构造演算（The Calculus of Constructions）

读书笔记

许博

1 λC 系统

λC 组合了第二章到第五章中介绍的系统，拥有四种可能的选择，即依赖于项/类型的项/类型。

λP 与 λC 只有一处不同，但足以扩展 λP 到 $\lambda\mathrm{C} = \lambda 2 + \lambda\underline{\omega} + \lambda\mathrm{P}$：

$$(form_{\lambda\mathrm{P}})\ \frac{\Gamma \vdash A : * \quad \Gamma, x : A \vdash B : s}{\Gamma \vdash \Pi x : A.B : s}$$

在这条规则中，关键点是 $A : *$，为了保证类型 $\Pi x : A.B$ 的成员（inhabitant）是项或者依赖于项的类型。但在舍弃了这个限制之后，我们就获得了我们想要的泛化：依赖于项/类型的项/类型。

看起来将 $A : *$ 替换为 $A : s$，其中 s 为 $*$ 或 $\square$，就足够了，但是规则中已经出现了 s，观察 $\lambda\underline{\omega}$ 的 $(form)$-规则：

$$(form_{\lambda\underline{\omega}})\ \frac{\Gamma \vdash A : s \quad \Gamma \vdash B : s}{\Gamma \vdash A \to B : s}$$

只能表示依赖于项的项和依赖于类型的类型，而不能相互交叉（cross-over）。

因此在 λC 的 $(from)$-规则中，使用了两个 s：s_1 和 s_2：

$$(form_{\lambda\mathrm{C}})\ \frac{\Gamma \vdash A : s_1 \quad \Gamma, x : A \vdash B : s_2}{\Gamma \vdash \Pi x : A.B : s_2}$$

$\Pi x : A.B$ 的类型继承自 B，也即依赖于项/类型（1）的项/类型（2）依然是项/类型（与 2 统一）。因此有一个有趣的事实是：假设 A 中不存在与

抽象的类型变量相同的自由类型变量，$* \to A$ 是一个类型，而 $A \to *$ 是一个种类 (kind)。

假设有一个函数 $\lambda x : A.b$，它的类型是 $\Pi x : A.B$，可以通过 $(abst)$-规则构建（$b : B$），通过 $(form_{\lambda\mathrm{C}})$-规则可知 A 的类型是 s_1，B 的类型是 s_2，则有以下可能：

$x : A : s_1$	$b : B : s_2$	(s_1, s_2)	$\lambda x : A \,.\, b$	from
$*$	$*$	$(*, *)$	term-depending-on-term	$\lambda\!\to$
$\square$	$*$	$(\square, *)$	term-depending-on-type	$\lambda 2$
$\square$	$\square$	$(\square, \square)$	type-depending-on-type	$\lambda\underline{\omega}$
$*$	$\square$	$(*, \square)$	type-depending-on-term	$\lambda\mathrm{P}$

2 λ-cube

对于 $\lambda\!\to$ 基础上的三个扩展，彼此之间相互独立，可以被看作是扩展时的三个相互垂直的方向，给出坐标轴的三维系统：

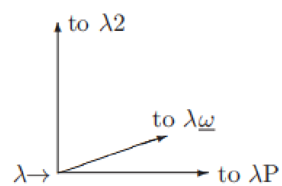

Figure 6.1 Directions of extending $\lambda\!\to$

三个扩展共同组成了 $\lambda\mathrm{C}$，而将 $\lambda\!\to$ 与这三种扩展中的两个相组合而成的系统分别叫做 $\lambda\omega$（$\lambda\!\to + \lambda 2 + \lambda\underline{\omega}$），$\lambda\mathrm{P}2$ 和 $\lambda\mathrm{P}\underline{\omega}$。

所有的八个系统可以在一个立方体中定位，也即所谓的 λ*−cube* 或者巴伦德雷格立方体（*Barendregt cube*）：

system:	combinations (s_1, s_2) allowed:			
$\lambda{\rightarrow}$	$(*,*)$			
$\lambda 2$	$(*,*)$	$(\square,*)$		
$\lambda\underline{\omega}$	$(*,*)$		$(\square,\square)$	
$\lambda\mathrm{P}$	$(*,*)$			$(*,\square)$
$\lambda\omega$	$(*,*)$	$(\square,*)$	$(\square,\square)$	
$\lambda\mathrm{P}2$	$(*,*)$	$(\square,*)$		$(*,\square)$
$\lambda\mathrm{P}\underline{\omega}$	$(*,*)$		$(\square,\square)$	$(*,\square)$
$\lambda\mathrm{P}\omega = \lambda\mathrm{C}$	$(*,*)$	$(\square,*)$	$(\square,\square)$	$(*,\square)$

