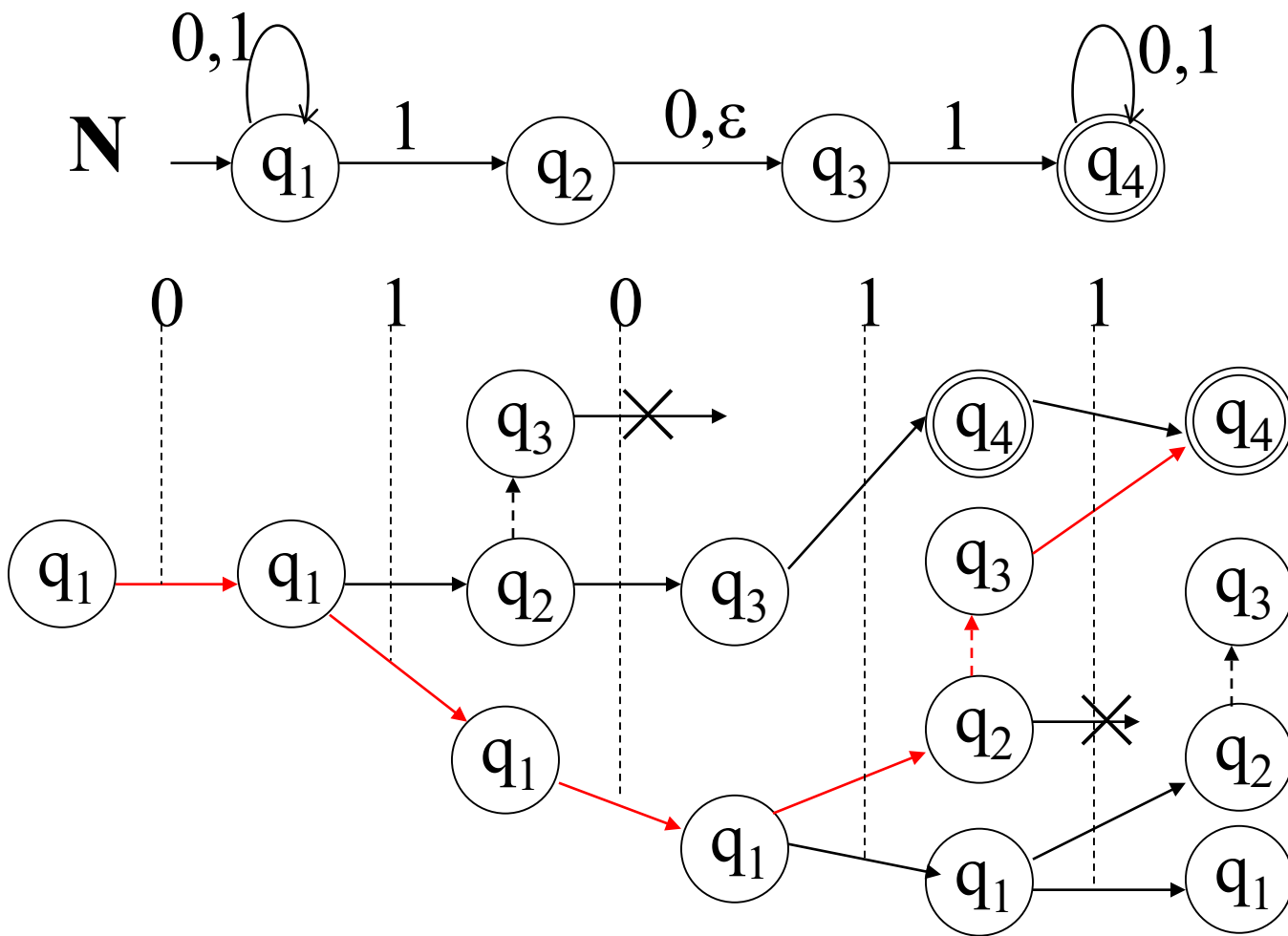


DFA

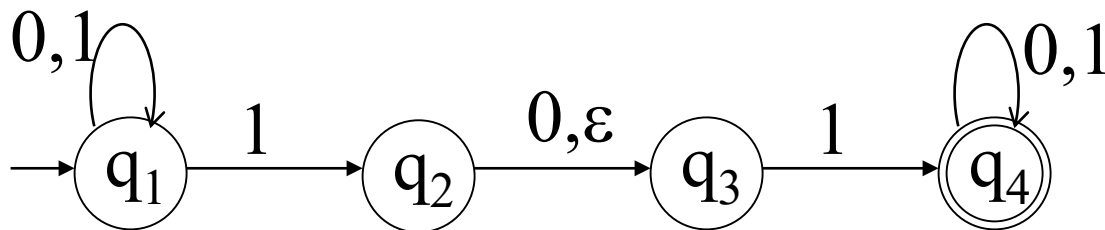


NFA

NFA的计算



每个NFA都有等价的DFA



以原状态的子集
为新机器的状态

编号	δ	0	1
1	$\{q_1\}$ 1	$\{q_1\}$	$\{q_1, q_2, q_3\}$ 2
2	$\{q_1, q_2, q_3\}$	$\{q_1, q_3\}$ 3	$\{q_1, q_2, q_3, q_4\}$ 4
3	$\{q_1, q_3\}$	$\{q_1\}$	$\{q_1, q_2, q_3, q_4\}$
4*	$\{q_1, q_2, q_3, q_4\}$	$\{q_1, q_3, q_4\}$ 5	$\{q_1, q_2, q_3, q_4\}$
5*	$\{q_1, q_3, q_4\}$	$\{q_1, q_4\}$ 6	$\{q_1, q_2, q_3, q_4\}$
6*	$\{q_1, q_4\}$	$\{q_1, q_4\}$	$\{q_1, q_2, q_3, q_4\}$

正则运算的封闭性

定理：正则语言对并运算封闭。

定理：正则语言对连接运算封闭。

定理：正则语言对星号运算封闭。

正则表达式

定义：称 R 是一个正则表达式, 若 R 是

1) $a, a \in \Sigma$;

2) ε ;

3) \emptyset ;

4) $(R_1 \cup R_2)$, R_1 和 R_2 是正则表达式;

5) $(R_1 \circ R_2)$, R_1 和 R_2 是正则表达式;

6) (R_1^*) , R_1 是正则表达式;

每个正则表达式 R 表示一个语言, 记为 $L(R)$.

例: 0^*10^* , $01 \cup 10$, $(\Sigma\Sigma)^*$, $1^*\emptyset$, \emptyset^* .

