数理解析研究所講究録
1533 巻 2007 年 64-85

CPS-translation as adjoint
– extended abstract –

Ken-etsu Fujita (藤田 憲悦)
Gunma University (群馬大学)
fujita@cs.gunma-u.ac.jp

Abstract

We show that there exist translations between polymorphic λ-calculus and a subsystem of minimal logic with existential types, which form a Galois insertion (embedding). The translation from polymorphic λ-calculus into the existential type system is the so-called call-by-name CPS-translation that can be expounded as an adjoint from the neat connection. The duality appears not only in the reduction relations but also in the proof structures such as paths between the source and the target calculi. From a programming point of view, this result means that abstract data types can interpret polymorphic functions under the CPS-translation. We may regard abstract data types as a dual notion of polymorphic functions.

1 Introduction

Galois connections arise, even if we do not aware of, in many parts of computer science [8, 12]. For instance, examples from logics are demonstrated in Backhouse [1], where provability or implication relation is a partial order on the set of formulae. Other kinds of examples come from reduction systems, whcih are shown by Danvy-Lawall [2] and Sabry-Wadler [16], where reduction relation forms a preorder over terms.

On the other hand, the term CPS-translation, in general, denotes a program translation method into continuation passing style that is the meaning of the program as a function taking the rest of the computation. The method has been studied for program transformation, definitional interpreter and denotational semantics [13].

We prove that there exist translations between polymorphic λ-calculus $\lambda 2$ (Girard-Reynolds) and a subsystem of minimal logic $\lambda^{\exists}$ with existential types, which form a Galois connection and moreover a Galois insertion (embedding). The translation from $\lambda 2$ into $\lambda^{\exists}$ is the so-called call-by-name CPS-translation [10, 15] that can be expounded as the adjoint of the inverse translation. From a programming point of view, this result also means that abstract data types [7] can interpret polymorphic functions under the CPS-translation. We may regard the notion of abstract data types as a dual notion of polymorphic functions.

Our main interest is a neat connection and proof duality between polymorphic types (2nd order universally quantified formulae) and existential types (2nd order existentially quantified formulae). It is logically quite natural like de Morgan's duality, and computationally still interesting, since dual of polymorphic functions with universal type can be regarded as abstract data types with existential type [7]. Although one can guess the existence of such a duality from the work of Selinger [14], instead of classical systems like [9, 14, 17], even intuitionistic systems can enjoy that polymorphic types can be interpreted by existential types. That is, computationally polymorphic function with universal type $\forall X.A$ can be interpreted by abstract data types with existential type, such that the parametric polymorphic function $\lambda X.M$ for X can be viewed, under the CPS-translation $*$, as an abstract data type $(\lambda X.M)^*$ for X, which is waiting for an implementation with type $\exists X.A^* = (\forall X.A)^*$. This interpretation also contains proof duality, such that the universal formulae introduction rule is interpreted by the use of the existential formulae elimination rule, and the universal elimination by the existential introduction. Moreover, with respect to reduction relations, we established not only a Galois connection but also a Galois insertion (embedding) from polymorphic λ-calculus (Girard-Reynolds) into the calculus with existential types. From the neat connection between the calculi, the fundamental properties such as normalization and Church-Rosser are related each other.

The paper is organized as follows: Section 2 provides our source and target calculi, respectively denoted by $\lambda 2$ and $\lambda^{\exists}$. Section 3 is devoted to the CPS-translation $*$ from $\lambda 2$ into $\lambda^{\exists}$. Here we demonstrate that the CPS-translation can be expounded as a lower adjoint of the inverse translation. Then the translations constitute a Galois insertion (embedding) from $\lambda 2$ into $\lambda^{\exists}$. Section 4 gives typing relation correspondence between the calculi and proof duality between sequences of formulae called paths.

2 Source and target calculi

2.1 Source calculus: $\lambda 2$

We introduce our source calculus of 2nd order λ-calculus (Girard-Reynolds), denoted by $\lambda 2$. For simplicity, we first adopt its domain-free style, and next the Church-style when we discuss proof terms.

Definition 1 (Types)

$$A ::= X \mid A \Rightarrow A \mid \forall X.A$$

Definition 2 ((Pseudo)$\lambda 2$-terms)

$$\Lambda 2 \ni M ::= x \mid \lambda x.M \mid \lambda x.M \mid MM \mid \lambda X.M \mid MA$$

Definition 3 (Reduction rules) (β) $(\lambda x.M_1)M_2 \to M_1[x := M_2]$

(η) $\lambda x.Mx \to M,$ *if* $x \notin FV(M)$

(β_t) $(\lambda X.M)A \to M[X := A]$

(η_t) $\lambda X.MX \to M,$ *if* $X \notin FV(M)$

$FV(M)$ denotes a set of free variables in M.

We write $\twoheadrightarrow_{\lambda 2}$ for the compatible relation obtained from the reflexivie and transitive closure of the one step reduction relation, and $\twoheadrightarrow^{+}_{\lambda 2}$ for that from the transitive closure. In particular, $\twoheadrightarrow_R$ denotes the subrelation of $\twoheadrightarrow$ restricted to the reduction rules $R \subseteq \{\beta, \eta, \beta_t, \eta_t\}$. We may write simply (β) for either (β) or (β_t), and (η) for either (η) or (η_t), if clear from the context. We employ the notation $\equiv$ to indicate the syntactic identity under renaming of bound variables.

2.2 Target calculus: $\lambda^{\exists}$

We next define our target calculus denoted by $\lambda^{\exists}$, which is logically a subsystem of minimal logic consisting of constant $\bot$, negation, conjunction and 2nd order existential quantification[1].

Definition 4 (Types)

$$A ::= \bot \mid X \mid \neg A \mid A \wedge A \mid \exists X.A$$

Definition 5 ((Pseudo)$\lambda^{\exists}$-terms)

$$\begin{aligned} \Lambda^{\exists} \ni M \ ::= \ & x \mid \lambda x.M \mid MM \mid \langle M, M\rangle \mid \mathtt{let}\ \langle x, x\rangle = M\ \mathtt{in}\ M \\ & \mid \langle A, M\rangle \mid \mathtt{let}\ \langle X, x\rangle = M\ \mathtt{in}\ M \end{aligned}$$

Definition 6 (Reduction rules) (β) $(\lambda x.M_1)M_2 \to M_1[x := M_2]$

(η) $\lambda x.Mx \to M,\ \textit{if}\ x \notin FV(M)$

$(\mathtt{let}_{\wedge})$ $\mathtt{let}\ \langle x_1, x_2\rangle = \langle M_1, M_2\rangle\ \mathtt{in}\ M \to M[x_1 := M_1, x_2 := M_2]$

$(\mathtt{let}_{\wedge_\eta})$ $\mathtt{let}\ \langle x_1, x_2\rangle = M_1\ \mathtt{in}\ M[z := \langle x_1, x_2\rangle] \to M[z := M_1],$

$\textit{if}\ x_1, x_2 \notin FV(M)$

$(\mathtt{let}_{\exists})$ $\mathtt{let}\ \langle X, x\rangle = \langle A, M_1\rangle\ \mathtt{in}\ M \to M[X := A, x := M_1]$

$(\mathtt{let}_{\exists_\eta})$ $\mathtt{let}\ \langle X, x\rangle = M_1\ \mathtt{in}\ M[z := \langle X, x\rangle] \to M_2[z := M_1],$ $\quad \textit{if}\ X, x \notin FV(M_2)$

We also write simply $(\mathtt{let})$ for either $(\mathtt{let}_{\wedge})$ or $(\mathtt{let}_{\exists})$, and $(\mathtt{let}_{\eta})$ for $(\mathtt{let}_{\wedge_\eta})$ or $(\mathtt{let}_{\exists_\eta})$. Similarly we write $\twoheadrightarrow_{\lambda^{\exists}}$ and $\twoheadrightarrow^{+}_{\lambda^{\exists}}$ as done for $\lambda 2$.

