

# 工学部専門科目「計算と論理」配布資料

## 多相ラムダ計算 (System F)

五十嵐 淳

京都大学 大学院情報学研究科 通信情報システム専攻

igarashi@kuis.kyoto-u.ac.jp

November 19, 2013

### 1 単純型付ラムダ計算 (補足・訂正)

(再帰) 関数, 自然数に関する場合分けの型付け規則において, 関数の仮引数など新たに型環境に追加される変数の名前がその外側で既に宣言されている (つまり型環境に既に現れている) 変数の名前と衝突しないように条件が必要であった. つまり T-FUN 規則は

$$\frac{\Gamma, x : S \vdash M : T \quad (x \notin \text{dom}(\Gamma))}{\Gamma \vdash \text{fun } x : S \Rightarrow M : S \rightarrow T} \quad (\text{T-FUN})$$

となる. ここで  $\text{dom}(\Gamma)$  は  $\Gamma$  中のコロンの左側に現れる変数からなる集合である.

すなわち,  $M \stackrel{\text{def}}{=} \text{fun } x : \text{nat} \Rightarrow \text{fun } x : \text{bool} \Rightarrow x$  に対し,

$$\frac{\frac{\frac{}{x : \text{nat}, x : \text{bool} \vdash x : \text{bool}} \text{T-VAR}}{x : \text{nat} \vdash \text{fun } x : \text{bool} \Rightarrow x : \text{bool} \rightarrow \text{bool}} \text{T-ABS}}{\vdash M : \text{nat} \rightarrow \text{bool} \rightarrow \text{bool}} \text{T-ABS}}$$

や, ましてや,

$$\frac{\frac{\frac{}{x : \text{nat}, x : \text{bool} \vdash x : \text{nat}} \text{T-VAR}}{x : \text{nat} \vdash \text{fun } x : \text{bool} \Rightarrow x : \text{bool} \rightarrow \text{nat}} \text{T-ABS}}{\vdash M : \text{nat} \rightarrow \text{bool} \rightarrow \text{nat}} \text{T-ABS}}$$

といった導出木を作ることはいできない. ただし, 引数の名前替えをして,

$$N \stackrel{\text{def}}{=} \text{fun } x : \text{nat} \Rightarrow \text{fun } y : \text{bool} \Rightarrow y$$

とすれば,

$$\frac{\frac{\frac{}{x : \text{nat}, y : \text{bool} \vdash y : \text{bool}} \text{T-VAR}}{x : \text{nat} \vdash \text{fun } y : \text{bool} \Rightarrow y : \text{bool} \rightarrow \text{bool}} \text{T-ABS}}{\vdash N : \text{nat} \rightarrow \text{bool} \rightarrow \text{bool}} \text{T-ABS}}$$

という関係を導出することができる. (ただし, 引数の名前替えをした項同士は同一視するのが慣習なので,  $M$  と  $N$  をわざわざ区別する意味はここではあまりない.)

## 2 単純型付ラムダ計算+自然数リスト

### 2.1 構文

(types)  $S, T ::= \dots$   
          | natlist

(terms)  $M, N ::= \dots$   
          | nil  
          | cons  
          | match  $M$  with nil =>  $N_1$  | cons  $x y$  =>  $N_2$  end

### 2.2 簡約

match nil with nil =>  $N_1$  | cons  $x y$  =>  $N_2$  end  $\longrightarrow N_1$  (R-MATCHN)

match cons  $M_1 M_2$  with nil =>  $N_1$  | cons  $x y$  =>  $N_2[x, y]$  end  $\longrightarrow N_2[M_1, M_2]$   
(R-MATCHC)

$$\frac{M \longrightarrow M'}{\text{match } M \text{ with nil } \Rightarrow N_1 \mid \text{cons } x y \Rightarrow N_2 \text{ end} \longrightarrow \text{match } M' \text{ with nil } \Rightarrow N_1 \mid \text{cons } x y \Rightarrow N_2 \text{ end}}$$
 (RC-MATCHL1)

$$\frac{N_1 \longrightarrow N'_1}{\text{match } M \text{ with nil } \Rightarrow N_1 \mid \text{cons } x y \Rightarrow N_2 \text{ end} \longrightarrow \text{match } M \text{ with nil } \Rightarrow N'_1 \mid \text{cons } x y \Rightarrow N_2 \text{ end}}$$
 (RC-MATCHL2)

$$\frac{N_2 \longrightarrow N'_2}{\text{match } M \text{ with nil } \Rightarrow N_1 \mid \text{cons } x y \Rightarrow N_2 \text{ end} \longrightarrow \text{match } M \text{ with nil } \Rightarrow N_1 \mid \text{cons } x y \Rightarrow N'_2 \text{ end}}$$
 (RC-MATCHL3)

