23rd Annual IEEE Symposium on Logic in Computer Science

Types for Hereditary Permutators

Makoto Tatsuta
National Institute of Informatics
2-1-2 Hitotsubashi, Tokyo 101-8430, Japan
tatsuta@nii.ac.jp

Abstract

This paper answers the open problem of finding a type system that characterizes hereditary permutators. First this paper shows that there does not exist such a type system by showing that the set of hereditary permutators is not recursively enumerable. The set of positive primitive recursive functions is used to prove it. Secondly this paper gives a best-possible solution by providing a countably infinite set of types such that a term has every type in the set if and only if the term is a hereditary permutator. By the same technique for the first claim, this paper also shows that a set of normalizing terms in infinite lambda-calculus is not recursively enumerable if it contains some term having a computable infinite path, and shows the set of streams is not recursively enumerable.

1. Introduction

A hereditary permutator is a lambda term that represents finite or infinite nests of permutations. Finite hereditary permutators have been proved to characterize the invertible terms in λ-calculus with $\beta\eta$-reduction [4]. [2] proved that hereditary permutators characterize the invertible terms in Scott's model D_∞. Invertible terms were used to characterize type isomorphisms [3].

A characterization of some class of lambda terms by a type system is often an interesting question [5, 9]. The question of finding a characterization of hereditary permutators by a type system has been studied intensively and is the problem 20 on the TLCA list of open problems [6].

This paper answers this question. First this paper shows that there does not exist such a type system by showing that the set of hereditary permutators is not recursively enumerable. Thus the set of hereditary permutators cannot be characterized by any type system and any type with a recursively enumerable language and a recursively enumerable set of inference rules. Secondly this paper gives a best-possible solution by providing a countably infinite set of types such that a term has every type in the set if and only if the term is a hereditary permutator.

The set of positive primitive recursive functions is used to prove the first claim. This set will be shown to be not recursively enumerable. For a given primitive recursive function f, we will construct a lambda term obtained from some hereditary permutator by modifying some node in its Böhm tree for each n such that the modified node is the same as the original node if and only if $f(n)$ is positive. Then we could decide whether f is positive by deciding whether this term is a hereditary permutator.

For the second claim, we will use intersection types and the Omega type to handle infinite computation in lambda terms. They will also enable us to use the subject reduction property and the subject expansion property.

We will apply the same technique for proving the first claim to infinite lambda calculus [8, 1] and stream types. We will prove that a set of normalizing terms in infinite lambda-calculus is not recursively enumerable if it contains some term having a computable infinite path. We will also show that the set of streams is not recursively enumerable.

Section 2 describes basic definitions and the problem. Non-existence of the solution is proved in Section 3. Section 4 gives permutator schemes that neatly characterize hereditary permutators. Section 5 defines the type system with the set of types that characterizes hereditary permutators. One direction of the characterization is proved in Section 6. Section 7 proves the other direction and completes the characterization theorem. Section 8 studies infinite lambda-calculus and Section 9 discusses stream types.

2. Hereditary Permutators

In this section, we will give the problem 20 in the TLCA list of open problems [6] as well as basic definitions.

Definition 2.1 (λ-Calculus) We have variables $x, y, z, \ldots$. λ-terms $M, N, \ldots$ are defined by:

$M, N, \ldots ::= x|\lambda x.M|MM.$

FV(M) denotes the set of free variables in M. $M[x := N]$ denotes a standard substitution. $M = N$ denotes the

1043-6871/08 $25.00 © 2008 IEEE
DOI 10.1109/LICS.2008.18

syntactical equality modulo renaming bound variables. Vars is the set of variables. Λ is the set of λ-terms.

One-step β-reduction $M \to_\beta N$ is defined by the compatible closure of

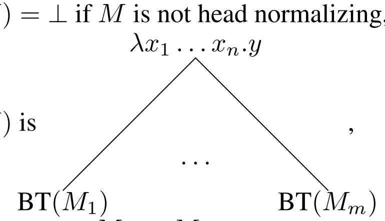
$$(\lambda x.M)N \to_\beta M[x := N].$$

β-reduction $\to_\beta^*$ is defined as the reflexive transitive closure of the relation $\to_\beta$. β-equality $M =_\beta N$ is defined as the least equivalence relation including $\to_\beta^*$. We will write $M_0 \to_\beta^n M_n$ when $M_0 \to_\beta M_1 \to_\beta \ldots \to_\beta M_n$ for some $M_1, \ldots, M_{n-1}$. We say M reduces to N if $M \to_\beta^* N$. A λ-term of the shape $(\lambda x.M)N$ is called a *redex*. A λ-term M is called *normal* if there is no λ-term N such that $M \to_\beta N$.

A λ-term M is called *head normal* if M is of the shape $\lambda x_1 \ldots x_n.yN_1 \ldots N_m$. A λ-term M is called *head normalizing* if there is some head normal term N such that $M \to_\beta^* N$.

Definition 2.2 (Böhm Trees) We suppose $\perp$ is a constant. A Böhm tree is defined as a (possibly infinite) tree with labels in $\{\lambda x_1 \ldots x_n.y | x_1, \ldots, x_n, y \in \text{Vars}\} \cup \{\perp\}$. Böhm tree BT$(M)$ of a λ-term M is defined by:

(1) BT$(M) = \perp$ if M is not head normalizing,

(2) BT(M) is 

,

if $M =_\beta \lambda x_1 \ldots x_n.yM_1 \ldots M_m$.

Definition 2.3 (Hereditary Permutators) A λ-term M is called a *hereditary permutator* if BT(M) satisfies the following conditions:

(H1) Its root has the shape $\lambda z x_1 \ldots x_n.z$ and the multiset of its child nodes is $\{\lambda w_1^i \ldots w_{n_i}^i.x_i | 1 \le i \le n\}$ for some variables $w_j^i \ne x_i$.

(H2) A node except the root has the shape $\lambda x_1 \ldots x_n.y$ and the multiset of its child nodes is $\{\lambda w_1^i \ldots w_{n_i}^i.x_i | 1 \le i \le n\}$ for some variables $w_j^i \ne x_i$.

The problem 20 in the TLCA list of open problems is finding a type that characterizes the set of hereditary permutators. This question expects that there is some type system T with some type A such that $M : A$ is provable in T if and only if M is a hereditary permutator.

We will answer this question. First, in Section 3 we will show that the set of hereditary permutators is not recursively enumerable. Hence we will conclude that there does not exist any type that characterizes hereditary permutators if the system has a recursively enumerable language and a recursively enumerable set of inference rules. Secondly, in Section 5 we will present an intersection type system with a countably infinite set of types which characterizes hereditary permutators.

3. Non-existence of a type for hereditary permutators

We will show that there does not exist any type that characterizes the set of hereditary permutators.

Notation. We will write HP for the set of hereditary permutators. N is the set of natural numbers. We will use a vector notation $\vec{e}$ to denote a sequence $e_1, \ldots, e_n$ $(n \ge 0)$. For example, we will use $\vec{M}$ to denote a sequence of λ-terms $M_1, \ldots, M_n$ $(n \ge 0)$. $M\vec{N}$ denotes $MN_1 \ldots N_n$. $\lambda\vec{x}.M$ denotes $\lambda x_1 \ldots x_n.M$. We will write $\overline{n}$ for the n-th Church numeral $\lambda fx.f^n x$ where $f^n x$ denotes $f(f(\ldots(fx)\ldots))$ (n times of f).

First, we give several notations for primitive recursive functions and partial recursive functions.

Definition 3.1 We write $\{n\}^{pr}(x)$ for the n-th unary primitive recursive function. $\langle x, y\rangle$ denotes the standard primitive recursive surjective pairing, and $\pi_0(x)$ and $\pi_1(x)$ are the first and second projections respectively. The n-th unary partial recursive function $\{n\}(x)$ is defined by $\{\pi_1(n)\}^{pr}(\mu y.(\{\pi_0(n)\}^{pr}(\langle x, y\rangle) = 0))$. We also define $u(x, y) = \{x\}^{pr}(y)$.

Remark. The function u is a universal function for unary primitive recursive functions. u is a total recursive function.

