构造主义与可计算性的桥梁

又名：如何编译你的证明

Trebor

构造主义与可计算性

- 构造性的证明 = 可计算?
  - 存在可计算函数 $\mathbb{N} \rightharpoonup (\mathbb{N} \rightharpoonup \mathbb{N})$ 使值域恰好是全体（一元）可计算部分函数.
  - 构造主义中可以证明不存在满射 $\mathbb{N} \to (\mathbb{N} \to \mathbb{N})$ （Cantor 对角化）
- 脑筋急转弯：语言 $L_p = \{11\ldots1 \mid \text{此字符串在 } p \text{ 的十进制展开中出现}\}$
  - 对于任何实数 p，这都是可判定的，甚至是正则的！
  - 同理：对于每个程序，它是否停机都是可判定的.
  - 需要描述“问题 $P(x, y)$ 可计算，但是对于每个 y，计算的程序不一样”.
- 高阶函数何时可计算? $(\mathbb{N} \to \mathbb{N}) \to \mathbb{N}$
- 不是自然数的对象如何定义可计算性?

编码问题

- **停机问题**：输入一个 λ–表达式，输出它是否停机
  - 只定义了关于自然数的函数的可计算性的概念
  - λ–表达式不是自然数 $\Longrightarrow$ 需要选择一个编码
    - 从表达式的集合 Λ 到 $\mathbb{N}$ 的任意一个单射？双射？
    - 考虑某个单射，将停机的表达式依次编码到偶数，不停机依次编码到奇数
      - $0 \leftrightarrow \lambda x\,.\,x, \quad 1 \leftrightarrow (\lambda x\,.\,xx)(\lambda x\,.\,xx), \quad 2 \leftrightarrow \lambda xy\,.\,x, \cdots$
    - 此时停机问题可以判定，因为自然数的奇偶性可以判定！
  - 在某某编码下停机问题不可判定？有点烦

有效意象

- （初等）意象：长得很像集合范畴的范畴
  - 这意味着可以用类似集合论（类型论）的语言
  - 高阶构造主义逻辑 / 依值类型论 + 外延相等 + 命题宇宙
    - 可以根据所研究的意象添加额外的条件
  - 给定意象 $\mathscr{E}$，内语言的叙述 $\Longrightarrow$ 关于 $\mathscr{E}$ 的叙述
- 有效意象 Eff：关于可有效计算的事物的意象
  - 高阶函数：自动解决
  - 编码问题：泛性质在同构意义下唯一确定
  - 各种杂症：后面会讲（如果有时间）

部分组合代数

- 我们需要从某种计算模型出发
- 定义**部分组合代数**（**P**artial **C**ombinatory **A**lgebra）为一个集合 A
  - 附带一个部分二元运算 $A \times A \rightharpoonup A$ （写成左结合乘法）
  - 满足存在 $s, k \in A$, $kxy = x$, $sxyz = (xz)(yz)$ 成立.
- 直观：fx 描述了程序 f 输入数据 x 的运算结果（可能不停机/出错）
- 喜欢范畴论的同学：可以等价地定义一种范畴，叫做 Turing 范畴.

- 这是有偏定义，无偏定义要如何写？

部分组合代数

- 定义**部分组合代数**（**P**artial **C**ombinatory **A**lgebra）为一个集合 A
  - 附带一个部分二元运算 $A \times A \rightharpoonup A$
  - 满足存在 $s, k \in A$，$kxy = x$，$sxyz = (xz)(yz)$ 成立.

- 选择你最喜爱的编程语言
  - fx 表示将 f 输入 x 运算的结果，如果类型错误/抛出异常/不停机，则不定义
- λ–演算、组合子演算、递归函数构成部分组合代数
  - 一种思路：考虑所有表达式的集合 $\{\lambda x\,.\,x, \lambda xy\,.\,x, \ldots\}$，商去 β, η 等价
    - 这样定义出的二元运算是全函数
  - 另一种思路：考虑所有正规形式的集合，不需要商去等价
    - 好处：因为没有商，判断相等是简单的
    - 坏处：二元运算是部分函数，可能不停机
- 简单类型 λ–演算（如果类型不匹配则不定义），称作 Kreisel PCA

部分组合代数

- 定义**部分组合代数**（**P**artial **C**ombinatory **A**lgebra）为一个集合 A
  - 附带一个部分二元运算 $A \times A \rightharpoonup A$
  - 满足存在 $s, k \in A$，$kxy = x$，$sxyz = (xz)(yz)$ 成立.