[1]For further introduction of the CPS target calculus $\lambda^{\exists}$ with let-expressions, see also the previous version [4] of this paper.

3 CPS-translation and Galois connection

3.1 CPS-translation * from $\Lambda 2$ into $\Lambda^{\exists}$

We define a translation, so-called modified CPS-translation * from pseudo $\lambda 2$-terms into pseudo $\lambda^{\exists}$-terms, which preserves not only reduction relations but also typing relations introduced later. In each case, a fresh and free variable a is introduced, which is called a continuation variable.

Definition 7 *1.* $x^* = xa$

2. $(\lambda x.M)^* = \texttt{let}\ \langle x, a\rangle = a\ \texttt{in}\ M^*$

3. $(M_1 M_2)^* = \begin{cases} M_1^*[a := \langle x, a\rangle] & \text{for } M_2 \equiv x \\ M_1^*[a := \langle \lambda a.M_2^*, a\rangle] & \text{otherwise} \end{cases}$

4. $(\lambda X.M)^* = \texttt{let}\ \langle X, a\rangle = a\ \texttt{in}\ M^*$

5. $(MA)^* = M^*[a := \langle A^*, a\rangle]$

6. $X^* = X;\quad (A_1 \Rightarrow A_2)^* = \neg A_1^* \wedge A_2^*;\quad (\forall X.A)^* = \exists X.A^*$

Remarked that M^* contains exactly one free occurrence of a continuation variable a, and M^* has neither β-redex nor η-redex. Let $\lambda X.M$ have type $\forall X.A$. Then, under the translation, the parametric polymorphic function $\lambda X.M$ with respect to X becomes an abstract data type $(\lambda X.M)^*$ for X, which is waiting for an implementation a with type $\exists X.A^*$ together with an interface (a signature) with type A^*, i.e., $(\lambda X.M)^*$ is

$$\texttt{abstype}\ X\ \texttt{with}\ a{:}A^*\ \texttt{is}\ a\ \texttt{in}\ M^*$$

in a familiar notation.

Lemma 1 *1. We have $M_1^*[x := \lambda a.M_2^*] \twoheadrightarrow_{\beta\eta} (M_1[x := M_2])^*$.*
In particular, $M_1^*[x := \lambda a.M_2^*] \twoheadrightarrow_{\beta} (M_1[x := M_2])^*$ provided that M_2 is not a variable.

2. If $M_1 \to_\beta M_2$, then $M_1^* \twoheadrightarrow^+_{\beta\eta\texttt{let}} M_2^*$.

3. If $M_1 \to_\eta M_2$, then $M_1^* \twoheadrightarrow^+_{\eta\texttt{let}_\eta} M_2^*$

Proof. By straightforward inductions. □

Proposition 1 *If we have $M_1 \to_{\lambda 2} M_2$, then $M_1^* \twoheadrightarrow^+_{\lambda^{\exists}} M_2^*$.*

Proof. By induction on the derivation. □

3.2 CPS-translation as adjoint

The main problem is how to define or expound an inverse translation. Sabry and Felleisen [15] have defined the universe of CPS terms, to say, $cps(\Lambda) = \{P \mid M^* \twoheadrightarrow P \text{ for some } M \in \Lambda\}$, for mapping canonical CPS terms back to the original ones. In our terminology, for $P \in cps(\Lambda)$, the downset $\downarrow P \stackrel{\text{def}}{=} \{Q \mid P \twoheadrightarrow Q\}$ is a subset of $cps(\Lambda)$, and $\downarrow P_2 \subseteq \downarrow P_1$ if $P_1 \twoheadrightarrow P_2$. From the definition, for each $P \in cps(\Lambda)$ there exists some $M \in \Lambda$ such that $M^* \twoheadrightarrow P$. If we have $M^* \equiv P$, then the inverse of P, denoted by $P^\sharp$, can be defined as M fortunately. Otherwise, we would take an approximation $P_1 \in \downarrow P$ to P such that $P \to P_1$, where $\downarrow P_1 \subseteq \downarrow P$. Then there also exists $M_1 \in \Lambda$ such that $M_1^* \twoheadrightarrow P_1$, and this process could be continued. In order to make the plan workable, we should have such a downclosed set as $cps(\Lambda)$, and moreover an inverse of $P \in cps(\Lambda)$ should be obtained by $P^\sharp \equiv M$ from the inverse image $[\downarrow P]^{-*} = \downarrow M$ for some $M \in \Lambda$. Here we must guarantee that $[\downarrow P]^{-*}$ is principal, i.e., $[\downarrow P]^{-*}$ is generated by a single $M \in \Lambda$ such that $[\downarrow P]^{-*} = \downarrow M$. That is,

for any $P \in cps(\Lambda)$ there uniquely exists $M \in \Lambda$ such that $\downarrow M = [\downarrow P]^{-*}$.

We say that the translation $*$ is monotonic if $M_1 \twoheadrightarrow M_2$ implies $M_1^* \twoheadrightarrow M_2^*$. It is observed that there may not exists $P^\sharp$ for some $P \in cps(\Lambda)$ unless $*$ is monotonic. For instance, assume that $M_1 \to M_2$ but $M_1^* \to P \leftarrow M_2^*$ for some P, and no other reductions are possible. Then $P^\sharp$ cannot be defined along the above. Moreover, for every normal form P_{nf} there should be uniquely exists M with $P_{nf} \equiv M^*$.

In order to give an inverse translation following the plan above, first we provide the mutual inductive definitions, respectively for denotations $Univ$ and continuations $\mathcal{C}$, as follows:

$$\frac{C \in \mathcal{C}}{xC \in Univ} \qquad \frac{C \in \mathcal{C} \quad P \in Univ}{(\lambda a.P)C \in Univ}$$

$$\frac{C \in \mathcal{C} \quad P \in Univ}{\texttt{let } \langle x, a\rangle = C \texttt{ in } P \in Univ} \qquad \frac{C \in \mathcal{C} \quad P \in Univ}{\texttt{let } \langle X, a\rangle = C \texttt{ in } P \in Univ}$$

$$a \in \mathcal{C} \qquad \frac{C \in \mathcal{C}}{\langle x, C\rangle \in \mathcal{C}}$$

$$\frac{C \in \mathcal{C} \quad P \in Univ}{\langle \lambda a.P, C\rangle \in \mathcal{C}} \qquad \frac{C \in \mathcal{C}}{\langle A^*, C\rangle \in \mathcal{C}}$$

We write $\langle R_1, R_2, \ldots, R_n\rangle$ for $\langle R_1, \langle R_2, \ldots, R_n\rangle\rangle$ with $n > 1$, and $\langle R_1\rangle$ for R_1 with $n = 1$. $C \in \mathcal{C}$ is in the form of $\langle R_1, \ldots, R_n, a\rangle$ where R_i $(1 \le i \le n)$ is x, $\lambda a.P$, or A^* with $n \ge 0$. We explicitly mention that $C \in \mathcal{C}$ has exactly one occurence of free variable a such that $C \equiv \langle R_1, \ldots, R_n, a\rangle$ with $n \ge 0$. $P \in Univ$ also has exactly one occurence of free variable a in such C as a proper subterm of P.

Lemma 2 *1. If $P_1 \in Univ$ and $P_1 \twoheadrightarrow_{\lambda\exists} P_2$, then $P_2 \in Univ$.*

2. If $C_1 \in \mathcal{C}$ and $C_1 \twoheadrightarrow_{\lambda\exists} C_2$, then $C_2 \in \mathcal{C}$.

Proof. Let $P, P_1 \in Univ$ and $C, C_1 \in \mathcal{C}$. Then $P[a := C_1], P[x := \lambda a.P_1], P[X := A^*] \in Univ$, and $C[a := C_1], C[x := \lambda a.P_1], C[X := A^*] \in \mathcal{C}$. □

Hence, both $Univ$ and $\mathcal{C}$ are closed under $\twoheadrightarrow_{\lambda\exists}$. Although $\twoheadrightarrow_{\lambda\exists}$ is defined over $\Lambda^{\exists}$, the binary relation $\twoheadrightarrow_{\lambda\exists}$ is well-defined over $Univ$ and $\mathcal{C}$ as well.

