ラムダ計算の型問題について[1]

– 数学基礎論からプログラミング言語の構造へ –

藤田 憲悦 (群馬大学)*

平成24年3月28日

1. はじめに

ラムダ計算 [7] は，Turing 機械と同様に万能な計算モデルである [27]．ラムダ計算は ”関数 ”の一般的な理論であり，関数適用と関数抽象の二つの操作よりラムダ式は構成される．そして，集合の概念に基づくグラフとしての関数ではなく，関数の値を求めていく過程を計算として扱う理論となっている．その背景にある思想は創造当時の Church 自身による次の文章に表れている．

『Underlying the formal calculi which we shall develop is *the concept iof a function*, $\cdots$. A function is *a rule of correspondence* by which when anything is given (as argument) another thing (the value of the function for that argument) may be obtained. That is, a function is an operation $\cdots$. In particular it is not excluded that *one of the elements of the range of arguments of a function f should be the function f itself.* This possibility has frequently been denied, $\cdots$. Here, however, we regard the operation or rule of correspondence, which constitutes the function, as being first given, $\cdots$.』

(Church [9], Chapter I より引用：*Italic face* はこの拙著による)

この理論は，現在では「型のないラムダ計算」と呼ばれている．そして，計算理論のみならず関数型プログラミング言語などの基礎の一つになっている．

一方,「型付きラムダ計算」の体系もある．ここで，型は ”関数 ”の定義域や値域に相当する概念を形式化したものである．型付きラムダ計算も数学・計算の基礎付けにとどまらず，プログラミング言語や定理証明器の基礎理論として研究されてきた．ラムダ計算の研究は，構文論と数学的モデルを扱う意味論とに大別される．構文論の研究の中でも，”関数の計算 ”的観点から，正規化定理，合流性 (Church-Rosser の定理)，Subject reduction 性など動的性質に関する研究と，型検査，型推論，証明可能性 (Inhabitation) など静的性質に関する研究に分類されるであろう．特に，型付きラムダ計算の静的性質は，型付きプログラミング言語や定理証明システムにおける ”計算 ”と独立な性質の解析にとって重要である．さらに，論理式 A はラムダ計算の型と解釈（同一視）でき，論理式 A の証明は型 A を持つラムダ式と解釈され得る [19]．この解釈は，Curry-Howard 同型 (または，Propositions-as-types，proofs-as-terms など)[16] と呼ばれ，論理とラムダ計算との橋渡しをしている．

キーワード： 2nd-order λ-calculus, type-checking, type-inference

* 〒376-8515 群馬県桐生市天神町 1-5-1 群馬大学 大学院工学研究科
e-mail: fujita@cs.gunma-u.ac.jp
web: http://www.cs.gunma-u.ac.jp/~fujita/

[1] 本稿は，2012 年度日本数学会年会特別講演（数学基礎論および歴史）の講演アブストラクトの前半部分を書き直したものです.

型付きラムダ計算の体系には，Curry 流 [10] と Church 流 [8] の 2 種類がある．Church 流のラムダ式は，その中に型の情報がコメントされた式になっている．これによって，Church 流のラムダ式の型検査問題が容易になる．さらに，Church 流のラムダ式が型を持つならばその型は唯一つ存在することになり，ラムダ式は証明記述言語に適している．他方，Curry 流のラムダ式は型情報を全く含まず，型のないラムダ式と同じである．これにより，型から解放されてラムダ式（プログラム）を書くことができる．そして，式にきちんと型が付くかどうかの判断は，別途定義される型割り当てアルゴリズムに任せることになる．

ラムダ計算の型体系が 1 階の体系 (simply typed lambda-calculus) の場合，Curry 流と Church 流のスタイルの違いは型検査・型推論問題の可解性に影響を及ぼさない．しかし，型体系を 2 階の体系（2 階直観主義論理）に拡張すると大きな違いが生じてくる．例えば，Church 流の型検査問題は決定可能であるが，Curry 流のその問題は決定不能 (Wells (1999) [28]) となってしまう．さらに，Curry 流と Church 流の中間的な構造を持つ式やそれらを組み合わせた型問題 [5] なども考えることができる．このような中間構造を持つラムダ式の型問題を統一的な枠組みの中で扱った研究として Fujita-Schubert (2000) [13] がある．その研究動機は，2 階ラムダ計算の型問題の観点から決定可能・決定不能の境界構造を解明することにあった．そこでは，中間構造の一つであるドメインフリー (domain-free) ラムダ式の型検査・型推論問題 [2] が決定不能であることが示された．最近になり，もう一つの中間構造であるタイプフリー (type-free) ラムダ式の型問題 [3] も決定不能であることが明らかになった [15].

本稿では，これまで得られた結果を踏まえて，2 階ラムダ計算の型問題，特に，型検査，型推論問題に関連する結果を紹介する．2 節では，2 階ラムダ計算の定義を，Church 流，Curry 流を含む五つのスタイルに対して与える．そして，型問題の定義，及び基本命題について述べる．3.1 節では，ドメインフリー・ラムダ式の型問題が決定不能であることを示す．決定不能である単一化問題が型推論問題へ還元されることを概観する．3.2 節では，タイプフリー・ラムダ式の型問題が決定不能である結果を述べる．3.3 節では，ホールアプリケーション・ラムダ式の型推論問題を解くアルゴリズムを与える．そして，型検査・型推論問題が決定可能であることを示す．4 節では，以上の結果をまとめる．型検査・型推論問題の決定不能性は，関数抽象 $(\lambda x\!:\!A.M)$ におけるドメインの型 A の削除が本質的な要因になっていることを検証する．さらに，型の複雑さである型ランクの観点から型問題の可解性を分類する．

なお，ラムダ計算の全般については，例えば，Barendregt [1, 2]，Barendregt-Dekkers-Statman [3]，高橋 [18] などが詳しい．また，ラムダ計算 (Church) とコンビネーター (Curry) の歴史については，Cardone-Hindley [6]，Seldin [17] などが興味深い．

2. ラムダ式のスタイルと 2 階ラムダ計算 $\lambda 2$

最初に，単純型付きラムダ計算 $\lambda^{\to}$ の型とラムダ式の構文を導入する．ここで，型は直観主義命題論理の含意 $(\to)$ 断片である．また，ラムダ式は，変数，関数抽象，関数適用の 3 種類から次のように構成される．