- 这样够了吗?
- 有没有元素 i 满足 $ix = x$?
  - 有：$skkx = kx(kx) = x$.
- 有没有办法将数据组成有序对?
  - 即需要程序 $\mathrm{pair}, \mathrm{fst}, \mathrm{snd}$，满足 $\mathrm{fst}(\mathrm{pair}\, x\, y) = x$ etc.
  - 有：$\mathrm{pair} = s(s(ks)(s(kk)(s(ks)(s(k(s(skk)))(s(kk)(skk))))))(k(s(kk)(skk)))$
  - 略，如果想知道这怎么想出来的问我或者其他讲师（其实是我在这抄的）
  - 把配对写成 $\langle x, y\rangle$，只要知道有就行了.
- 事实上 s, k 已经足够完成任何运算.（当然，可以替换成更高效的办法）

Eff 青春版：PER

- 定义**部分组合代数**（**P**artial **C**ombinatory **A**lgebra）为一个集合 A
  - 附带一个部分二元运算 $A \times A \rightharpoonup A$
  - 满足存在 $s, k \in A$，$kxy = x$，$sxyz = (xz)(yz)$ 成立.

- 我们想要编码数学对象（$\mathbb{Q}, \mathbb{R}, \cdots$）
  - 例如用两个整数表示一个有理数的分数形式
  - 一份数据可能不编码合法的对象（除以零）
  - 多份数据可能表示相同的对象（约分）
- 在固定的部分组合代数 A 上取一个子集 $R \subseteq A$，上面定义等价关系
  - 也可以表述成**部分等价关系**：等价关系去掉自反性 $x \sim x$ 要求.
    - $x \sim x$ 成立时表示这个元素编码合法的对象，$x \sim y$ 表示编码同一个对象.
  - 此时 $A/\sim$ 就表示它编码的数学对象的集合
- 考虑所有 A 上的部分等价关系构成的范畴 $\mathrm{PER}(A)$（以后省略 A）
- 态射是什么?

Eff 青春版：PER

- 定义**部分组合代数**（**P**artial **C**ombinatory **A**lgebra）为一个集合 A
  - 附带一个部分二元运算 $A \times A \rightharpoonup A$
  - 满足存在 $s, k \in A$，$kxy = x$，$sxyz = (xz)(yz)$ 成立.

- 如果有两个等价关系 $\sim$, $\approx$
  - 表示用 A 编码两种不同的数学对象的方法
- 有一个映射 $\varphi : (A/\sim) \to (A/\approx)$ 表示编码的数学对象之间的映射
- **可计算性**：存在程序 $f \in A$，使得 $f(\square) : A \rightharpoonup A$ **实现**了 φ.
  - $\sim$ 没有的地方不用管
  - 如果 $x \sim x$，那么 $f(x)$ 有定义，并且在 $\varphi[x]$ 的等价类中.
- 恒同态射 $\mathrm{id} = skk$
- 态射复合 $f \circ g = s(kf)g$
- 构成范畴 PER，又叫 Mod（Modest Set）

Eff 青春版：Asm 与 PER

- PER 升级版：允许同一个程序编码多个不同的元素
- **汇编**是一个集合 X 上配备一个关系 $\Vdash_X$（$a \Vdash_X x$ 表示程序 a 编码了元素 x）
  - 每个元素都至少要有一个程序编码它
  - 但是不要求一个程序编码唯一的元素
- 态射类似定义，即函数 $\varphi : X \to Y$，使得存在程序 f 满足
  - $a \Vdash_X x \implies fa \Vdash_Y \varphi(x).$
- 得到范畴 Asm，PER $\hookrightarrow$ Asm

Eff 青春版：Asm

- Asm 支持了很多集合（类型）的操作
- **乘积**:
  - 给定 $(X, \Vdash_X), (Y, \Vdash_Y)$，其乘积为 $(X \times Y, \Vdash_{X \times Y})$
  - 其中 $\langle a, b \rangle \Vdash_{X \times Y} (x, y) \iff (a \Vdash_X x) \wedge (b \Vdash_Y y)$.
- 集合的 $\{x \mid F(x) = G(x)\}$，称作等化集（范畴论的**等化子**）：
  - 直接把不符合条件的元素踢出去即可
- **函数**:
  - 对应的集合是可计算函数的集合
  - 其中 $f \Vdash_{X \to Y} \varphi$ 当且仅当 f 实现了 φ.