We employ a preorder $Q \sqsubseteq P$ defined by $P \twoheadrightarrow Q$, the reflexive and transitive closure $\twoheadrightarrow$ of one step reduction $\rightarrow$. Then an inverse of $P \in Univ$, denoted by $u(P)$ is defined as an upper adjoint (left adjoint) of $*$, as follows:

$u(P) \stackrel{\text{def}}{=} \sup\{M \in \Lambda2 \mid M^* \sqsubseteq P\}$

The existence of sup is not trivial, since $\sqsubseteq$ is a preorder here rather than a partial order in complete lattices [8, 1, 12]. In fact, this definition works well, which can be verified by case analysis on $P \in Univ$, in the following way:

- Case $P \equiv xC \equiv x\langle R_1, \ldots, R_n, a\rangle$ with $n \geq 0$

  From the definition of $*$, $u(P)$ is in the form of $xN_1 \ldots N_n$ for some term or type N_i, where

  - If $R_i \equiv x_i$, then $N_i \equiv x_i$ from the definition of $*$.
  - If $R_i \equiv \lambda a.P_i$, then similarly find the maximum N_i such that $N_i^* \sqsubseteq P_i$.
  - If $R_i \equiv A_i^*$, then we take $N_i \equiv A_i$.

- Case $P \equiv (\lambda a.P')C$

  We have no M such that $M^* \equiv (\lambda a.P')C$. Then we should find the greatest M' such that $M'^* \sqsubseteq P'[a := C] \sqsubseteq (\lambda a.P')C$, where a is a linear variable.

- Case $P \equiv \texttt{let}\ \langle x, a\rangle = C\ \texttt{in}\ P'$ with $C = \langle R_1, \ldots, R_n, a\rangle$ $(n \geq 0)$

  $u(P)$ is in the form of $(\lambda x.M)N_1 \ldots N_n$ for some M and N_i, where we should find the greatest M such that $M^* \sqsubseteq P'$, and:

  - If $R_i \equiv x_i$ then $N_i \equiv x_i$.
  - If $R_i \equiv \lambda a.P_i$ then find the greatest N_i such that $N_i^* \sqsubseteq P_i$.
  - If $R_i \equiv A_i^*$ then $N_i \equiv A_i$.

- Case $P \equiv \texttt{let}\ \langle X, a\rangle = C\ \texttt{in}\ P'$ is handled simiarly.

Here we have a valid induction measure, since continuation variable is linear and we always choose strictly smaller subterms to find an upper adjoint. This definition u is summarized as follows:

1. $u(x) = x$; $u(\lambda a.P) = u(P)$; $u(A^*) = A$
2. $u(x\langle R_1, \ldots, R_n, a\rangle) = x(u(R_1)) \ldots (u(R_n))$
3. $u((\lambda a.P)C) = u(P[a := C])$
4. $u(\texttt{let}\ \langle x, a\rangle = \langle R_1, \ldots, R_n, a\rangle\ \texttt{in}\ P) = (\lambda x.u(P))(u(R_1)) \ldots (u(R_n))$

5. $u(\texttt{let}\ \langle X, a\rangle = \langle R_1, \ldots, R_n, a\rangle\ \texttt{in}\ P) = (\lambda X.u(P))(u(R_1))\ldots(u(R_n))$

where the clause 1 is for R_i such that $\langle R_1, \ldots, R_n, a\rangle \in \mathcal{C}$, and the clause 2 through 5 are for $P \in Univ$.
The discussion above essentially gives a proof to the statement that for any $P \in Univ$, there uniquely exists $M \in \Lambda 2$ such that $\downarrow M = [\downarrow P]^{-*}$.

On the other hand, usually the definition of inverse translation $\sharp$ can be inductively given as follows [6, 5], where we write $C[\ ]$ for $C \in \mathcal{C}$ with a hole $[\ ]$:

Definition 8 *(0)* $x^\sharp = x$; $(\lambda a.P)^\sharp = P^\sharp$; $(A^*)^\sharp = A$

(1) $(xC)^\sharp = C^\sharp[x^\sharp]$

(2) $((\lambda a.P)C)^\sharp = C^\sharp[(\lambda a.P)^\sharp]$

(3) $(\texttt{let}\ \langle x, a\rangle = C\ \texttt{in}\ P)^\sharp = C^\sharp[\lambda x.P^\sharp]$

(4) $(\texttt{let}\ \langle X, a\rangle = C\ \texttt{in}\ P)^\sharp = C^\sharp[\lambda X.P^\sharp]$

(5) $a^\sharp = [\ ]$

(6) $\langle x, C\rangle^\sharp = C^\sharp[[\ ]x^\sharp]$

(7) $\langle \lambda a.P, C\rangle^\sharp = C^\sharp[[\ ](\lambda a.P)^\sharp]$

(8) $\langle A^*, C\rangle^\sharp = C^\sharp[[\ ](A^*)^\sharp]$

Note that we have $C^\sharp = [\ ]R_1^\sharp \ldots R_n^\sharp$ with left associativity, if $C \in \mathcal{C}$ is in the form of $\langle R_1, \ldots, R_n, a\rangle$.

Lemma 3 *1.* $(P[a := C])^\sharp = C^\sharp[P^\sharp]$

2. Let $P, P_1 \in Univ$ *and* $C \in \mathcal{C}$.
$(P[x := \lambda a.P_1])^\sharp = P^\sharp[x := P_1^\sharp]$
$(C[x := \lambda a.P_1])^\sharp = C^\sharp[x := P_1^\sharp]$

Proof. By induction on the structures of P and C. □

Proposition 2 (Inverse translation as adjoint) *For any* $P \in Univ$, *we have* $u(P) = P^\sharp$.

Proof. By induction on the structure of $P \in Univ$. □

In turn, given $\sharp$ as above, a lower adjoint (right adjoint) of $\sharp$ is defined as follows:
$l(M) \stackrel{\text{def}}{=} \inf\{P \in Univ \mid M \sqsubseteq P^\sharp\}$
Then the recursive procedure to find $l(M)$ is provided by case analysis on M.

- Case M of x:

  We have $l(x) = xa \in Univ$, since $x \sqsubseteq x = (xa)^\sharp$.

- Case M of $\lambda x.M'$:
  From $\lambda x.M' \sqsubseteq \lambda x.P^\sharp = (\texttt{let } \langle x, a\rangle = a \texttt{ in } P)^\sharp$, we should find the minimal $P \in \mathit{Univ}$ such that $M' \sqsubseteq P^\sharp$. Then $l(\lambda x.M')$ is in the form of $\texttt{let } \langle x, a\rangle = a \texttt{ in } P$.
- Case M of $M_1 M_2$:
  - Case $M \equiv x M_2$:
    We have $xM_2 \sqsubseteq xR^\sharp = (x\langle R, a\rangle)^\sharp$. Here R is either x' or $\lambda a.P$ for some $P \in \mathit{Univ}$.
    * If $M_2 \equiv x'$, then $l(xx') = x\langle x', a\rangle$ from $x'^\sharp = x'$.
    * Otherwise, find the least $P \in \mathit{Univ}$ such that $M_2 \sqsubseteq P^\sharp$. Then $l(xM_2)$ is in the form of $x\langle P, a\rangle$.
  - Case $M \equiv (\lambda x.M_3)M_2$:
    From $(\lambda x.M_3)M_2 \sqsubseteq (\lambda x.P^\sharp)R^\sharp = (\texttt{let } \langle x, a\rangle = \langle R, a\rangle \texttt{ in } P)^\sharp$, we should find the least P and R, respectively such that $M_3 \sqsubseteq P^\sharp$ and $M_2 \sqsubseteq R^\sharp$. As in the previous case, R is either x' or $\lambda a.P'$ for some $P' \in \mathit{Univ}$. Then $l((\lambda x.M_3)M_2)$ is in the form of $\texttt{let } \langle x, a\rangle = \langle x', a\rangle \texttt{ in } P$ or $\texttt{let } \langle x, a\rangle = \langle P', a\rangle \texttt{ in } P$.
  - Case $M \equiv M_4 M_3 M_2$:
    From $M_4 M_3 M_2 \sqsubseteq P^\sharp R_3^\sharp R_2^\sharp = (P[a := \langle R_3, R_2, a\rangle])^\sharp = ((\lambda a.P)\langle R_3, R_2, a\rangle)^\sharp$, we should find the least P, R_3, R_2, respectively, such that $M_4 \sqsubseteq P^\sharp$, $M_2 \sqsubseteq R_3^\sharp$, and $M_2 \sqsubseteq R_2^\sharp$. Then $l(M_4 M_3 M_2)$ is in the form of $P[a := \langle R_3', R_2', a\rangle]$, where R_i' is either x_i' or $\lambda a.P_i'$ for $P_i' \in \mathit{Univ}$ together with $M_i \sqsubseteq P_i'^\sharp$ for $i = 2, 3$.
- Case M of $\lambda X.M$:
  From $\lambda X.M' \sqsubseteq \lambda X.P^\sharp = (\texttt{let } \langle X, a\rangle = a \texttt{ in } P)^\sharp$, we should find the least $P \in \mathit{Univ}$ such that $M' \sqsubseteq P^\sharp$. Then $l(\lambda X.M')$ is in the form of $\texttt{let } \langle X, a\rangle = a \texttt{ in } P$.
- Case M of $M_1 A$:
  - Case $M \equiv xA$:
    Since $xA \sqsubseteq xA = (x\langle A^*, a\rangle)^\sharp$, we have $l(x\langle A^*, a\rangle) = xA$.
  - Other cases can be confirmed similarly.