正则表达式与DFA等价

定理2.3.1: 语言A正则 \Leftrightarrow A可用正则表达式描述.

A正则 \Rightarrow A有正则表达式

构造广义非确定有限自动机(GNFA)

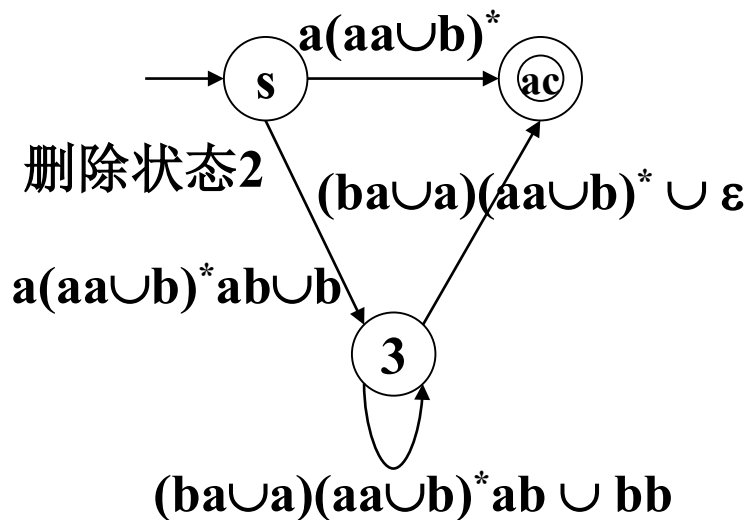
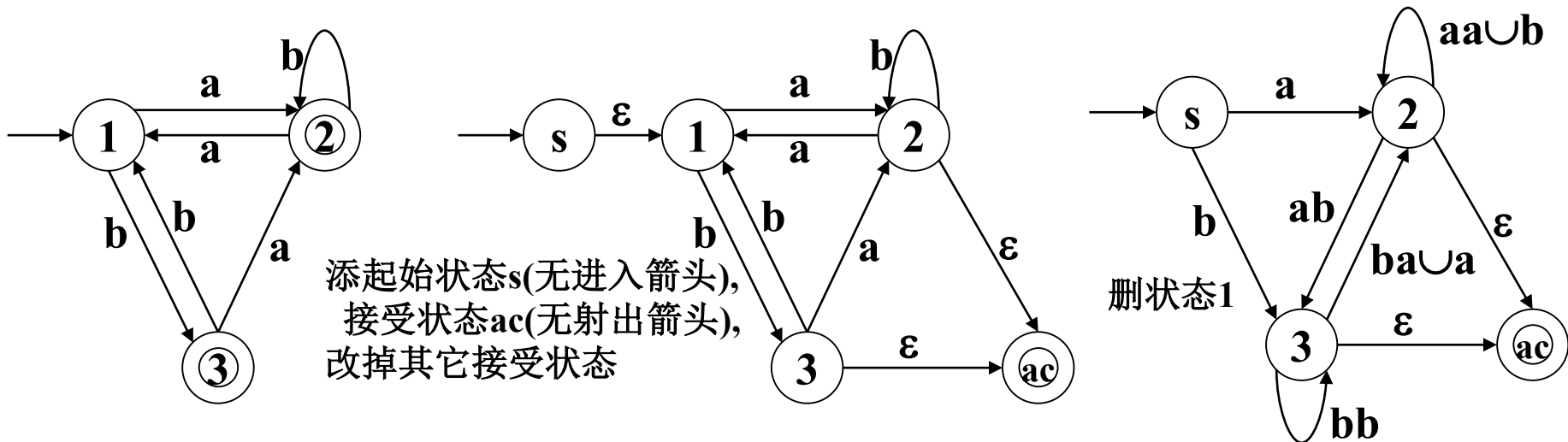
- 非确定有限自动机
- 转移箭头可以用任何正则表达式作标号

证明中的特殊要求:

- 起始状态无射入箭头.
- 唯一接受状态(无射出箭头).

手段: 一个一个地去掉中间状态.

举例: A 正则 $\Rightarrow A$ 有正则表达式



$$((a(aa \cup b)^* ab \cup b)((ba \cup a)(aa \cup b)^* ab \cup bb)^* ((ba \cup a)(aa \cup b)^* \cup \epsilon)) \cup (a(aa \cup b)^*)$$



非正则语言：泵引理的等价描述

定理(泵引理): 设A是正则语言, 则**存在** $p > 0$ 使得

对**任意** $w \in A$, $|w| \geq p$, **存在**分割 $w = xyz$ 满足

1) 对**任意** $i \geq 0$, $xy^iz \in A$;

2) $|y| > 0$;

3) $|xy| \leq p$.

若 A 是正则语言,

则 $\exists p > 0$

$\forall w \in A (|w| \geq p)$

$\exists x, y, z (|y| > 0, |xy| \leq p, w = xyz)$

$\forall i \geq 0,$

$xy^iz \in A.$

若 $\forall p > 0$

$\exists w \in A (|w| \geq p)$

$\forall x, y, z (|y| > 0, |xy| \leq p, w = xyz)$

$\exists i \geq 0,$

$xy^iz \notin A.$

则 A 非正则语言

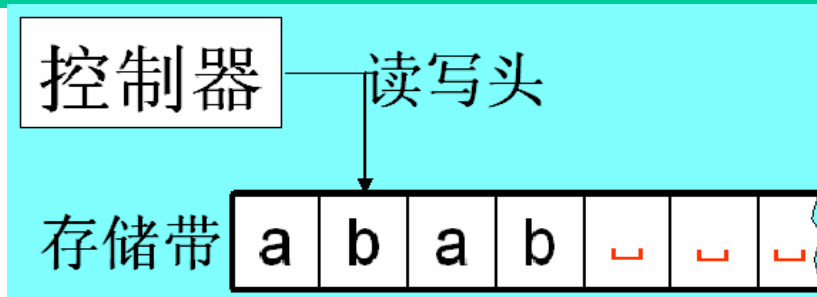
非正则语言: $B = \{ 0^n 1^n \mid n \geq 0 \}$ 非正则

$\therefore \forall p > 0,$
令 $w = 0^p 1^p,$
 $\forall x, y, z (|y| > 0, |xy| \leq p, w = xyz)$
令 $i = 0,$
 $xz = 0^{p-|y|} 1^p \notin B$
 $\therefore B$ 非正则语言

若 $\forall p > 0$
 $\exists w \in A (|w| \geq p)$
 $\forall x, y, z (|y| > 0, |xy| \leq p, w = xyz)$
 $\exists i \geq 0,$
 $xy^i z \notin A.$
则 A 非正则语言

第3章 图灵机

图灵机(TM)的形式化定义



TM是一个7元组 $(Q, \Sigma, \Gamma, \delta, q_0, q_a, q_r)$

1) Q 是状态集.