Definition 3.2 For a function $f : N^n \to N$, we say that a λ-term F represents f when $f(m_1, \ldots, m_n) = m$ iff $F\overline{m_1} \ldots \overline{m_n} \to_\beta^* \overline{m}$ for all $m_1, \ldots, m_n, m$.

Theorem 4.15 in Page 53 in [7] showed the following claim.

Theorem 3.3 ([7]) *For every recursive function f, there is some λ-term F such that F represents f.*

Definition 3.4 PPR is defined to be the set $\{n \in N | \forall x(\{n\}^{pr}(x) > 0)\}$.

PPR is the set of indexes for positive primitive recursive functions.

Proposition 3.5 *The set* PPR *is not recursively enumerable.*

Proof. By standard results in recursion theory, we have the recursive function $S : N^2 \to N$ defined by $\{S(n,m)\}^{pr}(x) = \{n\}^{pr}(\{m\}^{pr}(x))$, and the recursive function $P : N \to N$ defined by $\{P(n)\}^{pr}(m) = \langle n, m\rangle$.

Assume that PPR is recursively enumerable. We will show contradiction.

Define a partial function $f : N \to N$ by $f(x) = 1$ if $S(\pi_0(x), P(x))$ is in PPR, and $f(x)$ is undefined otherwise. Then f is partial recursive. There is a number e such that for all x, both $\{e\}(x)$ and $f(x)$ has the same value or both are undefined.

Then we show that $f(x)$ is defined if and only if $\{x\}(x)$ is undefined. It is proved as follows: $f(x)$ is defined iff $S(\pi_0(x), P(x))$ is in PPR by the definition of f, iff $\forall y(\{\pi_0(x)\}^{pr}(\{P(x)\}^{pr}(y)) > 0)$ by the definition of S and PPR, iff $\forall y(\{\pi_0(x)\}^{pr}(\langle x, y\rangle) > 0)$ by the definition of P, iff $\{\pi_1(x)\}^{pr}(\mu y.(\{\pi_0(x)\}^{pr}(\langle x, y\rangle) = 0))$ is undefined, iff $\{x\}(x)$ is undefined.

If $\{e\}(e)$ is defined, then $f(e)$ is defined by the definition of e, and hence $\{e\}(e)$ is undefined by the above. Hence $\{e\}(e)$ is undefined. However, $f(e)$ is undefined by the definition of e, and hence $\{e\}(e)$ is defined by the above, which is contradiction.

Consequently, the set PPR is not recursively enumerable. $\Box$

We define

$$\begin{array}{l} \mathrm{S} = \lambda yfx.f(yfx), \\ Y_0 = \lambda xy.y(xxy), \\ Y = Y_0Y_0. \end{array}$$

The term S is the successor for Church numerals. Y is Turing's fixed point operator. Remark that $YM \to_\beta^* M(YM)$.

We now prove the first main theorem by using PPR.

Theorem 3.6 *The set* HP *of hereditary permutators is not recursively enumerable.*

Proof. Assume HP is recursively enumerable. We will show contradiction.

Define P as $Y(\lambda pz_0z_1.z_0(pz_1))$. Then $\mathrm{BT}(\lambda z_0.Pz_0)$ is $\begin{array}{c} \lambda z_0z_1.z_0 \\ \mid \\ \lambda z_2.z_1 \\ \mid \\ \lambda z_3.z_2 \\ \mid \\ \vdots \end{array}$, since $Pz_n \to_\beta^* (\lambda z_0z_1.z_0(Pz_1))z_n =_\alpha (\lambda z_nz_{n+1}.z_n(Pz_{n+1}))z_n \to_\beta \lambda z_{n+1}.z_n(Pz_{n+1})$. Hence $\lambda z_0.Pz_0$ is in HP.

By Theorem 3.3, we have a λ-term U that represents the function u. Define T by

$$\begin{array}{l} \Delta = \lambda x.xx, \\ T = Y(\lambda txyz_0z_1.Uxy(\lambda w.z_0(tx(\mathrm{S}y)z_1))(\Delta\Delta)). \end{array}$$

We will write $\tilde{n}$ for $S^n\overline{0}$ for a number n. Then we have $T\overline{e}\tilde{n}z_n \to_\beta^* \lambda z_{n+1}.U\overline{e}\tilde{n}(\lambda w.z_n(T\overline{e}(\widetilde{n+1})z_{n+1}))(\Delta\Delta)$ in a similar way to P. Hence $\mathrm{BT}(T\overline{e}\tilde{n}z_n)$ is $\begin{array}{c} \lambda z_{n+1}.z_n \\ \mid \\ \mathrm{BT}(T\overline{e}(\widetilde{n+1})z_{n+1}) \end{array}$ if $\{e\}^{pr}(n) > 0$ since $U\overline{e}\tilde{n} =_\beta \overline{k+1}$ for some k, and $\mathrm{BT}(T\overline{e}\tilde{n}z_n)$ is $\perp$ if $\{e\}^{pr}(n) = 0$ since $U\overline{e}\tilde{n} =_\beta \overline{0}$.

We show that $\lambda z_0.T\overline{e}\overline{0}z_0$ is in HP if and only if e is in PPR. The direction from the right to the left is proved by $\mathrm{BT}(\lambda z_0.T\overline{e}\overline{0}z_0) = \mathrm{BT}(\lambda z_0.Pz_0)$ when e is in PPR, and hence $\lambda z_0.T\overline{e}\overline{0}z_0$ is in HP. In order to show the direction from the left to the right, first we assume e is not in PPR and will show $\lambda z_0.T\overline{e}\overline{0}z_0$ is not in HP. From $e \notin$ PPR, we have a number m_0 such that $\{e\}^{pr}(m_0) = 0$ and $\{e\}^{pr}(m) > 0$ for all $m < m_0$. Then $\mathrm{BT}(\lambda z_0.T\overline{e}\overline{0}z_0) = \begin{array}{c} \lambda z_0z_1.z_0 \\ \mid \\ \lambda z_2.z_1 \\ \mid \\ \vdots \\ \mid \\ \lambda z_{m_0}.z_{m_0-1} \\ \mid \\ \perp \end{array}$, and hence $\lambda z_0.T\overline{e}\overline{0}z_0$ is not in HP.

If HP were recursively enumerable, then PPR would be recursively enumerable, which would lead to contradiction. Therefore HP is not recursively enumerable. $\Box$

Non-existence of solutions for the problem follows immediately from the previous theorem.

Theorem 3.7 (No type for hereditary permutators) *There does not exist any type system T with any type A such that its language and the set of its inference rules are recursively enumerable, and the set of hereditary permutators is the same as $\{M \in \Lambda | \Gamma \vdash M : A$ is provable in T for some $\Gamma\}$.*

Proof. If we had such a type system T, then $\{M \in \Lambda | \Gamma \vdash M : A$ is provable in T for some $\Gamma\}$ would be recursively enumerable, and therefore HP would be recursively enumerable, which would contradict to Theorem 3.6. $\Box$

4. Permutator Schemes

We will define a permutator scheme that has the same Böhm tree as the application of a hereditary permutator to a variable. Permutator schemes will neatly characterize hereditary permutators and we will discuss some set-theoretic properties of them.

Definition 4.1 We write $\mathcal{S}_m$ for the symmetric group of order m. We define the set $\mathrm{PS}_n(z)$ for $n \geq 0$ and a variable z by

$$\begin{array}{l} \mathrm{PS}_0(z) = \Lambda, \\ \mathrm{PS}_{n+1}(z) = \{M \in \Lambda | \\ \quad M \to_\beta^* \lambda x_1 \ldots x_m.zM_{\pi(1)} \ldots M_{\pi(m)}, \\ \quad \pi \in \mathcal{S}_m, M_i \in \mathrm{PS}_n(x_i) \quad (1 \leq i \leq m)\}. \end{array}$$

Remark. (1) $M \in \mathrm{PS}_k(z)$ iff there is a hereditary permutator N such that $\mathrm{BT}(\lambda z.M)$ and $\mathrm{BT}(N)$ are the same at any node at depth $< k$. We call a term in $\mathrm{PS}_k(z)$ a permutator scheme at k.

(2) $\mathrm{PS}_{k+1}(z) \subseteq \mathrm{PS}_k(z)$.

Lemma 4.2 *(1) Let $n > 0$. M is in $\mathrm{PS}_n(z)$ if and only if the following hold in* $\mathrm{BT}(M)$.