That is, the procedure l is summarized in the following, where we write N for either a term or a type:

1. $l(x) = xa$
2. $l(\lambda x.M) = \texttt{let } \langle x, a\rangle = a \texttt{ in } l(M)$
3. $l(\lambda X.M) = \texttt{let } \langle X, a\rangle = a \texttt{ in } l(M)$
4. $l(MN_1 \dots N_n) = l(M)[a := \langle l'(N_1), \dots, l'(N_n), a\rangle]$
   where $l'(x) = x$; $l'(A) = A^*$; $l'(M) = \lambda a.l(M)$ otherwise.

The CPS-translation $*$ is of course the lower adjoint of $\sharp$ from the definitions.

Proposition 3 (CPS-translation as adjoint) *For any $M \in \Lambda 2$, we have $l(M) = M^*$.*

3.3 Galois insertion (embedding)

As expected from the previous propositions, the translations form the so-called Galois connection between $\lambda 2$ and *Univ*.

Lemma 4 *Let $P_1, P_2 \in Univ$.*

1. *If $P_1 \to_\beta P_2$, then $P_1^\sharp \equiv P_2^\sharp$.*
2. *If $P_1 \to_\eta P_2$, then $P_1^\sharp \equiv P_2^\sharp$.*
3. *If $P_1 \to_{\mathtt{let}} P_2$, then $P_1^\sharp \to_\beta P_2^\sharp$.*
4. *If $P_1 \to_{\mathtt{let}_\eta} P_2$, then $P_1^\sharp \to_\eta P_2^\sharp$.*

Proof. By induction on the derivations. □

Lemma 5 *Let $M \in \Lambda 2$ and $P \in Univ$.*

1. *$M^{*\sharp} \equiv M$ and $P \twoheadrightarrow_{\beta\eta} P^{\sharp *}$*
2. *If M is in $\lambda 2$-normal, then M^* is in $\lambda^\exists$-normal.*
   If P is in $\lambda^\exists$-normal, then $P^\sharp$ is in $\lambda 2$-normal.

Proof. By induction on the structures of $M \in \Lambda 2$ and $P \in Univ$. □

Theorem 1 (Galois insertion) *$\langle \Lambda 2, Univ, *, \sharp \rangle$ forms a Galois connection, in particular Galois insertion (embedding) such that $M^{*\sharp} \equiv M$. That is, let $M, M_1, M_2 \in \Lambda 2$ and $P, P_1, P_2 \in Univ$. Then we have the following properties:*

1. *If $M_1 \twoheadrightarrow_{\lambda 2} M_2$ then $M_1^* \twoheadrightarrow_{\lambda^\exists} M_2^*$.*
2. *If $P_1 \twoheadrightarrow_{\lambda^\exists} P_2$ then $P_1^\sharp \twoheadrightarrow_{\lambda 2} P_2^\sharp$.*
3. *If $M^{*\sharp} \twoheadrightarrow_{\lambda 2} M$ and $P \twoheadrightarrow_{\lambda^\exists} P^{\sharp *}$.*

In other words:

$$P^\sharp \twoheadrightarrow_{\lambda 2} M \text{ if and only if } P \twoheadrightarrow_{\lambda^\exists} M^*$$

Proof. From Lemmata 4 and 5. □
We summarize results induced from the discussion above.

Corollary 1 1. *Strong normalization of Univ implies that of $\lambda 2$.*

2. *$\lambda 2$ is weakly normalizing iff Univ is weakly normalizing.*
3. *There exists a one-to-one correspondence between $\lambda 2$-normal forms and Univ-normal forms.*
4. *$\lambda 2$ is Church-Rosser iff Univ is Church-Rosser.*
   We remark that $\Lambda^\exists$ itself is not Church-Rosser.

5. *Given the translation $*$. Then an inverse translation which satisfies the properties of Lemma 4 and Lemma 5 (1) above is unique under renaming of bound variables.*

6. *Let $\downarrow P$ be $\{Q \mid P \twoheadrightarrow_{\lambda^{\exists}} Q\}$ for $P \in Univ$. Then an inverse image of $\downarrow P$ is principal, in the sense that the inverse image of $\downarrow P$ is equal to $\downarrow (P^{\sharp})$ that is generated by a single $P^{\sharp} \in \Lambda 2$.*

7. *Let $\downarrow_{\lambda^{\exists}} [\Lambda 2]^*$ be $\{P \mid M^* \twoheadrightarrow_{\lambda^{\exists}} P$ for some $M \in \Lambda 2\}$. Let $\uparrow_{\beta\eta} [\Lambda 2]^*$ be $\{P \in Univ \mid P \twoheadrightarrow_{\beta\eta} M^*$ for some $M \in \Lambda 2\}$.*

   Then we have $\downarrow_{\lambda^{\exists}} [\Lambda 2]^* \subseteq Univ = \uparrow_{\beta\eta} [\Lambda 2]^*$.

   We remark that $\subseteq$ is strict, for instance,

   $(\lambda a.xa)a \in Univ$, but $(\lambda a.xa)a \notin \downarrow_{\lambda^{\exists}} [\Lambda 2]^*$.

4 Proof duality

4.1 Typing relation correspondence

From now on, we consider proof terms in the Church-style, and so are terms in *Univ* or $\mathcal{C}$. In particular, we write $\lambda a : A^*.P$ for λ-terms in *Univ*, and $\langle A_1^*, C\rangle_{\exists X.A^*}$ for pairs in $\mathcal{C}$. We give type assignment rules for $\lambda 2$ and $\lambda^{\exists}$, respectively, as follows.

$\lambda 2$:

$$\frac{x{:}A \in \Gamma}{\Gamma \vdash x : A}$$

$$\frac{\Gamma, x{:}A_1 \vdash M : A_2}{\Gamma \vdash \lambda x{:}A_1.M : A_1 \Rightarrow A_2}\ (\Rightarrow I) \qquad \frac{\Gamma \vdash M_1 : A_1 \Rightarrow A_2 \quad \Gamma \vdash M_2 : A_1}{\Gamma \vdash M_1 M_2 : A_2}\ (\Rightarrow E)$$

$$\frac{\Gamma \vdash M : A}{\Gamma \vdash \lambda X.M : \forall X.A}\ (\forall I)^* \qquad \frac{\Gamma \vdash M : \forall X.A}{\Gamma \vdash M A_1 : A[X := A_1]}\ (\forall E)$$

where $(\forall I)^*$ denotes the eigenvariable condition $X \notin FV(\Gamma)$.