2) Σ 是输入字母表,不包括空白符 \sqcup .

3) Γ 是带字母表,其中 $\sqcup \in \Gamma, \Sigma \subset \Gamma$.

4) $\delta: Q \times \Gamma \rightarrow Q \times \Gamma \times \{L, R\}$ 是转移函数.

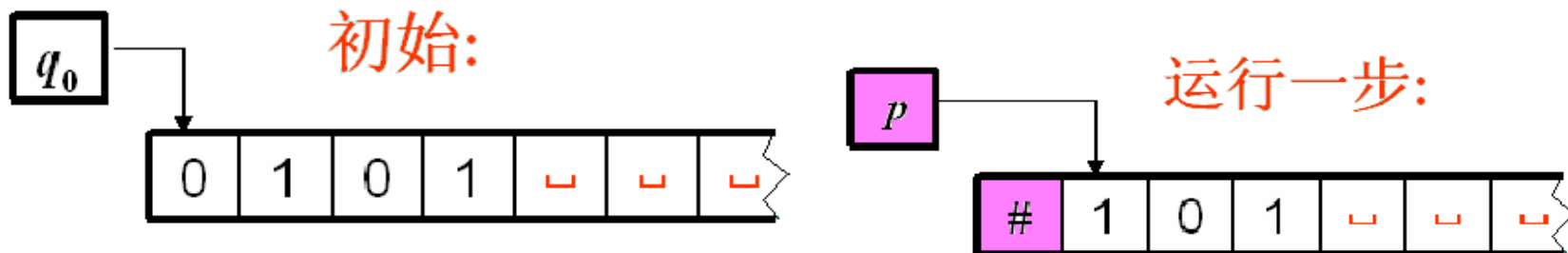
5) $q_0 \in Q$ 是起始状态. 6) $q_a \in Q$ 是接受状态.

7) $q_r \in Q$ 是拒绝状态, $q_a \neq q_r$.

图灵机的运行

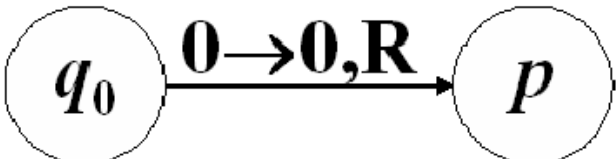
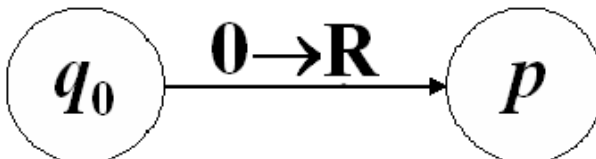
- 图灵机根据转移函数运行.

例: 设输入串为0101, 且 $\delta(q_0, 0) = (p, \#, R)$, 则有



- 注: 若要在最左端左移, 读写头保持不动.

$\delta(q_0, 0) = (p, \#, R)$ 的状态图表示:

 简记为 

判定器与语言分类

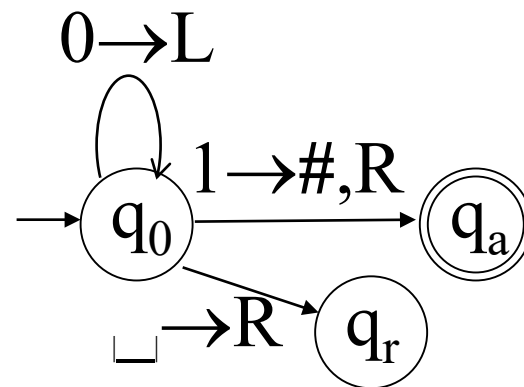
- 图灵机运行的三种结果
 1. 若TM进入**接受状态**,则停机且接受输入,
 2. 若TM进入**拒绝状态**,则停机且拒绝输入,
 3. 否则TM一直运行,不停机.

- 定义: 称图灵机M为**判定器**,
若M对所有输入都停机.

- 定义不同语言类:

图灵可判定语言: 某个判定器的语言

图灵可识别语言: 某个图灵机的语言,



图灵机的描述

- (1) 形式水平的描述(状态图或转移函数)
- (2) 实现水平的描述(读写头的移动,改写)
- (3) 高水平描述(使用日常语言)

用带引号的文字段来表示图灵机. 例如:

M=“对于输入串 w ,

- 1) 若 $w=\varepsilon$, 则拒绝.
- 2) 若只有1个0, 则接受.
- 3) 若0的个数为奇数, 则拒绝.
- 4) 从带左端隔一个0, 删一个0. 转(2).”

图灵机的变形

图灵机有多种变形:

例如多带图灵机, 非确定图灵机

还有如枚举器, 带停留的图灵机等等

只要满足必要特征,

它们都与这里定义的图灵机等价.

非确定型图灵机(NTM)

- NTM的转移函数

$$\delta: Q \times \Gamma \rightarrow P(Q \times \Gamma \times \{L, R\})$$

- NTM转移函数举例

$$\delta(q_3, 0) = \{(q_2, x, R), (q_1, 1, L), (q_3, \$, R)\}$$

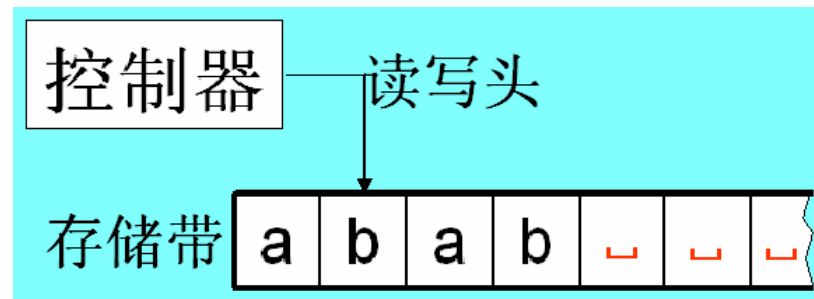
- 称NTM M接受x, 若在x上运行M时有接受分支.

- 称一NTM为判定的,

若它对所有输入,所有分支都停机.

- 定理: 每个NTM都有等价的确定TM.

