

# 同倫類型論

**JoJo**

jojoid@duck.com

# 目录

1 $\lambda$ 演算 .....	3
1.1 項 .....	3
1.2 自由和綁定變量 .....	3
1.3 $\alpha$ 等價 .....	4
1.4 代入 .....	4
2 類型論 .....	6
2.1 項 .....	6
2.2 語境 .....	6
2.3 結構規則 .....	6
2.4 類型宇宙 .....	6
2.5 依賴函數類型 .....	7
2.6 依賴序偶類型 .....	7
2.7 餘積類型 .....	8
2.8 空類型 $0$ .....	8
2.9 單元類型 $1$ .....	8
2.10 自然數類型 .....	9
2.11 恆等類型 .....	9
2.12 定義 .....	9
3 同倫類型論 .....	10
3.1 類型是高維羣胚 .....	10
3.2 函數是函子 .....	12

# 1 $\lambda$ 演算

## 1.1 項

### 定義 1.1 項

所有項的集合  $\Lambda$  的遞歸定義如下

1. (變量)  $\Lambda$  中有無窮個變量;
2. (抽象) 如果  $u$  是一個變量且  $M \in \Lambda$ , 則  $(u.M) \in \Lambda$ ;
3. (應用) 如果  $M, N \in \Lambda$ , 則  $(MN) \in \Lambda$ .

更簡短的表述是

$$\Lambda := V \mid (V.\Lambda) \mid (\Lambda\Lambda)$$

或

$$M := u \mid (u.M) \mid (MN)$$

其中  $V$  是變量集.

### 定義 1.2 子項

項  $M$  的所有子項的集合定義為  $Sub(M)$ ,  $Sub$  的遞歸定義如下

1. (基礎) 對於任何變量  $x$ ,  $Sub(x) := \{x\}$ ;
2. (抽象)  $Sub(x.M) := Sub(M) \cup \{(x.M)\}$ ;
3. (應用)  $Sub(MN) := Sub(M) \cup Sub(N) \cup \{(MN)\}$ .

- 引理 1.1
1. (自反性) 對於任何項  $M$ , 有  $M \in Sub(M)$ ;
  2. (傳遞性) 如果  $L \in Sub(M)$  且  $M \in Sub(N)$ , 則  $L \in Sub(N)$ .

引理 1.2 項可以以樹表示給出, 如下圖中的例子



項的子項對應於項的樹表示的子樹.

- 慣例 1.1
1. 最外層括號可以省略;
  2. (抽象是右結合的)  $x.y.M$  是  $x.(y.M)$  的一個縮寫;
  3. (應用是左結合的)  $MNL$  是  $((MN)L)$  的一個縮寫;
  4. (應用優先於抽象)  $x.MN$  是  $x.(MN)$  的一個縮寫.

## 1.2 自由和綁定變量

### 定義 1.3 自由變量

項  $M$  的所有自由變量的集合定義為  $FV(M)$ ， $FV$  的遞歸定義如下

1. (變量)  $FV(x) := \{x\}$ ;
2. (抽象)  $FV(\lambda x.M) := FV(M) \setminus \{x\}$ ;
3. (應用)  $FV(MN) := FV(M) \cup FV(N)$ .

例子 1.1  $(y(x.(xz)))$  的樹表示如下圖所示



$$FV(y(x.(xz))) = \{y, z\}.$$

### 定義 1.4 閉項

一個項  $M$  是閉的  $:\Leftrightarrow FV(M) = \emptyset$ .

所有閉項的集合記為  $\Lambda^0$ .

## 1.3 $\alpha$ 等價

### 定義 1.5 重命名

將項  $M$  中  $x$  的每個自由出現都替換為  $y$ ，結果記為  $M^{x \rightarrow y}$ .

### 定義 1.6 $\alpha$ 等價

定義  $\alpha$  等價  $=_\alpha$  為符合如下性質的關係

1. (重命名) 如果  $y$  不在  $M$  中出現，則  $x.M =_\alpha y.M^{x \rightarrow y}$ ;
2. (兼容性) 如果  $M =_\alpha N$ ，則  $ML =_\alpha NL$ ， $LM =_\alpha LN$  且對於任何變量  $z$  有  $z.M =_\alpha z.N$ ;
3. (自反性)  $M =_\alpha M$ ;
4. (對稱性) 如果  $M =_\alpha N$ ，則  $N =_\alpha M$ ;
5. (傳遞性) 如果  $L =_\alpha M$  且  $M =_\alpha N$ ，則  $L =_\alpha N$ .

## 1.4 代入

### 定義 1.7 代入

- (1a)  $x[N/x] := N$ ;
- (1b) 如果  $x \neq y$ ，則  $y[N/x] := y$ ;
- (2)  $(PQ)[N/x] := (P[N/x])(Q[N/x])$ ;
- (3) 如果  $z.P^{y \rightarrow z} =_\alpha y.P$  且  $z \notin FV(N)$ ，則  $(y.P)[N/x] := z.(P^{y \rightarrow z}[N/x])$ .

引理 1.3 設  $x \neq y$  且  $x \notin FV(N)$ ，則  $L[M, N/x, y] = L[N, M[N/y]/x, y]$ .

**定義 1.8**   **同時代人**

$M[N_1, \dots, N_n / x_1, \dots, x_n]$  表示把項  $N_1, \dots, N_n$  **同時代人**到變量  $x_1, \dots, x_n$ .