定義 1 ($\lambda^{\to}$ **：型，Church 流ラムダ式と Curry 流ラムダ式)**

---

[2] この問題は Barth-Sørensen の論文 (1997) [4] で未解決問題として提示されていた．

[3] この問題は Pfenning の論文 (1993) [25] で未解決のまま残されていた．

- 型

$$A ::= X \mid (A \to A)$$

- Church 流ラムダ式

$$M ::= x \mid (\lambda x\!:\!A.M) \mid (MM)$$

- Curry 流ラムダ式

$$M ::= x \mid (\lambda x.M) \mid (MM)$$

Church 流のラムダ式の呼び方は，Barendregt[2]，Seldin[17] に従っている．ただし，Church のオリジナル [8] では，きちんと型の付く式だけが well-formed formulas として次のように定義されている．

1. 全ての変数には型が一意に割り当てられている．
2. 変数 x_β は型 β を持つ式であり，M_α は型 α を持つ式ならば，$(\lambda x_\beta M_\alpha)$ は型 $(\beta \to \alpha)$ を持つ式である．
3. 式 $M_{(\beta\to\alpha)}$ は型 $(\beta \to \alpha)$ を持ち，N_β は型 β を持つ式ならば，$(M_{(\beta\to\alpha)}N_\beta)$ は型 α を持つ式である．

最近の文献 Barendregt-Dekkers-Statman[3] では，オリジナルの Church 流の式と区別するために，本稿で Church 流と定義しているラムダ式は de Bruijn 流の pseudo-term と呼ばれている．また，pseudo-terms や pre-terms を最初に定義して，その中できちんと型の付く式だけを Church 流のラムダ式と定義することもある．しかし，本稿では定義 1 をもって Church 流のラムダ式を導入する．

次に，2 階ラムダ計算の体系 λ2 を導入する．ここで，型は 2 階直観主義命題論理の $(\to, \forall)$ 断片である．文脈とは，変数から型への写像である．また，Church 流の 2 階のラムダ式は，変数，変数に関する関数抽象，関数適用，型変数に関する関数抽象，及び型への関数適用の 5 種類から構成される．

定義 2 (λ2：**2 階の型，文脈，Church 流ラムダ式と Curry 流ラムダ式**)

- 型

$$A ::= X \mid (A \to A) \mid \forall X.A$$

- 文脈

$$\Gamma ::= \langle\ \rangle \mid x\!:\!A, \Gamma$$

- Church 流ラムダ式

$$M ::= x \mid (\lambda x\!:\!A.M) \mid (MM) \mid (\Lambda X.M) \mid (M[A])$$

Church 流の式に対する推論規則を定義する．ここで，$\Gamma \vdash_{\mathrm{Ch}} M : A$ は，“文脈 Γ のもとで Church 流の式 M が型 A を持つ” と読む．

$$\frac{}{\Gamma, x\!:\!A \vdash_{\mathrm{Ch}} x : A}\ (\mathrm{var})$$

$$\frac{\Gamma, x\!:\!A_1 \vdash_{\mathrm{Ch}} M : A_2}{\Gamma \vdash_{\mathrm{Ch}} \lambda x\!:\!A_1.M : A_1 \to A_2}\ (\to I) \qquad \frac{\Gamma \vdash_{\mathrm{Ch}} M_1 : A_1 \to A_2 \quad \Gamma \vdash_{\mathrm{Ch}} M_2 : A_1}{\Gamma \vdash_{\mathrm{Ch}} M_1 M_2 : A_2}\ (\to E)$$

$$\frac{\Gamma \vdash_{\mathrm{Ch}} M : A}{\Gamma \vdash_{\mathrm{Ch}} \Lambda X.M : \forall X.A}\ (\forall I)^\star \qquad \frac{\Gamma \vdash_{\mathrm{Ch}} M : \forall X.A}{\Gamma \vdash_{\mathrm{Ch}} M[A_1] : A[X := A_1]}\ (\forall E)$$

但し，$(\forall I)^\star$ は変数条件 $X \notin \mathrm{FV}(\Gamma)$ が満たされていることを指す.

- Curry 流ラムダ式

$$M ::= x \mid (\lambda x.M) \mid (MM)$$

Curry 流の式に対する推論規則を次に定義する.

$$\frac{}{\Gamma, x\!:\!A \vdash_{\mathrm{Cu}} x : A}\ (\mathrm{var})$$

$$\frac{\Gamma, x\!:\!A_1 \vdash_{\mathrm{Cu}} M : A_2}{\Gamma \vdash_{\mathrm{Cu}} \lambda x.M : A_1 \to A_2}\ (\to I) \qquad \frac{\Gamma \vdash_{\mathrm{Cu}} M_1 : A_1 \to A_2 \quad \Gamma \vdash_{\mathrm{Cu}} M_2 : A_1}{\Gamma \vdash_{\mathrm{Cu}} M_1 M_2 : A_2}\ (\to E)$$

$$\frac{\Gamma \vdash_{\mathrm{Cu}} M : A}{\Gamma \vdash_{\mathrm{Cu}} M : \forall X.A}\ (\forall I)^\star \qquad \frac{\Gamma \vdash_{\mathrm{Cu}} M : \forall X.A}{\Gamma \vdash_{\mathrm{Cu}} M : A[X := A_1]}\ (\forall E)$$

但し，$(\forall I)^\star$ は変数条件 $X \notin \mathrm{FV}(\Gamma)$ が満たされていることを指す.

次に，Church 流と Curry 流の中間的構造を持つ式，domain-free，hole-application，及び type-free のラムダ式を導入する．そして，Church，Curry，domain-free，hole-application，type-free をラムダ式のスタイルと呼び，それぞれ，$\mathrm{Ch}, \mathrm{Cu}, \mathrm{Df}, [\,], \mathrm{Tf}$ と書く．残りのスタイル s を持つ式についても，$\Gamma \vdash_s M : A$ に対して推論規則が同様に定義されるがここでは省略する.