- 定理: 每个判定NTM都有等价的判定TM.



计算理论

第二部分 可计算理论

第4章 可判定性

定理:停机问题Halt是图灵可识别的

$\text{Halt} = \{ \langle M, x \rangle \mid \text{图灵机 } M \text{ 在串 } x \text{ 上会停机} \}$

证明: 构造识别Halt的图灵机T,

T = “对于输入 $\langle M, x \rangle$, M是图灵机, x是串

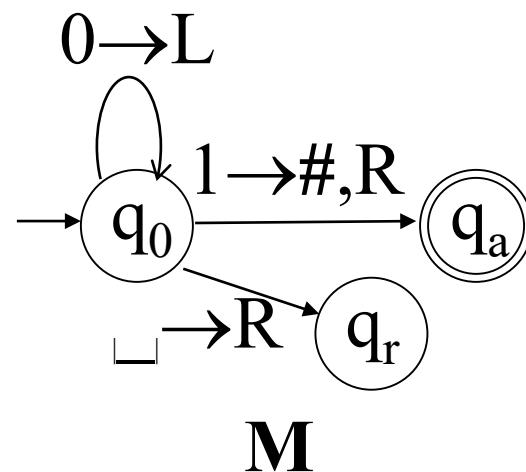
1. 在x上模拟M,

2. 若M停机(接受或拒绝), 则接受.”

T的语言是Halt, 证毕.

注: T不是判定器 (?)

例输入 $\langle M, 01 \rangle$



定理:停机问题Halt不可判定

$\text{Halt} = \{ \langle M, x \rangle \mid \text{图灵机 } M \text{ 在 } x \text{ 上会停机} \}$

证明: 假设Halt有判定器H, 构造图灵机D使用H:

Diagonal = “对于输入 $\langle M \rangle$, M是图灵机,

1. 在 $\langle M, \langle M \rangle \rangle$ 上运行H,
2. 若H接受, 则返回1;
3. 若H拒绝, 则停机.”

- 在Diagonal上输入 $\langle \text{Diagonal} \rangle$ ”是否会停机?
- 若D停机, 即 $\langle D, \langle D \rangle \rangle \in \text{HALT}$, H接受 $\langle D, \langle D \rangle \rangle$, 则由2, D不停机
- 若D不停机, 即 $\langle D, \langle D \rangle \rangle \notin \text{HALT}$, H拒绝 $\langle D, \langle D \rangle \rangle$, 则由3, D停机
- 矛盾, 所以H不存在.

定理: A_{TM} 的补不是图灵可识别的

定理: 若 A 和 A 的补都是图灵可识别, 则 A 图灵可判定

证明: 设图灵机 T 和 Q 分别识别 A 和 A 的补, 构造 R :

R = “对于输入 x , x 是串,

1. 在 x 上同步模拟 T 和 Q , 直到有一个接受,
2. 若 T 接受 x , 则接受; 若 Q 接受 x , 则拒绝.”

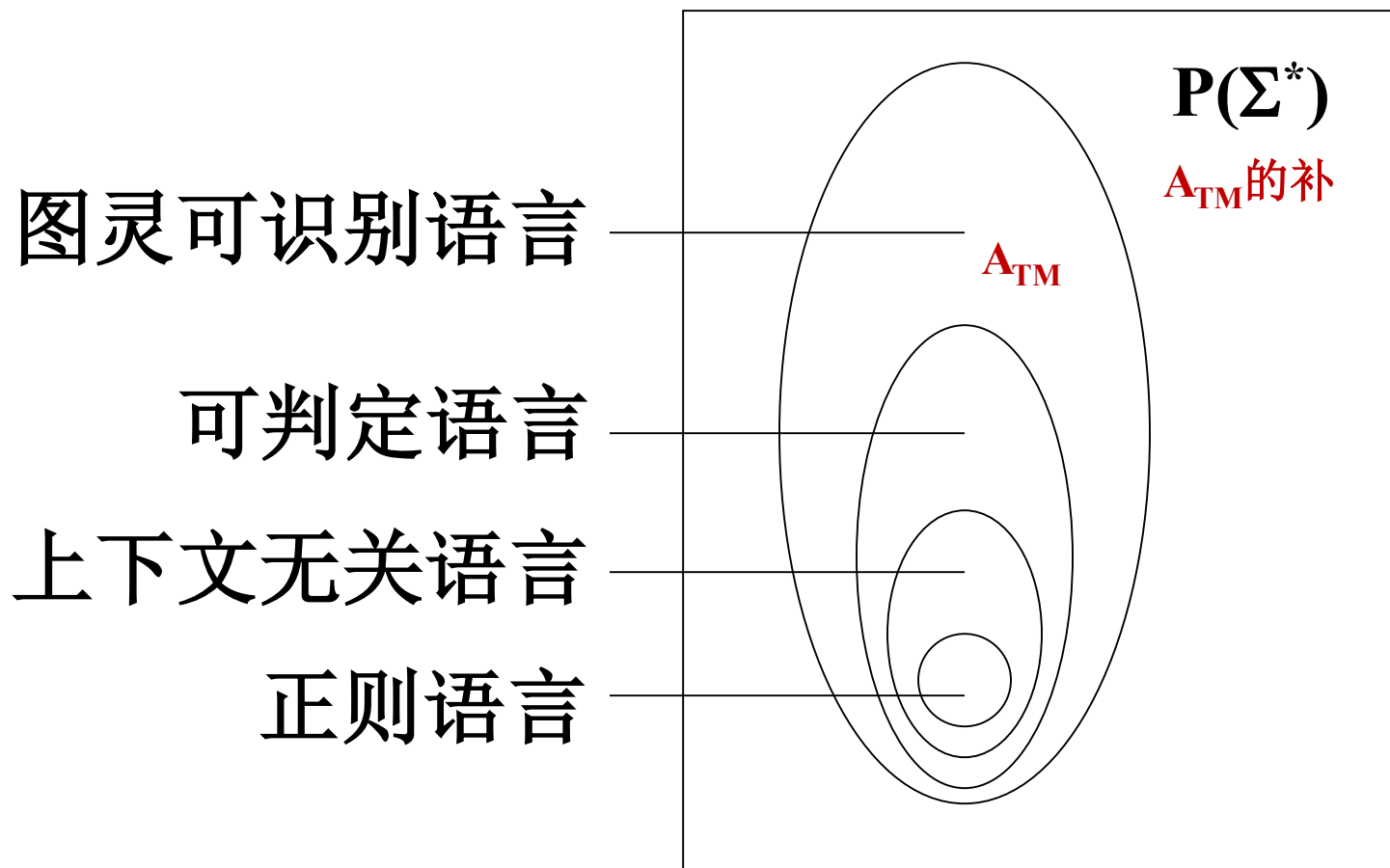
$x \in A \Rightarrow T \text{ 接受 } x \Rightarrow R \text{ 接受 } x$

$x \notin A \Rightarrow Q \text{ 接受 } x \Rightarrow R \text{ 拒绝 } x$

1. R 是判定器
2. R 的语言是 A .

推论: A_{TM} 的补不是图灵可识别的.