- 这样足够证明 Asm 局部积闭：拉回可以用乘积与等化子构造；依值函数对象可以通过函数与等化子构造<u>截面对象</u>.

Eff 青春版：Asm

- 固定一下 PCA，这里定成一种弱类型的编程语言（伪代码）
- $(\mathbb{N}, \Vdash_{\mathbb{N}})$ 表示自然数，其中语言中的自然数实现了数学上的自然数
  - 整体写作 $\mathbb{N}$，是 Asm 中的**自然数对象**（满足泛性质）
  - $\mathbb{N} \to \mathbb{N}$ 的态射正好与可计算函数一一对应
- $\nabla\mathbb{N}$ 集合仍然是自然数，但是任何程序都实现了任何自然数
  - 表示“无计算内涵”的自然数
    - 用于描述“问题 $P(x, y)$ 可计算，但是对于每个 y，计算的程序不一样”
- 类似地对任何集合都有 ∇X.
  - $\nabla X \to \nabla Y$ 的态射正好与纯集合的映射 $X \to Y$ 一一对应

Eff 青春版：Asm

- 三个二元集合
  - $\mathbf{2}$ 表示 Boole 值
  - $\nabla\mathbf{2}$ 无计算内涵的二元集
  - $\mathbb{S}$ 中 $a \Vdash_{\mathbb{S}} \mathrm{true}$ 当且仅当 a 停机（或者用别的编码停机问题的方式）
    - 名字：Sierpiński / **半可判定**（Semidecidable）
- 三种子集
  - 可判定子集：$X \to \mathbf{2}$
  - 半可判定子集：$X \to \mathbb{S}$
  - 任意子集：$X \to \nabla\mathbf{2}$

Eff 青春版：Asm

- Asm 还缺少的东西：在范畴内部讨论命题的能力
  - 范畴语言：缺少子对象分类器
  - 完全体：Eff **有效意象**
- 我们在 Asm 外部讨论命题（在 Eff 中可以等价地在内部表述）
  - $a \Vdash p \wedge q$ 当且仅当 $a = \langle b, c \rangle$，且 $b \Vdash p, c \Vdash q$.
  - $a \Vdash p \vee q$ 当且仅当 $b \Vdash p$ 且 $a = \langle \text{false}, b \rangle$ 或者 $b \Vdash q$ 且 $a = \langle \text{true}, b \rangle$.
  - $f \Vdash p \Rightarrow q$ 当且仅当对于任何 $a \Vdash p$，$fa \Vdash q$.
  - $a \Vdash x = y$ 当且仅当 $x = y$ 且 $a \Vdash_X x$.

Eff 青春版：Asm

- 我们在 Asm 外部讨论命题（在 Eff 中可以等价地在内部表述）
  - $a \Vdash p \wedge q$ 当且仅当 $a = \langle b, c \rangle$，且 $b \Vdash p, c \Vdash q$.
  - $a \Vdash p \vee q$ 当且仅当 $b \Vdash p$ 且 $a = \langle \mathrm{false}, b \rangle$ 或者 $b \Vdash q$ 且 $a = \langle \mathrm{true}, b \rangle$.
  - $f \Vdash p \Rightarrow q$ 当且仅当对于任何 $a \Vdash p$，$fa \Vdash q$.
  - $a \Vdash x = y$ 当且仅当 $x = y$ 且 $a \Vdash_X x$.
  - $f \Vdash \forall (x : X) \,.\, p(x)$ 当且仅当对于任何 $a \Vdash_X x$，$fa \Vdash p(x)$.
  - 存在：留作练习
- 在 Eff 中这些也都由泛性质唯一确定，Asm 将就一下
- 构成一种构造主义逻辑

