プログラム意味論とファイブレーション（仮）

眞田 嵩大
数理解析研究所 D1

Yota25，2022 年 3 月 26 日

Contents

ファイブレーション

プログラム意味論

プログラム意味論とファイブレーション

ファイブレーション

- $\mathcal{E}$，$\mathcal{B}$: 圏
- $p : \mathcal{E} \to \mathcal{B}$: 関手

$p : \mathcal{E} \to \mathcal{B}$ が**ファイブレーション**であるとは，

$$
\begin{tikzcd}
\mathcal{E} \arrow[d, "p"] & u^*X \arrow[r, "\overline{u}X"] & X \\
\mathcal{B} & J \arrow[r, "u"] & pX
\end{tikzcd}
$$

任意の対象 $J \in \mathcal{B}$，$X \in \mathcal{E}$ と射 $u : J \to pX$ に対して，ある対象 $u^*X \in \mathcal{E}$ と射 $\overline{u}X : u^*X \to X$ が存在して $p(\overline{u}X) = u$ である．さらに $\overline{u}X$ は次の普遍性を持つ．

ファイブレーション

- $\mathcal{E}$, $\mathcal{B}$: 圏
- $p : \mathcal{E} \to \mathcal{B}$: 関手

$p : \mathcal{E} \to \mathcal{B}$ が**ファイブレーション**であるとは,

$$\begin{tikzcd} \mathcal{E} \arrow[d, "p"] & u^*X \arrow[r, "\overline{u}X"] & X \\ \mathcal{B} & J \arrow[r, "u"] & pX \end{tikzcd}$$

任意の対象 $J \in \mathcal{B}$, $X \in \mathcal{E}$ と射 $u : J \to pX$ に対して, ある対象 $u^*X \in \mathcal{E}$ と射 $\overline{u}X : u^*X \to X$ が存在して $p(\overline{u}X) = u$ である. さらに $\overline{u}X$ は次の普遍性を持つ.

ファイブレーション

- $\mathcal{E}$, $\mathcal{B}$: 圏
- $p : \mathcal{E} \to \mathcal{B}$: 関手

$p : \mathcal{E} \to \mathcal{B}$ が**ファイブレーション**であるとは,

$$\begin{array}{ccc} \mathcal{E} & \quad & u^*X \xrightarrow{\overline{u}X} X \\ \downarrow p & & \\ \mathcal{B} & & J \xrightarrow{u} pX \end{array}$$

任意の対象 $J \in \mathcal{B}$, $X \in \mathcal{E}$ と射 $u : J \to pX$ に対して，ある対象 $u^*X \in \mathcal{E}$ と射 $\overline{u}X : u^*X \to X$ が存在して $p(\overline{u}X) = u$ である．さらに $\overline{u}X$ は次の普遍性を持つ．

カルテシアン射の普遍性

$\overline{u}X$ は次の普遍性を持つ．

$$\begin{tikzcd}
\mathcal{E} \arrow[dd, "p"] & Z \arrow[dr, dashed, "h"'] \arrow[drr, bend left, "g"] & & \\
 & & u^*X \arrow[r, "\overline{u}X"] & X \\
\mathcal{B} & pZ \arrow[dr, "v"'] \arrow[drr, bend left, "u\circ v"] & & \\
 & & J \arrow[r, "u"] & pX
\end{tikzcd}$$

任意の $Z \in \mathcal{E}$ と $g : Z \to X$，$v : pZ \to J$ に対して $p(g) = u \circ v$ ならば，$p(h) = v$ を満たす $h : Z \to u^*X$ が一意的に存在する．このような普遍性を持つ射をカルテシアン射と呼ぶ．

カルテシアン射の普遍性

$\overline{u}X$ は次の普遍性を持つ．

$$
\begin{tikzcd}
\mathcal{E} \arrow[dd, "p"] & Z \arrow[drr, bend left, "g"] \arrow[dr, dashed, "\color{red}h"'] & & \\
 & & u^*X \arrow[r, "\overline{u}X"] & X \\
\mathcal{B} & pZ \arrow[drr, bend left, "u\circ v"] \arrow[dr, "v"'] & & \\
 & & J \arrow[r, "u"] & pX
\end{tikzcd}
$$