(a) Its root has the shape $\lambda\vec{x}.z$,

(b) If it has a node $\lambda x_1 \dots x_m.y$ at depth $< n-1$, the multiset of its child nodes is $\{\lambda\vec{w}_i.x_i | 1 \le i \le m\}$ for some variables $\vec{w}_i$ different from x_i,

(c) If it has a node $\lambda x_1 \dots x_m.y$ at depth $n-1$, the node has m child nodes.

(2) $\lambda z.M$ is in HP *if and only if M is in* $\mathrm{PS}_n(z)$ *for all $n > 0$.*

Proof. (1) The left-hand side to the right-hand side. By induction on n.

Case $n = 1$. (a) $M \to_\beta^* \lambda x_1 \dots x_m.zM_{\pi(1)} \dots M_{\pi(m)}$, so the claim holds.

(b) There is no node at depth < 0.

(c) The node at depth 0 is the root and has m child nodes.

Case $n > 1$. Suppose $M \in \mathrm{PS}_n(z)$. Let $M \to_\beta^* \lambda x_1 \dots x_m.zM_{\pi(1)} \dots M_{\pi(m)}$ and $M_i \in \mathrm{PS}_{n-1}(x_i)$.

(a) The claim holds since its head variable is z.

(b) If the node is the root, the claim holds because the root of $\mathrm{BT}(M_i)$ is $\lambda\vec{w}_i.x_i$ by induction hypothesis (a) for $n-1$ with $M_i \in \mathrm{PS}_{n-1}(x_i)$. If the node is not the root, the node at depth $< n-1$ is some node at depth $< n-2$ of $\mathrm{BT}(M_i)$ for some i. By induction hypothesis (b) for $n-1$ with $M_i \in \mathrm{PS}_{n-1}(x_i)$, the claim holds.

(c) Any node at depth $n-1$ is some node at depth $n-2$ of $\mathrm{BT}(M_i)$ for some i. By induction hypothesis (c) for $n-1$ with $M_i \in \mathrm{PS}_{n-1}(x_i)$, the claim holds.

The right-hand side to the left-hand side. By induction on n.

Case $n = 1$. The claim holds from (a) and (c).

Case $n > 1$. By (a), we have $M \to_\beta^* \lambda x_1 \dots x_m.z\vec{M}$. By (b), $z\vec{M}$ is $zM_{\pi(1)} \dots M_{\pi(m)}$ for some $\pi \in \mathcal{S}_m$ and $M_i \to_\beta^* \lambda\vec{w}_i.x_i\vec{N}_i$.

We show that $\mathrm{BT}(M_i)$ satisfies (a) to (c) for x_i and $n-1$.

(a) Its root is $\lambda\vec{w}_i.x_i$.

(b) Any node at depth $< n-2$ in $\mathrm{BT}(M_i)$ is some node at depth $< n-1$ in $\mathrm{BT}(M)$, so the claim holds.

(c) Any node at depth $n-2$ in $\mathrm{BT}(M_i)$ is some node at depth $n-1$ in $\mathrm{BT}(M)$, so the claim holds.

By induction hypothesis for $n-1$, we have $M_i \in \mathrm{PS}_{n-1}(x_i)$. Hence M is in $\mathrm{PS}_n(z)$

(2) (A) The left-hand side to the right-hand side. For each n, we will show $M \in \mathrm{PS}_n(z)$. If $n = 0$, then the claim holds since $\mathrm{PS}_0(z) = \Lambda$. Suppose $n > 0$. We will show that $\mathrm{BT}(M)$ satisfies (a) to (c) in (1) for z and n.

(a) The claim holds by (H1).

(b)(c) If the node is the root, the claim follows from (H1). If the node is not the root, the claim follows from (H2).

Hence we have $M \in \mathrm{PS}_n(z)$ from (1).

(B) The right-hand side to the left-hand side.

(H1) The claim follows from $M \in \mathrm{PS}_2(z)$ and (a) and (b) in (1).

(H2) Let d be the depth of the node. The claim follows from $M \in \mathrm{PS}_{d+2}(z)$ and (b) in (1). $\square$

5. Types for hereditary permutators

This section will present a type system with a countably infinite set of types which characterizes hereditary permutators.

Definition 5.1 We define the type system $\mathcal{T}$.

We have type constants $p_n, q_m \quad (n \ge 0, m \ge 1)$, and Ω. Types $A, B, \dots$ are defined by:

$A, B, \dots ::= p_n | q_m | \Omega | A \to A | A \cap A \quad (n \ge 1, m \ge 0)$.

$\mathrm{TC}(\vec{A})$ is defined as the set of type constants in the types $\vec{A}$.

Type partial equivalence $A \sim_n B$ for $n > 0$ is defined by:

$$\Omega \sim_0 \Omega$$

$$\frac{A_i \sim_n B_i \qquad (1 \le i \le m)}{B_{\pi(1)} \to \dots \to B_{\pi(m)} \to q_k \sim_{n+1} A_1 \to \dots \to A_m \to q_k}$$

where $m \ge 0$, $\pi \in \mathcal{S}_m$, and $\mathrm{TC}(A_i, B_i) - \{\Omega\} \quad (1 \le i \le m)$, $\{q_k\}$ are disjoint in the second rule.

A type declaration is a finite set of the form $\{x_1 : A_1, \dots, x_n : A_n\}$ where x_i's are distinct variables and A_i's are types. We will write $\Gamma, \Delta, \dots$ for a type declaration. A judgment is $\Gamma \vdash M : A$. We will also write $x_1 : B_1, \dots, x_n : B_n \vdash M : A$ for $\{x_1 : B_1, \dots, x_n : B_n\} \vdash M : A$, and $\Gamma, y : C \vdash M : A$ for $x_1 : B_1, \dots, x_n : B_n, y : C \vdash M : A$, when Γ is $\{x_1 : B_1, \dots, x_n : B_n\}$.

Typing rules are given by:

$$\frac{}{\Gamma, x : A \vdash x : A}\ (Ass) \qquad \frac{\Gamma, x : A \vdash M : B}{\Gamma \vdash \lambda x.M : A \to B}\ (\to I)$$

$$\frac{\Gamma \vdash M : A \to B \quad \Gamma \vdash N : A}{\Gamma \vdash MN : B}\ (\to E)$$

$$\frac{\Gamma \vdash M : A \quad \Gamma \vdash M : B}{\Gamma \vdash M : A \cap B}\ (\cap I)$$

$$\frac{\Gamma \vdash M : A \cap B}{\Gamma \vdash M : A}\ (\cap E_1) \qquad \frac{\Gamma \vdash M : A \cap B}{\Gamma \vdash M : B}\ (\cap E_2)$$

$$\frac{}{\Gamma \vdash M : \Omega}\ (\Omega) \qquad \frac{\Gamma, z : A \vdash M : B \quad A \sim_n B}{\Gamma \vdash \lambda z.M : p_n}\ (p_n I)$$

Remark. The relation $\sim_n$ is proved to be a partial equivalence relation.

This is a standard intersection type system except for the type partial equivalence $\sim_n$ and the constants p_n, q_m. The intended meaning of the relation $\sim_n$ and the constants p_n are the set $\{(A, B) | z : A \vdash M : B \text{ iff } M \in \mathrm{PS}_n(z)\}$ and the set $\{\lambda z.M | M \in \mathrm{PS}_n(z)\}$ respectively. The constants q_m are used for locality in $\sim_n$. Intersection types and the type Ω are necessary since we need the subject expansion property in our proof.

We have a characterization theorem of HP by this type system with the set of the types p_n.

Theorem 5.2 (Characterization Theorem) *M is a hereditary permutator if and only if $\vdash M : p_n$ is provable in the type system $\mathcal{T}$ for all n.*

We will finish the proof of this theorem in Section 7. The soundness of this characterization will be proved in Section 6 and its completeness will be shown in Section 7.

6. Soundness

We will prove the soundness part of Theorem 5.2 by using permutator schemes.

First we will show basic properties for the system $\mathcal{T}$.

Definition 6.1 $[A]$ is defined as a type of the shape defined by:

$$[A] ::= A|[A] \cap B|B \cap [A].$$

$[A]$ ambiguously denotes some type obtained from A by intersection.