实战：Rice 定理

- **定义**：某个集合 X 满足任何映射 $X \to X$ 都有不动点，就称其满足**不动点性质**.
- 经典数学中有不动点性质的集合只有 $\{ \star \}$
- **定理**（Rice）：如果 X 有不动点性质，那么任何映射 $X \to \mathbf{2}$ 都是常函数.
- 证明. 对于任何映射 $f : X \to \mathbf{2}$ 与 $x, y \in X$，我们需要证明 $f(x) = f(y)$.
  - 定义函数 $g(z) =$ if $f(z) = f(y)$ then x else y.
  - 它有不动点 $u = g(u)$，分两种情况：
    - $f(u) = f(y)$，那么 $u = g(u) = x$，因此 $f(x) = f(y)$ 成立；
    - $f(u) \neq f(y)$，那么 $u = g(u) = y$，$f(y) \neq f(y)$ 矛盾，证毕.
- Asm 的哪些集合有不动点性质呢?

实战：Rice 定理

- **定义**：某个集合 X 满足任何映射 $X \to X$ 都有不动点，就称其满足**不动点性质**.
- **定理**（Lawvere）：如果有满射 $e : X \twoheadrightarrow (X \to Y)$，那么 Y 有不动点性质.
- 其实就是 **Cantor 对角化**的核心部分，Cantor 定理说满射 $X \twoheadrightarrow (X \to \mathbf{2})$ 不存在，即先用 Lawvere 定理后因为 $\mathbf{2}$ 不满足不动点性质得到矛盾.
- 证明. 对于任何映射 $f : Y \to Y$，考虑 $g(x) = f(e(x)(x))$，则 $g \in X \to Y$，因为 e 是满射，存在 x_0 使得 $e(x_0) = g$. 因此 $e(x_0)(x_0) = g(x_0) = f(e(x_0)(x_0))$，这就构造出了不动点.

实战：Rice 定理

- 已知：半可判定集的集合 $\mathbb{N} \to \mathbb{S}$ 是可枚举的.
  - 即：存在满射 $\mathbb{N} \twoheadrightarrow (\mathbb{N} \to \mathbb{S})$.
- 由 Lawvere 定理得到 $\mathbb{S}$ 有不动点性质
  - 同理，$(\mathbb{N} \to \mathbb{S}) \cong (\mathbb{N}^2 \to \mathbb{S}) \cong (\mathbb{N} \to (\mathbb{N} \to \mathbb{S}))$
  - 因此 $(\mathbb{N} \to \mathbb{S})$ 也有不动点性质.
- 由 Rice 定理得到推论：
  - 任何映射 $\mathbb{S} \to \mathbf{2}$ 都是常函数
    - 翻译到经典语言：停机问题不可判定
  - 任何映射 $(\mathbb{N} \to \mathbb{S}) \to \mathbf{2}$ 都是常函数（Rice 定理，经典皮肤）

豪华升级版：Eff

- PER $\hookrightarrow$ Asm $\hookrightarrow$ Eff.
  - 实际上是某种完备化
- 通俗易懂的构造介绍：Realizability Toposes. （topos 的复数到底是啥）
- 性质：
  - 选择公理、排中律不成立
  - 可数选择公理成立：若 $\forall(n:\mathbb{N}),\exists(y:Y),P(n,y)$，可以取出函数 $\mathbb{N}\to Y$.
  - Markov 原理成立（这个 Markov 和他爸同名，他爸提出了 Markov 链）
    - 如果一个算法不不停机，那么它停机
    - $\mathbb{S}\hookrightarrow\nabla\mathbf{2}$.

豪华升级版：Eff

- 相对可计算
  - 假如给你一个魔法机器可以解决某问题，那么你现在可计算的问题有哪些?
  - 例如：假如你的机器可以调用魔法解决停机问题……
  - 在 Eff 中对应某个满子范畴
    - 用到了 Lawvere–Tierney 拓扑
- 用作类型论的模型
  - 忘记花里胡哨的东西得到一些程序 $f \in A$
  - 即得到类型论的计算内涵（如立方汇编）
- 将数学理论放进去，自动得到程序：如 RZ.