

第2部分 可计算性理论

图灵机

- 第3章 丘奇-图灵论题
- 第6章 可计算性理论的高级专题(递归定理)

(不)可计算问题

- 第4章 可判定性
- 第5章 可归约性

第3章 丘奇-图灵论题

图灵机(TM)

- 多带图灵机
- 非确定型图灵机(NTM)
- 枚举器
- 图灵可识别, 图灵可判定

丘奇-图灵论题

- 算法、希尔伯特第10问题

图灵机的描述

- 形式描述; 实现描述; 高层描述
- 编码: $\langle O \rangle$

第6章 可计算性高级专题

★ 递归定理

★ 自引用

★ 应用递归定理的术语

- ★ 可以在M的定义中使用 $\langle M \rangle$

★ 应用

- ★ 图灵机接受性问题

- ★ 图灵机极小性问题

★ 通用机

第4章 可判定性

★ 可判定语言

- ★ 关于正则语言的可判定问题
- ★ 关于上下文无关语言的可判定问题

★ 停机问题

- ★ 对角化方法
- ★ 停机问题是不可判定的
- ★ 非图灵可识别语言

第5章 可归约性

★ 语言理论中的不可判定问题

- ★ 利用计算历史的归约

- ★ 线性界限自动机

★ 波斯特对应问题

★ 映射归约(多一归约, m 归约)



单带图灵机的例子

单带图灵机的示意图



- ✦ 双向读写头
- ✦ 无穷带
- ✦ 停机状态

识别 $w#w$ 的单带图灵机 M_1

★ $B = \{ w#w \mid w \in \{0,1\}^* \}$

★ $M_1 =$ “对于输入字符串 x :

1) 扫描输入,确认只含一个#.

否则拒绝.

2) 在#两边对应位置来回移动,

检查是否含相同符号.

若不是,则拒绝.

消去已检查过符号.

3) 当消去#左边所有符号时,

检查#右边是否还有符号,

若是,则拒绝. 否则接受.”