## 2 類型論

### 2.1 項

#### 定義 2.1 項

比  $\lambda$  演算多了一些常量以及新的構造.

### 2.2 語境

#### 定義 2.2 語境

一個語境是一個列表

$$x_1 : A_1, x_2 : A_2, \dots, x_n : A_n$$

其中  $x_1, \dots, x_n$  是不同的變量，它們分別擁有類型  $A_1, \dots, A_n$ . 我們用  $\Gamma, \Delta$  等字母來縮寫語境.

#### 定義 2.3 語境規則

$\Gamma \text{ ctx}$  是一個判斷，表示“ $\Gamma$  是良構的語境.”有如下規則

$$\frac{}{\cdot \text{ ctx}} \text{ ctx-EMP}$$
$$\frac{x_1 : A_1, x_2 : A_2, \dots, x_{n-1} : A_{n-1} \vdash A_n : \mathcal{U}_i}{(x_1 : A_1, \dots, x_n : A_n) \text{ ctx}} \text{ ctx-EXT}$$

其中，變量  $x_n$  與變量  $x_1, \dots, x_n$  中的任何一個都不同.

### 2.3 結構規則

#### 定義 2.4 Vble 規則

$$\frac{(x_1 : A_1, \dots, x_n : A_n) \text{ ctx}}{x_1 : A_1, \dots, x_n : A_n \vdash x_i : A_i} \text{ Vble}$$

#### 定義 2.5 判斷相等

如果  $a =_{\alpha} b$ ，則  $a \equiv b$ .

$$\frac{\Gamma \vdash a : A}{\Gamma \vdash a \equiv a : A}$$
$$\frac{\Gamma \vdash a \equiv b : A}{\Gamma \vdash b \equiv a : A}$$
$$\frac{\Gamma \vdash a \equiv b : A \quad \Gamma \vdash b \equiv c : A}{\Gamma \vdash a \equiv c : A}$$
$$\frac{\Gamma \vdash a : A \quad \Gamma \vdash A \equiv B : \mathcal{U}_i}{\Gamma \vdash a : B}$$
$$\frac{\Gamma \vdash a \equiv b : A \quad \Gamma \vdash A \equiv B : \mathcal{U}_i}{\Gamma \vdash a \equiv b : B}$$

### 2.4 類型宇宙

## 定義 2.6 類型宇宙層級

$$\mathcal{U}_0, \mathcal{U}_1, \mathcal{U}_2, \dots$$

有如下規則

$$\frac{\Gamma \text{ ctx}}{\Gamma \vdash \mathcal{U}_i : \mathcal{U}_{i+1}} \mathcal{U}\text{-INTRO}$$

$$\frac{\Gamma \vdash A : \mathcal{U}_i}{\Gamma \vdash A : \mathcal{U}_{i+1}} \mathcal{U}\text{-CUMUL}$$

## 2.5 依賴函數類型

### 定義 2.7 依賴函數類型

$$\frac{\Gamma \vdash A : \mathcal{U}_i \quad \Gamma, x : A \vdash B : \mathcal{U}_i}{\Gamma \vdash (x : A) \rightarrow B : \mathcal{U}_i} \Pi\text{-FORM}$$

$$\frac{\Gamma \vdash A_1 \equiv A_2 : \mathcal{U}_i \quad \Gamma, x : A_1 \vdash B_1 \equiv B_2 : \mathcal{U}_i}{\Gamma \vdash (x : A_1) \rightarrow B_1 \equiv (x : A_2) \rightarrow B_2 : \mathcal{U}_i} \Pi\text{-FORM-EQ}$$

$$\frac{\Gamma, x : A \vdash b : B}{\Gamma \vdash (x : A) \mapsto b : (x : A) \rightarrow B} \Pi\text{-INTRO}$$

$$\frac{\Gamma, x : A \vdash b_1 \equiv b_2 : B}{\Gamma \vdash (x : A) \mapsto b_1 \equiv (x : A) \mapsto b_2 : (x : A) \rightarrow B} \Pi\text{-INTRO-EQ}$$

$$\frac{\Gamma \vdash f : (x : A) \rightarrow B \quad \Gamma \vdash a : A}{\Gamma \vdash f(a) : B[a/x]} \Pi\text{-ELIM}$$

$$\frac{\Gamma \vdash f_1 \equiv f_2 : (x : A) \rightarrow B \quad \Gamma \vdash a : A}{\Gamma \vdash f_1(a) \equiv f_2(a) : B[a/x]} \Pi\text{-ELIM-EQ}$$

$$\frac{\Gamma, x : A \vdash b : B \quad \Gamma \vdash a : A}{\Gamma \vdash ((x : A) \mapsto b)(a) \equiv b[a/x] : B[a/x]} \Pi\text{-COMP}$$

$$\frac{\Gamma \vdash f : (x : A) \rightarrow B}{\Gamma \vdash f \equiv (x \mapsto f(x)) : (x : A) \rightarrow B} \Pi\text{-UNIQ}$$