任意の $Z \in \mathcal{E}$ と $g : Z \to X$，$v : pZ \to J$ に対して $p(g) = u \circ v$ ならば，$p(h) = v$ を満たす $h : Z \to u^*X$ が一意的に存在する．このような普遍性を持つ射をカルテシアン射と呼ぶ．

カルテシアン射の普遍性

$\overline{u}X$ は次の普遍性を持つ．

$$
\begin{tikzcd}
\mathcal{E} \arrow[dd, "p"] & Z \arrow[dr, dashed, "h"'] \arrow[drr, bend left, "g"] & & \\
& & u^*X \arrow[r, "\overline{u}X"] & X \\
\mathcal{B} & pZ \arrow[dr, "v"'] \arrow[drr, bend left, "u\circ v"] & & \\
& & J \arrow[r, "u"] & pX
\end{tikzcd}
$$

任意の $Z \in \mathcal{E}$ と $g : Z \to X$，$v : pZ \to J$ に対して $p(g) = u \circ v$ ならば，$p(h) = v$ を満たす $h : Z \to u^*X$ が一意的に存在する．このような普遍性を持つ射をカルテシアン射と呼ぶ．

ファイブレーションの気持ち

ファイブレーション $p : \mathcal{E} \to \mathcal{B}$ を以下のような気持ちで眺める．

- $\mathcal{B}$ は基本的な構造の圏
- $\mathcal{E}$ は $\mathcal{B}$ により細かな構造が付加された圏
- p は $X \in \mathcal{E}$ に対してそのベースの構造 $pX \in \mathcal{B}$ を抽出する
- $u : J \to pX$ に対して $u^*X \in \mathcal{E}$ は，u を $\mathcal{E}$ に持ち上げられるような最も弱い構造

ファイブレーションの具体例：Pred → Set

Set を集合と写像の圏とする．圏 Pred を次で定める．

- 対象は組 (I, X) たち．ここで $I \in$ Set かつ $X \subseteq I$．
- 射 $f : (J, Y) \to (I, X)$ は写像 $f : J \to I$ であって Y の元を X にうつすもの．

関手 $p :$ Pred → Set を

$$\begin{aligned}(I, X) &\mapsto I \\ f &\mapsto f\end{aligned}$$

と定める．このとき p はファイブレーションとなる．
$u : J \to p(I, X)$ に対して

$$u^*(I, X) = (J, u^{-1}(X)) = (J, \{j \in J \mid uj \in X\})$$

である．このことから u^* のことを逆像関手と呼ぶ．

ファイブレーションの具体例：Pred → Set

Set を集合と写像の圏とする．圏 Pred を次で定める．

- 対象は組 (I, X) たち．ここで $I \in \mathbf{Set}$ かつ $X \subseteq I$．
- 射 $f : (J, Y) \to (I, X)$ は写像 $f : J \to I$ であって Y の元を X にうつすもの．

関手 $p : \mathbf{Pred} \to \mathbf{Set}$ を

$$\begin{aligned} (I, X) &\mapsto I \\ f &\mapsto f \end{aligned}$$

と定める．このとき p はファイブレーションとなる．

$u : J \to p(I, X)$ に対して

$$u^*(I, X) = (J, u^{-1}(X)) = (J, \{j \in J \mid uj \in X\})$$

である．このことから u^* のことを**逆像関手**と呼ぶ．

ファイブレーションの具体例：Pred $\to$ Set

Set を集合と写像の圏とする．圏 Pred を次で定める．

- 対象は組 (I, X) たち．ここで $I \in$ Set かつ $X \subseteq I$．
- 射 $f : (J, Y) \to (I, X)$ は写像 $f : J \to I$ であって Y の元を X にうつすもの．

関手 p : Pred $\to$ Set を

$$\begin{aligned}(I, X) &\mapsto I \\ f &\mapsto f\end{aligned}$$

と定める．このとき p はファイブレーションとなる．
$u : J \to p(I, X)$ に対して

$$u^*(I, X) = (J, u^{-1}(X)) = (J, \{j \in J \mid uj \in X\})$$

である．このことから u^* のことを逆像関手と呼ぶ．

ファイブレーション

プログラム意味論

プログラム意味論とファイブレーション

プログラム意味論

単なる文字列としてのプログラムに何らか意味を与えることで計算の性質を研究する.

