メタラムダ計算の定式化

理学専攻　情報科学コース　岩井亜里紗　(指導教員：浅井健一)

1　はじめに

メタラムダ計算 (Meta Lambda Calculus, MLC)[1] とは、通常のラムダ式に加えて、メタレベルのラムダ式を扱うことができる計算体系である。これはプログラミング言語の視点から捉えると、2 レベルの式を扱うことのできるステージ言語の一つと考えられる。先行研究として、[2] においてメタラムダ計算の圏論的意味付けが与えられている。この中で、紙上による健全性の証明は行われているが、本研究では定理証明系である Isabelle/HOL における証明を行うことにより、より確かな定式化を得たいと考えている。

2　メタラムダ計算

以下で、本研究で扱う MLC の型、構文、自由変数、変数の代入、型規則について述べる。

2.1　型

MLC の型は以下のように与えられる。

$$\begin{array}{llcl} \mathit{type} & \alpha & ::= & \mathit{Int} \mid \alpha \to \alpha \\ \mathit{meta\_type} & \tau & ::= & (\Gamma \vdash \alpha) \mid \tau \Rightarrow \tau \end{array}$$

MLC では、ベースレベルの項とメタレベルの項を扱うため、型も 2 レベルの形で与えられている。$\to$ と $\Rightarrow$ は、それぞれベースレベルとメタレベルの関数型を表わす。Γ は、ベースレベルのコンテキスト（ベースレベルの変数と型のペアの集合）であり、項 M が型 $(\Gamma \vdash \alpha)$ を持つというのは、コンテキスト Γ のもとで、M が α 型を持つという意味である。

2.2　構文

MLC の構文は、以下のように与えられる。

$$\begin{array}{lcl} \overline{X} & ::= & X \mid \overline{X}[\Lambda/x] \\ \Lambda & ::= & c \mid x \mid \lambda x.\,\Lambda \mid \Lambda\,\Lambda \mid \overline{X} \mid \zeta X.\,\Lambda \mid \Lambda \lightning \Lambda \end{array}$$

単純型付きラムダ計算 (Simply Typed Lambda Calculus, STLC) の構文に加え、メタレベルの構文が定義されている。メタレベルの項については、変数、関数抽象、関数適用を表わす構文を導入した。また、メタ変数に関しては、メタ変数に対する変数の代入も $\overline{X}[\Lambda/x]$ のように構文に導入されている。これは、X が具体的な項に置き換わった時点で、実際に変数の代入を行うためである。

2.3　自由変数

MLC の自由変数は、以下のように与えられる。

$$\begin{array}{rcl} fv(x) & = & \{x\} \\ fv(c) & = & \{\} \\ fv(\lambda x.\,M) & = & fv(M) - \{x\} \\ fv(M\,N) & = & fv(M) \cup fv(N) \\ fv(X) & = & G(X) \\ fv(\overline{X}[L/x]) & = & (fv(\overline{X}) - \{x\}) \cup fv(L) \\ fv(\zeta X.\,M) & = & fv(M) \\ fv(M \lightning N) & = & fv(M) \end{array}$$

$$\begin{array}{rcl} mfv(x) & = & \{\} \\ mfv(c) & = & \{\} \\ mfv(\lambda x.\,M) & = & mfv(M) \\ mfv(M\,N) & = & mfv(M) \cup mfv(N) \\ mfv(X) & = & \{X\} \\ mfv(\overline{X}[L/x]) & = & mfv(\overline{X}) \cup mfv(L) \\ mfv(\zeta X.\,M) & = & mfv(M) - \{X\} \\ mfv(M \lightning N) & = & mfv(M) \cup mfv(N) \end{array}$$

関数 G は、各メタ変数に自由変数の集合を割り当てる関数である。MLC では、この関数が与えられているものと考える。また、MLC の構文が相互再帰的に定義されていることから、ベースレベル、メタレベルともに自由変数の定義についても実際は相互再帰的に定義している。