M_1 对011000#011000计算

0 1 1 0 0 0 # 0 1 1 0 0 0



M_1 对011000#011000计算

X 1 1 0 0 0 # 0 1 1 0 0 0

↑₀

M_1 对011000#011000计算

X 1 1 0 0 0 # 0 1 1 0 0 0

↑₀

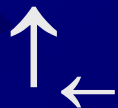
M_1 对011000#011000计算

X 1 1 0 0 0 # 0 1 1 0 0 0

↑
0,#

M_1 对011000#011000计算

X 1 1 0 0 0 # X 1 1 0 0 0



M_1 对011000#011000计算

X 1 1 0 0 0 # X 1 1 0 0 0

↑
←,#

M_1 对011000#011000计算

X 1 1 0 0 0 # X 1 1 0 0 0

↑
←,#

M_1 对011000#011000计算

X 1 1 0 0 0 # X 1 1 0 0 0



M_1 对011000#011000计算

XX1000#X11000

↑₁

M_1 对011000#011000计算

XX1000#X11000

↑₁

M_1 对011000#011000计算

XX1000#X11000

↑₁

M_1 对011000#011000计算

XX1000#X11000

↑
1,#

M_1 对011000#011000计算

XX1000#X11000

↑
1,#

M_1 对011000#011000计算

XX1000#XX1000



M_1 对011000#011000计算

XX1000#XX1000

↑
←,#

M_1 对011000#011000计算

XX1000#XX1000

↑
←,#

M_1 对011000#011000计算

XX1000#XX1000



M_1 对011000#011000计算

XX X 000#XX1000

\uparrow_1

M_1 对011000#011000计算

X X X 0 0 0 # X X 1 0 0 0

↑
1,#

M_1 对011000#011000计算

XXX000#XX1000

↑
1,#

M_1 对011000#011000计算

X X X 0 0 0 # X X X 0 0 0



M_1 对011000#011000计算

X X X X X 0 # X X X X X 0



M_1 对011000#011000计算

XXXXXXXX#XXXXXX0

↑₀

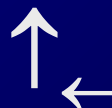
M_1 对011000#011000计算

XXXXXXXX#XXXXX0

↑
0,#

M_1 对011000#011000计算

XXXXXXXX#XXXXXXXX



M_1 对011000#011000计算

X X X X X X # X X X X X X

↑
←,#

M_1 对011000#011000计算

XXXXXXXX#XXXXXXXX



M_1 对011000#011000计算

XXXXXXXX#XXXXXXXX

↑
B

M_1 对011000#011000计算

XXXXXXXX#XXXXXXXX

↑
B

M_1 对011000#011000计算

X X X X X X # X X X X X X

↑
accept



单带图灵机的定义

单带图灵机的定义

★ **定义4.1:** TM $M=(Q,\Sigma,\Gamma,\delta,q_0,q_{acc},q_{rej})$

1) Q 是有穷状态集

2) Σ 是输入字母表, 空格符 $B \notin \Sigma$

3) Γ 是带字母表, $\Sigma \cup \{B\} \subseteq \Gamma$,

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

5) $q_0 \in Q$ 是初始状态

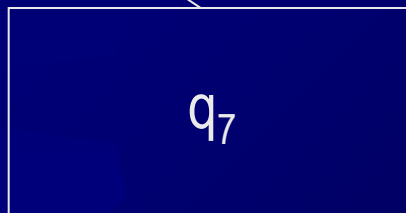
6) $q_{acc} \in Q$ 是**停机接受**状态

7) $q_{rej} \in Q$ 是**停机拒绝**状态, $q_{acc} \neq q_{rej}$

单带图灵机的格局

- ✦ 格局: $uqav$,
 - ✦ 当前状态: q ,
 - ✦ 当前带内容: uav ,
 - ✦ 当前扫描符号: a
- ✦ 初始格局: q_0w , w 是输入串
- ✦ 接受格局: $uq_{acc}av$,
- ✦ 拒绝格局: $uq_{rej}av$,
- ✦ 停机格局: $uq_{acc}av, uq_{rej}av$,

单带图灵机格局的例子



1011 q_7 01111

单带图灵机格局的产生

★ C_1 产生 C_2 :

- ★ 如果 $\delta(q_i, b) = (q_j, c, L)$, 则
 $uaq_i b v$ 产生 $uq_j a c v$
 $q_i b v$ 产生 $q_j c v$
 (在带左端不能向左移)
- ★ 如果 $\delta(q_i, b) = (q_j, c, R)$, 则
 $uaq_i b v$ 产生 $u a c q_j v$

单带图灵机M接受输入w

★ M接受输入w:

存在格局序列 C_1, C_2, \dots, C_k , 使得

1) $C_1 = q_0 w$ 是初始格局

2) 每个 C_i 产生 C_{i+1}

($i=1, \dots, k-1$)

3) C_k 是接受格局

★ M(识别,接受)的语言:

$$L(M) = \{ x \mid M \text{ 接受 } x \}$$

图灵机的计算结果

- ✦ 停机接受
- ✦ 停机拒绝
- ✦ 不停机

图灵可识别, 图灵可判定

★ 图灵可识别: $A=L(M)$.

- ★ $x \in A$ 时, M 在 x 上停机接受
- ★ $x \notin A$ 时, M 在 x 上停机拒绝
或不停机

★ 图灵可判定: $A=L(M)$

- ★ $x \in A$ 时, M 在 x 上停机接受
- ★ $x \notin A$ 时, M 在 x 上停机拒绝
(处处停机)

一些等价术语

★ 图灵可识别 = 递归可枚举 =
计算可枚举 = 半可判定 =
半可计算

★ 图灵可判定 = 递归 =
可解 = 能行 = 可判定 =
可计算

★ 下一章证明：
图灵可识别 \neq 图灵可判定



图灵机判定语言的例子

例4.4 $A = \{0^{2^n} \mid n \geq 0\}$

★ M_2 = “对输入串 w :

- 1) 从左向右扫描带子，隔一个消去一个0.
- 2) 若第一步之后，带子上只剩下1个0，则接受.
- 3) 若第一步之后，带子上剩下1个以上奇数个0，则拒绝.
- 4) 让读写头返回到带子最左端.
- 5) 转到第一步.”