- **操作的意味論**：プログラムを**どう実行するか**を指定する（＝簡約規則を与える）ことで意味を定める
- **表示的意味論**：プログラムが**何を表すか**を指定する（＝プログラムの集合から数学的構造への割り当てを定める）ことで意味を定める.

今回は表示的意味論，特にプログラムの意味を圏によって解釈する**圏論的意味論**について考える.

型とプログラム

データの種類のことを型と呼ぶ．プログラムはデータを加工するレシピである．一般に型がついたプログラムを以下のように書く．

$$x_1 : \tau_1, \ldots, x_n : \tau_n \vdash M : \tau$$

プログラムの例

Int **を整数を表す型とする．**

- $x : \mathsf{Int}, y : \mathsf{Int} \vdash \mathsf{add}(x, y) : \mathsf{Int}$ **は，**Int **型の変数** x**，**y **が与えられている状況で** $\mathsf{add}(x, y)$ **というプログラムは** Int **型であることを主張している．**
- $x : \mathsf{Int} \vdash \lambda y^{\mathsf{Int}}.\mathsf{add}(x, y) : \mathsf{Int} \to \mathsf{Int}$

プログラムの解釈

プログラムに圏 Set における解釈を与えたい．

- 型 τ に Set の対象，つまり集合 $[\![\tau]\!]$ を割り当てる．

$$[\![\mathsf{Unit}]\!] = \{*\}$$
$$[\![\mathsf{Int}]\!] = \mathbb{N}$$
$$[\![\tau_1 \to \tau_2]\!] = [\![\tau_2]\!]^{[\![\tau_1]\!]}$$

- プログラム $x_1 : \tau_1, \ldots, x_n : \tau_n \vdash M : \tau$ は Set における射

$$[\![M]\!] : [\![\tau_1]\!] \times \cdots \times [\![\tau_n]\!] \to [\![\tau]\!]$$

  として解釈する．

ここでは Set における解釈しか考えないが，一般には適当な構造（カルテシアン閉，モノイダル閉など）を持つ圏でプログラムを解釈できる．

プログラムの解釈の例

- $x : \mathsf{Int}, y : \mathsf{Int} \vdash \mathsf{add}(x, y) : \mathsf{Int}$ **の解釈は**

$$\begin{aligned} [\![\mathsf{add}(x, y)]\!] : \mathbb{N} \times \mathbb{N} &\to \mathbb{N} \\ (m, n) &\mapsto m + n \end{aligned}$$

- $x : \mathsf{Int} \vdash \lambda y^{\mathsf{Int}}.\mathsf{add}(x, y) : \mathsf{Int} \to \mathsf{Int}$ **の解釈は**

$$\begin{aligned} [\![\lambda y^{\mathsf{Int}}.\mathsf{add}(x, y)]\!] : \mathbb{N} &\to \mathbb{N}^{\mathbb{N}} \\ m &\mapsto (n \mapsto m + n) \end{aligned}$$

ファイブレーション

プログラム意味論

プログラム意味論とファイブレーション

ファイブレーションの気持ち（改）

ファイブレーション $p : \mathcal{E} \to \mathcal{B}$ を以下のような気持ちで眺める．

- $\mathcal{B}$ は基本的な構造の圏
  - 型とプログラムが解釈される圏
- $\mathcal{E}$ は $\mathcal{B}$ により細かな構造が付加された圏
  - 型とプログラムの性質の圏
- p は $X \in \mathcal{E}$ に対してそのベースの構造 $pX \in \mathcal{B}$ を抽出する
  - 性質を記述する領域の抽出
- $u : J \to pX$ に対して $u^*X \in \mathcal{E}$ は，u を $\mathcal{E}$ に持ち上げられるような最も弱い構造
  - プログラムの最弱事前条件