## 2.6 依賴序偶類型

### 定義 2.8 依賴序偶類型

$$\frac{\Gamma \vdash A : \mathcal{U}_i \quad \Gamma, x : A \vdash B : \mathcal{U}_i}{\Gamma \vdash (x : A) \times B : \mathcal{U}_i} \Sigma\text{-FORM}$$

$$\frac{\Gamma \vdash A_1 \equiv A_2 : \mathcal{U}_i \quad \Gamma, x : A_1 \vdash B_1 \equiv B_2 : \mathcal{U}_i}{\Gamma \vdash (x : A_1) \times B_1 \equiv (x : A_2) \times B_2 : \mathcal{U}_i} \Sigma\text{-FORM-EQ}$$

$$\frac{\Gamma, x : A \vdash B : \mathcal{U}_i \quad \Gamma \vdash a : A \quad \Gamma \vdash b : B[a/x]}{\Gamma \vdash (a, b) : (x : A) \times B} \Sigma\text{-INTRO}$$

$$\frac{\Gamma, x : A \vdash B : \mathcal{U}_i \quad \Gamma \vdash a_1 \equiv a_2 : A \quad \Gamma \vdash b_1 \equiv b_2 : B[a/x]}{\Gamma \vdash (a_1, b_1) \equiv (a_2, b_2) : (x : A) \times B} \Sigma\text{-INTRO-EQ}$$

$$\frac{\Gamma, z : (x : A) \times B \vdash C : \mathcal{U}_i \quad \Gamma, x : A, y : B \vdash g : C[(x, y)/z] \quad \Gamma \vdash p : (x : A) \times B}{\Gamma \vdash \text{ind}_{(x:A) \times B}(z.C, x.y.g, p) : C[p/z]} \Sigma\text{-ELIM}$$

$$\frac{\Gamma, z : (x : A) \times B \vdash C : \mathcal{U}_i \quad \Gamma, x : A, y : B \vdash g : C[(x, y)/z] \quad \Gamma \vdash p_1 \equiv p_2 : (x : A) \times B}{\Gamma \vdash \text{ind}_{(x:A) \times B}(z.C, x.y.g, p_1) \equiv \text{ind}_{(x:A) \times B}(z.C, x.y.g, p_2) : C[p_1/z] \equiv C[p_2/z]} \Sigma\text{-ELIM-EQ}$$

$$\frac{\Gamma, z : (x : A) \times B \vdash C : \mathcal{U}_i \quad \Gamma, x : A, y : B \vdash g : C[(x, y)/z] \quad \Gamma \vdash a : A \quad \Gamma \vdash b : B[a/x]}{\Gamma \vdash \text{ind}_{(x:A) \times B}(z.C, x.y.g, (a, b)) \equiv g[a, b/x, y] : C[p/z]} \Sigma\text{-COMP}$$

## 2.7 餘積類型

### 定義 2.9 餘積類型

$$\begin{array}{c}
\frac{\Gamma \vdash A : \mathcal{U}_i \quad \Gamma \vdash B : \mathcal{U}_i}{\Gamma \vdash A + B : \mathcal{U}_i} \text{+-FORM} \\
\\
\frac{\Gamma \vdash A_1 \equiv A_2 : \mathcal{U}_i \quad \Gamma \vdash B_1 \equiv B_2 : \mathcal{U}_i}{\Gamma \vdash A_1 + B_1 \equiv A_2 + B_2 : \mathcal{U}_i} \text{+-FORM-EQ} \\
\\
\frac{\Gamma \vdash A : \mathcal{U}_i \quad \Gamma \vdash B : \mathcal{U}_i \quad \Gamma \vdash a : A}{\Gamma \vdash \text{inl}(a) : A + B} \text{+-INTRO}_1 \\
\\
\frac{\Gamma \vdash A : \mathcal{U}_i \quad \Gamma \vdash B : \mathcal{U}_i \quad \Gamma \vdash b : B}{\Gamma \vdash \text{inr}(b) : A + B} \text{+-INTRO}_2 \\
\\
\frac{\Gamma \vdash A : \mathcal{U}_i \quad \Gamma \vdash B : \mathcal{U}_i \quad \Gamma \vdash a_1 \equiv a_2 : A}{\Gamma \vdash \text{inl}(a_1) \equiv \text{inl}(a_2) : A + B} \text{+-INTRO}_1\text{-EQ} \\
\\
\frac{\Gamma \vdash A : \mathcal{U}_i \quad \Gamma \vdash B : \mathcal{U}_i \quad \Gamma \vdash b_1 \equiv b_2 : B}{\Gamma \vdash \text{inr}(b_1) \equiv \text{inr}(b_2) : A + B} \text{+-INTRO}_2\text{-EQ} \\
\\
\frac{\Gamma, z : (A + B) \vdash C : \mathcal{U}_i \quad \Gamma, x : A \vdash c : C[\text{inl}(x)/z] \quad \Gamma, y : B \vdash d : C[\text{inr}(y)/z] \quad \Gamma \vdash e : (A + B)}{\Gamma \vdash \text{ind}_{A+B}(z.C, x.c, y.d, e) : C[e/z]} \text{+-ELIM} \\
\\
\frac{\Gamma, z : (A + B) \vdash C : \mathcal{U}_i \quad \Gamma, x : A \vdash c : C[\text{inl}(x)/z] \quad \Gamma, y : B \vdash d : C[\text{inr}(y)/z] \quad \Gamma \vdash e_1 \equiv e_2 : (A + B)}{\Gamma \vdash \text{ind}_{A+B}(z.C, x.c, y.d, e_1) \equiv \text{ind}_{A+B}(z.C, x.c, y.d, e_2) : C[e_1/z] \equiv C[e_2/z]} \text{+-ELIM-EQ} \\
\\
\frac{\Gamma, z : (A + B) \vdash C : \mathcal{U}_i \quad \Gamma, x : A \vdash c : C[\text{inl}(x)/z] \quad \Gamma, y : B \vdash d : C[\text{inr}(y)/z] \quad \Gamma \vdash a : A}{\Gamma \vdash \text{ind}_{A+B}(z.C, x.c, y.d, \text{inl}(a)) \equiv c[a/x] : C[\text{inl}(a)/z]} \text{+-COMP}_1 \\
\\
\frac{\Gamma, z : (A + B) \vdash C : \mathcal{U}_i \quad \Gamma, x : A \vdash c : C[\text{inl}(x)/z] \quad \Gamma, y : B \vdash d : C[\text{inr}(y)/z] \quad \Gamma \vdash b : B}{\Gamma \vdash \text{ind}_{A+B}(z.C, x.c, y.d, \text{inr}(b)) \equiv d[b/y] : C[\text{inr}(b)/z]} \text{+-COMP}_2
\end{array}$$