定義 3 (Domain-free，hole-application，type-free スタイル)

- Domain-free ラムダ式

$$M ::= x \mid (\lambda x.M) \mid (MM) \mid (\Lambda X.M) \mid (M[A])$$

- Hole-application ラムダ式

$$M ::= x \mid (\lambda x\!:\!A.M) \mid (MM) \mid (\Lambda X.M) \mid (M[\,])$$

- Type-free ラムダ式

$$M ::= x \mid (\lambda x.M) \mid (MM) \mid (\Lambda.M) \mid (M[\,])$$

Curry-Howard 同型のもとで，きちんと型の付くラムダ式は直観主義論理の証明図のコードになっている．Church 流のラムダ式を例にとると，変数は推論規則 (var) が使用されていることを，そして関数抽象の式は含意の導入規則 $(\to I)$ が使用されていることを意味している．また，関数適用の式は $(\to E)$，型変数の抽象は $(\forall I)$，そして型の適用は $(\forall E)$ の推論規則がそれぞれ使用されていることを指している．ここで，ラムダ式に含まれている証明図の情報は，次の三つに分類される．(i) どの推論規則 (what) を，(ii) どのように (how, which instance)，(iii) どこで (where) 適用しているか．そして，これら 3 種類の情報を捨象することによって，Church 流から Curry 流のラムダ式に段階的に移行できる．これに型問題を重ねると，Church 流のラムダ式の型検査・型推論問題は決定可能であるが，Curry 流のその問題は決定不能である．三つの中間構造を全て混合した式の型推論問題は一般的には決定不能 [25] であるが，Church 流と Curry 流の間に決定可能な構造が他にも存在する可能性は否めない．そこで,「ラムダ式の中のどの情報が，型問題の可解性・非可解性に本質的に寄与しているか」という問題について議論する．

定義 4 (Erasure mapping) 五つのスタイルに次の順序を定義する．

$$\mathrm{Cu} < \mathrm{Tf} < \mathrm{Df} < \mathrm{Ch}, \quad \mathrm{Tf} < [] < \mathrm{Ch}$$

スタイル $s, t \in \{\mathrm{Cu}, \mathrm{Tf}, [], \mathrm{Df}, \mathrm{Ch}\}$ に対して $s < t$ であるとき，t-スタイルのラムダ式を s-スタイルのラムダ式に変換する写像 $|\cdot|_s^t$ を定義する．

- $|x|_{\mathrm{Df}}^{\mathrm{Ch}} = x$, $|\lambda x{:}A.M|_{\mathrm{Df}}^{\mathrm{Ch}} = \lambda x.|M|_{\mathrm{Df}}^{\mathrm{Ch}}$, $|M_1 M_2|_{\mathrm{Df}}^{\mathrm{Ch}} = |M_1|_{\mathrm{Df}}^{\mathrm{Ch}} |M_2|_{\mathrm{Df}}^{\mathrm{Ch}}$,
  $|\Lambda X.M|_{\mathrm{Df}}^{\mathrm{Ch}} = \Lambda X.|M|_{\mathrm{Df}}^{\mathrm{Ch}}$, $|M[A]|_{\mathrm{Df}}^{\mathrm{Ch}} = |M|_{\mathrm{Df}}^{\mathrm{Ch}}[A]$;
- $|\Lambda X.M|_{\mathrm{Tf}}^{\mathrm{Df}} = \Lambda.|M|_{\mathrm{Tf}}^{\mathrm{Df}}$, $|M[A]|_{\mathrm{Tf}}^{\mathrm{Df}} = |M|_{\mathrm{Tf}}^{\mathrm{Df}}[]$;
- $|x|_{\mathrm{Cu}}^{\mathrm{Tf}} = x$, $|\lambda x.M|_{\mathrm{Cu}}^{\mathrm{Tf}} = \lambda x.|M|_{\mathrm{Cu}}^{\mathrm{Tf}}$, $|M_1 M_2|_{\mathrm{Cu}}^{\mathrm{Tf}} = |M_1|_{\mathrm{Cu}}^{\mathrm{Tf}} |M_2|_{\mathrm{Cu}}^{\mathrm{Tf}}$,
  $|\Lambda.M|_{\mathrm{Cu}}^{\mathrm{Tf}} = |M|_{\mathrm{Cu}}^{\mathrm{Tf}}$, $|M[]|_{\mathrm{Cu}}^{\mathrm{Tf}} = |M|_{\mathrm{Cu}}^{\mathrm{Tf}}$, 以下同様に定義する．

命題 1 (Church スタイルにおける型の一意性) $\Gamma \vdash_{\mathrm{Ch}} M : A_1$ かつ $\Gamma \vdash_{\mathrm{Ch}} M : A_2$ ならば $A_1 = A_2$ である．

命題 2 (Erasure and lifting) スタイル $s, t \in \{\mathrm{Cu}, \mathrm{Tf}, [], \mathrm{Df}, \mathrm{Ch}\}$ に対して $s < t$ とする．

1. $\Gamma \vdash_t M : A$ ならば $\Gamma \vdash_s |M|_s^t : A$ である．
2. $\Gamma \vdash_s M : A$ ならば，t-スタイルのラムダ式 N が存在して $|N|_s^t = M$ かつ $\Gamma \vdash_t N : A$ である．

命題 3 (Generation lemma) 各スタイル $s \in \{\mathrm{Tf}, \mathrm{Df}, [], \mathrm{Ch}\}$ に対して次が成立する．

1. $\Gamma \vdash_s x : A$ ならば $\Gamma(x) = A$ である．
2. $\Gamma \vdash_s \lambda x : A_0.M : A_1$ ならば，ある型 A_2 が存在して $\Gamma, x : A_0 \vdash_s M : A_2$ かつ $A_1 = (A_0 \to A_2)$ である．但し，$s \geq []$ である．

3. $\Gamma \vdash_s \lambda x.M : A_1$ ならば，ある型 A_0, A_2 が存在して $\Gamma, x : A_0 \vdash_s M : A_2$ かつ $A_1 = (A_0 \to A_2)$ である．但し，$s \leq \mathrm{Df}$ である．

4. $\Gamma \vdash_s M_1 M_2 : A_1$ ならば，ある型 A_0 が存在して $\Gamma \vdash_s M_1 : A_0 \to A_1$ かつ $\Gamma \vdash_s M_2 : A_0$ である．

5. $\Gamma \vdash_s \Lambda X.M : A_1$ ならば，ある型 A_2 が存在して $\Gamma \vdash_s M : A_2$ かつ $A_1 = \forall X.A_2$ である．但し，$X \notin \mathrm{FV}(\Gamma)$ かつ $s > \mathrm{Tf}$ である．