### 2.3 型付け規則

$$\Gamma \vdash \text{nil} : \text{natlist} \quad (\text{T-NIL})$$

$$\Gamma \vdash \text{cons} : \text{nat} \rightarrow \text{natlist} \rightarrow \text{natlist} \quad (\text{T-CONS})$$

$$\frac{\Gamma \vdash M : \text{natlist} \quad \Gamma \vdash N_1 : T \quad \Gamma, x : \text{nat}, y : \text{natlist} \vdash N_2 : T}{\Gamma \vdash \text{match } M \text{ with nil } \Rightarrow N_1 \mid \text{cons } x y \Rightarrow N_2 \text{ end} : T} \quad (\text{T-MATCHL})$$

### 3 System F: 多相ラムダ計算

教科書 Poly.v で見たような多相関数 (*polymorphic function*) (型引数を取り, 引数によって型が変化する関数) によって単純型付ラムダ計算を拡張した計算体系が多相ラムダ計算 (*polymorphic  $\lambda$ -calculus*) である. 多相ラムダ計算は System F と呼ばれている.<sup>1</sup>

System F の (単純型付ラムダ計算に対しての) 本質的な拡張は項 (プログラム) への多相関数<sup>2</sup>

$$\text{fun } X : \text{Type} \Rightarrow M$$

と型適用  $M T$  の導入, そして多相型  $\forall X : \text{Type}, T$  の導入である.

(ふつうの) 関数の  $\beta$  簡約と同様に型抽象が型に適用された項は簡約によって計算が進む.

$$(\text{fun } X : \text{Type} \Rightarrow M[X]) T \longrightarrow M[T]$$

関数本体中の (型の) 仮引数  $X$  が実引数  $T$  で置き換わるところは  $\beta$  簡約と全く同じである. 例えば, 与えられた関数  $f$  を  $x$  に二回適用する多相高階関数

$$M \stackrel{\text{def}}{=} \text{fun } X : \text{Type} \Rightarrow \text{fun } f : X \rightarrow X \Rightarrow \text{fun } x : X \Rightarrow f (f x)$$

について,

$$M \text{ nat} \longrightarrow \text{fun } f : \text{nat} \rightarrow \text{nat} \Rightarrow \text{fun } x : \text{nat} \Rightarrow f (f x)$$

$$M \text{ bool} \longrightarrow \text{fun } f : \text{bool} \rightarrow \text{bool} \Rightarrow \text{fun } x : \text{bool} \Rightarrow f (f x)$$

となり, (機能は同じだが) 自然数, 真偽値に特化した関数が得られる.

型付け関係は単純型付ラムダ計算と同様に

$$\Gamma \vdash M : T$$

という形で表されるが, 型環境には変数の型宣言  $x : T$  だけでなく, 型変数の宣言  $X : \text{Type}$  も並ぶことになる. また (ふつうの) 関数や関数適用のための規則は変わらない.

多相関数についての型付け規則は

$$\frac{\Gamma, X : \text{Type} \vdash M : T \quad X \notin \text{dom}(\Gamma)}{\Gamma \vdash \text{fun } X : \text{Type} \Rightarrow M : \forall X : \text{Type}, T} \quad (\text{T-TFUN})$$

と与えられる. 関数全体に与えられる型の違いはあるが, T-FUN 規則とよく似ている. この  $X$  は後で多相関数が型引数に適用された時に具体的な型に置き換えられるものである. 実際, 型適用の型付け規則は

$$\frac{\Gamma \vdash M : \forall X : \text{Type}, T[X]}{\Gamma \vdash M S : T[S]} \quad (\text{T-TAPP})$$

<sup>1</sup>アメリカの計算機科学者ジョン・レイノルズ (John Reynolds) とフランスの論理学者ジャン・イヴ・ジラルール (Jean-Yves Girard) が 1970 年代始めに独立に考え出した. System F の名前はジラルールによる.