## 2.8 空類型 0

### 定義 2.10 空類型 0

$$\begin{array}{c}
\frac{\Gamma \text{ ctx}}{\Gamma \vdash \mathbf{0} : \mathcal{U}_i} \mathbf{0}\text{-FORM} \\
\\
\frac{\Gamma, x : \mathbf{0} \vdash C : \mathcal{U}_i \quad \Gamma \vdash a : \mathbf{0}}{\Gamma \vdash \text{ind}_{\mathbf{0}}(x.C, a) : C[a/x]} \mathbf{0}\text{-ELIM} \\
\\
\frac{\Gamma, x : \mathbf{0} \vdash C : \mathcal{U}_i \quad \Gamma \vdash a_1 \equiv a_2 : \mathbf{0}}{\Gamma \vdash \text{ind}_{\mathbf{0}}(x.C, a_1) \equiv \text{ind}_{\mathbf{0}}(x.C, a_2) : C[a_1/x] \equiv C[a_2/x]} \mathbf{0}\text{-ELIM-EQ}
\end{array}$$

## 2.9 單元類型 1

### 定義 2.11 單元類型 1

$$\begin{array}{c}
\frac{\Gamma \text{ ctx}}{\Gamma \vdash \mathbf{1} : \mathcal{U}_i} \mathbf{1}\text{-FORM} \\
\\
\frac{\Gamma \text{ ctx}}{\Gamma \vdash \star : \mathbf{1}} \mathbf{1}\text{-INTRO} \\
\\
\frac{\Gamma, x : \mathbf{1} \vdash C : \mathcal{U}_i \quad \Gamma \vdash c : C[\star/x] \quad \Gamma \vdash a : \mathbf{1}}{\Gamma \vdash \text{ind}_{\mathbf{1}}(x.C, c, a) : C[a/x]} \mathbf{1}\text{-ELIM} \\
\\
\frac{\Gamma, x : \mathbf{1} \vdash C : \mathcal{U}_i \quad \Gamma \vdash c : C[\star/x] \quad \Gamma \vdash a_1 \equiv a_2 : \mathbf{1}}{\Gamma \vdash \text{ind}_{\mathbf{1}}(x.C, c, a_1) \equiv \text{ind}_{\mathbf{1}}(x.C, c, a_2) : C[a_1/x] \equiv C[a_2/x]} \mathbf{1}\text{-ELIM-EQ} \\
\\
\frac{\Gamma, x : \mathbf{1} \vdash C : \mathcal{U}_i \quad \Gamma \vdash c : C[\star/x]}{\Gamma \vdash \text{ind}_{\mathbf{1}}(x.C, c, \star) \equiv c : C[\star/x]} \mathbf{1}\text{-COMP}
\end{array}$$



## 2.10 自然數類型

### 定義 2.12 自然數類型

$$\begin{array}{c}
\frac{\Gamma \text{ ctx}}{\Gamma \vdash \mathbb{N} : \mathcal{U}_i} \text{N-FORM} \\
\\
\frac{\Gamma \text{ ctx}}{\Gamma \vdash 0 : \mathbb{N}} \text{N-INTRO}_1 \\
\\
\frac{\Gamma \vdash n : \mathbb{N}}{\Gamma \vdash \text{succ}(n) : \mathbb{N}} \text{N-INTRO}_2 \\
\\
\frac{\Gamma \vdash n_1 \equiv n_2 : \mathbb{N}}{\Gamma \vdash \text{succ}(n_1) \equiv \text{succ}(n_2) : \mathbb{N}} \text{N-INTRO}_2\text{-EQ} \\
\\
\frac{\Gamma, x : \mathbb{N} \vdash C : \mathcal{U}_i \quad \Gamma \vdash c_0 : C[0/x] \quad \Gamma, x : \mathbb{N}, y : C \vdash c_s : C[\text{succ}(x)/x] \quad \Gamma \vdash n : \mathbb{N}}{\Gamma \vdash \text{ind}_{\mathbb{N}}(x.C, c_0, x.y.c_s, n) : C[n/x]} \text{N-ELIM} \\
\\
\frac{\Gamma, x : \mathbb{N} \vdash C : \mathcal{U}_i \quad \Gamma \vdash c_0 : C[0/x] \quad \Gamma, x : \mathbb{N}, y : C \vdash c_s : C[\text{succ}(x)/x] \quad \Gamma \vdash n_1 \equiv n_2 : \mathbb{N}}{\Gamma \vdash \text{ind}_{\mathbb{N}}(x.C, c_0, x.y.c_s, n_1) \equiv \text{ind}_{\mathbb{N}}(x.C, c_0, x.y.c_s, n_2) : C[n_1/x] \equiv C[n_2/x]} \text{N-ELIM-EQ} \\
\\
\frac{\Gamma, x : \mathbb{N} \vdash C : \mathcal{U}_i \quad \Gamma \vdash c_0 : C[0/x] \quad \Gamma, x : \mathbb{N}, y : C \vdash c_s : C[\text{succ}(x)/x]}{\Gamma \vdash \text{ind}_{\mathbb{N}}(x.C, c_0, x.y.c_s, 0) \equiv c_0 : C[0/x]} \text{N-COMP}_1 \\
\\
\frac{\Gamma, x : \mathbb{N} \vdash C : \mathcal{U}_i \quad \Gamma \vdash c_0 : C[0/x] \quad \Gamma, x : \mathbb{N}, y : C \vdash c_s : C[\text{succ}(x)/x] \quad \Gamma \vdash n : \mathbb{N}}{\Gamma \vdash \text{ind}_{\mathbb{N}}(x.C, c_0, x.y.c_s, \text{succ}(n)) \equiv c_s[n, \text{ind}_{\mathbb{N}}(x.C, c_0, x.y.c_s, n)/x, y] : C[\text{succ}(n)/x]} \text{N-COMP}_2
\end{array}$$