$\lambda^{\exists}$:

$$\frac{x{:}A \in \Gamma}{\Gamma \vdash x : A}$$

$$\frac{\Gamma, x{:}A \vdash M : \bot}{\Gamma \vdash \lambda x{:}A.M : \neg A}\ (\neg I) \qquad \frac{\Gamma \vdash M_1 : \neg A \quad \Gamma \vdash M_2 : A}{\Gamma \vdash M_1 M_2 : \bot}\ (\neg E)$$

$$\frac{\Gamma \vdash M_1 : A_1 \quad \Gamma \vdash M_2 : A_2}{\Gamma \vdash \langle M_1, M_2\rangle : A_1 \wedge A_2}\ (\wedge I) \qquad \frac{\Gamma \vdash M_1 : A_1 \wedge A_2 \quad \Gamma, x_1{:}A_1, x_2{:}A_2 \vdash M : A}{\Gamma \vdash \mathtt{let}\ \langle x_1, x_2\rangle = M_1\ \mathtt{in}\ M : A}\ (\wedge E)$$

$$\frac{\Gamma \vdash M : A[X := A_1]}{\Gamma \vdash \langle A_1, M\rangle_{\exists X.A} : \exists X.A}\ (\exists I) \qquad \frac{\Gamma \vdash M : \exists X.A \quad \Gamma, x{:}A \vdash M_1 : A_1}{\Gamma \vdash \texttt{let}\ \langle X, x\rangle = M\ \texttt{in}\ M_1 : A_1}\ (\exists E)^\star$$

where $(\exists E)^\star$ denotes the eigenvariable condition $X \notin FV(\Gamma, A_1)$.
The typability problem for $\lambda^\exists$ is decidable, i.e., given Γ and M, we can find A such that $\Gamma \vdash_{\lambda\exists} M : A$. We give a certain typability for terms *à la* Church in *Univ* or $\mathcal{C}$.

Lemma 6 *For $P \in$ Univ and $C \in \mathcal{C}$,*

1. *if we have $\neg\Gamma^*, a{:}A^* \vdash_{\lambda\exists} P : B_1$ then $B_1 \equiv \bot$; and*
2. *if we have $\neg\Gamma^*, a{:}A^* \vdash_{\lambda\exists} C : B_2$ then $B_2 \equiv A_1^*$ for some A_1.*

This lemma means that if the unique variable a has some type in the form of A^* and other free variables in P or C, denoted by x, have type $\neg A_1^*$ for some A_1, then $P \in$ *Univ* has type $\bot$ and $C \in \mathcal{C}$ has type A_2^* for some A_2.

Proof. By simultaneous induction on the structures of P and C.

1. Case of $\neg\Gamma^*, a{:}A^* \vdash_{\lambda\exists} xC : B$

   $B \equiv \bot$, $x : \neg A'^* \in \neg\Gamma^*$, and $\neg\Gamma^*, a{:}A^* \vdash_{\lambda\exists} C : A'^*$ for some A'.

2. Case of $\neg\Gamma^*, a{:}A^* \vdash_{\lambda\exists} (\lambda a'.P)C : B$

   $B \equiv \bot$, $\neg\Gamma^*, a{:}A^* \vdash_{\lambda\exists} \lambda a'.P : \neg B'$, and $\neg\Gamma^*, a{:}A^* \vdash_{\lambda\exists} C : B'$ for some B'.

   By the second induction hypothesis, we have $B' \equiv A'^*$ for some A'.

3. Case of $\neg\Gamma^*, a{:}A^* \vdash_{\lambda\exists} \texttt{let}\ \langle x, a'\rangle = C\ \texttt{in}\ P : B$

   $\neg\Gamma^*, a{:}A^* \vdash_{\lambda\exists} C : A_1 \wedge A_2$, and

   $\neg\Gamma^*, a{:}A^*, x{:}A_1, a'{:}A_2 \vdash_{\lambda\exists} P : B$ for some A_1, A_2.

   From the first induction hypothesis, we have $B \equiv \bot$, and from the second induction hypothesis, we have $A_1 \wedge A_2 \equiv A'^*$ for some A'. From the definition, $A_1 \equiv \neg A_3^*$ and $A_2 \equiv A_4^*$ for some A_3, A_4.

4. Case of $\neg\Gamma^*, a{:}A^* \vdash_{\lambda\exists} \texttt{let}\ \langle X, a'\rangle = C\ \texttt{in}\ P : B$

   $\neg\Gamma^*, a{:}A^* \vdash_{\lambda\exists} C : \exists X.A_1$, and

   $\neg\Gamma^*, a{:}A^*, x{:}A_1 \vdash_{\lambda\exists} P : B$ for some A_1.

   From the first induction hypothesis, we have $B \equiv \bot$, and from the second induction hypothesis, we have $\exists X.A_1 \equiv A_2^*$ for some A_2. From the definition, $A_1 \equiv A_3^*$ for some A_3.

5. Case of $\neg\Gamma^*, a{:}A^* \vdash_{\lambda\exists} a : B$

   We have $B \equiv A^*$.

6. Case of $\neg\Gamma^*, a\!:\!A^* \vdash_{\lambda\exists} \langle x, C\rangle : B$

   $\neg\Gamma^*, a\!:\!A^* \vdash_{\lambda\exists} x : B_1$, and $\neg\Gamma^*, a\!:\!A^* \vdash_{\lambda\exists} C : B_2$ for some B_1, B_2.

   $B_1 \equiv \neg A_1^* \in \neg\Gamma^*$ for some A_1, and $B_2 \equiv A_2^*$ by the second induction hypothesis.

7. Case of $\neg\Gamma^*, a\!:\!A^* \vdash_{\lambda\exists} \langle\lambda a'\!:\!A'^*.P, C\rangle : B$

   $\neg\Gamma^*, a\!:\!A^* \vdash_{\lambda\exists} \lambda a'\!:\!A'^*.P : \neg A'^*$, and $\neg\Gamma^*, a\!:\!A^* \vdash_{\lambda\exists} C : B_2$ for some B_1, B_2. From the second induction hypothesis, we have $B_2 \equiv B_3^*$ for some B_3.

8. Case of $\neg\Gamma^*, a\!:\!A^* \vdash_{\lambda\exists} \langle A_1^*, C\rangle_{\exists X.A_2^*} : B$

   We have $\neg\Gamma^*, a\!:\!A^* \vdash_{\lambda\exists} C : A_2^*[X := A_1^*]$, and $B \equiv \exists X.A_2^*$.

From the above, the following set of typing rules, denoted by $\lambda_U^{\exists}$, is enough for *Univ* and $\mathcal{C}$:

$$\frac{x\!:\!\neg A_1^* \in \neg\Gamma^* \quad \neg\Gamma^*, a\!:\!A^* \vdash C : A_1^*}{\neg\Gamma^*, a\!:\!A^* \vdash xC : \bot}$$

$$\frac{\neg\Gamma^*, a\!:\!A^* \vdash C : \neg A_1^* \wedge A_2^* \quad \neg\Gamma^*, x\!:\!\neg A_1^*, a\!:\!A_2^* \vdash P : \bot}{\neg\Gamma^*, a\!:\!A^* \vdash \mathtt{let}\ \langle x, a\rangle = C\ \mathtt{in}\ P : \bot}\ (\wedge E)$$

$$\frac{\neg\Gamma^*, a\!:\!A^* \vdash C : \exists X.A_1^* \quad \neg\Gamma^*, a\!:\!A_1^* \vdash P : \bot}{\neg\Gamma^*, a\!:\!A^* \vdash \mathtt{let}\ \langle X, a\rangle = C\ \mathtt{in}\ P : \bot}\ (\exists E)^*$$