We have a standard generation lemma.

Lemma 6.2 (Generation Lemma) *(1) If $\Gamma \vdash x : [B]$ and B is not Ω nor an intersection type, then Γ includes $x : A$ for some A of the shape $[B]$.*

(2) If $\Gamma \vdash \lambda x.M : [A \to B]$, then $\Gamma, x : A \vdash M : B$.

(3) If $\Gamma \vdash MN : [B]$ and B is not Ω nor an intersection type, then there is some A such that $\Gamma \vdash M : A \to B$ and $\Gamma \vdash N : A$.

(4) If $\Gamma, x : A_1 \to \ldots \to A_n \to B \vdash x M_1 \ldots M_m : C_{l+1} \to \ldots \to C_n \to B$, $n > 0$, and $C_{l+1} \to \ldots \to C_n \to B$ is not Ω nor an intersection type, then $l = m$, $A_i = C_i \quad (l < i \le n)$, and $\Gamma, x : A_1 \to \ldots \to A_n \to B \vdash M_i : A_i \quad (1 \le i \le l)$.

Proof. (1)(2)(3) By induction on the proof.

(4) The claim is proved by induction on m from (1) and (3). □

Next we will show the subject reduction property. The next lemma is used for that.

Lemma 6.3 *$\Gamma, x : A \vdash M : B$ and $\Gamma \vdash N : A$ imply $\Gamma \vdash M[x := N] : B$.*

Proof. By induction on $\Gamma, x : A \vdash M : B$ □

Proposition 6.4 (Subject Reduction) *If $\Gamma \vdash M : A$ and $M \to_\beta M'$, then $\Gamma \vdash M' : A$.*

Proof. Induction on the proof. Consider cases according to the last rule. By induction hypothesis, we can immediately show cases where M is not the redex. We will show only interesting cases.

Case $(\to E)$ where M is the redex. The proof is

$$\frac{\Gamma \vdash \lambda x.M : B \to A, \quad \Gamma \vdash N : B}{\Gamma \vdash (\lambda x.M)N : A}.$$

and M' is $M[x := N]$.

By Lemma 6.2 (2) for $\Gamma \vdash \lambda x.M : B \to A$, we have $\Gamma, x : B \vdash M : A$. By Lemma 6.3, we have $\Gamma \vdash M[x := N] : A$.

Case $(p_n I)$. The proof is

$$\frac{\Gamma, z : A \vdash N : B \quad A \sim_n B}{\Gamma \vdash \lambda z.N : p_n}$$

and M' is $\lambda z.N'$ where $N \to_\beta N'$. By induction hypothesis, we have $\Gamma, z : A \vdash N' : B$. Hence $\Gamma \vdash \lambda z.N' : p_n$ holds. □

Then we will show the subject expansion property. For that, we need the following lemma.

Lemma 6.5 *If $\Gamma \vdash M[x := N] : A$, then there is some type B such that $\Gamma, x : B \vdash M : A$ and $\Gamma \vdash N : B$.*

Proof. Induction on the proof. Consider cases according to the last rule. We will show only interesting cases.

Case $x \notin \mathrm{FV}(M)$. Let B be Ω.

Case $(\to E)$. The proof is

$$\frac{\Gamma \vdash M_1[x := N] : C \to A \quad \Gamma \vdash M_2[x := N] : C}{\Gamma \vdash (M_1 M_2)[x := N] : A}.$$

By induction hypothesis, we have B_1 such that $\Gamma, x : B_1 \vdash M_1 : C \to A$ and $\Gamma \vdash N : B_1$. By induction hypothesis, we also have B_2 such that $\Gamma, x : B_2 \vdash M_2 : C$ and $\Gamma \vdash N : B_2$. Let B be $B_1 \cap B_2$.

Case $(\cap I)$. Similar to Case $(\to E)$.

Case (Ω). Let B be Ω. □

In order to have this lemma, we need intersection types and the Ω type. The subject expansion property is proved by using this lemma.

Proposition 6.6 (Subject Expansion) *$\Gamma \vdash M : A$ and $M' \to_\beta M$ implies $\Gamma \vdash M' : A$.*

Proof. Induction on the proof. Consider cases according to the last rule. We will discuss only interesting cases.

Case M' is the redex. Let M' be $(\lambda x.L)N$ and M be $L[x := N]$. By Lemma 6.5, there is a type B such that $\Gamma, x : B \vdash L : A$ and $\Gamma \vdash N : B$. Then we have $\Gamma \vdash \lambda x.L : B \to A$ and hence we have $\Gamma \vdash M' : A$.

Case $(\to I)$ and M' is not the redex. We can put $M' = \lambda x.N'$, $M = \lambda x.N$, and $N' \to_\beta N$. By induction hypothesis for $\Gamma, x : A \vdash N : B$, we have $\Gamma, x : A \vdash N' : B$. Hence $\Gamma \vdash M' : A$ holds.

Case $(\to E)$ and M' is not the redex. We can put $M' = N'L'$, $M = NL$. The claim follows from induction hypothesis.

Cases $(\cap I)$, $(\cap E_1)$, and $(\cap E_2)$. By induction hypothesis.

Case (Ω). The claim holds trivially.

Case (p_nI) and M' is not the redex. We can put $M' = \lambda z.N'$, $M = \lambda z.N$, and $N' \to_\beta N$. By induction hypothesis, $\Gamma, z : A \vdash N' : B$ holds. Hence we have $\Gamma \vdash M' : p_n$. □

Remark. We will use the subject reduction property and the subject expansion property for proving soundness and completeness respectively.

Definition 6.7 right(A) is defined as the rightmost type constant in A. HN is defined to be the set of head normalizing terms. $X \to Y$ is defined as the set $\{M \in \Lambda | \forall N \in X(MN \in Y)\}$ for sets $X, Y \subseteq \Lambda$.

Example. right$((C \to q_1) \cap (C \to q_2)) = q_2$.

The interpretation $[A]$ of a type A is defined by:

$$\begin{array}{ll} [q_n] = [p_{n+1}] = \text{HN} & (n \geq 0), \\ [\Omega] = \Lambda, & \\ [A \to B] = [A] \to [B], & \\ [A \cap B] = [A] \cap [B]. & \end{array}$$

The interpretation $[\sim_n]$ is defined as $p(\Lambda) \times p(\Lambda)$.

Lemma 6.8 *(1) $x\vec{M}$ is in $[A]$.*

(2) right$(A) \neq \Omega$ *implies* $[A] \subseteq$ HN.

(3) $[A]$ is closed under $=_\beta$.

Proof. (1) By induction on A.

(2) By induction on A.

Case $A \to B$. Suppose right$(A \to B) \neq \Omega$ and $M \in [A \to B]$. By (1) we have $x \in [A]$. Hence Mx is in $[B]$. By induction hypothesis for B, we have $Mx \in$ HN. Hence M is in HN.

Case $A \cap B$. From right$(A \cap B) \neq \Omega$, we have right$(B) \neq \Omega$. By induction hypothesis for B, we have $[B] \subseteq$ HN. Hence we get $[A \cap B] \subseteq$ HN.

(3) By induction on A. □

Definition 6.9 A variable assignment ρ is defined by $\rho :$ Vars $\to \Lambda$. A variable assignment $\rho[x := M]$ is defined by $(\rho[x := M])(x) = M$ and $(\rho[x := M])(y) = \rho(y)$ if x is not y. The interpretation $[M]\rho$ of a term M with ρ is defined as $M[x_1 := \rho(x_1), \ldots, x_n := \rho(x_n)]$ where FV$(M) = \{x_1, \ldots, x_n\}$.

Lemma 6.10 *If we have $\overrightarrow{x : B} \vdash M : A$ and $\rho(x_i) \in [B_i]$ $(\forall i)$, then we have $[M]\rho \in [A]$.*

Proof. It is proved by induction on the proof. We consider cases according to the last rule. We will show only interesting cases.

Case $(\to I)$. Assume $N \in [A]$. We will show $[\lambda x.M]\rho N \in [B]$. Let ρ' be $\rho[x := N]$. By induction hypothesis, we have $[M]\rho' \in [B]$. Since we have $[\lambda x.M]\rho N \to_\beta [M]\rho'$, from Lemma 6.8 (3), we get $[\lambda x.M]\rho N \in [B]$. Hence $[\lambda x.M]\rho$ is in $[A \to B]$.