6. $\Gamma \vdash_{\mathrm{Tf}} \Lambda.M : A_1$ ならば，ある型 A_2 が存在して $\Gamma \vdash_{\mathrm{Tf}} M : A_2$ かつ $A_1 = \forall X.A_2$ である．但し，$X \notin \mathrm{FV}(\Gamma)$ である．

7. $\Gamma \vdash_s M[A] : A_1$ ならば，ある型 A_2 が存在して $\Gamma \vdash_s M : \forall X.A_2$ かつ $A_1 = A_2[X := A]$ である．ただし，$s \geq \mathrm{Df}$ である．

8. $\Gamma \vdash_s M[] : A_1$ ならば，ある型 A, A_2 が存在して $\Gamma \vdash_s M : \forall X.A_2$ かつ $A_1 = A_2[X := A]$ である．ただし，$s \leq []$ である．

定義 5 **(スタイルでパラメタ化された型問題)** 各スタイル $s \in \{\mathrm{Cu}, \mathrm{Tf}, [], \mathrm{Df}, \mathrm{Ch}\}$ における型問題を定義する．

1. s-スタイル式の型検査問題 ($\Gamma \vdash_s M : A?$ or $\mathrm{TCP}(s)$):

   文脈 Γ，s-スタイルのラムダ式 M，型 A が与えられたとき，$\Gamma \vdash_s M : A$ かどうかを決定せよ．

2. s-スタイル式の型推論問題 ($\Gamma \vdash_s M :?$ or TIP(s)):

   文脈 Γ，s-スタイルのラムダ式 M が与えられたとき，$\Gamma \vdash_s M : A$ を満たす型 A が存在するかどうかを決定せよ．

3. s-スタイル式の強型推論問題 ($?, \Gamma \vdash_s M :?$ or STIP(s)):

   文脈 Γ，s-スタイルのラムダ式 M が与えられたとき，$\Sigma, \Gamma \vdash_s M : A$ を満たす型 A と文脈 Σ が存在するかどうかを決定せよ．

4. s-スタイル式の型付け問題 ($? \vdash_s M :?$ or TP(s)):

   s-スタイルのラムダ式 M が与えられたとき，$\Gamma \vdash_s M : A$ を満たす型 A と文脈 Γ が存在するかどうかを決定せよ．

5. s-スタイル式による証明問題 ($\vdash ? : A$ or IHP(s)):

   型 A が与えられたときに，$\vdash_s M : A$ を満たす s-スタイルの閉ラムダ式 M が存在するかどうかを決定せよ．

命題 4 **(型問題間の還元関係)**

1. $\mathrm{TCP}(s) \hookrightarrow \mathrm{TIP}(s)$，但し $s \in \{\mathrm{Cu}, \mathrm{Tf}, \mathrm{Df}, [], \mathrm{Ch}\}$.

2. $\mathrm{TIP}(s) \hookrightarrow \mathrm{TCP}(s)$，但し $s \in \{\mathrm{Cu}, \mathrm{Tf}, \mathrm{Df}\}$.

3. $\mathrm{TIP}(s) \hookrightarrow \mathrm{TCP}(s)$，但し $s \in \{\mathrm{Tf}, []\}$.

4. $\mathrm{TIP}(s) \hookrightarrow \mathrm{TP}(s)$，但し $s \in \{[], \mathrm{Ch}\}$.
5. $\mathrm{TIP}(\mathrm{Df}) \hookrightarrow \mathrm{TP}(\mathrm{Df})$
6. $\mathrm{TP}(s) \hookrightarrow \mathrm{TIP}(s)$，但し $s \in \{\mathrm{Cu}, \mathrm{Tf}, \mathrm{Df}\}$.
7. $\mathrm{STIP}(s) \hookrightarrow \mathrm{TIP}(s)$，但し $s \in \{\mathrm{Cu}, \mathrm{Tf}, \mathrm{Df}\}$.
8. $\mathrm{STIP}(s) \hookrightarrow \mathrm{TP}(s)$，但し $s \in \{[], \mathrm{Ch}\}$.
9. $\mathrm{TIP}(s) \hookrightarrow \mathrm{STIP}(s)$，但し $s \in \{\mathrm{Cu}, \mathrm{Tf}, \mathrm{Df}, [], \mathrm{Ch}\}$.
10. $\mathrm{IHP}(s) \longleftrightarrow \mathrm{IHP}(t)$，但し $s, t \in \{\mathrm{Cu}, \mathrm{Tf}, \mathrm{Df}, [], \mathrm{Ch}\}$.

証明.

1. $\Gamma \vdash_s M : A \Longleftrightarrow \Gamma, z{:}A \to Z \vdash_s zM : B$ を満たす型 B が存在する.
2. $\Gamma \vdash_s M : B$ を満たす型 B が存在する $\Longleftrightarrow \Gamma, z{:}Z \vdash_s (\lambda v.z)M : Z$.
3. $\Gamma \vdash_s M : B$ を満たす型 B が存在する $\Longleftrightarrow \Gamma, z{:}\forall X.(X \to Z) \vdash_s z[]M : Z$.
4. $\Gamma = \{a_1 : A_1, \ldots, a_n : A_n\}$ とする．$\Gamma \vdash_s M : B$ を満たす型 B が存在する
   $\Longleftrightarrow \Sigma \vdash_s z(\lambda a_1{:}A_n \ldots \lambda a_n{:}A_n.M) : B$ を満たす型 B と文脈 Σ が存在する.
5. $\Gamma = \{a_1 : A_1, \ldots, a_n : A_n\}$ とする．$\Gamma \vdash_s M : B$ を満たす型 B が存在する
   $\Longleftrightarrow \Sigma \vdash_s M_0 : B$ を満たす型 B と文脈 Σ が存在する．但し，
   $M_0 = z_0(z_1(z[\forall X.X]))(z_1 z)(z[(A_1 \to \cdots \to A_n \to Y) \to Y](\lambda a_1 \ldots \lambda a_n.yM))$
   である.