## 2.11 恆等類型

### 定義 2.13 恆等類型

$$\begin{array}{c}
\frac{\Gamma \vdash A : \mathcal{U}_i \quad \Gamma \vdash a : A \quad \Gamma \vdash b : A}{\Gamma \vdash a =_A b : \mathcal{U}_i} =\text{-FORM} \\
\\
\frac{\Gamma \vdash A : \mathcal{U}_i \quad \Gamma \vdash a_1 \equiv a_2 : A \quad \Gamma \vdash b_1 \equiv b_2 : A}{\Gamma \vdash a_1 =_A b_1 \equiv a_2 =_A b_2 : \mathcal{U}_i} =\text{-FORM-EQ} \\
\\
\frac{\Gamma \vdash A : \mathcal{U}_i \quad \Gamma \vdash a : A}{\Gamma \vdash \text{refl}_a : a =_A a} =\text{-INTRO} \\
\\
\frac{\Gamma \vdash A : \mathcal{U}_i \quad \Gamma \vdash a_1 \equiv a_2 : A}{\Gamma \vdash \text{refl}_{a_1} \equiv \text{refl}_{a_2} : a_1 =_A a_1 \equiv a_2 =_A a_2} =\text{-INTRO-EQ} \\
\\
\frac{\Gamma, x : A, y : A, p : x =_A y \vdash C : \mathcal{U}_i \quad \Gamma, z : A \vdash c : C[z, z, \text{refl}_z/x, y, p] \quad \Gamma \vdash a : A \quad \Gamma \vdash b : A \quad \Gamma \vdash q : a =_A b}{\Gamma \vdash \text{ind}_{=_A}(x.y.p.C, z.c, a, b, q) : C[a, b, q/x, y, p]} =\text{-ELIM} \\
\\
\frac{\Gamma, x : A, y : A, p : x =_A y \vdash C : \mathcal{U}_i \quad \Gamma, z : A \vdash c : C[z, z, \text{refl}_z/x, y, p] \quad \Gamma \vdash a : A \quad \Gamma \vdash b : A \quad \Gamma \vdash q_1 \equiv q_2 : a =_A b}{\Gamma \vdash \text{ind}_{=_A}(x.y.p.C, z.c, a, b, q_1) \equiv \text{ind}_{=_A}(x.y.p.C, z.c, a, b, q_2) : C[a, b, q_1/x, y, p] \equiv C[a, b, q_2/x, y, p]} =\text{-ELIM-EQ} \\
\\
\frac{\Gamma, x : A, y : A, p : x =_A y \vdash C : \mathcal{U}_i \quad \Gamma, z : A \vdash c : C[z, z, \text{refl}_z/x, y, p] \quad \Gamma \vdash a : A}{\Gamma \vdash \text{ind}_{=_A}(x.y.p.C, z.c, a, a, \text{refl}_a) \equiv c[a/z] : C[a, a, \text{refl}_a/x, y, p]} =\text{-COMP}
\end{array}$$

恆等類型的項稱為**道路**；恆等類型的消除規則稱為**道路歸納**。

## 2.12 定義

**例子 2.1**  $\circ := (A : \mathcal{U}_i) \mapsto (B : \mathcal{U}_i) \mapsto (C : \mathcal{U}_i) \mapsto (g : B \rightarrow C) \mapsto (f : A \rightarrow B) \mapsto (x : A) \mapsto g(f(x))$ .

### 3 同倫類型論

#### 3.1 類型是高維羣胚

**引理 3.1** 對於任何  $A : \mathcal{U}_i, x, y : A$ ，都能構造一個函數  $_-^{-1} : (x =_A y) \rightarrow (y =_A x)$  使得  $(\text{refl}_x)^{-1} \equiv \text{refl}_x$ .

$p^{-1}$  稱為  $p$  的逆.

*Proof.* 第一種證明

設  $A : \mathcal{U}_i, D : (x, y : A) \rightarrow (x =_A y) \rightarrow \mathcal{U}_i, D(x, y, p) := (y =_A x)$ .

隨即我們就能構造一個函數  $d := x \mapsto \text{refl}_x : (x : A) \rightarrow D(x, x, \text{refl}_x)$ .

然後根據恆等類型的消除規則我們有，對於任何  $x, y : A, p : (x =_A y)$ ，可以構造項  $\text{ind}_{=_A}(D, d, x, y, p) : (y =_A x)$ .

