

可计算性理论

杨睿之

复旦大学哲学学院

2021 年春季

课程信息

- 时间地点:

- 周二 18:30 - 21:05, HGX401

- 网站:

<https://aplacenearby.ggr.fun/recursion2021>

- 教材: André Nies, *Computability and Randomness*,
Oxford University Press, 2009

《递归论：算法与随机性基础》，复旦大学出版社

- 参考: 《数理逻辑：证明及其限度》(第二版)

课程信息

考核

- 论文：带回去做的一组证明题

课程团队

- 杨睿之

邮箱：yangruizhi@fudan.edu.cn

- 刘桢

邮箱：19210160030@fudan.edu.cn

研究对象

这门学科所探讨的**可计算**、**随机**是关于什么的概念？

- 问题？

例： π （十进制表示）的第 x 位是几？

- 函数？

例：哈希函数（Hash function）？ $y = x^2$ ？

- 过程？

例：抛硬币？伯努利过程（Bernoulli process）

口号

万物皆可（用自然数）编码

例

- 图片 (.bmp)
- 声音 (.wav)
- 文章、程序.....
- 虚幻引擎、3D 打印.....

什么是编码？

- 编码 (encoding) 是一个以自然数为值域的函数, 解码 (decoding) 是以自然数为定义域的函数。
- 编码和解码应该是能行的 (effective)
- 成功的编码不仅仅是数学问题

用自然数编码数学对象

记法

- 自然数集: $\mathbb{N} = \omega$
- 卡氏积: 例: $\mathbb{N} \times \mathbb{N} = \mathbb{N}^2 = \{(n, m) \mid n, m \in \mathbb{N}\}$
- 函数/序列: $A^B = \{f \mid f: A \rightarrow B\}$
例: 01 序列组成的集合: 2^ω
- 有穷序列: 例: $\omega^{<\omega} = \{s \mid \text{存在 } n \in \omega \text{ 使得 } s: n \rightarrow \omega\}$

用自然数编码数学对象

编码自然数有序对

事实

存在双射 $f : \mathbb{N}^2 \rightarrow \mathbb{N}$, f 和它的逆都是能行可计算的

证明.

考虑

$$f(m, n) = n + \frac{(m+n)(m+n+1)}{2}$$

任给自然数 k , 如何算出 (m, n) 使得 $f(m, n) = k$?

用自然数编码数学对象

编码有穷自然数序列

事实

存在双射 $f: \omega^{<\omega} \rightarrow \omega$, f 和 f^{-1} 都是能行可计算的

证明.

考虑:

$$f(\langle \rangle) = 0$$

$$f(\langle x_0, \dots, x_n \rangle) = p_0^{x_0+1} \cdot \dots \cdot p_n^{x_n+1} - 1$$

用自然数编码数学对象

编码遗传有穷集合

回忆: $V_\omega = \bigcup_n V_n = \{x \mid x \text{ 是遗传有穷的集合} \}$

定义 (Ackermann 翻译)

递归定义 $\Gamma: V_\omega \rightarrow \mathbb{N}$, 使得

$$\Gamma(x) = \sum_{y \in x} 2^{\Gamma(y)}$$

定义自然数上关系 nEm , 当且仅当 $2 \nmid \lfloor \frac{m}{2^n} \rfloor$

用自然数编码数学对象

- 编码整数、有理数
- 编码形式语言——哥德尔编码
- 编码程序、计算过程.....

通过编码，我们可以将许多数学问题（系列）或函数运算转化为关于自然数集的问题

研究对象

这门学科所探讨的可计算、随机是关于什么的概念？

- 自然数集
- 自然数集上的函数
- 无穷 01 序列
特征函数
- $[0, 1]$ 间的实数（二进制表示）

二进制表示实数

考虑函数

$$F : \{Z \in P(\mathbb{N}) \mid Z \text{ 是余无穷的}\} \rightarrow [0, 1)_{\mathbb{R}}$$

使得 $F(Z) = 0.Z = \sum_{i \in Z} 2^{-(i+1)}$

例: $F(\{1, 2, 3\}) = 0.0111$ (二进制) = $7/16$

注意: 令 $Y = \{1, 2\} \cup \{n \in \mathbb{N} \mid n \geq 4\}$, 则

$$\sum_{i \in Y} 2^{-(i+1)} = 0.110111 \cdots = 7/16$$

二进有理数

定义

定义 二进有理数 (dyadic rational) 集为

$$\mathbb{Q}_2 = \{z2^{-n} \mid z \in \mathbb{Z}, n \in \mathbb{N}\}$$

注意:

- 有理数不一定是二进有理数, 例如: $1/3$
- 二进有理数在实数中稠密

二叉树

记法

- 我们用 $2^{<\omega}$ 、 $\{0,1\}^*$ 表示有穷 01 序列（或 01 字符串）组成的集合。一般用 $\sigma, \tau, \rho \dots$ 表示 01 字符串
- 我们用 $\sigma \leq \tau / \sigma < \tau$ 表示 σ 是 τ 的前段/真前段

$(2^{<\omega}, \leq)$ 构成了一颗完全二叉树

二叉树

记法

- 用 $\sigma\tau$ / σa 表示两个字符串 / 一个字符串和一个字符的首尾连接
- 用 $\sigma | \tau$ 表示两个字符串是不相容的
- 用 $\sigma <_L \tau$ 表示“ σ 在 τ 的左边”。 $<_L$ 是字典序的一个子序
- 用 $|\sigma|$ 表示字符串 σ 的长度
- 用 \emptyset 或 $\langle \rangle$ 表示空字符串