例4.4 $A = \{0^{2^n} \mid n \geq 0\}$

★ $M_2 = (Q, \Sigma, \Gamma, \delta, q_1, q_{acc}, q_{rej})$

★ $Q = \{q_1, q_2, q_3, q_4, q_5, q_{acc}, q_{rej}\}$

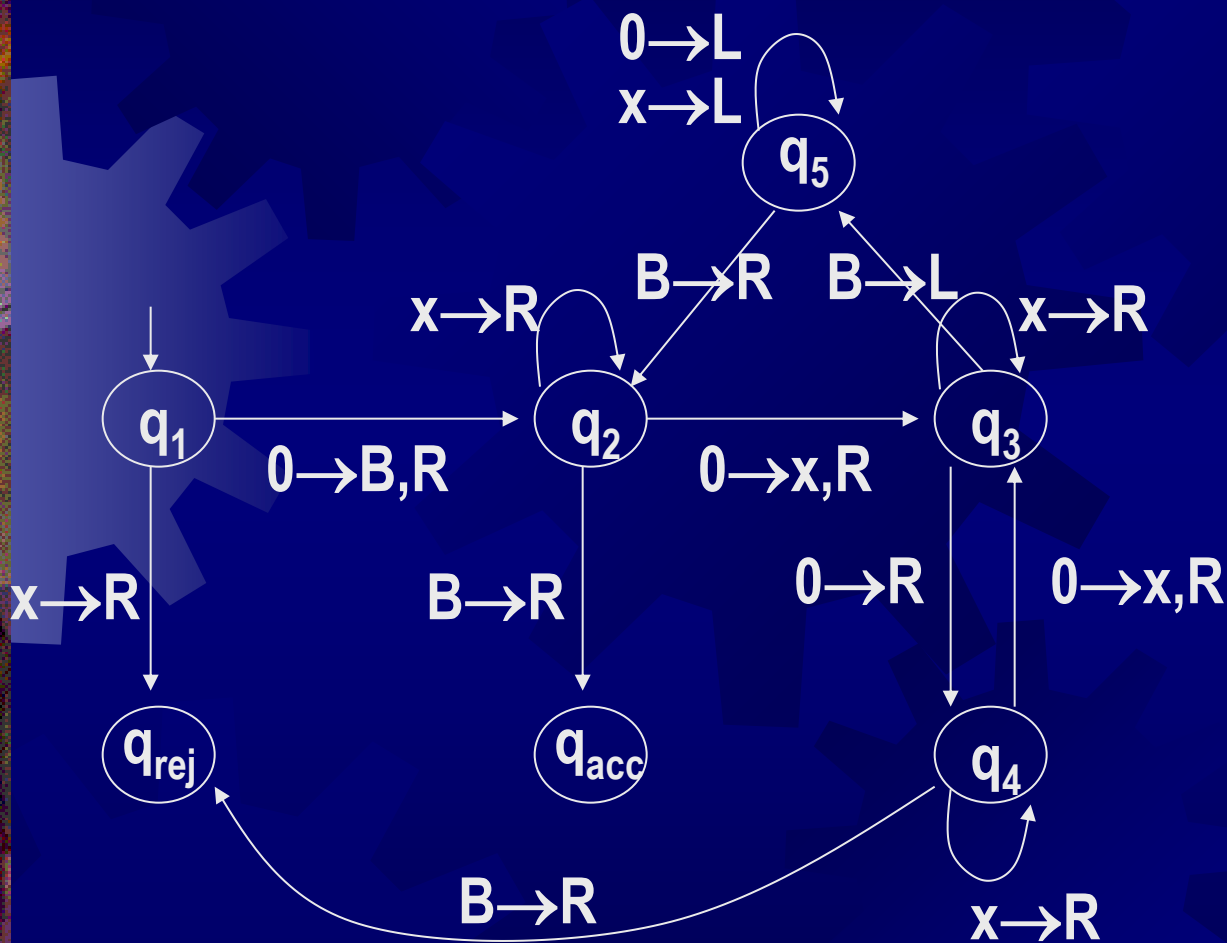
★ $\Sigma = \{0\}$

★ $\Gamma = \{0, X, B\}$

★ δ 如下页图示

★ 用空白符**B**作为左端点定界符

例4.4 M_2 的状态转移图



例4.4 M_2 在0000上的计算