   $\because$ $(\Longrightarrow)$ 変数 z の型を $\forall X.X$ とすると，明らかに成立する.
   $(\Longleftarrow)$ 式 M_0 が型付け可能であると仮定する．M_0 の定義より，z の型は2階の型であるから $\forall X.\mathsf{F}(X)$ とおく．ここで，F は引数1の関数変数である．M_0 中に z_1 の出現が2箇所あり，それらに矛盾なく型が付くことから，F に関する単一化方程式 $\mathsf{F}(\forall X.X) \doteq \forall X.\mathsf{F}(X)$ が解をもつ．ここで，高階単一化の(半)手続き[4]により，この方程式は唯一の解 $[\mathsf{F} := \lambda x.x]$，即ち恒等関数をもつ．従って，$z$ の型は $\forall X.X$ となり，M_0 の型付け可能性から文脈 Γ を求めることができた.
6. $\mathrm{FV}(M) = \{x_1, \ldots, x_n\}$ とする.
   $\Sigma \vdash_s M : B$ を満たす型 B と文脈 Σ が存在する
   $\Longleftrightarrow\ \vdash_s \lambda x_1 \ldots \lambda x_n.M : B$ を満たす型 B が存在する.
7. 上の場合6と同様.　8. 上の場合4と同様.
9. $\Gamma \vdash M : A$ を満たす型 A が存在する
   $\Longleftrightarrow \Gamma, z{:}B \vdash zM : A$ を満たす型 A, B が存在する
10. 命題2 (erasure and lifting) による. □

[4] 例えば，Dowek[11] に解説がある.

3. 2階ラムダ計算の型問題

3.1. Domain-free $\lambda 2$ と ML

Domain-free $\lambda 2$ の型推論問題 TIP(Df) が決定不能であることを示す．そのためにまず，叙述的な断片である domain-free ML を導入する．次に，domain-free ML の強型付け問題が決定不能であることを示す．そして，domina-free $\lambda 2$ が domain-free ML の保存的拡大であることより，TIP(Df) の決定不能性が従う．

定義 6 (Domain-free ML)

- 単相型　$\tau ::= X \mid (\tau \to \tau)$
- 多相型　$\sigma ::= \tau \mid \forall X.\sigma$
- 文脈　$\Gamma ::= \langle\ \rangle \mid x : \sigma, \Gamma$
- Domain-free ML 式　$M ::= x \mid (\lambda x.M) \mid (MM) \mid (x[\tau_1]\cdots[\tau_n])$
- Domain-free ML の推論規則

$$\frac{\Gamma(x) = \forall X_1 \cdots X_n.\tau \quad (n \geq 0)}{\Gamma \vdash_{\mathrm{dfML}} x[\tau_1]\cdots[\tau_n] : \tau[X_1 := \tau_1, \ldots, X_n := \tau_n]}\ (\mathrm{var})$$

$$\frac{\Gamma, x : \tau_1 \vdash_{\mathrm{dfML}} M : \tau_2}{\Gamma \vdash_{\mathrm{dfML}} \lambda x.M : \tau_1 \to \tau_2}\ (\to I) \qquad \frac{\Gamma \vdash_{\mathrm{dfML}} M_1 : \tau_1 \to \tau_2 \quad \Gamma \vdash_{\mathrm{dfML}} M_2 : \tau_1}{\Gamma \vdash_{\mathrm{dfML}} M_1 M_2 : \tau_2}\ (\to E)$$

次に，2階単一化問題 (simple instances) を定義する．この問題は関数変数を含む2階の単一化問題であるが，関数変数の引数部分には単一化子が作用する変数を含まないように制限された問題である．これは，決定不能性がいえるぎりぎりの線まで単純化された単一化問題と考えられる．制限されたこの問題の採用により，domain-free $\lambda 2$ の単純な断片に対する型検査問題でさえ決定不能となることがわかる．

定義 7 (2階単一化問題 (simple instances))

- 2階単一化問題 (simple instances) の表現は，n-引数の関数変数 $\mathsf{F}^{(n)}$ $(n \geq 0)$，及び2-引数の定数 $\to$ から次のように構成される．ここで，τ_i は単相型である．$\mathsf{F}^{(0)}$ は型変数とし，単に X と書く．

$$T ::= \mathsf{F}^{(n)}\tau_1 \cdots \tau_n \mid (T \to T)$$

- 2階単一化問題の表現に含まれている変数の集合を定義する．
  $\mathrm{Var}(\mathsf{F}^{(n)}\tau_1 \cdots \tau_n) = \{\mathsf{F}^{(n)}\}$,　$\mathrm{Var}(T_1 \to T_2) = \mathrm{Var}(T_1) \cup \mathrm{Var}(T_2)$
- T_1，T_2 は2階単一化問題の表現であり，関数変数の集合を $\{\mathsf{F}_1^{(k_1)}, \ldots, \mathsf{F}_n^{(k_n)}\} = \{\mathsf{F}_i^{(k_i)} \in \mathrm{Var}(T_1, T_2) \mid k_i \geq 1\}$ とする．2階単一化問題のインスタンス $T_1 \doteq T_2$ が解を持つとは，$S(T_1) = S(T_2)$ となる代入 S

$$S = [\mathsf{F}_1^{(k_1)} := \lambda X_1 \ldots X_{k_1}.\tau_1, \ldots, \mathsf{F}_n^{(k_n)} := \lambda X_1 \ldots X_{k_n}.\tau_n]$$

  が存在することである．ここで，各 τ_i は単相型である．

定理 1 (Schubert [26]) 2階単一化問題(simple instances)は決定不能である．

2階単一化問題(simple instances)からdomain-free MLの強型付け問題を構築する．

定義 8 (2階単一化問題のコード化) 2階単一化問題(simple instances)のインスタンスを$T_1 \doteq T_2$とする．ここで，代入のもとで定まる文脈$\Sigma(T_1, T_2)$とラムダ式$M_{T_{1,2}}$を以下に定義する．以下において，$\Sigma(T_1, T_2)$を単にΣと書くことがある．

- $\Sigma(T_1, T_2) \stackrel{\text{def}}{=} \Sigma_0 \cup \Sigma_1$，但し

  $\Sigma_0 = \{x : X \mid X \in \mathrm{Var}(T_1, T_2), i.e., X = \mathsf{F}^{(0)}\}$,

  $\Sigma_1 = \{x_{\mathsf{F}} : \forall X_1 \ldots X_n.\mathsf{F}X_1 \ldots X_n \mid \mathsf{F}^{(n)} \in \mathrm{Var}(T_1, T_2) \text{ for } n \geq 1\}$.