Cases $(\to E)$, $(\cap I)$, $(\cap E_i)$ are proved by induction hypothesis.

Case (Ω) is proved by $[\Omega] = \Lambda$.

Case (p_nI). The proof is

$$\frac{\Gamma, z : A \vdash M : B \quad A \sim_n B}{\Gamma \vdash \lambda z.M : p_n}.$$

Let ρ' be $\rho[z := z]$. By Lemma 6.8 (1), $\rho'(z)$ is in $[A]$. By induction hypothesis, we have $M\rho' \in [B]$. Since $n > 0$ and $A \sim_n B$, we have right$(B) \neq \Omega$. By Lemma 6.8 (2), we have $[B] \subseteq$ HN. Since $(\lambda z.M)\rho$ is $\lambda z.M\rho'$, we have $(\lambda z.M)\rho \in$ HN. □

Proposition 6.11 *If $\overrightarrow{x : B} \vdash M : A$ and* right$(A) \neq \Omega$, *M is head normalizing.*

Proof. Let $\rho(x) = x$. By Lemma 6.8 (1), $\rho(x_i) = x_i$ is in $[B_i]$. By Lemma 6.10, we have $M\rho \in [A]$. By Lemma 6.8 (2) we have $[A] \subseteq$ HN. Since $M\rho = M$, we have $M \in$ HN. □

Definition 6.12 We define the set core(A) of type constants for a type A by induction on A by

$$\begin{array}{ll} \text{core}(c) = \{c\} & (c = q_n, p_n, \Omega), \\ \text{core}(A \to B) = \text{core}(B), & \\ \text{core}(A \cap B) = \text{core}(A) \cup \text{core}(B). & \end{array}$$

We will also write core$(A_1, \ldots, A_n)$ and core$(x_1 : A_1, \ldots, x_n : A_n)$ for $\cup\{\text{core}(A_i) | 1 \leq i \leq n\}$.

Example. core$(A \to B \to q_0, (C \to q_1) \cap (C \to q_2)) = \{q_0, q_1, q_2\}$.

Lemma 6.13 *If $\Gamma, x : A \vdash x\vec{M} : B$, then* core$(A) \supseteq$ core(B).

Proof. By induction on the proof. □

Lemma 6.14 *If $A \sim_n B$ and $\Gamma, z : A \vdash M : B$ are provable and* core$(\Gamma) \cap (\text{TC}(A, B) - \{\Omega\}) = \phi$, *then M is in* PS$_n(z)$.

Proof. By induction on n.

Case $n = 0$. The claim holds since PS$_0(z) = \Lambda$.

Case $n + 1$. Let A be $B_{\pi(1)} \to \ldots \to B_{\pi(m)} \to q_k$ and B be $A_1 \to \ldots \to A_m \to q_k$. By Proposition 6.11, M is head normalizing. Let $M \to_\beta^* \lambda\vec{x}.y\vec{M}$ and $\vec{x}$ be $x_1, \ldots, x_l$.

By Proposition 6.4, we have $\Gamma, z : A \vdash \lambda\vec{x}.y\vec{M} : B$. By Lemma 6.2 (2), we have $\Gamma, z : A, \overrightarrow{x : A} \vdash y\vec{M} : C$ where we put $C = A_{l+1} \to \ldots \to A_m \to q_k$.

Since core$(\Gamma) \cap (\text{TC}(A, B) - \{\Omega\}) = \phi$, we have $q_k \notin$ core(Γ) and hence y is z or x_i from Lemma 6.13. Since $q_k \notin \text{TC}(A_i, B_i) - \{\Omega\}$, we have $q_k \notin$ core(A_i) and hence y is z from Lemma 6.13. Then we have $\Gamma, z : A, \overrightarrow{x : A} \vdash z\vec{M} : C$.

If $m = 0$, then the claim holds since $l = 0$, $\vec{M}$ is empty, and $z \in \mathrm{PS}_n(z)$. Suppose $m > 0$. By Lemma 6.2 (4), the length of $\vec{M}$ is l and we have $B_{\pi(i)} = A_i \quad (l+1 \le i \le m)$ and $\Gamma, z : A, \overrightarrow{x : A} \vdash M_i : B_i \quad (1 \le i \le l)$ where we put $z\vec{M} = zM_{\pi(1)} \dots M_{\pi(l)}$.

By definition, $M \in \mathrm{PS}_1(z)$ iff $M \to_\beta^* \lambda\vec{x}.z\vec{M}$ and the lengths of $\vec{x}$ and $\vec{M}$ are the same. Hence we have $M \in \mathrm{PS}_1(z)$. If $n = 0$, the claim immediately follows from this.

Suppose $n \ge 1$. When $i \ne j$, $\mathrm{core}(A_i) \ne \mathrm{core}(B_j)$ holds since these cores are some type constants other than Ω and $\mathrm{TC}(A_i, B_i) - \{\Omega\}$ and $\mathrm{TC}(A_j, B_j) - \{\Omega\}$ are disjoint. From $B_{\pi(i)} = A_i \quad (l+1 \le i \le m)$ we have $\pi(i) = i$ for $l+1 \le i \le m$. Therefore π is in $\mathcal{S}_l$.

Since we already have $A_i \sim_n B_i, \Gamma, z : A, \overrightarrow{x : A} \vdash M_i : B_i$, and $\mathrm{core}(\Gamma, z : A, x_j : A_j \quad (i \ne j)) \cap (\mathrm{TC}(A_i, B_i) - \{\Omega\}) = \phi$, by induction hypothesis for n, we have $M_i \in \mathrm{PS}_n(x_i)$. From $M \to_\beta^* \lambda x_1 \dots x_l.zM_{\pi(1)} \dots M_{\pi(l)}$, $\pi \in \mathcal{S}_l$, and $M_i \in \mathrm{PS}_n(x_i)$, we have $M \in \mathrm{PS}_{n+1}(z)$. $\square$

Lemma 6.15 $\vdash \lambda z.M : [p_n]$ *implies* $M \in \mathrm{PS}_n(z)$.

Proof. Induction on the proof. Consider cases according to the last rule.

We do not have cases (Ass), $(\to I)$, $(\to E)$, nor (Ω).

Cases $(\cap I)$ and $(\cap E_i)$ are proved by induction hypothesis.

Case $(p_n I)$. The proof is

$$\frac{z : A \vdash M : B \quad A \sim_n B}{\vdash \lambda z.M : p_n} .$$

Let Γ be ϕ in Lemma 6.14, we have $M \in \mathrm{PS}_n(z)$. $\square$

Proposition 6.16 (Soundness) *If* $\vdash M : p_n$ *for all* n, *then* M *is in* HP.

Proof. By Proposition 6.11, we have $M \to_\beta^* \lambda x_1 \dots x_l.y\vec{M}$. By Proposition 6.4, we have $\vdash \lambda x_1 \dots x_l.y\vec{M} : p_n$.

If $l = 0$, we have contradiction from Lemma 6.2 (1) and (3). So we have $l > 0$. Let $\lambda z.N$ be $\lambda x_1 \dots x_l.y\vec{M}$.

We have $\vdash \lambda z.N : p_n$ for all n. By Lemma 6.15, we have $N \in \mathrm{PS}_n(z)$ for all n. By Lemma 4.2 (2), we have $\lambda z.N \in \mathrm{HP}$. Hence M is in HP. $\square$

7. Completeness

We will show the completeness and finish the proof of the characterization theorem.

Lemma 7.1 *If* $M \in \mathrm{PS}_n(z)$, *there are* A *and* B *such that* $z : A \vdash M : B$ *and* $A \sim_n B$.

Proof. Induction on n.

Case $n = 0$. Take Ω for A, B.

Case $n + 1$. Suppose $M \to_\beta^* \lambda x_1 \dots x_m.zM_{\pi(1)} \dots M_{\pi(m)}$, $\pi \in \mathcal{S}_m$, and $M_i \in \mathrm{PS}_n(x_i)$.