$q_10000, Bq_2000, BXq_300, BX0q_40,$
 $BX0Xq_3B, BX0q_5XB, BXq_50XB,$
 $Bq_5X0XB, q_5BX0XB, Bq_2X0XB,$
 $BXq_20XB, BXXq_3XB, BXXXq_3B,$
 $BXXq_5XB, BXq_5XXB, Bq_5XXXB,$
 $q_5BXXXB, Bq_2XXXB, BXq_2XXB,$
 $BXXq_2XB, BXXXq_2B, BXXXBq_{acc}$

例4.5 $B = \{ w\#w \mid w \in \{0,1\}^* \}$

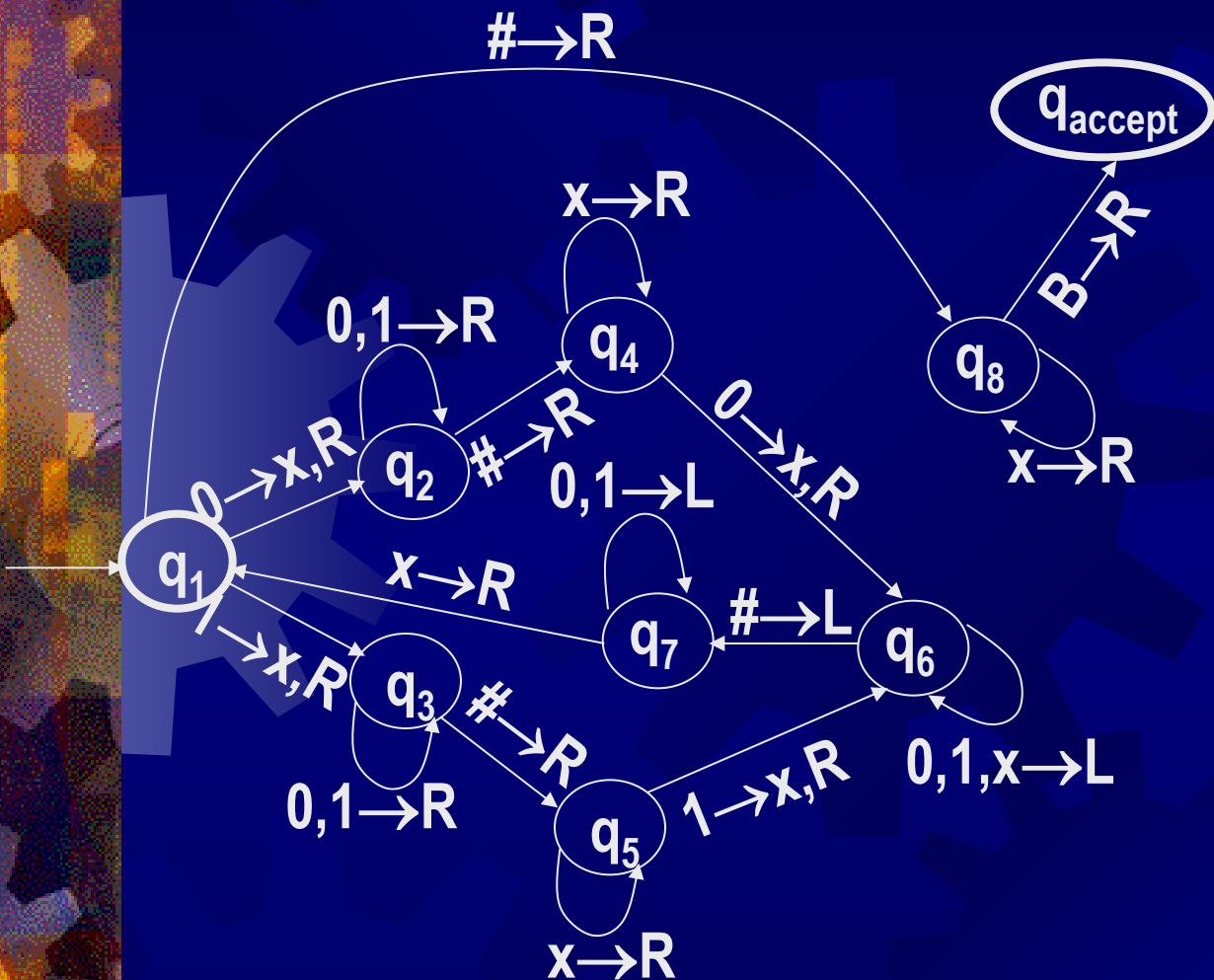
★ $M_1 =$ “对于输入字符串 w :

