

## 计算理论导引第二周作业

**Question 1.** 设  $T(n), T'(n)$  为时间可构造. 证明  $T(n) + T'(n)$ 、 $T(n) \cdot T'(n)$ 、 $T(n)^{T'(n)}$  均为时间可构造. 能证明  $T(n)/T'(n)$  是时间可构造吗?  $\log T(n)$  呢?

设  $M_T$  和  $M_{T'}$  分别是计算  $T(n)$  和  $T'(n)$  的图灵机.

- $T(n) + T'(n)$ : 考察这样的  $M_+$ , 它先和  $M_T$  执行一样的运算操作, 在执行完成后, 它不清空带子, 而是继续进行和  $M_{T'}$  一样的操作, 最后把他们的计算结果加起来, 这个二进制加法操作的开销是  $O(\log(n))$  的. 所以总运行时间是  $O(T(n) + T'(n))$  步. 因此,  $T(n) + T'(n)$  是时间可构造的.
- $T(n) \cdot T'(n)$ : 考察这样的  $M_\cdot$ , 它进行如下操作: 先和  $M_T$  执行一样的运算操作, 待完成后, 再一个独立的带子上写下一个 1, 这个循环持续  $T'(n)$  遍. 由于  $T'(n)$  时间可构造, 这个操作是可以被图灵机实现的, 所以  $T(n) \cdot T'(n)$  是时间可构造的.
- $T(n)^{T'(n)}$ : 类似上一个, 构建一个循环.  
先计算  $T(n)$  (需要  $T(n)$  步), 再计算  $T'(n)$  (需要  $T'(n)$  步); 因为根据定义, 我们需要最后  $T(n)^{T'(n)}$  的二进制表示, 所以需要执行一个  $T'(n)$  次的循环, 在循环体中, 执行一个  $T(n)$  次的乘法操作. 乘法操作参考 (2), 是时间可构造的.
- $T(n)/T'(n)$ : 不行. 若  $T(n) = 2^{2^n}$ ,  $T'(n) = 2^n$ , 需要用  $M_T$ , 先算出  $T(n)$  的值是多少, 但是  $T(n) \gg T(n)/T'(n)$ , 不能被写成大  $O$  的形式.
- $\log T(n)$ : 同理不行.

**Question 2.** 在通用图灵机的证明中, 我们让  $|R_i| = 2 \cdot 2^{i^2}$ . 证明在哪一步会出问题? 如果没有问题的话, 我们会得到一个更高效的通用图灵机!

由于更改, 我们必须在原来的证明中把冗余信息的存放方式修正为: 要么全空, 要么存  $\log(\frac{R_i}{2})/2$  (替代半满), 要么存  $\log(\frac{R_i}{2})$  (替代全满).

问题出在最后的移动冗余信息中. 考察一个简单的图灵机  $M$ , 它只是简单地将读写头向右移动  $T(n)$  个单元格, 每步都在纸带上写入一个符号. 令  $S(i)$  为从中心点 0 到段  $R_{i-1}$  末尾的总容量.  $S(i) = \sum_{j=0}^{i-1} |R_j| = \sum_{j=0}^{i-1} 2 \cdot 2^{j^2}$ .

为了让  $M$  的读写头移动到第  $S(k) + 1$  个单元格,  $U$  必须已经执行了所有  $i = 1, 2, \dots, k$  的重组操作, 把  $R_i$  的冗余信息搬到前面  $1 \sim i - 1$  个格子内. 则:

$$T(n) \geq \sum_{i=1}^k C_i = \sum_{i=1}^k O(2^{i^2}) = O(2^{k^2})$$

取  $k = \sqrt{\log_2 T(n)} + 1$ , 代回  $T_U(n)$ :

$$(1) \quad T'(n) = O(2^{\log_2 T(n) + 2\sqrt{\log_2 T(n)} + 1}) = O(2^{\log_2 T(n)} \cdot 2^{2\sqrt{\log_2 T(n)}} \cdot 2^1) = O(T(n) \cdot 2^{2\sqrt{\log_2 T(n)}})$$

他的效率很低下.

**Question 3.** 说明: 若忽略不终止性, 在第 28 页上定义的非确定的“通用”图灵机是正确的.

( $\Rightarrow$ ). 根据非确定性计算的定义, 如果  $N_\alpha$  接受输入  $x$ , 那么必然存在至少一条从初始格局到接受格局的合法计算路径. 设这条接受路径为格局序列  $C = (C_0, C_1, \dots, C_m)$ , 其对应的非确定性选择序列为  $D = (\delta_1, \dots, \delta_m)$ .

$\forall$  本身是一台非确定图灵机, 它能够探索所有可能的计算分支. 因此, 必然存在一  $\forall$  的计算路径, 在该路径上, 它所猜测的格局序列恰好是  $C$ , 猜测的选择序列恰好是  $D$ . 此时被猜测的序列本身就是一条合法的接受路径, 所有验证步骤都会成功,  $\forall$  将停机并成功模拟了运算.

( $\Leftarrow$ ). 如果  $\mathbb{V}$  接受输入  $\langle \alpha, x \rangle$ , 那么必然存在至少一条  $\mathbb{V}$  的计算路径使其进入接受状态. 在这条使  $\mathbb{V}$  接受的路径上, 它必须成功地完成“猜测”和“验证”两个阶段.  $\mathbb{V}$  所猜测的格局序列  $\mathcal{C} = (C_0, C_1, \dots, C_m)$  和选择序列  $\mathcal{D} = (\delta_1, \dots, \delta_m)$  必须满足以下所有条件:

- $C_0$  被验证为是  $\mathbb{N}_\alpha$  在输入  $x$  上的初始格局.
- $C_m$  被验证为是一个接受格局.
- 对所有  $i \in \{0, \dots, m-1\}$ , 转移  $C_i \rightarrow C_{i+1}$  被验证为是  $\mathbb{N}_\alpha$  在选择  $\delta_{i+1}$  下的一步合法操作.

根据定义,  $\mathbb{N}_\alpha$  接受  $x$ .

**Question 4. 证明:** 设图灵机  $\mathbb{M}$  在  $T(n) = \omega(n)$  步内判定  $L$ . 对任意  $\varepsilon > 0$ , 有图灵机  $\mathbb{M}', \mathbb{M}'$  能在  $\varepsilon T(n)$  步内判定  $L$ .

根据课堂上讲过的 ( $T(n) = \omega(n)$  不一定成立的) 线性加速定理, 我们知道, 有图灵机  $\mathbb{M}'$ , 它能在  $F(n) \triangleq \varepsilon_1 T(n) + n + 2$  步内判定  $L$ , 其中  $\varepsilon_1 > 0$ .

由于  $T(n) = \omega(n)$ , 知  $\forall \delta > 0, \exists N \in \mathbb{N}, \forall n > N$ ,

$$0 < \frac{n+2}{T(n)} < \delta \implies F(n) < (\varepsilon_1 + \delta)T(n).$$

我们取  $\varepsilon_1 = \delta = \frac{\varepsilon}{2}$  即可.

使用了这个模版:<https://github.com/simurgh9/hw>