Induction hypothesis for n, we have A_i and B_i such that $x_i : A_i \vdash M_i : B_i$ and $A_i \sim_n B_i$. By choosing appropriate q_l for each (A_i, B_i), we can suppose that $\mathrm{TC}(A_i, B_i) - \{\Omega\} \quad (1 \le i \le m)$ are disjoint.

Let q_k be fresh in A_i, B_i and $A = B_{\pi(1)} \to \dots \to B_{\pi(m)} \to q_k$ and $B = A_1 \to \dots \to A_m \to q_k$. Then we have $A \sim_{n+1} B$.

Then we have $z : A, \overrightarrow{x : A} \vdash zM_{\pi(1)} \dots M_{\pi(m)} : q_k$. Hence we get $z : A \vdash \lambda\vec{x}.zM_{\pi(1)} \dots M_{\pi(m)} : B$. By Proposition 6.6, we have $z : A \vdash M : B$. $\square$

Proposition 7.2 (Completeness) *If* $M \in \mathrm{HP}$, *then we have* $\vdash M : p_n \quad (n > 0)$.

Proof. Let $M \to_\beta^* \lambda z.N$. By Lemma 4.2 (2), we have $N \in \mathrm{PS}_n(z)$ for all $n > 0$. By Lemma 7.1, there are A and B such that $z : A \vdash N : B$ and $A \sim_n B$. By the rule $(p_n I)$, we get $\vdash \lambda z.N : p_n$. By Proposition 6.6, we have $\vdash M : p_n$. $\square$

Now we complete the proof of the characterization theorem.

Proof of Theorem 5.2. The implication from the right-hand side to the left-hand side is proved by Proposition 6.16. The implication from the left-hand side to the right-hand side is proved by Proposition 7.2. $\square$

Example. We will think linear hereditary permutators, where every node has at most one child node. They are described by P and P_m below and typed by $A_n \to A_n$ below. We use Y given in Section 3. Let

$$Q = Y(\lambda fxy.x(fy)),$$
$$P = \lambda x_0.Qx_0.$$

We have $Qx_0 \to_\beta^* \lambda x_1.x_0(Qx_1) \to_\beta^* \lambda x_1.x_0(\lambda x_2.x_1(Qx_2)) \to_\beta^* \lambda x_1.x_0(\lambda x_2.x_1(\lambda x_3.x_2(Qx_3))) \to_\beta^* \dots$. Then $\mathrm{BT}(P)$ is

$$\begin{array}{l} \lambda x_0 x_1.x_0 \\ \quad | \\ \lambda x_2.x_1 \\ \quad | \\ \lambda x_3.x_2 \\ \quad | \\ \quad \vdots \end{array}$$

and P is a hereditary permutator.

Let

$$A_0 = \Omega,$$
$$A_{n+1} = A_n \to q_{n+1}.$$

Define P_m by

$$P_0 = \lambda z.z,$$
$$P_{n+1} = \lambda zx_1.z(P_n x_1).$$

They are finite hereditary permutators. Then we have $\vdash P : A_n \to A_n$ and $\vdash P_m : A_n \to A_n$ for all n. On the other hand, we can show that if $\vdash M : A_n \to A_n$ for all n, then we have $\mathrm{BT}(M) = \mathrm{BT}(P)$ or $M =_\beta P_m$ for some m.

8. Normalizing Terms in Infinite λ-Calculus

We will discuss infinite λ-calculus [8, 1]. We will show that any set of normalizing λ-terms in infinite λ-calculus is not recursively enumerable if it contains some term having a computable infinite path.

We will give the definition of the infinite λ-calculi Λ^{abc} in [8], where a, b, c are parameters ranging over $\{0, 1\}$.

Definition 8.1 A *position*, denoted by u, is defined to be a finite string of positive integers.

For a λ-term M and a position u, the subterm $M|u$ of M is defined by induction on u by

$$\begin{aligned}
&M|\langle\rangle = M,\\
&(\lambda x.M)|1 \cdot u = M|u,\\
&(MN)|1 \cdot u = M|u,\\
&(MN)|2 \cdot u = N|u.
\end{aligned}$$

The depth $D^{abc}(M, u)$ of the subterm $M|u$ in M is defined by induction on u by:

$$\begin{aligned}
&D^{abc}(M, \langle\rangle) = 0,\\
&D^{abc}(\lambda x.M, 1 \cdot u) = a + D^{abc}(M, u),\\
&D^{abc}(MN, 1 \cdot u) = b + D^{abc}(M, u),\\
&D^{abc}(MN, 2 \cdot u) = c + D^{abc}(N, u).
\end{aligned}$$

The distance $d^{abc}(M, N)$ for $M, N \in \Lambda$ is defined as 0 if $M = N$, and $\frac{1}{2^l}$ if $M \neq N$ and l is the minimum of $D^{abc}(M, u)$ where $M|u$ and $N|u$ are both defined, and they are distinct variables, or of different syntactic types.

The set Λ^{abc} is defined as $\{(M_0, M_1, M_2, \ldots) | M_i \in \Lambda, \forall\epsilon > 0 \exists n \forall i, j \geq n(d^{abc}(M_i, M_j) < \epsilon)\}$ where $(M_0, M_1, M_2, \ldots)$ is a countably infinite sequence of λ-terms.

The equality $(M_0, M_1, M_2, \ldots) \equiv_{abc} (N_0, N_1, N_2, \ldots)$ on Λ^{abc} is defined by $\lim_{n\to\infty} d^{abc}(M_n, N_n) = 0$.

Remark. (1) d^{abc} is proved to be actually a distance [8].

(2) The quotient set $\Lambda^{abc}/\equiv_{abc}$ is the completion of Λ with the distance d^{abc}.

Definition 8.2 For $M \in \Lambda$ and $N \in \Lambda^{abc}$, $M \to^\infty_{abc} N$ is defined to hold if $N = (M_0, M_1, \ldots)$, $M = M_0 \to_\beta M_1 \to_\beta \ldots$, and $\lim_{i\to\infty} d_i = \infty$ where d_i is the depth of the redex for $M_i \to_\beta M_{i+1}$.

Proposition 8.3 *If we have $M_0 \to_\beta M_1 \to_\beta \ldots$ and $\lim_{i\to\infty} d_i = \infty$ where d_i is the depth of the redex for $M_i \to_\beta M_{i+1}$, then $(M_0, M_1, \ldots)$ is in Λ^{abc}.*

Proof. For a given ϵ, take k such that $\frac{1}{2^k} < \epsilon$. From $\lim d_i = \infty$, there is n_0 such that $d_i > k$ for all $i \geq n_0$. Then for all $i, j \geq n_0$, we have $d^{abc}(M_i, M_j) < \frac{1}{2^k} < \epsilon$. $\square$

Definition 8.4 We will abbreviate $D^{abc}(M, u)$, $d^{abc}(M, N)$, Λ^{abc}, $\equiv_{abc}$, and $\to^\infty_{abc}$ as $D(M, u)$, $d(M, N)$, Λ^∞, $\equiv_\infty$, and $\to^\infty$ respectively, when they are not ambiguous.

$M \to^{\leq\infty} N$ is defined to hold for $M \in \Lambda$ and $N \in \Lambda^\infty$ if (1) $M \to^\infty N$, or (2) $M \to^*_\beta L$ and $N \equiv_\infty (L, L, \ldots)$.

We say that $M \in \Lambda^\infty$ has a redex when M is $(M_0, M_1, \ldots)$ and we have u and n such that $M_i|u = (\lambda x.P_i)Q_i$ for all $i \geq n$. $M \in \Lambda^\infty$ is a normal form if M does not have any redex. We say that $M \in \Lambda$ is normalizing in Λ^∞ when there is a normal form $N \in \Lambda^\infty$ such that $M \to^{\leq\infty} N$. $\mathrm{NF}_{\leq\infty}$ is defined to be the set $\{M \in \Lambda | M \text{ is normalizing in } \Lambda^\infty\}$. $M =_\infty N$ is defined to hold for $M, N \in \Lambda$, if $M \to^\infty L_1$ and $N \to^\infty L_2$ and $L_1 \equiv_\infty L_2$ for some L_1, L_2. We say $M_0 \to_\beta M_1 \to_\beta M_2 \to_\beta \ldots$ converges to a normal form when $(M_0, M_1, M_2, \ldots)$ is a normal form in Λ^∞.