各语言类之间的关系



可判定性

成员测试:

$A_{\text{DFA}} = \{ \langle B, w \rangle \mid B \text{ 是 DFA, } w \text{ 是串, } B \text{ 接受 } w \}$ 可判定

$A_{\text{TM}} = \{ \langle M, w \rangle \mid M \text{ 是一个 TM, 且接受 } w \}$ 不可判定

空性质测试: $E_{\text{DFA}} = \{ \langle A \rangle \mid A \text{ 是 DFA, } L(A) = \emptyset \}$ 可判定

等价性质测试:

$EQ_{\text{DFA}} = \{ \langle A, B \rangle \mid A \text{ 和 } B \text{ 都是 DFA, 且 } L(A) = L(B) \}$ 可判定

计算理论

第三部分 计算复杂性

第7章 时间复杂性

1. 时间复杂性

$\{0^k1^k \mid k \geq 0\}$ 的时间复杂性分析

2. 不同模型的运行时间比较

单带与多带 确定与非确定

3. P类与NP类

4. NP完全性及NP完全问题

时间复杂性

- 判定器M的**运行时间或时间复杂度**是 $f:N \rightarrow N$,
 $f(n)$ 是M在**所有长为n的输入**上运行的最大步数.
- 若 $f(n)$ 是M的运行时间, 则称
M在时间 $f(n)$ 内运行 或 M是 $f(n)$ 时间图灵机

分析算法

讨论语言 $A = \{ 0^k 1^k \mid k \geq 0 \}$ 的复杂性:

M_1 = “对输入串 w :

- 1) 扫描带, 如果在1的右边发现0, 则拒绝.
- 2) 如果0和1都在带上, 就重复下一步.
- 3) 扫描带, 删除一个0和一个1.
- 4) 如果带上同时没有0和1, 就接受.”

时间分析: (1) $2n = O(n)$, (4) $n = O(n)$,

$$\{ (2) 2n = O(n) + (3) 2n = O(n) \} \times (n/2) = O(n^2)$$

所以 M_1 的运行时间是 $O(n^2)$.

图灵机 M_2

M_2 = “对输入串 w :

1) 扫描带, 若1的右边有0, 则拒绝. $O(n)$

2) 若0,1都在带上, 重复以下步骤. $O(n)$

3) 检查带上0,1总数的奇偶性,
若是奇数, 就拒绝. $O(n)$

4) 再次扫描带,
第1个0开始, 隔1个0删除1个0; $O(n)$
第1个1开始, 隔1个1删除1个1.

5) 若带上同时没有0和1, 则接受. $O(n)$

否则拒绝.”

$\times \log n$

总时间:

$O(n \log n)$

单带与多带运行时间比较

$\{ 0^k 1^k \mid k \geq 0 \}$ 有 $O(n)$ 时间双带图灵机

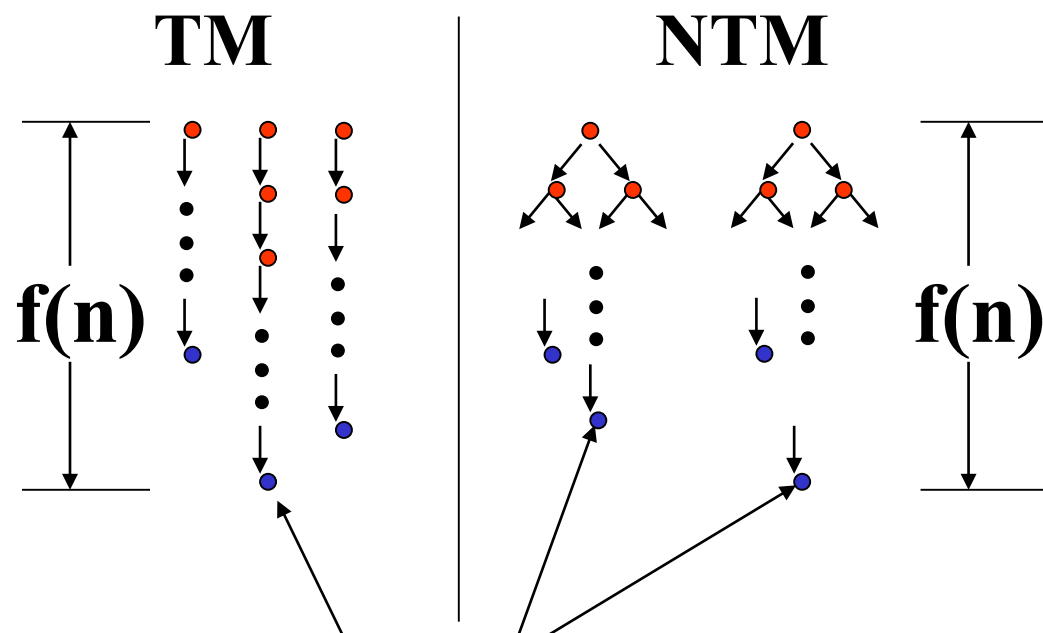
M_3 = “对输入串 w :

- 1) 扫描1带,如果在1的右边发现0,则拒绝.
- 2) 将1带的1复制到2带上.
- 3) 每删除一个1带的0就删除一个2带的1.
- 4) 如果两带上同时没有0和1,就接受.”

定理: 设函数 $t(n) \geq n$, 则每个 $t(n)$ 时间多带TM
和某个 $O(t^2(n))$ 时间单带TM等价.

NTM的运行时间

定义: 对非确定型判定器 N , 其运行时间 $f(n)$ 是在**所有**长为 n 的输入上, **所有**分支的最大步数.