プログラムのファイブレーション意味論

プログラム $x_1 : \tau_1, \ldots, x_n : \tau_n \vdash M : \tau$ **に対して**

$$x_1 : \tau_1, \ldots, x_n : \tau_n \, \{P\} \vdash M : \tau \{\upsilon.Q\}$$

で「プログラム実行前に条件 P **が満たされているとき，**M **を実行した結果を** υ **とすると，実行後に条件** Q **が成立する」ことを表す．**

これをファイブレーションを用いて以下のように表す．

$$\begin{tikzcd} \mathcal{E} \arrow[d, "p"'] & P \arrow[r] & Q \\ \mathcal{B} & \llbracket \tau_1 \rrbracket \times \cdots \times \llbracket \tau_n \rrbracket \arrow[r, "\llbracket M \rrbracket"] & \llbracket \tau \rrbracket \end{tikzcd}$$

プログラムのファイブレーション意味論

プログラム $x_1 : \tau_1, \ldots, x_n : \tau_n \vdash M : \tau$ に対して

$$x_1 : \tau_1, \ldots, x_n : \tau_n \, \{P\} \vdash M : \tau \{v.Q\}$$

で「プログラム実行前に条件 P が満たされているとき，M を実行した結果を v とすると，実行後に条件 Q が成立する」ことを表す．

これをファイブレーションを用いて以下のように表す．

$$\begin{array}{ccc} \mathcal{E} & \quad & P \longrightarrow Q \\ \downarrow p & & \\ \mathcal{B} & & [\![\tau_1]\!] \times \cdots \times [\![\tau_n]\!] \xrightarrow{[\![M]\!]} [\![\tau]\!] \end{array}$$

ファイブレーションを使ったプログラムの解釈の具体例

変数 x と y が $x = 0$ かつ $y = 2$ という条件を満たしているならば，プログラム $\mathsf{add}(x, y)$ を実行した結果の値は偶数である．

$$x : \mathsf{Int}, y : \mathsf{Int}\ \{x = 0 \wedge y = 2\}$$
$$\vdash \mathsf{add}(x, y) : \mathsf{Int}\ \{v.v \text{ is even}\}$$

この主張が正しいことは，プログラム $\mathsf{add}(x, y)$ の Set における解釈がファイブレーション p に沿って Pred に持ち上がることからわかる．

$$\begin{tikzcd}
\mathbf{Pred} \arrow[d, "p"'] & (\mathbb{N}^2, \{(0,2)\}) \arrow[r] & (\mathbb{N}, \{n \mid n \text{ is even}\}) \\
\mathbf{Set} & \mathbb{N}^2 \arrow[r, "{[\![\mathsf{add}(x,y)]\!]}"] & \mathbb{N}
\end{tikzcd}$$

ファイブレーションを使ったプログラムの解釈の具体例

変数 x と y が $x = 0$ かつ $y = 2$ という条件を満たしているならば，プログラム $\mathsf{add}(x, y)$ を実行した結果の値は偶数である．

$$x : \mathsf{Int}, y : \mathsf{Int}\ \{x = 0 \wedge y = 2\}$$
$$\vdash \mathsf{add}(x, y) : \mathsf{Int}\ \{v.v \text{ is even}\}$$

この主張が正しいことは，プログラム $\mathsf{add}(x, y)$ の Set における解釈がファイブレーション p に沿って Pred に持ち上がることからわかる．

$$\begin{array}{lccc} \textbf{Pred} & (\mathbb{N}^2, \{(0,2)\}) & \longrightarrow & (\mathbb{N}, \{n \mid n \text{ is even}\}) \\ \downarrow p & & & \\ \textbf{Set} & \mathbb{N}^2 & \xrightarrow{[\![\mathsf{add}(x,y)]\!]} & \mathbb{N} \end{array}$$

ファイブレーションを使ったプログラムの解釈の具体例

変数 x と y が $x = 0$ かつ $y = 2$ という条件を満たしているならば，プログラム $\mathsf{add}(x, y)$ を実行した結果の値は偶数である．

$$x : \mathsf{Int}, y : \mathsf{Int}\ \{x = 0 \land y = 2\}$$
$$\vdash \mathsf{add}(x, y) : \mathsf{Int}\ \{v. v \text{ is even}\}$$