これに付随して、各項に対してある変数 x が項の自由変数の集合に現れないことを示す、fresh の定義も行った。

Definition 1 (fresh variables)

$$\mathit{fresh}\ x\ M \ = \ x \notin fv(M)$$

Definition 2 (fresh meta-variables)

$$\mathit{meta\_fresh}\ X\ M \ = \ X \notin mfv(M)$$

2.4　変数の代入

MLC におけるベースレベルの変数の代入は、以下のように与えられる。ただし、$x \neq y$ とする。

$$\begin{array}{rcl} c[L/x] & \equiv & c \\ x[L/x] & \equiv & L \\ y[L/x] & \equiv & y \\ (\lambda x.\,M)[L/x] & \equiv & \lambda x.\,M \\ (\lambda y.\,M)[L/x] & \equiv & \lambda y.\,(M[L/x]) \\ & & \quad *\text{ただし、}\mathit{fresh}\ y\ L \\ (M\,N)[L/x] & \equiv & (M[L/x])\,(N[L/x]) \\ X[L/x] & \equiv & X[L/x] \\ (X[M/y])[L/x] & \equiv & X[M/y][L/x] \\ (\zeta X.\,M)[L/x] & \equiv & \zeta X.\,(M[L/x]) \\ & & \quad *\text{ただし、}\mathit{meta\_fresh}\ X\ L \\ (M \lightning N)[L/x] & \equiv & (M[L/x]) \lightning N \end{array}$$

2.5　型規則

MLC の judgment は以下のような形で表わされる。

$$\Delta \Vdash M : \tau$$

ここで、Δ は、メタレベルのコンテキスト（メタレベルの変数と型のペアの集合）、M は項、τ は、メタレベルの型を表わす。

実際に使用される型の推論規則は以下の通りである。

$$\frac{}{\Delta \Vdash c : (\Gamma \vdash Int)}\ (CON)$$

$$\frac{}{\Delta \Vdash x : (x : \alpha, \Gamma \vdash \alpha)}\ (VAR)$$

$$\frac{\Delta \Vdash M : (x : \alpha_1, \Gamma \vdash \alpha_2)}{\Delta \Vdash \lambda x.\, M : (\Gamma \vdash \alpha_1 \to \alpha_2)}\ (LAM)$$

$$\frac{\Delta \Vdash M : (\Gamma \vdash \alpha_1 \to \alpha_2) \quad \Delta \Vdash N : (\Gamma \vdash \alpha_1)}{\Delta \Vdash M\, N : (\Gamma \vdash \alpha_2)}\ (APP)$$

$$\frac{}{X : \tau, \Delta \Vdash X : \tau}\ (MVAR1)$$

$$\frac{\Delta \Vdash X : \tau_1 \quad x \notin fv(X) \quad weaken\ x\ \tau_1\ \tau_2}{\Delta \Vdash X : \tau_2}\ (MVAR2)$$

$$\frac{X : \tau_1, \Delta \Vdash M : \tau_2}{\Delta \Vdash \zeta X.\, M : \tau_1 \Rightarrow \tau_2}\ (MLAM)$$

$$\frac{\Delta \Vdash M : \tau_1 \Rightarrow \tau_2 \quad \Delta \Vdash N : \tau_1}{\Delta \Vdash M \between N : \tau_2}\ (MAPP)$$

$$\frac{\Delta \Vdash \overline{X} : \tau_1 \quad \Delta \Vdash M : \tau_2 \quad sub\_rel\ x\ \tau_1\ \tau_2\ \tau_3}{\Delta \Vdash \overline{X}[M/x] : \tau_3}\ (SUB)$$

(SUB) の型推論に含まれる $sub\_rel\ x\ \tau_1\ \tau_2\ \tau_3$ は、メタレベルの型が二通りあるために導入したものである。具体的には、以下のように定義している。

$$\frac{}{sub\_rel\ x\ (x : \alpha_1, \Gamma \vdash \alpha_2)\ (\Gamma \vdash \alpha_1)\ (\Gamma \vdash \alpha_2)}$$

$$\frac{sub\_rel\ x\ \tau_1\ \tau_2\ \tau_3}{sub\_rel\ x\ (\tau \Rightarrow \tau_1)\ \tau_2\ (\tau \Rightarrow \tau_3)}$$