$$\frac{\neg\Gamma^*, a\!:\!A^* \vdash C : A_1^* \quad \neg\Gamma^*, a\!:\!A_1^* \vdash P : \bot}{\neg\Gamma^*, a\!:\!A^* \vdash (\lambda a\!:\!A_1^*.P)C : \bot}\ (\neg IE)$$

$$\frac{}{\neg\Gamma^*, a\!:\!A^* \vdash a : A^*}$$

$$\frac{x\!:\!\neg A_1^* \in \neg\Gamma^* \quad \neg\Gamma^*, a\!:\!A^* \vdash C : A_2^*}{\neg\Gamma^*, a\!:\!A^* \vdash \langle x, C\rangle : \neg A_1^* \wedge A_2^*}\ (\wedge I_{var})$$

$$\frac{\neg\Gamma^*, a\!:\!A_1^* \vdash P : \bot \quad \neg\Gamma^*, a\!:\!A^* \vdash C : A_2^*}{\neg\Gamma^*, a\!:\!A^* \vdash \langle\lambda a\!:\!A_1^*.P, C\rangle : \neg A_1^* \wedge A_2^*}\ (\wedge I_\lambda)$$

$$\frac{\neg\Gamma^*, a\!:\!A^* \vdash C : A_1^*[X := A_2^*]}{\neg\Gamma^*, a\!:\!A^* \vdash \langle A_2^*, C\rangle_{\exists X.A_1^*} : \exists X.A_1^*}\ (\exists I)$$

where $(\exists E)^*$ denotes the eigenvariable condition $X \notin FV(\Gamma, A_1)$.

Lemma 7 *1.* $\neg\Gamma^*, a\!:\!A^* \vdash_{\lambda\exists} P : \bot$ *if and only if* $\neg\Gamma^*, a\!:\!A^* \vdash_{\lambda_U^{\exists}} P : \bot$

2. $\neg\Gamma^*, a\!:\!A^* \vdash_{\lambda\exists} C : A_1^*$ *if and only if* $\neg\Gamma^*, a\!:\!A^* \vdash_{\lambda_U^{\exists}} C : A_1^*$

Proof. If-part is clear. Only-if-part is by induction on the structures of P and C.

Proposition 4 *1. If we have* $\neg\Gamma^*, a{:}A^* \vdash_{\lambda^{\exists}_U} P : \bot$ *then* $\Gamma \vdash_{\lambda 2} P^\sharp : A$.

2. If we have $\neg\Gamma^*, a{:}A^* \vdash_{\lambda^{\exists}_U} C : A_1^*$ *then* $\Gamma, x{:}A_1 \vdash_{\lambda 2} (xC)^\sharp : A$.

Proof. By simultaneous induction on the derivations.

1. Case of $xC : \bot$

   By the induction hypothesis, we have $\Gamma, y{:}A_1 \vdash C^\sharp[y] : A$. Hence, we have $\Gamma, x{:}A_1 \vdash (xC)^\sharp : A$ from $x{:}A_1 \in \Gamma$.

2. Case of $\texttt{let}\ \langle x, a\rangle = C\ \texttt{in}\ P : \bot$

   By the induction hypotheses, we have $\Gamma, y{:}A_1 \Rightarrow A_2 \vdash C^\sharp[y] : A$ and $\Gamma, x{:}A_1 \vdash P^\sharp : A_2$. Hence, we have $\Gamma \vdash C^\sharp[\lambda x.P^\sharp] : A$.

3. Case of $\texttt{let}\ \langle X, a\rangle = C\ \texttt{in}\ P : \bot$

   By the induction hypotheses, we have $\Gamma, y{:}\forall X.A_1 \vdash C^\sharp[y] : A$ and $\Gamma \vdash P^\sharp : A_1$ where $X \notin FV(\Gamma, A_1)$. Hence, we have $\Gamma \vdash C^\sharp[\lambda X.P^\sharp] : A$.

4. Case of $(\lambda a.P)C : \bot$

   By the induction hypotheses, we have $\Gamma, y{:}A_1 \vdash C^\sharp[y] : A$ and $\Gamma \vdash P^\sharp : A_1$. Hence, we have $\Gamma \vdash C^\sharp[P^\sharp] : A$.

5. Case of $a : A^*$

   We have $x{:}A \vdash (xa)^\sharp : A$, where $(xa)^\sharp = x$.

6. Case of $\langle x, C\rangle : \neg A_1^* \wedge A_2^*$

   By the induction hypothesis, we have $\Gamma, y{:}A_2 \vdash C^\sharp[y] : A$. Hence, we have $\Gamma, z{:}A_1 \Rightarrow A_2 \vdash C^\sharp[zx] : A$ from $x{:}A_1 \in \Gamma$, where $(z\langle x, C\rangle)^\sharp = C^\sharp[zx]$.

7. Case of $\langle \lambda a.P, C\rangle : \neg A_1^* \wedge A_2^*$

   By the induction hypotheses, we have $\Gamma \vdash P^\sharp : A_1$ and $\Gamma, y{:}A_2 \vdash C^\sharp[y] : A$. Hence, we have $\Gamma, z{:}A_1 \Rightarrow A_2 \vdash C^\sharp[zP^\sharp] : A$, where $(z\langle \lambda a.P, C\rangle)^\sharp = \langle \lambda a.P, C\rangle^\sharp[z] = C^\sharp[zP^\sharp]$.

8. Case of $\langle A_2^*, C\rangle_{\exists X.A_1^*} : \exists X.A_1^*$

   By the induction hypothesis, we have $\Gamma, y{:}A_1[X := A_2] \vdash C^\sharp[y] : A$. Hence, we have $\Gamma, z{:}\forall X.A_1 \vdash C^\sharp[zA_2] : A$, where $(z\langle A_2^*, C\rangle)^\sharp = C^\sharp[zA_2]$.

Theorem 2 $\Gamma \vdash_{\lambda 2} M : A$ *if and only if* $\neg\Gamma^*, a{:}A^* \vdash_{\lambda\exists} M^* : \bot$

Proof. If we have $\Gamma \vdash_{\lambda 2} M : A$, then $\neg\Gamma^*, a : A^* \vdash_{\lambda\exists} M^* : \bot$ by induction on the derivation. In turn, if we have $\neg\Gamma^*, a : A^* \vdash_{\lambda\exists} M^* : \bot$, then we also have $\neg\Gamma^*, a : A^* \vdash_{\lambda^{\exists}_U} M^* : \bot$. Hence, from Proposition 4 above, we have $\Gamma \vdash_{\lambda 2} (M^*)^\sharp : A$ where $(M^*)^\sharp \equiv M$.

4.2 Duality on formulae, proofs and paths

Well-known duality like de Morgan's appears on the sets of formulae $Form$ with provability $\vdash$ or logical implication $\Rightarrow$ relation which forms a partial order. Such a duality is characterised as translations between the tuples $\langle Form, \Rightarrow\rangle$. In Gentzen's sequent calculus LK, switching formulae between antecident and succedent gives one example under the translation d: $X^d = X$; $(\neg A)^d = \neg A^d$; $(A \wedge B)^d = A^d \vee B^d$, $(A \vee B)^d = A^d \wedge B^d$, $(\forall x.A)^d = \exists x.A^d$, $(\exists x.A)^d = \forall x.A^d$. Then for $\langle Form, \vdash\rangle$ and $\langle Form, \dashv\rangle$, we have $\Gamma^d \vdash \Delta$ iff $\Gamma \dashv \Delta^d$.

Along this line, another translation is negation $\neg$ between $\langle Form, \Rightarrow\rangle$ and $\langle Form, \Leftarrow\rangle$. Then we have $\neg A \Rightarrow B$ iff $A \Leftarrow \neg B$.