現在對於任何  $x, y : A$  我們可以定義期望得到的函數  $_-^{-1} := p \mapsto \text{ind}_{=_A}(D, d, x, y, p)$ .

由恆等類型的計算規則，得  $(\text{refl}_x)^{-1} \equiv \text{refl}_x$ . □

*Proof.* 第二種證明

對於每個  $x, y : A$  和  $p : x = y$ ，我們想要構造一個項  $p^{-1} : y = x$ . 根據  $p$  的道路歸納，我們只需要給出  $y$  是  $x$  且  $p$  是  $\text{refl}_x$  時的構造. 在該情況下， $\text{refl}_x$  和  $\text{refl}_x^{-1}$  的類型都是  $x = x$ . 因此我們可以簡單地定義  $\text{refl}_x^{-1} := \text{refl}_x$ . 於是根據道路歸納，我們完成了構造. □

**引理 3.2** 對於任何  $A : \mathcal{U}_i, x, y, z : A$ ，都能構造一個函數  $\cdot : (x =_A y) \rightarrow (y =_A z) \rightarrow (x =_A z)$  使得  $\text{refl}_x \cdot \text{refl}_x \equiv \text{refl}_x$ .

$p \cdot q$  稱為  $p$  和  $q$  的連接.

*Proof.* 第一種證明

期望得到的函數擁有類型  $(x, y, z : A) \rightarrow (x =_A y) \rightarrow (y =_A z) \rightarrow (x =_A z)$ .

我們將改為定義一個函數，擁有和預期等價的類型  $(x, y : A) \rightarrow (x =_A y) \rightarrow (z : A) \rightarrow (y =_A z) \rightarrow (x =_A z)$ ，這允許我們使用兩次恆等類型的消除規則.

設  $D : (x, y : A) \rightarrow (x =_A y) \rightarrow \mathcal{U}_i, D(x, y, p) := (z : A) \rightarrow (q : y =_A z) \rightarrow (x =_A z)$ .

然後，為了對  $D$  應用恆等類型的消除規則，我們需要類型為  $(x : A) \rightarrow D(x, x, \text{refl}_x)$  的函數，也就是類型為  $(x, z : A) \rightarrow (q : x =_A z) \rightarrow (x =_A z)$ .

現在設  $E : (x, z : A) \rightarrow (q : x =_A z) \rightarrow \mathcal{U}_i, E(x, z, q) := (x =_A z)$ .

隨即我們能構造函數  $e := x \mapsto \text{refl}_x : (x : A) \rightarrow E(x, x, \text{refl}_x)$ .

對  $E$  應用恆等類型的消除規則，我們得到函數  $d : (x, z : A) \rightarrow (q : x =_A z) \rightarrow E(x, z, q), x \mapsto z \mapsto q \mapsto \text{ind}_{=_A}(E, e, x, z, q)$ .

因為  $E(x, z, q) \equiv (x =_A z)$ ，所以  $d : (x : A) \rightarrow D(x, x, \text{refl}_x)$ .

然後對  $D$  應用恆等類型的消除規則我們有，對於任何  $x, y : A, p : (x =_A y)$ ，可以構造項  $\text{ind}_{=_A}(D, d, x, y, p) \equiv \text{ind}_{=_A}(D, (x, z : A) \mapsto (q : y =_A z) \mapsto \text{ind}_{=_A}(E, e, x, z, q), x, y, p) : (z : A) \rightarrow (q : y =_A z) \rightarrow (x =_A z)$ .

於是我們有

$$(x, y : A) \mapsto (p : x =_A y) \mapsto \text{ind}_{=_A}(D, (x, z : A) \mapsto (q : y =_A z) \mapsto \text{ind}_{=_A}(E, e, x, z, q), x, y, p) : \\ (x, y : A) \rightarrow (x =_A y) \rightarrow (z : A) \rightarrow (y =_A z) \rightarrow (x =_A z)$$

現在對於任何  $a, b, c : A$  我們可以定義期望得到的函數

$$\cdot := (p : a =_A b) \mapsto \text{ind}_{=_A}(D, (x : A) \mapsto (q : b =_A c) \mapsto \text{ind}_{=_A}(E, e, x, c, q), a, b, p) : \\ (a, b, c : A) \rightarrow (a =_A b) \rightarrow (b =_A c) \rightarrow (a =_A c).$$

由恆等映射的計算規則，得

$$\text{refl}_a \cdot \text{refl}_a \equiv \text{ind}_{=_A}(D, (x : A) \mapsto \text{ind}_{=_A}(E, e, x, a, \text{refl}_a), a, a, \text{refl}_a) \equiv \text{ind}_{=_A}(E, e, a, a, \text{refl}_a) \equiv e(a) \equiv \text{refl}_a.$$

□

*Proof.* 第二種證明

對於每個  $x, y, z : A$ ,  $p : x = y$  和  $q : y = z$ , 我們想要構造一個項  $p \bullet q : x = z$ . 根據  $p$  的道路歸納, 我們只需要給出  $y$  是  $x$  且  $p$  是  $\text{refl}_x$  時的構造, 即對於每個  $x, z : A$  和  $q : x = z$ , 構造一個項  $\text{refl}_x \bullet q : x = z$ . 根據  $q$  的道路歸納, 只需給出  $z$  是  $x$  且  $q$  是  $\text{refl}_x$  時的構造, 即對於每個  $x : A$ , 構造一個項  $\text{refl}_x \bullet \text{refl}_x : x = x$ . 因此我們可以簡單地定義  $\text{refl}_x \bullet \text{refl}_x \equiv \text{refl}_x$ . 於是根據道路歸納, 我們完成了構造.  $\square$