Definition 8.5 We fix a fresh variable z. A λ-term C is called a context term if C has one free occurrence of the variable z. We write $C[M]$ for $C[x := M]$. We also write $C_1C_2 \ldots C_n[M]$ for $C_1[C_2[\ldots[C_n[M]]\ldots]]$ when $C_1, \ldots, C_n$ are context terms.

The following claim holds for any parameters a, b, c.

Theorem 8.6 *Suppose $S \subseteq \mathrm{NF}_{\leq\infty}$, S is closed under $=_\infty$, and there exists $M \in S$ such that*

(1) there are λ-terms P_i and context terms C_i $\quad(i \geq 0)$ such that $P_0 = M$, $C_0 = z$, and $P_0 \to^*_\beta C_1[P_1] \to^*_\beta C_1C_2[P_2] \to^*_\beta \ldots$ converges to a normal form where $C_1 \ldots C_n[P_n] \to^*_\beta C_1 \ldots C_{n+1}[P_{n+1}]$ is obtained by $P_n \to^*_\beta C_{n+1}[P_{n+1}]$ in $C_1 \ldots C_n$ $\quad(\forall n \geq 0)$,

(2) $D(C_n, u) > 0$ if $C_n|u = z$ $\quad(\forall n > 0)$,

(3) there is a λ-term F such that $F\overline{n} =_\beta \lambda z.C_n$ $\quad(\forall n \geq 0)$.

Then the set S is not recursively enumerable.

We say M has a computable infinite path when M satisfies the above conditions (1) to (3). Then a computation tree of M has the infinite path consisting of the nodes $C_0[z], C_0C_1[z], C_0C_1C_2[z], \ldots$ by (1). The depths of those nodes increase by (2) and this path is computable by (3). This theorem shows that a $=_\infty$-closed set of normalizing terms in Λ^∞ is not recursively enumerable if it contains some term having a computable infinite path.

Examples. (1) $\mathrm{NF}_{\leq\infty}$ is not recursively enumerable.

(2) The set $\{M \in \Lambda | M \to^\infty N \text{ for some normal } N\}$ of normalizing terms by infinite reduction is not recursively enumerable.

We will prove this theorem in this section after some preparation.

We will use the universal function u and its representation U as well as $\mathrm{S}, \overline{n}, Y$ and Δ defined in Section 3.

The next lemma is a standard result in λ-calculus where $\to_h$ is the head reduction, and $\to_i$ is the inner reduction defined as a β-reduction that is not the head reduction.

Lemma 8.7 *If $M \to_\beta^* N$ includes n steps of head reduction, then there is L such that $M \to_h^* L \to_i^* N$ and $M \to_h^* L$ has $\geq n$ steps.*

Lemma 8.8 *If $M_0 \to_\beta M_1 \to_\beta \ldots$ converges to a normal form, the depth of every redex is greater than k, and $M_0 \to_\beta^* L$, then L does not have any redex at depth $\leq k$.*

Proof. Assume L has some redex at depth $\leq k$. We will show contradiction. We can suppose any other term among $M \to_\beta^* L$ except L does not have a redex at depth $\leq k$ and L has a redex at depth k.

Let $M_\infty = (M_0, M_1, \ldots)$. Let the redex at depth k in L be $(\lambda x.Q)R$. Then we have the corresponding $P_0 R_0$ in M such that $P_0 \to_\beta^* \lambda x.Q$ in $M \to_\beta^* L$.

Case 1 where $b = 0$. The depth of P_0 in M is k. If P_0 is a redex, then it remains in M_∞ and it contradicts to the normality of M_∞. Therefore P_0 is not a redex. Hence $P_0 = \lambda x.Q_1$ for some Q_1. Then we have contradiction since M has the redex $(\lambda x.Q_1)R_0$ and it remains in M_∞.

Case 2 where $b = 1$.

Case 2.1 where P_0 has a normal form in Λ. Let $P_0 \to_\beta^* P_1$ for a normal term P_1. By Church-Rosser, we have $P_1 = \lambda x.P_2$ for some P_2. Hence we have contradiction since $P_0 \to_\beta^* P_1$ is eventually done in $M_0 \to_\beta M_1 \to_\beta \ldots$ and M_∞ has the redex $(\lambda x.P_2)R_{1,i}$ for some $R_{1,i}$ for all $i \geq n$ for some n.

Case 2.2 where P_0 does not have any normal form in Λ. Since M_∞ is normal, we have $P_0 \to_\beta P_1 \to_\beta \ldots$ done in $M_0 \to_\beta M_1 \to_\beta \ldots$ such that $P_\infty = (P_0, P_1, \ldots)$ is in Λ^∞ and normal.

Case 2.2.1 where $P_0 \to_\beta P_1 \to_\beta \ldots$ contains only finitely many steps of head reduction. There is n such that P_n is a head normal form. Then $P_n =_\beta \lambda x.Q$. Hence $P_n = \lambda x.R_1$ for some R_1. Then we have m and $R_{2,i}$ such that $P_i = \lambda x.R_{2,i}$ for $i \geq m$, which contradicts to the normality of M_∞.

Case 2.2.2 where $P_0 \to_\beta P_1 \to_\beta \ldots$ contains infinitely many steps of head reduction. By Lemma 8.7 for $P_0 \to_\beta^* \lambda x.Q$, we have $P_0 \to_h^{n_0} \lambda x.Q_0 \to_i^* \lambda x.Q$ for some n_0, Q_0. Then we have n_1 such that $P_0 \to_\beta^* P_{n_1}$ contains at least n_0 steps of head reduction. By Lemma 8.7, we have $P_0 \to_h^{n_0 \leq} Q_1 \to_i^* P_{n_1}$ for some Q_1. Hence $Q_1 = \lambda x.Q_2$ for some Q_2. Therefore $P_{n_1} = \lambda x.Q_3$ for some Q_3. Hence we have $P_i = \lambda x.Q_{4,i}$ for some $Q_{4,i}$ for $i \geq n_1$, which contradicts to the normality of M_∞. □

Proof of Theorem 8.6. Let $P_0 \to_\beta^* C_1[P_1] \to_\beta^* C_1C_2[P_2] \to_\beta^* \ldots$ be $M_0 \to_\beta M_1 \to_\beta \ldots$ and M_∞ be $(M_0, M_1, \ldots)$. Then M_∞ is in Λ^∞ and normal. Let $M_{p_n} = C_1 \ldots C_n[P_n]$.

Suppose w is a fresh variable. Let

$$T = Y(\lambda txy.Uxy(\lambda w.Fy(tx(\mathrm{S}y)))(\Delta\Delta)).$$

First we will show that $e \in$ PPR implies $T\overline{e}\overline{0} \in S$. Let Q_n be $T\overline{e}\overline{n}$. Let $L_0 \to_\beta L_1 \to_\beta L_2 \to_\beta \ldots$ be $Q_0 \to_\beta^* C_1[Q_1] \to_\beta^* C_1C_2[Q_2] \to_\beta^* \ldots$ where $C_1 \ldots C_n[Q_n] \to_\beta^* C_1 \ldots C_{n+1}[Q_{n+1}]$ is obtained by $Q_n \to_\beta^* C_{n+1}[Q_{n+1}]$ in $C_1 \ldots C_n$. Let L_∞ be $(L_0, L_1, \ldots)$. Then we have $L_\infty \in \Lambda^\infty$. Let $L_{q_n} = C_1 \ldots C_n[Q_n]$. For any given $\epsilon > 0$, we have k such that $\frac{1}{2^k} < \frac{\epsilon}{3}$. From (2), we have $D(C_1 \ldots C_k[z], u) \geq k$ if $C_1 \ldots C_k[z]|u = z$. Hence $d(M_{p_k}, L_{q_k}) \leq \frac{1}{2^k}$. Let n_0 be $\max(p_k, q_k)$. For all $i \geq n_0$, we have $d(M_i, L_i) \leq d(M_i, M_{p_k}) + d(M_{p_k}, L_{q_k}) + d(L_{q_k}, L_i) \leq \frac{1}{2^k} + \frac{1}{2^k} + \frac{1}{2^k} < \epsilon$. Hence $M_\infty \equiv_\infty L_\infty$. Since S is closed under $=_\infty$, we have $T\overline{e}\overline{0} \in S$.