Figure 6.2 The eight systems of the λ-cube

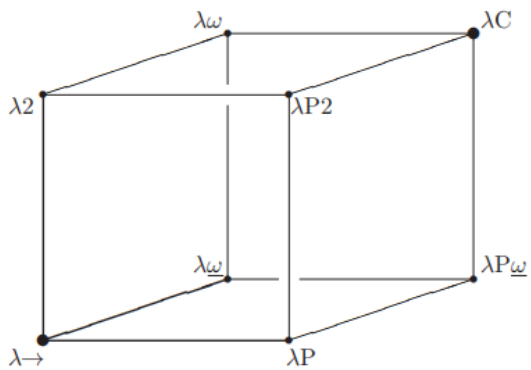

Figure 6.3 The λ-cube or Barendregt cube

这八个不同的系统可以通过推导规则的唯一个集合来描述：

$$(sort)\quad \emptyset \vdash * : \Box$$

$$(var)\ \frac{\Gamma \vdash A : s}{\Gamma,\ x : A \vdash x : A} \quad \text{if } x \notin \Gamma$$

$$(weak)\ \frac{\Gamma \vdash A : B \qquad \Gamma \vdash C : s}{\Gamma,\ x : C \vdash A : B} \quad \text{if } x \notin \Gamma$$

$$(form)\ \frac{\Gamma \vdash A : s_1 \qquad \Gamma,\ x : A \vdash B : s_2}{\Gamma \vdash \Pi x : A\,.\,B \ :\ s_2}$$

$$(appl)\ \frac{\Gamma \vdash M \ :\ \Pi x : A\,.\,B \qquad \Gamma \vdash N : A}{\Gamma \vdash MN \ :\ B[x := N]}$$

$$(abst)\ \frac{\Gamma,\ x : A \vdash M : B \qquad \Gamma \vdash \Pi x : A\,.\,B \ :\ s}{\Gamma \vdash \lambda x : A\,.\,M \ :\ \Pi x : A\,.\,B}$$

$$(conv)\ \frac{\Gamma \vdash A : B \qquad \Gamma \vdash B' : s}{\Gamma \vdash A : B'} \quad \text{if } B =_\beta B'$$

Figure 6.4 Derivation rules for the systems of the λ-cube

每一个系统都依赖于 (s_1, s_2) 的组合，根据表 Figure 6.2 可得。以 $\lambda\underline{\omega}$ 为例，只需要将 $A \to B$ 看作是 $\Pi x : A.B$ 的缩写，又因为 $(s_1, s_2) \in \{(*, *), (\Box, \Box)\}$，所以再令 $s_1 = s_2 = s$ 即可。

3 λC 的性质

之前描述的大部分性质，λC 依然保持，但措辞可能会有不同，对于某些直白的改变不再赘述。

定义 3.1（λC 的表达式，$\mathcal{E}$）

λC 的表达式的集合 $\mathcal{E} = V|\Box| * |(\mathcal{E}\mathcal{E})|(\lambda V : \mathcal{E}.\mathcal{E})|(\Pi V : \mathcal{E}.\mathcal{E})$

引理 3.2（自由变量引理）

如果 $\Gamma \vdash A : B$，则 $FV(A), FV(B) \subseteq dom(\Gamma)$

引理 3.3（良构的上下文）

如果存在 A 和 B 使得 $\Gamma \vdash A : B$，则上下文 Γ 是良构的。

引理 3.4（压缩引理，*Condensing Lemma*）

如果 $\Gamma', x : A, \Gamma'' \vdash B : C$ 且 x 不在 Γ'', B, C 中出现，则有 $\Gamma', \Gamma'' \vdash B : C$。

需要注意的是，压缩引理与 Lemma 2.10.5 不同：Lemma 2.10.5 中，所有多余声明可以一次性去掉，而新的引理中一次只能去除一个，因为 B, C 中的自由变量可能依赖于上下文中其它的变量。

引理 3.5（替换引理）

令 $\Gamma', x : A, \Gamma'' \vdash B : C$ 且 $\Gamma' \vdash D : A$，则 $\Gamma', \Gamma'' [x := D] \vdash B [x := D] : C [x := D]$

引理 3.6（*Subject Reduction*）

如果 $\Gamma \vdash A : B$ 且 $A \twoheadrightarrow_\beta A'$，则 $\Gamma \vdash A' : B$。

在第四章中提到过，$(SubjectReduction)$ 引理可以直接证明，而 $(conv)$-规则需要显式给出，因为出现过的项不能在声明中（以相同含义）出现第二次，无法同时引出两个类型，而相同类型可以多次出现，推导出不同的项即可。

定义 3.7（*Strong Normalisation Theorem / Termination Theorem*）

每一个合法的 M 都是 *strongly normalising*。

定义 3.8（良好定义和类型检查的可判定性）

在 λC 和它的子系统中，良好定义和类型检查都是可判定的。

项查找在 $\lambda\!\rightarrow$ 和 $\lambda\underline{\omega}$ 中是可判定的，但是在剩余的系统中是不可判定的。

尽管项查找在许多时候是不可判定的，但是计算机可以为定理证明提供大量的帮助，通过提供一些开放的目标，或者检查推导的过程是否符合规则。这样的程序被称为“证明助手（proof assistants）”，在帮助人们解决逻辑或者代数问题时越来越有用，它们同样被用于证明计算机程序的正确性，比如证明一个给定的计算机程序是否满足它的规范。