Yet another example of translations known as sectioning (Curry version of binary operators) are given in Backhouse [1], as follows:

$$\begin{aligned} (A\wedge) &\stackrel{\text{def}}{=} X \mapsto A \wedge X \\ (A\Rightarrow) &\stackrel{\text{def}}{=} X \mapsto A \Rightarrow X \end{aligned}$$

Then we have $B^{(A\wedge)} \Rightarrow C$ iff $B \Rightarrow C^{(A\Rightarrow)}$, and commutativity with quantifiers[2]: $(\exists X.B)^{(A\wedge)}$ iff $\exists X.B^{(A\wedge)}$, and $(\forall X.B)^{(A\Rightarrow)}$ iff $\forall X.B^{(A\Rightarrow)}$, where $X \notin FV(A)$. Moreover, $\langle Form, \Rightarrow\rangle$ is a poset, and the supremum can be regarded as existential quantification. We may write a partial order $\sqsubseteq$ instead of $\Rightarrow$. The supremum of the translation $(A\wedge)$ is thought of as the supremum of the range of the translation. Then, in fact, $\sup(A\wedge)$ is given by the following X:

1. For any $B \in Form$, $B^{(A\wedge)} \sqsubseteq X$.

2. For arbitrary $C \in Form$, if $B^{(A\wedge)} \sqsubseteq C$ for any $B \in Form$, then $X \sqsubseteq C$.

That is, we have $\sup(A\wedge) = \exists X.A \wedge X = \top^{(A\wedge)}$, where $\top \equiv \exists X.X$ and $\exists$ commutes with the translation $(A\wedge)$ from the commutativity. Similarly, we have $\inf(A\Rightarrow) = \forall X.A \Rightarrow X = \bot^{(A\Rightarrow)}$ where $\bot \equiv \forall X.X$, as the following X:

1. For any $B \in Form$, $X \sqsubseteq B^{(A\Rightarrow)}$.

2. For arbitrary $C \in Form$, if $C \sqsubseteq B^{(A\Rightarrow)}$ for any $B \in Form$, then $C \sqsubseteq X$.

Not only with provalibity but also with proof terms, Wadler [17] has introduced the dual calculus for classical propositional logic. The previous simple example $\Gamma^d \vdash \Delta \Longleftrightarrow \Gamma \dashv \Delta^d$ might be involved in the dual calculus. An involutive duality on $\lambda\mu$-calculus is revealed on the dual calculus via translations.

The control and co-control categories by Selinger [14] elegantly reveals, as internal languages, duality between call-by-value and call-by-name $\lambda\mu$-calculi with conjunctions and disjunctions.

Here we demonstrate another duality on a sequence of formulae from the viewpoint of proof structures. Following Prawitz [11], we define the notion of paths together with names of inference rules (R). In particular, introduction rules are denoted by (I), and eliminations are by (E).

[2]We refomulate his discussion [1] in the second order intuitionistic logic.

Definition 9 (Path) *A sequence consisting of formulae A_i and inference rules (R_i)*

$$A_1(R_1)A_2(R_2)\dots A_{n-1}(R_{n-1})A_n$$

is defined as a path in the deduction Π of $\lambda 2$ or $\lambda^{\exists}$, as follows:

(i) *A_1 is a top-formula in Π, which is not discharged by an application of $(\wedge E)$ or $(\exists E)$;*

(ii) *A_i $(i < n)$ is not the minor premiss of an application of $(\Rightarrow E)$ or $(\neg E)$, and either*

- *(a) A_i is not the major premiss of $(\wedge E)$ or $(\exists E)$, and A_{i+1} is the formula occurrence immediately below A_i by an application of (R_i), or*
- *(b) A_i is the major premiss of an application (R_i) of $(\wedge E)$ or $(\exists E)$, and A_{i+1} is an assumption discharged by (R_i); and*

(iii) *A_n is either a minor premiss of $(\Rightarrow E)$ or $(\neg E)$, or the end-formula of Π.*

We call a path a main path if the path ends with the end-formula of the deduction. We assign an order to each path π, denoted by $ord(\pi)$. A main path has the order 0. A path that ends with a minor premiss of an application $(\Rightarrow E)$ or $(\neg E)$ has order $n+1$ if the corresponding major premiss of this application belongs to a path with order n. A length of the path $\pi \equiv A_1(R_1)A_2(R_2)\dots A_{n-1}(R_{n-1})A_n$ is defined as n, denoted by $len(\pi)$.

Let χ be either x or X. We simply write $\mathtt{let}\ \langle\chi_1,\chi_2,x_3\rangle = M_1\ \mathtt{in}\ M_2$ for

$$\mathtt{let}\ \langle\chi_1,y\rangle = M_1\ \mathtt{in}\ \mathtt{let}\ \langle\chi_2,x_3\rangle = y\ \mathtt{in}\ M$$

where y is a fresh variable. Similarly, we write $\mathtt{let}\ \langle\chi_1,\dots,\chi_n,x\rangle = M_1\ \mathtt{in}\ M_2$, and so on. Let $M_{nf} \equiv \lambda\chi_1.\dots\lambda\chi_n.xN_1\dots N_m$ be a normal form of $\lambda 2$-terms with $n,m \geq 0$ and N_i is either a term or a type. Then,

$M^*_{nf} \equiv \mathtt{let}\ \langle\chi_1,\dots,\chi_n,a\rangle = a\ \mathtt{in}\ x\langle N^*_1,\dots,N^*_m,a\rangle$

is also normal in $\Lambda^{\exists}$. We analyze the proof structure of M^*_{nf} in terms of paths. We define the following inference rules correspondence between $\Lambda 2$ and $\Lambda^{\exists}$:

$(\Rightarrow I)^* = (\wedge E)$, $(\Rightarrow E)^* = (\wedge I)$, $(\forall I)^* = (\exists E)$, and $(\forall E)^* = (\exists I)$.

Theorem 3 (Proof duality) *Let Π be the normal deduction of $\Gamma \vdash_{\lambda 2} M : A$, and Π^* be the normal deduction of $\neg\Gamma^*, a{:}A^* \vdash_{\lambda^{\exists}} M^* : \bot$. We have a path π of Π, to say:*

$$A_1(E_1)A_2(E_2)\dots A_m(E_m)A_{m+1}(I_{m+1})\dots A_{m+n-1}(I_{m+n})A_{m+n+1}$$

with the proviso that $len(\pi) > 1$ if $ord(\pi) > 0$,
if, and only if, we have a path π^* of Π^* with the same length and $ord(\pi)+1$ order, such that

$$A^*_{m+n+1}(I_{m+n})^*A^*_{m+n-1}\dots(I_{m+1})^*A^*_{m+1}(E_m)^*A^*_m\dots(E_2)^*A^*_2(E_1)^*A^*_1.$$

For a path π beginning with a top-formula A_1 and ending with a conclusion A, we have the corresponding path π^* beginning with a top-formula A^* and ending with A^*_1. The side condition concerns a technical matter. Since the definition says that $(Mx)^* = M^*[a :=$

$\langle x, a\rangle]$, we have no corresponding path to the type of x. Although the definition of the CPS-translation can be simplified as $(Mx)^* = M^*[a := \langle \lambda a.xa, a\rangle]$ for removing the condition, this simplification might involve an extra η-redex. The path consisting of the type of such x is not a main path, i.e., the order is greater than 0 and the length is 1.

Proof. If-part is by induction on the derivation.

1. Case of $M \equiv x$ where $ord(\pi) = 0$ and $len(\pi) = 1$:

$$\frac{x : \neg A^* \quad a : A^*}{xa : \bot}\ (\neg E)$$

Then $\pi^* = A^*$ with $ord(\pi^*) = 1 = ord(\pi) + 1$ and $len(\pi^*) = 1 = len(\pi)$.

2. Case of $M \equiv M_1 M_2$:

$$\frac{\begin{array}{c}\Pi_1\\ M_1 : A \Rightarrow B\end{array} \quad \begin{array}{c}\Pi_2\\ M_2 : A\end{array}}{M_1 M_2 : B}\ (\Rightarrow E)$$

where $\pi = \pi_1(A \Rightarrow B)(\Rightarrow E)B$ with $ord(\pi) = n + 1$. Then we have

$$\begin{array}{c}\dfrac{\dfrac{\begin{array}{c}[a : A^*]\\ \Sigma_2\\ M_2^* : \bot\end{array}}{\lambda a.M_2^* : \neg A^*}\ (\neg I) \quad a : B^*}{\langle \lambda a.M_2^*, a\rangle : \neg A^* \wedge B^*}\ (\wedge I)\\ \Sigma_1\\ M^*[a := \langle \lambda a.M_2^*, a\rangle] : \bot\end{array}$$

where path $\pi^* = B^*(\Rightarrow E)^*(A \Rightarrow B)^*\pi_1^*$ with $ord(\pi^*) = ord(\pi)+1$ from the induction hypothesis $ord(\pi_1^*) = ord(\pi_1) + 1$.