- $M_{T_{1,2}} \stackrel{\text{def}}{=} \lambda z_1.\lambda z_2.\lambda f.f z_1(f z_2(\lambda g.g [\![T_1]\!]^{z_1}_{\Sigma} [\![T_2]\!]^{z_2}_{\Sigma}))$，ここで $[\![T]\!]^{z}_{\Sigma}$ は次に定義する．

  1. $T = \mathsf{F}^{(n)}\tau_1 \ldots \tau_n$の場合 $(n \geq 0)$
     - $[\![\mathsf{F}\tau_1 \ldots \tau_n]\!]^{z}_{\Sigma} = \lambda f.f z(f(x_{\mathsf{F}}[\tau_1] \ldots [\tau_n])(\lambda g.g))$
       但し $\Sigma(x_{\mathsf{F}}) = \forall X_1 \ldots X_n.\mathsf{F}X_1 \ldots X_n$,
     - $(\!|\, \mathsf{F}\tau_1 \ldots \tau_n \,|\!) = ((\mathsf{F}\tau_1 \ldots \tau_n) \to (\alpha \to \alpha) \to \alpha \to \alpha) \to \alpha \to \alpha$
       ここで α は新しい型変数；
  2. $T = (T_1 \to T_2)$の場合
     - $[\![T_1 \to T_2]\!]^{z}_{\Sigma} = \lambda z_1.\lambda z_2.\lambda f.f(z z_1)(f z_2(\lambda g.g([\![T_1]\!]^{z_1}_{\Sigma})([\![T_2]\!]^{z_2}_{\Sigma})))$,
     - $(\!|\, T_1 \to T_2 \,|\!) = T_1 \to T_2 \to (T_2 \to A \to A) \to A$
       ここで $A = ((\!|\, T_1 \,|\!) \to (\!|\, T_2 \,|\!) \to \alpha) \to \alpha$，但し α は新しい型変数.

上の定義より，インスタンスから構成したコード$M_{T_{1,2}}$は正規形(β-normal)である．

補題 1 1. 任意の代入Sに対して，$S(\Sigma), z : S(T) \vdash_{\mathrm{dfML}} [\![T]\!]^{z}_{\Sigma} : S(\!|\, T \,|\!)$が成立する．

2. $S(\Sigma), z : \tau_1 \vdash_{\mathrm{dfML}} [\![T]\!]^{z}_{\Sigma} : \tau$ ならば $\tau_1 = S(T)$ かつ $\tau = S(\!|\, T \,|\!)$ である．

3. $S(T_1) = S(T_2) \iff S(\Sigma(T_1, T_2)) \vdash_{\mathrm{dfML}} M_{T_{1,2}} : S(\!|\, T_1 \to T_2 \,|\!)$

命題 5 (2UNIF$\hookrightarrow$STIP(dfML)) インスタンス$T_1 \doteq T_2$が解を持つ
$\iff \Sigma_0, \Gamma \vdash_{\mathrm{dfML}} M_{T_{1,2}} : \tau$ を満たす型τと文脈Γが存在する．

命題 6 (保存的拡大) ラムダ式 M_{nf} を domain-free ML の正規形とする．また，Γ をdomain-free MLの文脈とする．$\Gamma \vdash_{\mathrm{dfML}} M_{\mathrm{nf}} : \tau \iff \Gamma \vdash_{\mathrm{Df\lambda 2}} M_{\mathrm{nf}} : \tau$.

定義 9 (型のランク) ランクkを持つ型の集合$R(k)$を次のように定義する．
$R(0) ::= \tau$ (単相型)
$R(k+1) ::= R(k) \mid R(k) \to R(k+1) \mid \forall X.R(k+1)$

定理 2 (Domain-free $\lambda 2$の非可解性[13]) 1. TCP(Df), TIP(Df), 及びTP(Df)は全て等価であり決定不能な問題である．

2. 特に，STIP(Df)，及びTIP(Df)は型のランク2で決定不能である．TCP(Df)は型のランク3で決定不能である．

証明．命題4より，STIP(dfML) が TIP(Df) に還元できることを示せば十分である．

2階単一化問題 (simple instances) のインスタンス $T_1 \doteq T_2$ が解を持つ

$\iff \Sigma_0, \Gamma \vdash_{\mathrm{dfML}} M_{T_{1,2}} : \tau$ を満たす型 τ と文脈 Γ が存在する (STIP(dfML)) $\because$ 命題9

$\iff \Sigma_0, \Gamma \vdash_{\mathrm{Df\lambda 2}} M_{T_{1,2}} : \tau$ を満たす型 $\tau \in R(0)$ と文脈 $\Gamma \in R(1)$ が存在する (STIP(Df)) $\because$ 命題10

$\iff \Sigma_0 \vdash_{\mathrm{Df\lambda 2}} \lambda\vec{x}.M_{T_{1,2}} : \sigma_1 \to \cdots \to \sigma_n \to \tau$ を満たす型 $\sigma_1, \ldots, \sigma_n \in R(1), \tau \in R(0)$ が存在する (TIP(Df)) $\because$ 命題3

ここで，$(\sigma_1 \to \cdots \to \sigma_n \to \tau) \in R(2)$ である．次に，TIP(Df)$\hookrightarrow$TCP(Df) を示す．

$\Gamma \vdash_{\mathrm{Df\lambda 2}} M : \rho$ を満たす型 $\rho \in R(2)$ が存在する

$\iff \Gamma, z\!:\!Z \vdash_{\mathrm{Df\lambda 2}} (\lambda v.z)M : Z$，ここで $\lambda v.z : (\rho \to Z) \in R(3)$ である． □

3.2. Type-free $\lambda 2$

Type-free $\lambda 2$ の型問題，TCP(Tf)，TIP(Tf)，TP(Tf) もそれぞれ決定不能となる [15]．

定理 3 (Type-free $\lambda 2$ の非可解性 [15]) Type-free $\lambda 2$ に対して，TCP(Tf)，TIP(Tf)，TP(Tf) は全て決定不能な問題である．

2階単一化問題 (simple instances) の変種である flat form と呼ばれる2階単一化問題を導入して [14]，その決定不能性より帰結される．