<sup>2</sup>型抽象とも呼ぶ

のようになり，多相関数型  $\forall X : \text{Type}, T[X]$  を持つ多相関数  $M$  型引数  $S$  を与えると， $X$  を  $S$  に置き換えたような型（すなわち  $T[S]$ ）が得られる．

例えば，上で定義した二回適用する関数の例は以下のように型付け関係を導出することができる．

$$\begin{array}{c}
M \stackrel{\text{def}}{=} \text{fun } X : \text{Type} \Rightarrow \text{fun } f : X \rightarrow X \Rightarrow \text{fun } x : X \Rightarrow f (f x) \\
\Gamma \stackrel{\text{def}}{=} X : \text{Type}, f : X \rightarrow X, x : X \rightarrow X \\
\\
\frac{\frac{\frac{\Gamma \vdash f : X \rightarrow X}{\Gamma \vdash f : X \rightarrow X} \text{T-VAR} \quad \frac{\Gamma \vdash f : X \rightarrow X \quad \Gamma \vdash x : X}{\Gamma \vdash f x : X} \text{T-APP} \quad \frac{\Gamma \vdash f : X \rightarrow X \quad \Gamma \vdash f x : X}{\Gamma \vdash f (f x) : X} \text{T-APP}}{\frac{\Gamma \vdash f : X \rightarrow X \quad \Gamma \vdash f (f x) : X}{X : \text{Type}, f : X \rightarrow X \vdash \text{fun } x : X \Rightarrow f (f x) : X \rightarrow X} \text{T-FUN}}{\frac{X : \text{Type} \vdash \text{fun } f : X \rightarrow X \Rightarrow \text{fun } x : X \Rightarrow f (f x) : (X \rightarrow X) \rightarrow X \rightarrow X}{\vdash M : \forall X : \text{Type}, (X \rightarrow X) \rightarrow X \rightarrow X} \text{T-TFUN}} \text{T-TAPP} \\
\frac{\vdash M : \forall X : \text{Type}, (X \rightarrow X) \rightarrow X \rightarrow X}{\vdash M \text{ nat} : (\text{nat} \rightarrow \text{nat}) \rightarrow \text{nat} \rightarrow \text{nat}} \text{T-TAPP}
\end{array}$$

以下に（自然数も真偽値も再帰関数も含まない）純粋な System F の定義をまとめる．

### 3.1 型，ラムダ項，型環境，判断の構文

$$\begin{array}{l}
\text{(types)} \quad S, T ::= X \\
\quad \quad \quad | S \rightarrow T \\
\quad \quad \quad | \forall X : \text{Type}, T \\
\\
\text{(terms)} \quad M, N ::= x \\
\quad \quad \quad | \text{fun } x : T \Rightarrow M \\
\quad \quad \quad | M_1 M_2 \\
\quad \quad \quad | \text{fun } X : \text{Type} \Rightarrow M \\
\quad \quad \quad | M T \\
\\
\text{(type environments)} \quad \Gamma ::= \bullet \\
\quad \quad \quad | \Gamma, x : T \\
\quad \quad \quad | \Gamma, X : \text{Type}
\end{array}$$

### 3.2 簡約

$$\begin{array}{l}
(\text{fun } x : T \Rightarrow M[x]) N \longrightarrow M[N] \quad \text{(R-BETA)} \\
(\text{fun } X : \text{Type} \Rightarrow M[X]) T \longrightarrow M[T] \quad \text{(R-TBETA)} \\
\frac{M \longrightarrow M'}{\text{fun } x : T \Rightarrow M \longrightarrow \text{fun } x : T \Rightarrow M'} \quad \text{(RC-FUN)}
\end{array}$$

$$\frac{M \longrightarrow M'}{M N \longrightarrow M' N} \quad (\text{RC-APP1})$$

$$\frac{N \longrightarrow N'}{M N \longrightarrow M N'} \quad (\text{RC-APP2})$$

$$\frac{M \longrightarrow M'}{\text{fun } X : \text{Type} \Rightarrow M \longrightarrow \text{fun } X : \text{Type} \Rightarrow M'} \quad (\text{RC-TFUN})$$