この主張が正しいことは，プログラム $\mathsf{add}(x, y)$ の Set における解釈がファイブレーション p に沿って Pred に持ち上がることからわかる．

$$\begin{tikzcd}
\mathbf{Pred} \arrow[d, "p"'] & (\mathbb{N}^2, \{(0,2)\}) \arrow[r] & (\mathbb{N}, \{n \mid n \text{ is even}\}) \\
\mathbf{Set} & \mathbb{N}^2 \arrow[r, "{[\![\mathsf{add}(x,y)]\!]}"] & \mathbb{N}
\end{tikzcd}$$

逆像関手の役割

ファイブレーションの定義に，$u : J \to pX$ に対して $\overline{u}X : u^*X \to X$ があってカルテシアン射となる，というものがあった．

$$
\begin{tikzcd}
\mathcal{E} \arrow[d, "p"'] & u^*X \arrow[r, "\overline{u}X"] & X \\
\mathcal{B} & J \arrow[r, "u"] & pX
\end{tikzcd}
$$

プログラムのファイブレーション意味論の文脈では，u^*X はプログラムの最弱事前条件に対応している．

$$
\begin{tikzcd}
\mathcal{E} \arrow[d, "p"'] & \llbracket M \rrbracket^* Q \arrow[r, "\overline{\llbracket M \rrbracket}Q"] & Q \\
\mathcal{B} & \llbracket \Gamma \rrbracket \arrow[r, "\llbracket M \rrbracket"] & \llbracket \tau \rrbracket
\end{tikzcd}
$$

逆像関手の役割

ファイブレーションの定義に，$u : J \to pX$ に対して $\overline{u}X : u^*X \to X$ があってカルテシアン射となる，というものがあった．

$$\begin{tikzcd} \mathcal{E} \arrow[d, "p"'] \\ \mathcal{B} \end{tikzcd} \qquad \begin{tikzcd} u^*X \arrow[r, "\overline{u}X"] & X \\ J \arrow[r, "u"] & pX \end{tikzcd}$$

プログラムのファイブレーション意味論の文脈では，u^*X はプログラムの最弱事前条件に対応している．

$$\begin{tikzcd} \mathcal{E} \arrow[d, "p"'] \\ \mathcal{B} \end{tikzcd} \qquad \begin{tikzcd} [\![M]\!]^*Q \arrow[r, "\overline{[\![M]\!]}Q"] & Q \\ [\![\Gamma]\!] \arrow[r, "[\![M]\!]"] & [\![\tau]\!] \end{tikzcd}$$

逆像関手の役割

ファイブレーションの定義に，$u : J \to pX$ に対して $\overline{u}X : u^*X \to X$ があってカルテシアン射となる，というものがあった．

$$\begin{array}{ccc} \mathcal{E} & \quad & u^*X \xrightarrow{\overline{u}X} X \\ \big\downarrow p & & \\ \mathcal{B} & & J \xrightarrow{u} pX \end{array}$$

プログラムのファイブレーション意味論の文脈では，u^*X はプログラムの最弱事前条件に対応している．

$$\begin{array}{ccc} \mathcal{E} & \quad & [\![M]\!]^*Q \xrightarrow{\overline{[\![M]\!]}Q} Q \\ \big\downarrow p & & \\ \mathcal{B} & & [\![\Gamma]\!] \xrightarrow{[\![M]\!]} [\![\tau]\!] \end{array}$$

逆像関手の役割

ファイブレーションの定義に，$u : J \to pX$ に対して $\overline{u}X : u^*X \to X$ があってカルテシアン射となる，というものがあった．

$$\begin{array}{ccc} \mathcal{E} & \qquad & u^*X \xrightarrow{\overline{u}X} X \\ \downarrow{\scriptstyle p} & & \\ \mathcal{B} & & J \xrightarrow{u} pX \end{array}$$