3.3. Hole-application $\lambda 2$

Hole-application $\lambda 2$ の推論規則をまとめる．

$$\frac{}{\Gamma \vdash_{\mathrm{hole}} x : \Gamma(x)}\ (\mathrm{var})$$

$$\frac{\Gamma, x\!:\!A_1 \vdash_{\mathrm{hole}} M : A_2}{\Gamma \vdash_{\mathrm{hole}} \lambda x\!:\!A_1.M : A_1 \to A_2}\ (\to I) \qquad \frac{\Gamma \vdash_{\mathrm{hole}} M_1 : A_1 \to A_2 \quad \Gamma \vdash_{\mathrm{hole}} M_2 : A_1}{\Gamma \vdash_{\mathrm{hole}} M_1 M_2 : A_2}\ (\to E)$$

$$\frac{\Gamma \vdash_{\mathrm{hole}} M : A}{\Gamma \vdash_{\mathrm{hole}} \Lambda X.M : \forall X.A}\ (\forall I)^* \qquad \frac{\Gamma \vdash_{\mathrm{hole}} M : \forall X.A}{\Gamma \vdash_{\mathrm{hole}} M[] : A[X := B]}\ (\forall E)$$

定理 4 (Hole-application $\lambda 2$) Hole-application $\lambda 2$ の TP([]) は決定不能である．

これは，文献 [15] で使われた技法が TP([]) にも適用されることによる．

次に，命題4より，型検査問題 TCP([]) と型推論問題 TIP([]) は等価である．文脈 Γ のもとで hole-application ラムダ式 M の型を計算するアルゴリズム $\mathsf{type}(\Gamma; M)$ を与える．そして，このアルゴリズムの健全性と完全性を示す．

定義 10 (型推論アルゴリズム type)

1. $\mathsf{type}(\Gamma; x) = \Gamma(x)$
2. $\mathsf{type}(\Gamma; \lambda x\!:\!A.M) = (A \to \mathsf{type}(\Gamma, x\!:\!A; M))$
3. $\mathsf{type}(\Gamma; MN) =$

   let $B_1 = \mathsf{type}(\Gamma; M)$ and

   let $B_2 = \mathsf{type}(\Gamma; N)$ and $S = \mathsf{unify}(B_1, B_2 \to \alpha)$

   in $S(\alpha)$ (* α は新しい単一化変数 *)

Styles	TCP		TIP		TP
Church	yes	$\hookrightarrow$	yes	$\hookrightarrow$	no [26]
Hole-application	Yes	$\longleftrightarrow$	Yes	$\hookrightarrow$	No
Domain-free	No	$\longleftrightarrow$	No	$\longleftrightarrow$	No [13]
Type-free	No	$\longleftrightarrow$	No	$\hookleftarrow$	No [15]
Curry	no	$\longleftrightarrow$	no	$\hookleftarrow$	no [28]

表 1: $\lambda 2$ に対する TCP，TIP，TP の決定可能性

4. $\mathsf{type}(\Gamma; \Lambda X.M) =$

   let $B = \mathsf{type}(\Gamma; M)$ and $X \notin \mathrm{FV}(\Gamma)$

   in $\forall X.S(B)$ (* S は単一化変数に対する任意の代入 *)

5. $\mathsf{type}(\Gamma; M[\,]) =$

   let $B = \mathsf{type}(\Gamma; M)$ and $S = \mathsf{unify}(\forall X.\mathsf{F}(X), B)$

   in $S(\mathsf{F})(\beta)$ (* β は新しい単一化変数，F は新しい単一化変数で関数変数 *)

アルゴリズム中では，単一化変数と呼ばれる変数を導入して，単一化問題を解きながら型を推論していく．ここで，単一化手続きを unify と書いている．この単一化問題には 2 階の関数変数が含まれているが，高階パターン [24] と呼ばれる決定可能な単一化問題になっている．

定理 5 (type **の健全性と完全性**)

1. 型推論アルゴリズムが停止して $\mathsf{type}(\Gamma; M) = A$ ならば，$\Gamma \vdash_{\text{hole}} M : A$ である．

2. 文脈 Γ と hole-application ラムダ式 M に対して，$\Gamma \vdash_{\text{hole}} M : A$ と仮定する．この時，型推論アルゴリズムは停止して $\mathsf{type}(\Gamma; M) = B$ であり，かつ $A = S(B)$ を満たす単一化変数に対する代入 S が存在する．

証明． 式 M の構成に関する帰納法による． □

4. まとめ

2 階ラムダ計算の型問題の可解性とラムダ式のスタイルとの関係を表 1 にまとめる．表で，“yes” は問題が決定可能であることを，“no” は決定不能であることを示している．ここで，no[26] は Schubert[26] により，no[28] は Wells[28] による．また，“No” は文献 [13, 15] の結果であり，“Yes” は本稿による．

型付け問題 (TP) はどのスタイルでも決定不能な問題である．他方，型推論問題 (TIP)，型検査問題 (TCP) は hole-application，domain-free を境に決定可能・不能が分かれている．これより，$(\forall E)$ のインスタンス情報が削除されても決定可能であるが，関数抽象の domain が省略されると決定不能となる．定理 2 の証明を精査すると，多相型 $(\forall X.A)$ の domain を求める問題，即ち多相型を含む文脈を求める問題は関数変数を含む決定不能な単一化問題と同等であることがわかる．以上より，多相型の domain 情報を削ることは型推論問題の決定不能の要因になっていると結論づけられる．

$? \vdash M :?$	Church	Domain-free	Type-free	Curry
$\lambda 2$	no	No	No	no [28]
rank 2	no	No		yes [21]
ML	no [26]	No		yes [22]

表 2: TP の決定可能性と型ランク

$\Gamma \vdash M :?$	Church	Domain-free	Type-free	Curry
$\lambda 2$	yes	No	No	no [28]
rank 2	yes	No		yes [21]
ML	yes	yes	yes	yes

表 3: TIP の決定可能性と型ランク

次に，表2，表3，表4は，型問題 (TP, TIP, TCP) の可解性を型の複雑さであるランクで分類したものである．興味深い点は，Curry 流の型推論問題 (TIP) は型ランク 2 まで決定可能であるが，domain-free の型推論問題 (TIP) は型ランク 2 で決定不能となってしまうことである．Curry 流に比べて domain-free スタイルは余計な情報を含んでおり，これをうまく使うと難しい問題が定義 8 のごとくコーディングされることによる．言い換えると，domain-free の型推論問題を解こうとすると，余計な情報から厄介な制約条件が生成されてしまう[5]．以上本稿の結果が，決定可能な型推論機構を持つプログラミング言語の設計に少しでも役立てば幸いである．