$$\frac{M \longrightarrow M'}{M T \longrightarrow M' T} \quad (\text{RC-TAPP})$$

### 3.3 型付け規則

$$\frac{(x : T \in \Gamma)}{\Gamma \vdash x : T} \quad (\text{T-VAR})$$

$$\frac{\Gamma, x : S \vdash M : T}{\Gamma \vdash \text{fun } x : S \Rightarrow M : S \rightarrow T} \quad (\text{T-FUN})$$

$$\frac{\Gamma \vdash M : S \rightarrow T \quad \Gamma \vdash N : S}{\Gamma \vdash M N : T} \quad (\text{T-APP})$$

$$\frac{\Gamma, X : \text{Type} \vdash M : T \quad X \notin \text{dom}(\Gamma)}{\Gamma \vdash \text{fun } X : \text{Type} \Rightarrow M : \forall X : \text{Type}, T} \quad (\text{T-TFUN})$$

$$\frac{\Gamma \vdash M : \forall X : \text{Type}, S[X]}{\Gamma \vdash M T : S[T]} \quad (\text{T-TAPP})$$

### 3.4 System F の重要な性質

定理 1 (合流性) 任意の項  $M_1, M_2, M_3$  に対して,  $M_1 \longrightarrow^* M_2$  かつ  $M_1 \longrightarrow^* M_3$  ならば, ある  $M_4$  が存在し,  $M_2 \longrightarrow^* M_4$  かつ  $M_3 \longrightarrow^* M_4$  が成り立つ.

定理 2 (型保存定理)  $\Gamma \vdash M : T$  かつ,  $M \longrightarrow M'$  ならば,  $\Gamma \vdash M' : T$  である.

項  $M$  が値である, とは,  $M$  が  $S(\dots(0)\dots)$ ,  $\text{true}$ ,  $\text{false}$ ,  $\text{fun } x : T \Rightarrow M$ ,  $\text{fun } X : \text{Type} \Rightarrow M$  もしくは,  $\text{fix } x (y : S) : T := M$  いずれかの形をしていることをいう.

定理 3 (前進性)  $\vdash M : T$  ならば,  $M$  は何らかの値であるか,  $M \longrightarrow M'$  なる  $M'$  が存在する.

定理 4 (強正規化性)  $M$  には  $\text{fix}$  が現れず, かつ,  $\Gamma \vdash M : T$  とする  $M \longrightarrow M_1 \longrightarrow M_2 \longrightarrow \dots$  なる項の無限列  $M_1, M_2, \dots$  は存在しない.

## 4 System F + 自然数, 真偽値, リスト, 再帰関数

(types)  $S, T ::= \dots$

- | nat
- | bool
- | list  $T$

(terms)  $M, N ::= \dots$

- | 0
- | S
- | match  $M$  with 0 =>  $N_1$  | S  $x$  =>  $N_2$  end
- | true
- | false
- | if  $M$  then  $N_1$  else  $N_2$
- | nil
- | cons
- | match  $M$  with nil =>  $N_1$  | cons  $x$   $y$  =>  $N_2$
- | fix  $x$  ( $y : S$ ) :  $T := M$

$\Gamma ::= \dots$

### 4.1 簡約

match 0 with 0 =>  $N_1$  | S  $x$  =>  $N_2$  end  $\longrightarrow N_1$  (R-MATCHZ)

match S  $M$  with 0 =>  $N_1$  | S  $x$  =>  $N_2[x]$  end  $\longrightarrow N_2[M]$  (R-MATCHS)

if true then  $N_1$  else  $N_2$   $\longrightarrow N_1$  (R-IFT)

if false then  $N_1$  else  $N_2$   $\longrightarrow N_2$  (R-IFF)

match nil  $T$  with nil =>  $N_1$  | cons  $x$   $y$  =>  $N_2[x, y]$   $\longrightarrow N_1$  (R-MATCHN)

match cons  $T$   $M_1$   $M_2$  with nil =>  $N_1$  | cons  $x$   $y$  =>  $N_2[x, y]$   $\longrightarrow N_2[M_1, M_2]$   
(R-MATCHC)

