

型付きλ計算

Jacques Garrigue, 2011年6月20日

λ計算の関数は数学的関数と違い、定義域と値域を定めていない。例えば、前に見た c_+ は整数以外の物に適用しても意味がないが、そのような項は正しくないと判断できない。

1 型 (type) と項

型付きλ計算では、そのような領域を**型**で表す。単純な計算系では全ての値は一つだけの型に属する。型には二種類があり、基底型と関数・構造型に分けられる。

$$\begin{aligned} b &::= \text{int} \mid \text{bool} \mid \dots \\ t &::= b \mid t \rightarrow t \mid t \times t \end{aligned}$$

項の中にも、型情報を入れ、定数も追加する。

$$M ::= x \mid c_t \mid \lambda x:t.M \mid (M M) \mid (M, M)$$

簡約規則には、 β に加えて定数に関する δ 規則も導入される。

$$\begin{aligned} (\lambda x:\tau.M) N &\rightarrow [N/x]M \\ (\text{fst}_{\tau \times \theta \rightarrow \tau} (M, N)) &\rightarrow M \\ (\text{snd}_{\tau \times \theta \rightarrow \theta} (M, N)) &\rightarrow N \\ (\text{s}_{\text{int} \rightarrow \text{int}} n_{\text{int}}) &\rightarrow (n + 1)_{\text{int}} \\ (\text{if0}_{\text{int} \rightarrow \tau \rightarrow \tau \rightarrow \tau} 0_{\text{int}} M N) &\rightarrow M \\ (\text{if0}_{\text{int} \rightarrow \tau \rightarrow \tau \rightarrow \tau} n_{\text{int}} M N) &\rightarrow N \\ \dots & \end{aligned}$$

上記の δ 規則では s は固定の型を持っているが、 fst , snd と if0 は様々な型について定義しなければならない。 τ や θ はそういう任意に決められる型を表している。

2 型推論

ある項の正しさ (well-formedness) を型判定式で表す。

$$\Gamma \vdash M : \tau$$

M は項であり、 τ は型である。そして Γ は型宣言列と呼ばれる変数名と型の集合 $(x_1 : \tau_1, \dots, x_n : \tau_n)$ 。

次の型推論規則で型判定式が証明された項を正しいとする。

| | |
|----|---|
| 変数 | $\Gamma \vdash x : \tau$ ($x : \tau$ は Γ に含まれる) |
| 定数 | $\Gamma \vdash c_\tau : \tau$ |
| 抽象 | $\frac{\Gamma, x : \theta \vdash M : \tau}{\Gamma \vdash \lambda x : \theta. M : \theta \rightarrow \tau}$ |
| 適用 | $\frac{\Gamma \vdash M : \theta \rightarrow \tau \quad \Gamma \vdash N : \theta}{\Gamma \vdash (M N) : \tau}$ |
| 直積 | $\frac{\Gamma \vdash M : \tau \quad \Gamma \vdash N : \theta}{\Gamma \vdash (M, N) : \tau \times \theta}$ |

証明の例

$$\frac{\frac{x : \text{int} \vdash s_{\text{int} \rightarrow \text{int}} : \text{int} \rightarrow \text{int} \quad x : \text{int} \vdash x : \text{int}}{x : \text{int} \vdash (s_{\text{int} \rightarrow \text{int}} x) : \text{int}}}{\frac{\vdash \lambda x : \text{int}. (s_{\text{int} \rightarrow \text{int}} x) : \text{int} \rightarrow \text{int} \quad \vdash 1_{\text{int}} : \text{int}}{\vdash ((\lambda x : \text{int}. (s_{\text{int} \rightarrow \text{int}} x)) 1_{\text{int}}) : \text{int}}}$$

2.1 性質

次の定理は fst, snd 以外の δ 規則を含まない計算系に関するものである。

定理 1 (主部簡約) $\Gamma \vdash M : \tau$ と $M \rightarrow N$ が成り立てば、 $\Gamma \vdash N : \tau$ が成り立つ。

定理 2 (停止性) $\Gamma \vdash M : \tau$ ならば、無限な簡約列 $(M \rightarrow M_1 \rightarrow M_2 \rightarrow \dots)$ は存在しない。

停止性が成り立てば、定義できないものがある。例えば、型のない λ 計算で停止しなかった $(\lambda x. x x)(\lambda x. x x)$ は定義できない。それを見せるために、 $\lambda x : \tau. x x$ の証明を書いてみよう。

$$\frac{\frac{x : \tau \vdash x : \tau \rightarrow \theta \quad x : \tau \vdash x : \tau}{x : \tau \vdash x x : \theta}}{\vdash \lambda x : \tau. x x : \tau \rightarrow \theta}$$

項の構造から、以上の形の証明しかできないが、条件として $\tau = (\tau \rightarrow \theta)$ を満たす必要がある。当然ながら、上記の型の定義では、こんな方程式を満たすような τ は存在しない。