- 1) 扫描输入, 确认只含一个#.
否则拒绝.
- 2) 在#两边对应位置来回移动,
检查是否含相同符号.
若不是, 则拒绝.
消去已检查过符号.
- 3) 当消去#左边所有符号时,
检查#右边是否还有符号,
若是, 则拒绝. 否则接受.”

例4.5

- ★ $M_1 = (Q, \Sigma, \Gamma, \delta, q_1, q_{acc}, q_{rej})$
 - ★ $Q = \{q_1, q_2, \dots, q_8, q_{acc}, q_{rej}\}$
 - ★ $\Sigma = \{0, 1, \#\}$
 - ★ $\Gamma = \{0, 1, \#, X, B\}$
 - ★ δ 如下页图示
- ★ 省略到拒绝状态的转移

例4.5 M_1 的状态转移图



例4.6 $C = \{ a^i b^j c^k \mid i \times j = k \}$

★ M_3 = “对输入串 w :

- 1) 从左向右扫描输入, 确认输入具有形式 $a^* b^* c^*$, 否则拒绝.
- 2) 让读写头回到带子左端.
- 3) 消去1个 a , 向右扫描直到 b 出现. 在 b 和 c 之间来回移动, 成对消去 b 和 c , 直到消去所有 b .
- 4) 若还有 a 未消去, 则恢复所有已消去的 b , 重复第三步. 若所有 a 已消去, 则检查是否消去所有 c , 若是则接受, 否则拒绝.”

例4.6发现带左端的技巧

★ 如何发现带左端

- ★ 当前位置上写下某个特殊符号，在控制中记住所取代的符号，让带头向左移动
- ★ 若带头还在特殊符号上，则已经到带子左端
- ★ 否则，恢复右边的符号，再向左移

例4.7 $E = \{ \#x_1\#x_2\#\dots\#x_k \mid x_i \in \{0,1\}^*, \forall i \neq j, x_i \neq x_j \}$

★ $M_4 =$ “对输入 w :

- 1) 在最左端带符号顶上做记号.
若此符号是空白符, 则接受,
若此符号是#, 则进行下一步. 否则拒绝.
- 2) 向右扫描下一个#, 在其顶上做第二个记号.
若在遇到空白符之前没有遇到#,
则只有 x_1 , 因此接受.
- 3) 来回移动比较做记号#右边两个串,
若相等, 则拒绝.
- 4) 将右边#上记号向右移到下一个#上.
若在遇到空白符之前没有遇到#,
则将左边#上记号向右移到下一个#上,
并将右边记号移到后面#上.
若此时右边仍然找不到#, 则接受.
- 5) 转到 第三步.”

例4.7(元素区分问题)

- ★ 把每个 x_i 与它右侧的所有 x_j 相比
- ★ 在#上做记号, 带字母表包含 $\{\#, \dot{\#}\}$
- ★ 上述语言都是可判定的
(处处停机)

作业

★ 3.1 3.2