定理: 设 $t(n) \geq n$,
则每个
时间 $t(n)$ NTM
都有一等价的
时间 $2^{O(t(n))}$ TM.

$$\text{NTIME}(t(n)) \subseteq \text{TIME}(2^{O(t(n))})$$

P类

定义:P是单带确定TM在
多项式时间内可判定的问题,即

$$P = \cup_k \text{TIME}(n^k)$$

P类的重要性在于:

- 1) 对于所有与单带确定TM等价的模型,P不变.
- 2) P大致对应于在计算机上实际可解的问题.
研究的核心是一个问题是否属于P类.

NP类

$\text{NTIME}(t(n)) = \{L \mid L \text{ 可被 } O(t(n)) \text{ 时间 NTM 判定.}\}$

定义: NP 是单带非确定 TM 在

多项式时间内可判定的问题, 即

$$\text{NP} = \bigcup_k \text{NTIME}(n^k)$$

NP问题

团:无向图的完全子图(所有节点都有边相连).

$\text{CLIQUE} = \{ \langle G, k \rangle \mid G \text{ 是有 } k \text{ 团的无向图} \}$

定理: $\text{CLIQUE} \in \text{NP}$.

N=“对于输入 $\langle G, k \rangle$,这里G是一个图:

- 1)非确定地选择G中k个节点的子集c.
- 2)检查G是否包含连接c中节点的所有边.
- 3)若是,则接受;否则,拒绝.”

P与NP

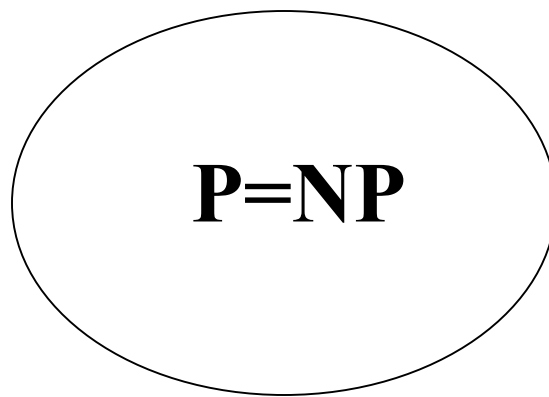
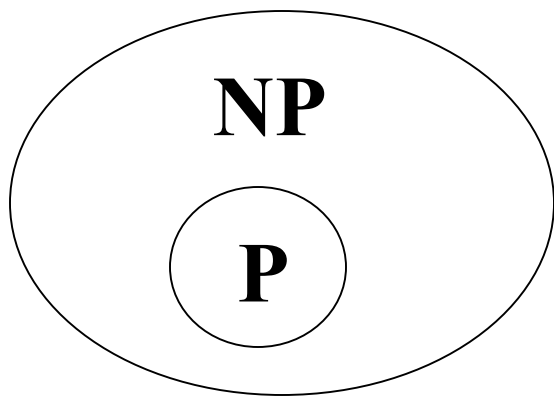
P=成员资格可以**快速判定**的语言类.

NP=成员资格可以**快速验证**的语言类.

显然有 $P \subseteq NP$

但是否有 $P = NP$?

看起来难以想象,但是现在没有发现反例.



当代数学与
理论计算机
共同的难题.

可满足问题SAT

- 可满足性问题:

$$\text{SAT} = \{ \langle \phi \rangle \mid \phi \text{ 是可满足的布尔公式} \}$$

- 二元可满足性问题:

$$\text{2SAT} = \{ \langle \phi \rangle \mid \phi \text{ 是可满足的2cnf} \}$$

- 三元可满足性问题:

$$\text{3SAT} = \{ \langle \phi \rangle \mid \phi \text{ 是可满足的3cnf} \}$$

二元可满足问题 2SAT $\in P$

1. 当2cnf中有子句是单文字 x , 则反复执行(直接)清洗

1.1 由 x 赋值, 1.2 删去含 x 的子句, 1.3 删去含 $\neg x$ 的文字
若清洗过程出现相反单文子子句, 则清洗失败并结束

$$(x_1 \vee x_2) \wedge (x_3 \vee \neg x_2) \wedge (x_1) \wedge (\neg x_1 \vee \neg x_2) \wedge (x_3 \vee x_4) \wedge (\neg x_3 \vee x_5) \wedge (\neg x_4 \vee \neg x_5) \wedge (\neg x_3 \vee x_4)$$

$$\rightarrow (x_3 \vee \neg x_2) \wedge (\neg x_2) \wedge (x_3 \vee x_4) \wedge (\neg x_3 \vee x_5) \wedge (\neg x_4 \vee \neg x_5) \wedge (\neg x_3 \vee x_4)$$

$$\rightarrow (x_3 \vee x_4) \wedge (\neg x_3 \vee x_5) \wedge (\neg x_4 \vee \neg x_5) \wedge (\neg x_3 \vee x_4)$$

2. 若无单文字子句, 则任选变量赋真/假值各(赋值)清洗一次
若两次都清洗失败, 则回答不可满足.

$$x_3=1 \rightarrow (x_5) \wedge (\neg x_4 \vee \neg x_5) \wedge (x_4) \rightarrow (\neg x_4) \wedge (x_4) \text{ 失败}$$

$$x_3=0 \rightarrow (x_4) \wedge (\neg x_4 \vee \neg x_5) \rightarrow (\neg x_5) \rightarrow \emptyset \text{ 成功}$$

3. 若成功清洗后有子句剩下, 则继续2. 否则, 回答可满足.

3SAT \in NP

三元可满足性问题:

$$3SAT = \{ \langle \phi \rangle \mid \phi \text{ 是可满足的 3cnf} \}$$

P时间内判定3SAT的NTM:

N=“对于输入 $\langle \phi \rangle$, ϕ 是一个 3cnf 公式,

1) 非确定地选择各变量的赋值 T .

2) 若在赋值 T 下 $\phi=1$, 则接受; 否则拒绝.”

第2步在公式长度的多项式时间内运行.

多项式时间映射归约与C-L定理

- Cook-Levin定理: $\text{SAT} \in \text{P} \Leftrightarrow \text{P} = \text{NP}$.
- 定义: 多项式时间可计算函数 $f: \Sigma^* \rightarrow \Sigma^*$.
- 定义: 称A可多项式时间映射归约到B ($A \leq_p B$),
若存在多项式时间可计算函数 $f: \Sigma^* \rightarrow \Sigma^*$,

$$\forall w \in \Sigma^*, w \in A \Leftrightarrow f(w) \in B.$$

函数f称为A到B的多项式时间归约.

通俗地说: f 将A的实例编码转换为B的实例编码.