**引理 3.3** 設  $A : \mathcal{U}_i$ ,  $x, y, z, w : A$ ,  $p : x = y$ ,  $q : y = z$  且  $r : z = w$ . 我們有以下結論:

1.  $p = p \bullet \text{refl}_y$  且  $p = \text{refl}_x \bullet p$ ;
2.  $p \bullet p^{-1} = \text{refl}_x$  且  $p^{-1} \bullet p = \text{refl}_y$ ;
3.  $(p^{-1})^{-1} = p$ ;
4.  $p \bullet (q \bullet r) = (p \bullet q) \bullet r$ .

*Proof.* 所有證明都使用道路歸納.

1. 第一種證明: 設  $D : (x, y : A) \rightarrow (p : x = y) \rightarrow \mathcal{U}$ ,  $D(x, y, p) \equiv (p = p \bullet \text{refl}_y)$ . 那麼  $D(x, x, \text{refl}_x)$  是  $\text{refl}_x = \text{refl}_x \bullet \text{refl}_x$ . 因為  $\text{refl}_x \bullet \text{refl}_x \equiv \text{refl}_x$ , 我們有  $D(x, x, \text{refl}_x) \equiv (\text{refl}_x = \text{refl}_x)$ . 因此可以構造函數  $d \equiv x \mapsto \text{refl}_{\text{refl}_x} : (x : A) \rightarrow D(x, x, \text{refl}_x)$ . 根據道路歸納, 對於每個  $x, y : A$  和  $p : x = y$ , 我們有項  $\text{ind}_{=A}(D, d, x, y, p) : p = p \bullet \text{refl}_y$ .

第二種證明: 根據  $p$  的道路歸納, 只需要假設  $y$  是  $x$  且  $p$  是  $\text{refl}_x$ . 在該情況下,  $p \bullet \text{refl}_y \equiv \text{refl}_x \bullet \text{refl}_x \equiv \text{refl}_x$ . 因此只需證明  $\text{refl}_x = \text{refl}_x$ , 這是簡單的, 即  $\text{refl}_{\text{refl}_x} : \text{refl}_x = \text{refl}_x$ .

2. 第一種證明: 設  $D : (x, y : A) \rightarrow (p : x = y) \rightarrow \mathcal{U}$ ,  $D(x, y, p) \equiv (p \bullet p^{-1} = \text{refl}_x)$ . 那麼  $D(x, x, \text{refl}_x)$  是  $\text{refl}_x \bullet \text{refl}_x^{-1} = \text{refl}_x$ . 因為  $\text{refl}_x^{-1} \equiv \text{refl}_x$  且  $\text{refl}_x \bullet \text{refl}_x \equiv \text{refl}_x$ , 我們有  $D(x, x, \text{refl}_x) \equiv (\text{refl}_x = \text{refl}_x)$ . 因此可以構造函數  $d \equiv x \mapsto \text{refl}_{\text{refl}_x} : (x : A) \rightarrow D(x, x, \text{refl}_x)$ . 根據道路歸納, 對於每個  $x, y : A$  和  $p : x = y$ , 我們有項  $\text{ind}_{=A}(D, d, x, y, p) : p \bullet p^{-1} = \text{refl}_x$ .

第二種證明: 根據  $p$  的道路歸納, 只需要假設  $y$  是  $x$  且  $p$  是  $\text{refl}_x$ . 在該情況下,  $p \bullet p^{-1} \equiv \text{refl}_x \bullet \text{refl}_x^{-1} \equiv \text{refl}_x$ .

3. 第一種證明: 設  $D : (x, y : A) \rightarrow (p : x = y) \rightarrow \mathcal{U}$ ,  $D(x, y, p) \equiv (p^{-1})^{-1} = p$ . 那麼  $D(x, x, p)$  是  $(\text{refl}_x^{-1})^{-1} = \text{refl}_x$ . 因為  $\text{refl}_x^{-1} \equiv \text{refl}_x$ , 所以  $(\text{refl}_x^{-1})^{-1} \equiv \text{refl}_x^{-1} \equiv \text{refl}_x$ , 那麼  $D(x, x, \text{refl}_x) \equiv (\text{refl}_x = \text{refl}_x)$ . 因此我們能構造函數  $d \equiv x \mapsto \text{refl}_{\text{refl}_x} : (x : A) \rightarrow D(x, x, \text{refl}_x)$ . 根據道路歸納, 對於每個  $x, y : A$  和  $p : x = y$ , 我們有項  $\text{ind}_{=A}(D, d, x, y, p) : (p^{-1})^{-1} = p$ .

第二種證明: 根據  $p$  的道路歸納, 只需要假設  $y$  是  $x$  且  $p$  是  $\text{refl}_x$ . 在該情況下,  $(p^{-1})^{-1} \equiv (\text{refl}_x^{-1})^{-1} \equiv \text{refl}_x$ .

4. 我們想要構造的函數的類型是  $(x, y, z, w : A) \rightarrow (p : x = y) \rightarrow (q : y = z) \rightarrow (r : z = w) \rightarrow (p \bullet (q \bullet r) = (p \bullet q) \bullet r)$ , 我們改為證明  $(x, y : A) \rightarrow (p : x = y) \rightarrow (z : A) \rightarrow (q : y = z) \rightarrow (w : A) \rightarrow (r : z = w) \rightarrow (p \bullet (q \bullet r) = (p \bullet q) \bullet r)$ .