プログラムのファイブレーション意味論の文脈では，u^*X はプログラムの最弱事前条件に対応している．

$$\begin{array}{ccc} \mathcal{E} & \qquad & [\![M]\!]^*Q \xrightarrow{\overline{[\![M]\!]}Q} Q \\ \downarrow{\scriptstyle p} & & \\ \mathcal{B} & & [\![\Gamma]\!] \xrightarrow{[\![M]\!]} [\![\tau]\!] \end{array}$$

逆像関手の役割を具体例で観察

W に入る**最も弱い**条件は何か．

$$x : \mathsf{Int}, y : \mathsf{Int}\,\{W\} \vdash \mathsf{add}(x, y) : \mathsf{Int}\,\{v.v \text{ is even}\}$$

「最も弱い」というのは

$$x : \mathsf{Int}, y : \mathsf{Int}\,\{P\} \vdash \mathsf{add}(x, y) : \mathsf{Int}\,\{v.v \text{ is even}\}$$

を満たす任意の条件 P に対して $P \Rightarrow W$ が成り立つこと．

$$\begin{tikzcd}
 & (\mathbb{N}^2, P) \arrow[d, dashed] \arrow[dr, bend left] & \\
\mathbf{Pred} \arrow[dd, "p"'] & (\mathbb{N}^2, W) \arrow[r] & (\mathbb{N}, \{n \mid n \text{ is even}\}) \\
 & & \\
\mathbf{Set} & \mathbb{N}^2 \arrow[r, "{[\![\mathsf{add}(x,y)]\!]}"] & \mathbb{N}
\end{tikzcd}$$

これは W が逆像の普遍性を持つということを言っている．

具体的な計算

$$x : \mathsf{Int}, y : \mathsf{Int}\,\{W\} \vdash \mathsf{add}(x, y) : \mathsf{Int}\,\{v.v \text{ is even}\}$$

$\{v.v \text{ is even}\}$ **に対するプログラム** $\mathsf{add}(x, y)$ **の最弱事前条件** W **は，**

$$\begin{aligned} &[\![\mathsf{add}(x, y)]\!]^*(\mathbb{N}, \{n \mid n \text{ is even}\}) \\ &= (\mathbb{N}^2, [\![\mathsf{add}(x, y)]\!]^{-1}\{n \mid n \text{ is even}\}) \\ &= (\mathbb{N}^2, \{(x, y) \mid x + y \text{ is even}\}) \end{aligned}$$

より「$x + y$ is even**」である．**

$$\begin{aligned} &x : \mathsf{Int}, y : \mathsf{Int}\,\{x = 0 \land y = 2\} \\ &\vdash \mathsf{add}(x, y) : \mathsf{Int}\,\{v.v \text{ is even}\} \end{aligned}$$

も成り立っていたが，このとき

$$x = 0 \land y = 2 \Rightarrow x + y \text{ is even}$$

が成り立っている．

具体的な計算

$$x : \mathsf{Int}, y : \mathsf{Int}\,\{W\} \vdash \mathsf{add}(x, y) : \mathsf{Int}\,\{v.v \text{ is even}\}$$

$\{v.v \text{ is even}\}$ に対するプログラム $\mathsf{add}(x, y)$ の最弱事前条件 W は，

$$\begin{aligned}
&[\![\mathsf{add}(x, y)]\!]^*(\mathbb{N}, \{n \mid n \text{ is even}\}) \\
&= (\mathbb{N}^2, [\![\mathsf{add}(x, y)]\!]^{-1}\{n \mid n \text{ is even}\}) \\
&= (\mathbb{N}^2, \{(x, y) \mid x + y \text{ is even}\})
\end{aligned}$$

より「$x + y$ is even」である．

$$\begin{aligned}
&x : \mathsf{Int}, y : \mathsf{Int}\,\{x = 0 \wedge y = 2\} \\
&\vdash \mathsf{add}(x, y) : \mathsf{Int}\,\{v.v \text{ is even}\}
\end{aligned}$$

も成り立っていたが，このとき

$$x = 0 \wedge y = 2 \Rightarrow x + y \text{ is even}$$

が成り立っている．

最近考えていること