また、$weaken\ x\ \tau\ \tau'$ とあるのは、型 τ をベースレベルの変数 x で weakening した結果が τ' になるという意味である。

3 Isabelle/HOL

Isabelle/HOL は、高階論理に基づく定理証明系である。Isabelle で使用される記述言語を Isar と言い、人が書く証明に近いスタイルの証明を記述することができる。また Nominal Isabelle という、Isabelle/HOL の拡張で、Nominal Logic に基づく定理証明系も存在する。これは、α 変換などの問題を上手く解決できるもので、STLC などはこの Nominal Isabelle を用いると比較的楽に証明することができる。本研究でも当初、こちらの Nominal Isabelle による証明を行っていたが、メタラムダ計算では通常のラムダ計算と自由変数の定義が異なるため、使用できないことがわかった。具体的な例として、MLC の式とその変換を提示する。

$$\begin{aligned} (\zeta X.\, (\lambda x.\, X)\, c) \between x &=_{M\beta} ((\lambda x.\, X)\, c)[x/X] \\ &= (\lambda x.\, x)\, c \\ &=_{\beta} x[c/x] \\ &= c \end{aligned}$$

このように、項 $(\zeta X.\, (\lambda x.\, X)\, c) \between x$ は定数 c へと変換されるが、メタレベルの関数適用の引数 x は、関数部分に含まれる λx に束縛されている。つまりこの場合、x はこの式の自由変数に含まれず、Nominal Isabelle を使用すると問題が生じてしまう。以上のような理由により、Nominal の機能を含まない Isabelle による証明へと変更した。

4 Isabelle による定式化

初めに、ベースレベルの α、β、η 変換の健全性を示すために必要な代入補題の証明を行った。

Lemma1 (Base-level Substitution Lemma)

$$\frac{\Gamma \Vdash M : \tau_1 \quad \Gamma \Vdash N : \tau_2 \quad sub\_rel\ x\ \tau_1\ \tau_2\ \tau_3}{\Gamma \Vdash M[N/x] : \tau_3}$$

Proof. 項 M に関する帰納法より。

また、代入補題の証明に必要な、メタレベルのコンテキストにおける weakening の証明も行った。

Lemma2 (Meta-level Weakening)

$$\frac{\Gamma \Vdash M : \tau_1 \quad meta\_fresh\, X\, M}{X : \tau_2, \Gamma \Vdash M : \tau_1}$$

Proof. 項 M に関する帰納法と、型の推論木に関する帰納法より。

ただし、MLC の構文がメタ変数とその他の項とで相互再帰的に定義されているため、提示したいずれの補題も相互再帰の形をとっている。

5 まとめと今後の課題

本研究では、ステージ言語の一つであるメタラムダ計算の定式化に向けて、構文などの各種定義とベースレベルの代入補題の証明、そしてメタレベルの weakening の証明を、Isabelle を用いて行った。残りのベースレベルの weakening、α、β、η 変換の証明、そしてメタレベルについても同様に証明を行い、定式化を完成させることが今後の課題である。また、[2] における証明と今回 Isabelle 上にて行った証明との違いについても考察したい。

参考文献

[1] Bekki, D. ”Monads and meta-lambda calculus”, In Hattori, H., editor, New Frontiers in Artificial Intelligence (JSAI 2008 Conference and Workshops, Asahikawa, Japan, June 2008, Revised Selected Papers), volume LNAI 5447, pp.193-208, Springer, 2009.

[2] 増子萌. 戸次大介. ”メタラムダ計算の圏論的意味論”, 第 13 回プログラミングおよびプログラミング言語ワークショップ (PPL2011), pp.60-74, 2011.