最後に，ラムダ計算 (の型問題) は様々なところで縁の下の力持ちになっている．一つは関数型プログラミング言語である．例えば，型付き関数型言語として

OCaml (http://caml.inria.fr/ocaml/)

Haskell (http://www.haskell.org/haskellwiki/Haskell)

Standard ML of New Jersey (http://www.smlnj.org/)

などがある．もう一つは，検証技術などで不可欠な定理証明器である．高階ラムダ計算に基づく Coq や Martin-Löf の型理論に基づく Agda などがある．

Coq (http://coq.inria.fr/)

Agda (http://wiki.portal.chalmers.se/agda/pmwiki.php)

合わせてご参考までに．

$\Gamma \vdash M : A?$	Church	Domain-free	Type-free	Curry
$\lambda 2$	yes	No	No	no
rank 3	yes	No		no [28]
rank 2	yes			yes [23]
ML	yes	yes	yes	yes

表 4: TCP の決定可能性と型ランク

[5]「余計なことを中途半端に言うと，物事がかえって難しくなる」状況に似ている．

参考文献

[1] H. P. Barendregt: *The lambda Calculus. Its Syntax and Semantics*, North-Holland, second, revised edition, 1984.

[2] H. P. Barendregt: *Lambda calculi with types*, In S. Abramsky, *et al.* editors, *Handbook of Logic in Computer Science*, Vol II, pp. 117–309, Oxford University Press, 1992.

[3] H. P. Barendregt, W. Dekkers, R. Statman: *Lambda Calculus with Types*, Cambridge University Press, 2012.

[4] G. Barthe, M. H. Sørensen: *Domain-Free Pure Type Systems*, Lecture Notes in Computer Science 1234, pp. 9–20, 1997.

[5] H.-J. Boehm: *Partial polymorphic type inference is undecidable*, Proc. IEEE 26th Annual Symposium on Foundations of Computer Science, pp. 339–345, 1985.

[6] F. Cardone, J. R. Hindley: *Lambda-Calculus and Combinators in the 20th Century*, In Dov M. Gabbay and John Woods editors, *Handbook of the History of Logic*, Vol. 5, 723–817, North-Holland, 2009.

[7] A. Church: *A set of postulates for the foundation of logic*, Annals of Math. 33(2), pp. 346–366, 1932 and 34(4), pp. 839–864, 1933.

[8] A. Church: *A formulation of the simple theory of types*, J. Symbolic Logic 5, pp. 56–68, 1940.

[9] A. Church: *The Calculi of Lambda-Conversion*, Princeton University Press, 1941.

[10] H. B. Curry: *Functionality in combinatory logic*, Proc. Nat. Acad. Science USA, 20, pp. 584–590, 1934.

[11] G. Dowek: *Higher-Order Unification and Matching*, In A. Robinson and A. Voronkov editors, *HANDBOOK OF AUTOMATED REASONING*, Elsevier Science Publishers, pp. 1009–1062, 2001.

[12] K. Fujita: *CPS-translation as adjoint*, Theoretical Computer Science 411 (2), pp. 324–340, 2010.

[13] K. Fujita, A. Schubert: *Partially typed terms between Church-style and Curry-style*, Lecture Notes in Computer Science 1872, pp. 505-520, 2000.

[14] K. Fujita, A. Schubert: *Existential type systems with no types in terms*, Lecture Notes in Computer Science 5608, pp. 112-125, 2009.

[15] K. Fujita, A. Schubert: *The undecidability of type related problems in type-free style System* F, Leibniz International Proceedings in Informatics 6, pp. 103–118, 2010.

[16] W. A. Howard: *The formulae-as-types notion of construction*, In J. P. Seldin and J. R. Hindley editors, *To H. B. Curry: Essays on Combinatory Logic, Lambda Calculus and Formalism*, Academic Press, 1980.

[17] J. P. Seldin: *The Logic of Church and Curry*, In Dov M. Gabbay and John Woods editors, *Handbook of the History of Logic*, Vol. 5, 819–873, North-Holland, 2009.

[18] 高橋正子: **計算論–計算可能性とラムダ計算–**, 近代科学社, 1991.

[19] H. Tonino, K. Fujita: *On the adequacy of representing higher order intuitionistic logic as a pure type system*, Ann. Pure Appl. Logic 57, pp. 251–276, 1992.

[20] A. J. Kfoury, J. Tiuryn, P. Urzyczyn: *The undecidability of the semi-unification problem*, STOC '90: Proc. 22nd Annual ACM Symposium on Theory of Computing, pp. 468–476, 1990.

[21] A. J. Kfoury, J. Tiuryn: *Type Reconstruction in Finite Rank Fragments of the Second-Order λ-Calculus*, Information and Computation 98, pp. 228–257, 1992.

[22] A. J. Kfoury, J. Tiuryn, P. Urzyczyn: *An Analysis of ML Typability*, Journal of the Association for Computing Machinery, Vol. 41, No. 2, pp. 368–398, 1994.

[23] A. J. Kfoury, J. B. Wells: *A direct algorithm for type inference in the rank-2 fragment of the second-order λ-calculus*, Proc. ACM LISP and Functional Programming, pp. 196–207, 1994.

[24] D. Miller: *A logic programming language with lambda-abstraction, function variables, and simple unification*, J. Logic and Computation 1 (4), pp. 497–536, 1991.

[25] F. Pfenning: *On the undecidability of partial polymorphic type reconstruction*, Fundamenta Informaticae 19 (1,2), pp. 185–199, 1993.

[26] A. Schubert: *Second-order unification and type inference for Church-style polymorphism*, POPL '98: Proc. 25th ACM Symposium on Principles of Programming Languages, pp. 279–288, 1998.

[27] A. M. Turing: *Computability and λ-definability*, J. Symbolic Logic 2, pp. 153–163, 1937.

[28] J. B. Wells: *Typability and type checking in system* F *are equivalent and undecidable*, Ann. Pure Appl. Logic 98, pp. 111–156, 1999.