Next we will show that $T\overline{e}\overline{0} \in S$ implies $e \in$ PPR. Assume $e \notin$ PPR. We will show a contradiction. There is m_0 such that $\{e\}^{pr}(m_0) = 0$ and $\{e\}^{pr}(m) > 0$ for all $m < m_0$. Then we have $T\overline{e}\overline{0} \to_\beta^* C_1 \ldots C_{m_0-1}[\Delta\Delta]$. Let $C_1 \ldots C_{m_0-1}[z]|u = z$ and $k = D(C_1 \ldots C_{m_0-1}[z], u)$. Since M_∞ is normal, $C_1 \ldots C_{m_0-1}[z]$ is normal.

Case 1 where $T\overline{e}\overline{0}$ has a normal form in Λ. Suppose $T\overline{e}\overline{0} \to_\beta^* L$ and L is normal. By Church-Rosser, $C_1 \ldots C_{m_0-1}[\Delta\Delta] \to_\beta^* L$, which leads to contradiction.

Case 2 where $T\overline{e}\overline{0}$ does not have any normal form in Λ. Since $T\overline{e}\overline{0} \in \mathrm{NF}_{\leq\infty}$, we can suppose $T\overline{e}\overline{0} = L_0 \to_\beta L_1 \to_\beta \ldots$, $L_\infty = (L_0, L_1, \ldots)$ is in Λ^∞ and normal. Let the depth of the redex for $L_i \to_\beta L_{i+1}$ be d_i. There is n_0 such that $d_i > k$ for all $i \geq n_0$. By Church-Rosser, there is R such that $C_1 \ldots C_{m_0-1}[\Delta\Delta] \to_\beta^* R$ and $L_{n_0} \to_\beta^* R$. Hence we have $L_{n_0} \to_\beta^* C_1 \ldots C_{m_0-1}[\Delta\Delta]$. Using Lemma 8.8 to $L_{n_0} \to_\beta L_{n_0+1} \to_\beta \ldots$ and $C_1 \ldots C_{m_0-1}[\Delta\Delta]$, $C_1 \ldots C_{m_0-1}[\Delta\Delta]$ does not have any redex at depth $\leq k$, which leads to contradiction.

Hence we have shown that $e \in$ PPR if and only if $T\overline{e}\overline{0} \in S$.

If S were recursively enumerable, the set $\{e | T\overline{e}\overline{0} \in S\}$ would be also recursively enumerable, but this set is the same as PPR, so PPR would be also recursively enumerable, which would contradict to Proposition 3.5. Therefore S is not recursively enumerable. □

9. Incompleteness of Stream Types

By the same technique used in Section 8, we will also show incompleteness of stream types. We will discuss with an example of binary streams.

Let us think the simply typed lambda calculus with the types bool and stream, which represent the boolean and the bit stream respectively. Its terms are defined by

$$M ::= x|0|1|\lambda x.M|MM|(M, M)|\pi_0 M|\pi_1 M$$

where $0, 1$ are bits, (M_1, M_2) is a pair, and π_0 and π_1 are the first and the second projections respectively.

We will use Λ^0 for the set of closed terms. Its types are defined by

$$A ::= X|\text{bit}|\text{stream}|A \to A|A \times A.$$

Its inference rules are (Ass), $(\to I)$, $(\to E)$ with the standard introduction and elimination rules for $\times$ and

$$\frac{}{0 : \text{bit}}\ (\text{bit}I_1) \qquad \frac{}{1 : \text{bit}}\ (\text{bit}I_2)$$

$$\frac{M : \text{bit} \quad 0 : A \quad 1 : A}{M : A}\ (\text{bit}E)$$

$$\frac{M : A \quad \begin{array}{c}[x : A]\\ \vdots \\ \pi_0 x : \text{bit}\end{array} \quad \begin{array}{c}[x : A]\\ \vdots \\ \pi_1 x : A\end{array}}{M : \text{stream}}\ (\text{stream}I)$$

$$\frac{M : \text{stream}}{\pi_0 M : \text{bit}}\ (\text{stream}E_1) \qquad \frac{M : \text{stream}}{\pi_1 M : \text{stream}}\ (\text{stream}E_2)$$

We will use $M \to_\beta N$ for the one-step reduction obtained from the one defined in Section 2 by adding

$$\pi_0(M, N) \to_\beta M,$$
$$\pi_1(M, N) \to_\beta N.$$

We will also use $=_\beta$ obtained from this β-reduction.

Our intended interpretation of types is set-theoretic. A type means a set of closed terms closed under $=_\beta$. An environment ρ is a map from type variables to $=_\beta$-closed subsets of Λ^0. The interpretation $[A]\rho$ of a type A is defined by induction on A by

$$\begin{aligned}
&[X]\rho = \rho(X),\\
&[\text{bit}]\rho = \text{BIT},\\
&[\text{stream}]\rho = \text{STREAM},\\
&[A \to B]\rho = [A]\rho \to [B]\rho,\\
&[A \times B]\rho = [A]\tilde{\times}[B],
\end{aligned}$$

where

$$\begin{aligned}
&\text{BIT} = \{M \in \Lambda^0 | M =_\beta 0 \text{ or } M =_\beta 1\},\\
&\text{STREAM} = \{M \in \Lambda^0 | \pi_0(\pi_1^n M) \in \text{BIT} \quad (\forall n)\},\\
&S_1 \tilde{\times} S_2 = \{M \in \Lambda^0 | M =_\beta (N, L), N \in S_1, L \in S_2\}.
\end{aligned}$$

We can prove that the type bit characterizes the set BIT. That is, M : bit is provable in the system if and only if M is in BIT for all $M \in \Lambda^0$.

On the other hand, the type stream does not characterize the set STREAM. That is, the set $\{M \in \Lambda^0 | M : \text{stream is provable}\}$ is not equal to STREAM. Indeed any formal system with a recursively enumerable language and a recursively enumerable set of inference rules cannot characterize the set STREAM. This fact immediately follows from the next theorem.

Theorem 9.1 (Incompleteness of Streams) *The set* STREAM *is not recursively enumerable.*

This is prove by letting

$$T = Y(\lambda txy.(Uxy(\lambda w.(0, tx(\mathrm{S}y)))(\Delta\Delta))$$

and showing that e is in PPR iff $T\overline{e}\overline{0}$ is in STREAM.

This discussion can be extended to show incompleteness of general coinductive types.

Acknowledgments

We would like to thank Mariangiola Dezani-Ciancaglini and the editorial board of the TLCA list of open problems for comments. We would also like to thank Koji Nakazawa for careful reading a draft of this paper.

References

[1] A. Berarducci, M. Dezani-Ciancaglini, Infinite lambda-calculus and Types, Theoretical Computer Science 212 (1999) 29–75.

[2] J.A. Bergstra and J.W. Klop, Invertible terms in the lambda calculus, *Theoretical Computer Science* **11** (1980) 19–37.

[3] K. Bruce, R. Di Cosmo, and G. Longo, Provable isomorphisms of types, *Mathematical Structures in Computer Science* **2** (2) (1992) 231–247.

[4] M. Dezani-Ciancaglini, Characterization of normal forms possessing inverse in the lambda-beta-eta-calculus, *Theoretical Computer Science* **2** (1976) 323–337.

[5] M. Dezani-Ciancaglini, F. Honsell, and Y. Motohama, Compositional Characterization of λ-terms using Intersection Types, *Theoretical Computer Science* **340** (3) (2005) 459–495.

[6] R. Hasegawa, L. Paolini, and P. Urzyczyn, The TLCA list of Open Problems, `http://tlca.di.unito.it/opltlca/`, 2007.

[7] J.R. Hindley and J.P. Seldin, *Introduction to Combinators and λ-Calculus* (Cambridge University Press, Cambridge, 1986).

[8] J.R. Kennaway, J.W. Klop, M.R. Sleep, and F.J de Vries, Infinitary lambda calculus, Theoretical Computer Science 175 (1) (1997) 93–125.

[9] M. Tatsuta and M. Dezani-Ciancaglini, Normalisation is Insensible to lambda-term Identity or Difference, In: Proceedings of Twenty First Annual IEEE Symposium on Logic in Computer Science (2006) 327–336.