設  $D_1 : (x, y : A) \rightarrow (p : x = y) \rightarrow \mathcal{U}$ ,  $D_1(x, y, p) \equiv (z : A) \rightarrow (q : y = z) \rightarrow (w : A) \rightarrow (r : z = w) \rightarrow (p \bullet (q \bullet r) = (p \bullet q) \bullet r)$ . 根據  $p$  的道路歸納, 只需要構造類型為  $(x : A) \rightarrow D_1(x, x, \text{refl}_x) \equiv (x, z : A) \rightarrow (q : x = z) \rightarrow (w : A) \rightarrow (r : z = w) \rightarrow (\text{refl}_x \bullet (q \bullet r) = (\text{refl}_x \bullet q) \bullet r)$  的函數.

為了構造這個類型的函數, 我們設  $D_2 : (x, z : A) \rightarrow (q : x = z) \rightarrow \mathcal{U}$ ,  $D_2(x, z, q) \equiv (w : A) \rightarrow (r : z = w) \rightarrow (\text{refl}_x \bullet (q \bullet r) = (\text{refl}_x \bullet q) \bullet r)$ . 根據  $q$  的道路歸納, 只需要構造類型為  $(x : A) \rightarrow D_2(x, x, \text{refl}_x) \equiv (x, w : A) \rightarrow (r : x = w) \rightarrow (\text{refl}_x \bullet (\text{refl}_x \bullet r) = (\text{refl}_x \bullet \text{refl}_x) \bullet r)$  的函數.

為了構造這個類型的函數, 我們設  $D_3 : (x, w : A) \rightarrow (r : x = w) \rightarrow \mathcal{U}$ ,  $D_3(x, w, r) \equiv (\text{refl}_x \bullet (\text{refl}_x \bullet r) = (\text{refl}_x \bullet \text{refl}_x) \bullet r)$ . 根據  $r$  的道路歸納, 只需要構造類型為  $(x : A) \rightarrow D_3(x, x, \text{refl}_x) \equiv (x : A) \rightarrow (\text{refl}_x \bullet (\text{refl}_x \bullet \text{refl}_x) = (\text{refl}_x \bullet \text{refl}_x) \bullet \text{refl}_x) \equiv (x : A) \rightarrow \text{refl}_x = \text{refl}_x$  的函數. 這是簡單的, 即  $\text{refl}_{\text{refl}_x}$ .

因此, 應用 3 此道路歸納, 我們就得到了想要的類型的函數.  $\square$

**定義 3.1** 設  $A : \mathcal{U}$ ,  $a : A$ . 序偶  $(A, a) : (A : \mathcal{U}) \times A$  稱為一個有點類型,  $a$  稱為它的基點. 類型  $(A : \mathcal{U}) \times A$  記為  $\mathcal{U}_\bullet$ .

### 定義 3.2 迴路空間

對於  $n : \mathbb{N}$ ，一個有點類型  $(A, a)$  的  $n$  重迭代迴路空間  $\Omega^n(A, a)$  遞歸地定義為

$$\Omega^0(A, a) := (A, a),$$

$$\Omega^1(A, a) := ((a =_A a), \text{refl}_a),$$

$$\Omega^{n+1}(A, a) := \Omega^n(\Omega(A, a)),$$

它的一個項稱為點  $a$  的一個  $n$  維迴路。

## 3.2 函數是函子

引理 3.4 對於任何  $A, B : \mathcal{U}, f : A \rightarrow B, x, y : A$ ，都能構造函數  $ap_f : (x =_A y) \rightarrow (f(x) =_B f(y))$ ,  $ap_f(\text{refl}_x) \equiv \text{refl}_{f(x)}$ .

*Proof.* 第一種證明：設  $D : (x, y : A) \rightarrow (x =_A y) \rightarrow \mathcal{U}, D(x, y, p) := (f(x) =_B f(y))$ . 那麼我們有  $d := (x : A) \mapsto \text{refl}_{f(x)} : (x : A) \rightarrow (f(x) =_B f(y))$ . 根據  $p$  的道路歸納，我們得到函數  $ap_f : (x =_A y) \rightarrow (f(x) =_B f(y))$ . 根據恆等類型的計算規則，對於任何  $x : A$ ，有  $ap_f(\text{refl}_x) \equiv \text{refl}_{f(x)}$ .

第二種證明：為了對任何  $p : x = y$  定義  $ap_f(p)$ ，根據  $p$  的道路歸納，只需要構造  $p$  是  $\text{refl}_x$  的情況。在該情況下，我們定義  $ap_f(\text{refl}_x) := \text{refl}_{f(x)} : f(x) = f(x)$ .

□

慣例 3.1 我們將經常將  $ap_f(p)$  簡寫為  $f(p)$ .

引理 3.5 對於任何函數  $f : A \rightarrow B, g : B \rightarrow C$  和道路  $p : x =_A y, q : y =_B z$ ，我們有：

1.  $ap_f(p \bullet q) = ap_f(p) \bullet ap_f(q)$ ;
2.  $ap_f(p^{-1}) = (ap_f(p))^{-1}$ ;
3.  $ap_g(ap_f(p)) = ap_{g \circ f}(p)$ ;
4.  $ap_{id_A}(p) = p$ .

*Proof.*

□