- Cook-Levin定理: 对任意 $A \in \text{NP}$ 都有 $A \leq_p \text{SAT}$.
- 定理1: 若 $A \leq_p B$, 且 $B \in \text{P}$, 则 $A \in \text{P}$.
- 注: 定理1说明, 若 $\text{SAT} \in \text{P}$, 则 $\text{NP} = \text{P}$.

定理: $3SAT \leq_p CLIQUE$

$3SAT = \{ \langle \phi \rangle \mid \phi \text{ 是可满足的 3cnf 公式} \}$

$CLIQUE = \{ \langle G, k \rangle \mid G \text{ 是有 } k \text{ 团的无向图} \}.$

NP完全性

- 定义:语言B称为NP完全的(NPC),若它满足:
1) $B \in NP$; 2) $\forall A \in NP$, 都有 $A \leq_p B$.
- 定理1:若 $A \leq_p B$, 且 $B \in P$, 则 $A \in P$.
- 定理2: 若B是NPC, 且 $B \in P$, 则 $P=NP$.
- 定理3: 若B是NPC, $B \leq_p C$, 且 $C \in NP$, 则C是NPC.
证明: $\forall A \in NP, (A \leq_p B) + (B \leq_p C) \Rightarrow A \leq_p C$
- Cook-Levin定理: SAT是NP完全问题.
- 推论: CLIQUE是NPC.

$\forall A \in \text{NP}$, 都有 $A \leq_p \text{SAT}$

其它NP完全问题



计算理论总结

计算模型

- 有限自动机 非确定有限自动机 正则表达式
正则语言 泵引理
- 图灵机 图灵可判定语言 图灵可识别语言

可计算理论

停机问题非图灵可判定,

停机问题的补不是图灵可识别

计算复杂性

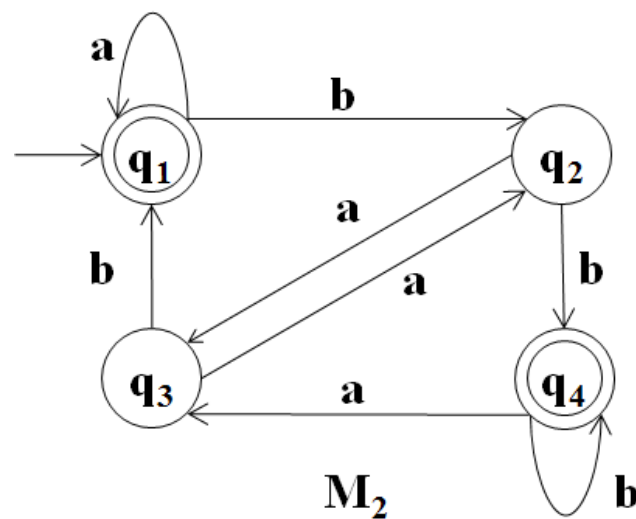
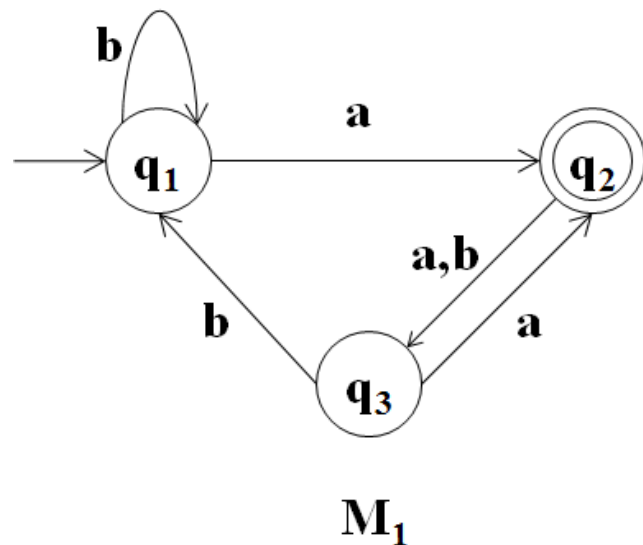
- **P, NP, NPC**

计算理论基础第1章作业

1.1 下图给出了两台DFA M_1 和 M_2 的状态图。

回答下述关于这两台机器的问题。

- a. 它们的起始状态是什么?
- b. 它们的接受状态集是什么?
- c. 对输入aabb, 它们经过的状态序列是什么?
- d. 它们接受字符串aabb吗?
- e. 它们接受字符串 ϵ 吗?



计算理论基础第1章作业

1.6 画出识别下述语言的DFA状态图. 字母表为 $\{0,1\}$

d. $\{ w \mid w \text{ 的长度不小于3, 并且第3个符号为0} \}$;

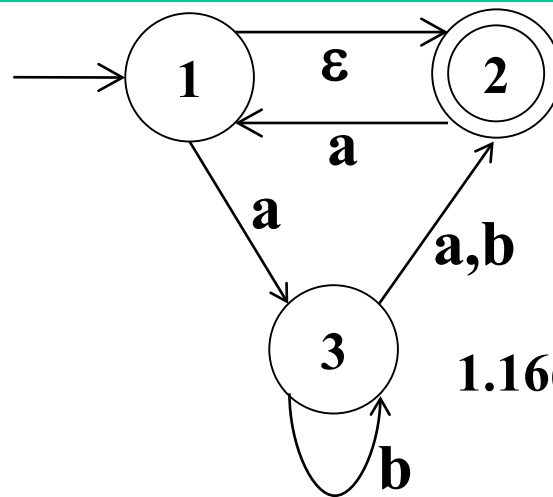
1.7. 给出下述语言的NFA, 并且符合规定的状态数.

字母表为 $\{0,1\}$

e. 语言 $0^*1^*0^*0$, 3个状态.

计算理论基础第1章作业

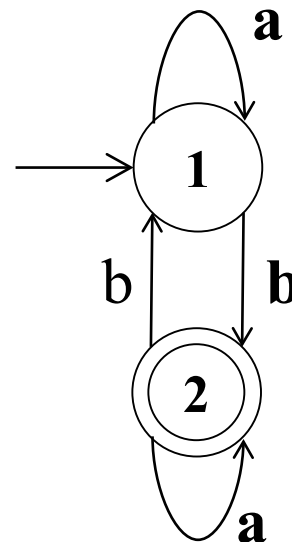
1.16(b) 将如右图的非确定有限自动机转换成等价的确定有限自动机.



1.16(b)题图

计算理论基础第1章作业

1.21(a) 将如右图的有限自动机转换成等价的正则表达式.