二叉树

考虑函数 $f : 2^{<\omega} \rightarrow \mathbb{N}$, 对 01 字符串 σ , 令 $f(\sigma) = n$, 其中 $n + 1$ 是 $\sigma 1$ (从左到右) 二进制表示自然数

例

- $f(\emptyset) = 0$
- $f(\langle 0 \rangle) = 1$
- $f(1100) = (2^0 + 2^1 + 2^4) - 1 = 17$

显然, f 是双射, f 和 f^{-1} 都是能行的

二叉树

定义

- 我们称 $T \subset 2^{<\omega}$ 是一颗 **二叉树** (binary tree), 当且仅当 T 在取前段下封闭 (若 $\sigma \in T$ 且 $\tau \leq \sigma$, 则 $\tau \in T$)。
- 称 $Z \subset \mathbb{N}$ 是 T 上的一条 **路径** (path), 当且仅当对任意 $n \in \mathbb{N}$, $Z \upharpoonright n \in T$ 。

因此, 二叉树和二叉树中的路径都可以被编码为自然数的子集

可计算性

对可计算性 (Computability) 的等价刻画

- 递归函数 (哥德尔, 1933)
- λ -演算 (丘奇, 1936)
- 图灵机可计算 (图灵, 1936)
- C, Python, Ethereum Smart Contracts, ...

丘奇-图灵论题 (Church-Turing Thesis): 一个函数

$f : \mathbb{N}^n \rightarrow \mathbb{N}$ 是能行可计算的, 当且仅当它是图灵机可计算的

可计算性

定义

假设 ψ 是自然数上的 k 元部分函数 ($\text{dom } \psi \subset \mathbb{N}^k$ 且 $\text{ran } \psi \subset \mathbb{N}$), 我们称 ψ 是部分可计算的 (partial computable), 当且仅当存在图灵机 (或计算机程序) P 满足: 对任意 $x_1, \dots, x_k \in \mathbb{N}$, 若 $(x_1, \dots, x_k) \in \text{dom } \psi$ (记作 $\psi(x_1, \dots, x_k) \downarrow$), 则有 $y \in \mathbb{N}$, $\psi(x_1, \dots, x_k) = y$ 且程序 P 在输入 (x_1, \dots, x_k) 后能停机并输出 y ; 若 $(x_1, \dots, x_k) \notin \text{dom } \psi$ (记作 $\psi(x_1, \dots, x_k) \uparrow$), 则程序 P 在输入为 (x_1, \dots, x_k) 后不停机或没有输出。

可计算性

定义

- 此时, 我们称 P 计算 ψ
- 如果 $\text{dom } \psi = \mathbb{N}^k$, 我们称 ψ 是可计算的 (computable)
- 我们称集合 $A \subset \mathbb{N}$ 是可计算的, 当且仅当它的特征函数是可计算的

枚举部分可计算函数

显然，我们可以能行地枚举给定计算机语言所写的所有程序： $\{P_e\}_{e \in \mathbb{N}}$ 。对每个 e ，存在唯一的部分函数 ψ ，使得 P_e 计算 ψ ，记作 Φ_e 。此时，称 e 是 ψ 的索引 (index)。在以后的讨论中，我们固定一个这样的枚举。

注意：部分函数的元数在这里并不重要 (为什么?)。为方便，可以对每个 $k \in \mathbb{N}$ 固定一个输入必须是 n 元组的程序的枚举 $\{P_e^k\}_{e \in \mathbb{N}}$ ，由此得到一个 k 元部分可计算函数的枚举 $\{\Phi_e^k\}_{e \in \mathbb{N}}$ 。

不可计算的集合

定义 (停机问题)

定义 停机问题 (halting problem) 为集合

$$\emptyset' = \{e \in \mathbb{N} \mid \Phi_e(e) \downarrow\}$$

\emptyset' 是不可计算的 [否则 $\mathbb{N} \setminus \emptyset'$ 也是可计算的.....]

相对可计算与图灵归约

定义

设想带信息源 (oracle) 的图灵机 / 外接 (无穷大) 硬盘的计算机。我们称集合 X 可图灵归约到 Y (记作 $X \leq_T Y$)，当且仅当存在带信息源的图灵机 P_e (或计算机程序运行在可读取外部存储的计算机上) 以 Y 为信息源计算 X ，即

$$\Phi_e^Y = X$$

相对可计算与图灵归约

例

- $\emptyset' \leq_T \{(e, x) \mid \Phi_e(x) \downarrow\}$
- 令 $\emptyset'' = \{e \in \mathbb{N} \mid \Phi_e^{\emptyset'}(e) \downarrow\}$, 则

$$\emptyset' \leq_T \emptyset''$$

相对可计算与图灵归约

定义

- 定义集合间等价关系 $A \equiv_T B$, 当且仅当 $A \leq_T B$ 且 $B \leq_T A$
- 对集合 A , 定义 $\text{deg}_T(A) = \{B \subset \mathbb{N} \mid B \equiv_T A\}$
- 定义 $\mathcal{D}_T = \{\text{deg}_T(A) \mid A \subset \mathbb{N}\}$. 定义 \mathcal{D}_T 上的关系 \leq_T 自然继承自 $P(\mathbb{N})$ 上的 \leq_T

我们称 (\mathcal{D}_T, \leq_T) 是一个 **度结构** (degree structure), 对各种度结构的研究是递归论的经典研究方向

可计算性、定义复杂度与随机性

下期预告

- 递归论基本概念
- $s-m-n$ 定理、递归定理等