(fix  $x$  ( $y : S$ ) :  $T := M[x, y]$ )  $N \longrightarrow M[\text{fix } x \text{ ( $y : S$ ) : } T := M[x, y], N]$  (R-FIX)

$$\frac{M \longrightarrow M'}{\text{match } M \text{ with } 0 \Rightarrow N_1 \mid S \ x \Rightarrow N_2 \text{ end} \longrightarrow \text{match } M' \text{ with } 0 \Rightarrow N_1 \mid S \ x \Rightarrow N_2 \text{ end}}$$
 (RC-MATCH1)

$$\frac{N_1 \longrightarrow N'_1}{\text{match } M \text{ with } 0 \Rightarrow N_1 \mid S \ x \Rightarrow N_2 \text{ end} \longrightarrow \text{match } M \text{ with } 0 \Rightarrow N'_1 \mid S \ x \Rightarrow N_2 \text{ end}}$$
 (RC-MATCH2)

$\frac{N_2 \longrightarrow N'_2}{\text{match } M \text{ with } 0 \Rightarrow N_1 \mid S \ x \Rightarrow N_2 \text{ end} \longrightarrow \text{match } M \text{ with } 0 \Rightarrow N_1 \mid S \ x \Rightarrow N'_2 \text{ end}}$	(RC-MATCH3)
$\frac{M \longrightarrow M'}{\text{if } M \text{ then } N_1 \text{ else } N_2 \longrightarrow \text{if } M' \text{ then } N_1 \text{ else } N_2}$	(RC-IF1)
$\frac{N_1 \longrightarrow N'_1}{\text{if } M \text{ then } N_1 \text{ else } N_2 \longrightarrow \text{if } M \text{ then } N'_1 \text{ else } N_2}$	(RC-IF2)
$\frac{N_2 \longrightarrow N'_2}{\text{if } M \text{ then } N_1 \text{ else } N_2 \longrightarrow \text{if } M \text{ then } N_1 \text{ else } N'_2}$	(RC-IF3)
$\frac{M \longrightarrow M'}{\text{fun } x : T \Rightarrow M \longrightarrow \text{fun } x : T \Rightarrow M'}$	(RC-FUN)
$\frac{M \longrightarrow M'}{\text{fix } x \ (y : S) : T := M' \longrightarrow \text{fix } x \ (y : S) : T := M'}$	(RC-FIX)

## 4.2 型付け規則

$\frac{}{\Gamma \vdash 0 : \text{nat}}$	(T-ZERO)
$\frac{}{\Gamma \vdash S : \text{nat} \rightarrow \text{nat}}$	(T-SUCC)
$\frac{\Gamma \vdash M : \text{nat} \quad \Gamma \vdash N_1 : T \quad \Gamma, x : \text{nat} \vdash N_2 : T}{\Gamma \vdash \text{match } M \text{ with } 0 \Rightarrow N_1 \mid S \ x \Rightarrow N_2 \text{ end} : T}$	(T-MATCH)
$\frac{}{\Gamma \vdash \text{true} : \text{bool}}$	(T-TRUE)
$\frac{}{\Gamma \vdash \text{false} : \text{bool}}$	(T-FALSE)
$\frac{\Gamma \vdash M : \text{bool} \quad \Gamma \vdash N_1 : T \quad \Gamma \vdash N_2 : T}{\Gamma \vdash \text{if } M \text{ then } N_1 \text{ else } N_2 : T}$	(T-IF)
$\frac{\Gamma, x : S \rightarrow T, y : S \vdash M : T}{\Gamma \vdash \text{fix } x \ (y : S) : T := M : S \rightarrow T}$	(T-FIX)
$\frac{}{\Gamma \vdash \text{nil} : \forall X : \text{Type}, \text{list } X}$	(T-NIL)

$$\frac{}{\Gamma \vdash \text{cons} : \forall X : \text{Type}, X \rightarrow \text{list } X \rightarrow \text{list } X} \quad (\text{T-CONS})$$
$$\frac{\Gamma \vdash M : \text{list } S \quad \Gamma \vdash N_1 : T \quad \Gamma, x : S, y : \text{list } S \vdash N_2 : T}{\Gamma \vdash \text{match } M \text{ with nil} \Rightarrow N_1 \mid \text{cons } x \ y \Rightarrow N_2 : T} \quad (\text{T-MATCHL})$$