同様に Y を導入すると、止らない計算ができるので、型付 λ 計算では Y は定義できない。

3 論理との関係

各推論規則の中で型のところだけを残せば、 λ 計算を作る動機となった直観主義論理の推論規則が得られる。定数は公理になる。

$$\begin{array}{c} [A]^{(n)} \\ \vdots \\ B \\ \supset \text{導入} \frac{A \supset B}{A \supset B}^{(n)} \end{array} \quad \supset \text{除去} \frac{A \supset B \quad A}{B}$$

$$\wedge \text{導入} \frac{A \quad B}{A \wedge B} \quad \wedge \text{除去} \frac{A \wedge B \supset A \quad A \wedge B \supset B}{A \wedge B \supset A \quad A \wedge B \supset B}$$

例えば、 $\lambda x : \tau \times \theta. (\text{snd } x, \text{fst } x)$ の証明は

$$\frac{\frac{\Gamma \vdash \text{snd} : \tau \times \theta \rightarrow \theta \quad \Gamma \vdash x : \tau \times \theta}{\Gamma \vdash (\text{snd } x) : \theta} \quad \frac{\vdash \text{fst} : \tau \times \theta \rightarrow \tau \quad \Gamma \vdash x : \tau \times \theta}{\Gamma \vdash (\text{fst } x) : \tau}}{\Gamma = x : \tau \times \theta \vdash (\text{snd } x, \text{fst } x) : \theta \times \tau}}{\vdash \lambda x : \tau \times \theta. (\text{snd } x, \text{fst } x) : \tau \times \theta \rightarrow \theta \times \tau}$$

を次の直観主義論理の証明に変えられる

$$\frac{\frac{A \wedge B \supset B \quad A \wedge B^{(1)}}{B} \quad \frac{A \wedge B \supset A \quad A \wedge B^{(1)}}{A}}{B \wedge A}^{(1)} \quad \frac{B \wedge A}{A \wedge B \supset B \wedge A}$$

しかも最初の証明は項だけから再構築できるので、ある項はその型の証明である、ともいう。

4 万能性

以上の δ 規則と停止性で分かるように、単体では型付 λ 計算は万能ではない。ただし、二種類の異なる問題がある。

整数などに関する δ 規則について、チャーチ数が型付けできないのは原因である。細かく言えば、型付けはできるが、汎用性が足りない。

$$\vdash \lambda f : \tau \rightarrow \tau. \lambda x : \tau. f^n x : (\tau \rightarrow \tau) \rightarrow \tau \rightarrow \tau$$

問題は、この関数はある特定の型 τ においてしか使えない。チャーチ数の計算は様々なものを x に代入することを利用していたので、これでは意味のあるエンコーディングはできない。この問題は、計算を δ 規則に移すことで解決される。

同様に、再帰的な計算をしたければ、 Y の δ 規則を入れる必要がある。

$$Y_{(\tau \rightarrow \tau) \rightarrow \tau} M \rightarrow M \quad (Y_{(\tau \rightarrow \tau) \rightarrow \tau} M)$$

整数と Y だけを加えれば、型付 λ 計算は万能になる。実行には δ 規則を使うか、定数を型のない λ 計算の項に変えて型のない λ 計算で実行するかのどちらもができる。

型体系を強くすれば、汎用性のある定義ができるようになり、チャーチ数の定義がそのまま使える。二次 λ 計算では型変数が用意されている。

$$\begin{aligned} t & ::= \dots \mid \tau \mid \forall \tau. t \\ M & ::= \dots \mid \Lambda \tau. M \mid M[t] \end{aligned}$$

例えば二次 λ 計算と言われる体系では、次のように表現する。

$$\begin{aligned} \vdash c_n &= \Lambda \tau. \lambda f : \tau \rightarrow \tau. \lambda x : \tau. f^n x : \forall \tau. (\tau \rightarrow \tau) \rightarrow \tau \rightarrow \tau \\ \vdash c_+ &= \lambda m : Int. \lambda n : Int. \Lambda \tau. \lambda f : \tau \rightarrow \tau. \lambda x : \tau. (m[\tau] x (n[\tau] f x)) : Int \rightarrow Int \rightarrow Int \\ \vdash c_\times &= \lambda m : Int. \lambda n : Int. \Lambda \tau. \lambda f : \tau \rightarrow \tau. (m[\tau] (n[\tau] f)) : Int \rightarrow Int \rightarrow Int \\ \vdash c_{\text{exp}} &= \lambda m : Int. \lambda n : Int. \Lambda \tau. n[\tau \rightarrow \tau] (m[\tau]) : Int \rightarrow Int \rightarrow Int \end{aligned}$$

ここでは $Int = \forall \tau. (\tau \rightarrow \tau) \rightarrow \tau \rightarrow \tau$ 。

ただし、二次 λ 計算は型付 λ 計算と同様に停止性を持っているので、相変わらず Y は定義できない。それは欠点とは限らない。全ての計算が止まるということは、全ての項に計算できる意味があるということでもある。

さらに進んで、再帰的な型を持った λ 計算を使えば、 Y も定義できるようになるが、停止性が失われる。