1.21(a)题图

1.29 使用泵引理证明下述语言不是正则的。

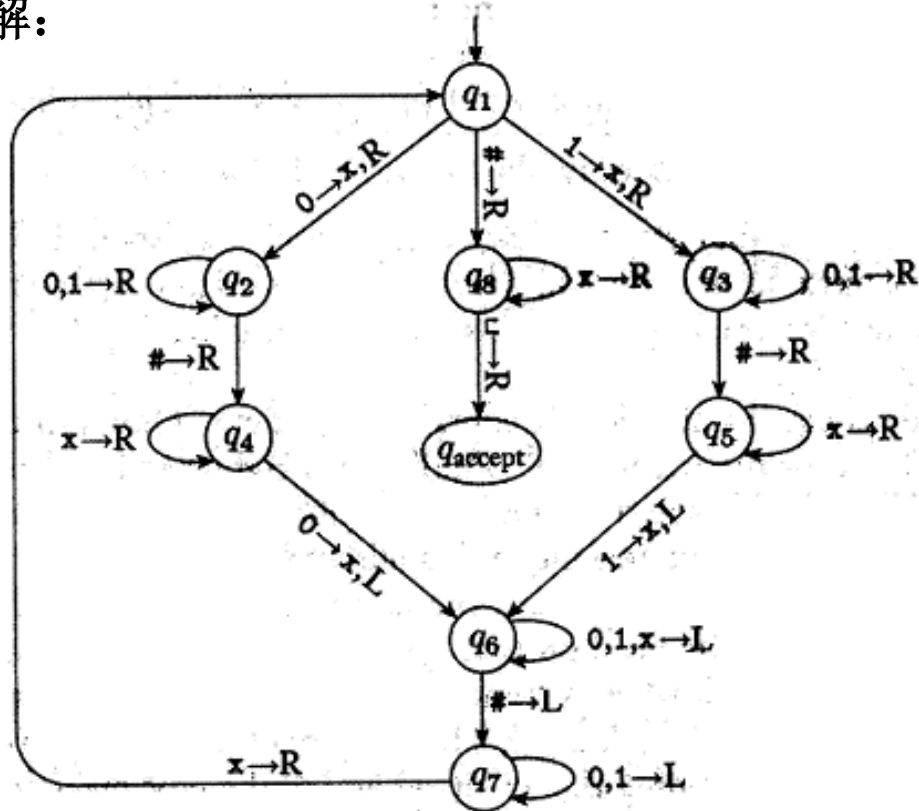
b. $A = \{ www \mid w \in \{a,b\}^* \}$

计算理论第3章作业

3.2 对于识别 $\{w|w=u\#u, u \in \{0,1\}^*\}$ 的图灵机 M_1 (见左图), 在下列输入串上, 给出 M 所进入的格局序列.

c. 1##1, d. 10#11, e. 10#10

解:



补充说明: 没有画出的箭头指向拒绝状态, 假设这些箭头都不改写右移且 q_r 是拒绝状态.

计算理论第3章作业

3.8 下面的语言都是字母表 $\{0,1\}$ 上的语言, 以实现水平的描述给出判定这些语言的图灵机:

b. $B = \{w | w \text{ 所包含的 } 0 \text{ 的个数是 } 1 \text{ 的个数的两倍}\}$

3.15b 证明图灵可判定语言类在连接运算下封闭.

3.16d 证明图灵可识别语言类在交运算下封闭.

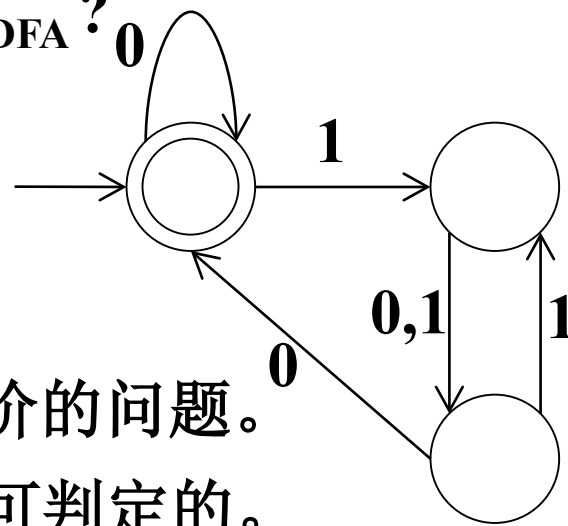
计算理论第4章作业

4.1 对于右图所示的DFA M , 回答下列问题, 并说明理由

a. $\langle M, 0100 \rangle \in A_{DFA}$? b. $\langle M, 011 \rangle \in A_{DFA}$?

c. $\langle M \rangle \in A_{DFA}$?

e. $\langle M \rangle \in E_{DFA}$? f. $\langle M, M \rangle \in EQ_{DFA}$?



4.2 考虑一个DFA和一个正则表达式是否等价的问题。

将这个问题描述为一个语言并证明它是可判定的。

4.3 设 $ALL_{DFA} = \{ \langle A \rangle \mid A \text{ 是一个识别 } \Sigma^* \text{ 的 DFA} \}$. 证明 ALL_{DFA} 可判定.

计算理论第7章作业

7.9 无向图中的三角形是一个3团。证明 $\text{TRIANGLE} \in \text{P}$ ，其中 $\text{TRIANGLE} = \{ \langle G \rangle \mid G \text{ 包含一个三角形} \}$ 。

计算理论第7章作业

7.11 若图G的节点重新排序后，G可以变得与H完全相同，则称G与H是同构的。令 $ISO = \{ \langle G, H \rangle \mid G \text{ 和 } H \text{ 是同构的图} \}$ 。证明 $ISO \in NP$ 。

证明:构造如下非确定图灵机

N= “对于输入 $\langle G, H \rangle$, G和H都是图,

- 1) 若G和H顶点数不同, 则拒绝.
- 2) 设G的顶点为 x_1, x_2, \dots, x_n , H的顶点为 y_1, y_2, \dots, y_n .
- 3) 非确定的选择1到n的排列p.
- 4) 对 $i = 1$ 到 $n-1$
- 5) 对 $j = i+1$ 到 n
- 6) 若 $(x_i, x_j) \in E(G)$ 异或 $(y_{p(i)}, y_{p(j)}) \in E(H)$ 为真, 则拒绝
- 7) 接受.”。

若G, H同构, 则 N一定有分支接受; 否则, N所有分支拒绝.

N的所有分支都在都在 $O(n^2)$ 时间内运行.

所以, N是ISO的多项式时间非确定判定器, $ISO \in NP$.