3. Case of $M \equiv \lambda x{:}A.M_1$:

$$\frac{\begin{array}{c}[x : A]\\ \Pi_1\\ M_1 : B\end{array}}{\lambda x{:}A.M_1 : A \Rightarrow B}\ (\Rightarrow I)$$

where $\pi = \pi_1 B(\Rightarrow I)(A \Rightarrow B)$. Then we have

$$\frac{a : \neg A^* \wedge B^* \qquad \begin{array}{c}[x : \neg A^*] \quad [a : B^*]\\ \Sigma_1\\ M_1^* : \bot\end{array}}{\mathtt{let}\ \langle x, a\rangle = a\ \mathtt{in}\ M^* : \bot}\ (\wedge E)$$

where $\pi^* = (A \Rightarrow B)^*(\Rightarrow I)^*B^*\pi_1^*$ with $ord(\pi^*) = ord(\pi) + 1$ from the induction hypothesis.

4. Case of $M \equiv M_1 B$:

$$\frac{\begin{array}{c}\Pi_1\\ M_1 : \forall X.A\end{array}}{M_1 B : A[X := B]}\ (\forall E)$$

where $\pi = \pi_1(\forall X.A)(\forall E)(A[X := B])$. Then we have

$$\begin{array}{c}\dfrac{a : A^*[X := B^*]}{\langle B^*, a\rangle : \exists X.A^*}\ (\exists I)\\ \Sigma_1\\ M_1^*[a := \langle B^*, a\rangle] : \bot\end{array}$$

where $\pi^* = (A[X := B])^*(\forall E)^*(\exists X.A^*)\pi_1^*$ with $order(\pi^*) = order(\pi) + 1$.

5. Case of $M \equiv \lambda X.M_1$:

$$\frac{\begin{array}{c}\Pi_1\\ M_1 : A\end{array}}{\lambda X.M_1 : \forall X.A}\ (\forall I)$$

where $\pi = \pi_1 A(\forall I)(\forall X.A)$. Then we have

$$\frac{a : \exists X.A^* \quad \begin{array}{c}[a : A^*]\\ \Sigma_1\\ M_1^* : \bot\end{array}}{\mathtt{let}\ \langle X, a\rangle = a\ \mathtt{in}\ M_1^* : \bot}\ (\exists E)$$

where $\pi^* = (\forall X.A)^*(\forall I)^*A^*\pi_1^*$ with $ord(\pi^*) = ord(\pi) + 1$.

Only-if-part:

Case of $len(\pi^*) = 1$ where $\pi^* = A^*$.

In this case, we have $M_{nf}^* = xa$, and hence $M_{nf} = x$ and $\pi = A$ with $ord(\pi) = 0$ and $len(\pi) = 1$.

Case of $len(\pi^*) > 1$.

We let

$\pi^* = A_{m+n+1}^*(I_{m+n})^*A_{m+n-1}^* \cdots (I_{m+1})^*A_{m+1}^*(E_m)^*A_m^* \cdots (E_2)^*A_2^*(E_1)^*A_1^*$ and $M_{nf}^* \equiv$ $\mathtt{let}\ \langle \chi_1, \ldots, \chi_n, a\rangle = a\ \mathtt{in}\ x\langle N_1^*, \ldots, N_m^*, a\rangle$, where N_i is either a term or a type.

From π^* and M_{nf}^*, we have the following structure as a part of the normal proof of

$\lambda^{\exists}_{U}$:

$$\dfrac{a : A^*_{m+n+1} \qquad \dfrac{[a : A^*_{m+n}] \qquad \begin{array}{c} \dfrac{[a : A^*_{m+2}] \qquad \dfrac{x : \neg A^*_1 \qquad \dfrac{N^*_1 : \neg B^*_2 \qquad \begin{array}{c} \dfrac{N^*_m : \neg B^*_{m+1} \quad [a : A^*_{m+1}]}{\langle N^*_m, a\rangle : A^*_m}\,(E_m)^* \\ \vdots \\ \langle N^*_2, \dots, N^*_m, a\rangle : A^*_2 \end{array}}{\langle N^*_1, \dots, N^*_m, a\rangle : A^*_1}\,(E_1)^*}{x\langle N^*_1, \dots, N^*_m, a\rangle : \bot}\,(\neg E)}{\bot}\,(I_{m+1})^* \\ \vdots \\ \bot \end{array}}{\bot}\,(I_{m+n-1})^*}{M^*_{nf} : \bot}\,(I_{m+n})^*$$

where $(I_i)^*$ is either $(\wedge E)$ or $(\exists E)$, and $(E_j)^*$ is either $(\wedge I)$ or $(\forall I)$. Here, we have $N^*_i : \neg B^*_{i+1}$ for some B_{i+1} $(1 \le i \le m)$ if N_i is a term, and then $A^*_i \equiv \neg B^*_{i+1} \wedge A^*_{i+1}$. Otherwise N_i is a type to say A'^*_i, and then $N^*_i : \neg B^*_{i+1}$ is to be deleted from the figure and $A^*_i \equiv \exists X.A'^*_{i+1}$ where $A^*_{i+1} \equiv A'^*_{i+1}[X := A'^*_i]$ for some A'_{i+1} $(1 \le i \le m)$.

Following the proof of Proposition 4, we have the desired derivation of $M_{nf} : A_{m+n+1}$ from that of $M^*_{nf} : \bot$ as follows:

(a) Case of I-part (A_i with $1 \le i \le m+1$):

i. Subcase of $N^*_m \equiv x_m$:
From the $\lambda^{\exists}_{U}$ deduction where $A^*_m \equiv (\neg B^*_{m+1} \wedge A^*_{m+1})$

$$\dfrac{x_m : \neg B^*_{m+1} \quad a : A^*_{m+1}}{\langle x_m, a\rangle : A^*_m}\,(\wedge I_{var})$$

we have the $\lambda 2$ deduction with $(B_{m+1} \Rightarrow A_{m+1}) \equiv A_m$

$$\dfrac{z_m : B_{m+1} \Rightarrow A_{m+1} \quad x_m : B_{m+1}}{z_m x_m : A_{m+1}}\,(\Rightarrow E)$$

ii. Subcase of $N^*_m \equiv \lambda a.P_m$:
From the $\lambda^{\exists}_{U}$ deduction where $A^*_m \equiv (\neg B^*_{m+1} \wedge A^*_{m+1})$

$$\dfrac{\begin{array}{c} a : B^*_{m+1} \\ \vdots \\ P_m : \bot \end{array} \quad a : A^*_{m+1}}{\langle \lambda a.P_m, a\rangle : A^*_m}\,(\wedge I_\lambda)$$

we have the $\lambda 2$ deduction with $(B_{m+1} \Rightarrow A_{m+1}) \equiv A_m$

$$\dfrac{z_m : B_{m+1} \Rightarrow A_{m+1} \quad P^\sharp_m : B_{m+1}}{z_m P^\sharp_m : A_{m+1}}\,(\Rightarrow E)$$

iii. Subcase of $N^*_m \equiv A'^*_m$:
From the $\lambda^{\exists}_U$ deduction where $A^*_{m+1} \equiv A'^*_{m+1}[X := A'^*_m]$ and $A^*_m \equiv \exists X.A'^*_{m+1}$

$$\frac{a : A^*_{m+1}}{\langle A'^*_m, a\rangle : \exists X.A'^*_{m+1}} \ (\exists I)$$

we have the $\lambda 2$ deduction with $\forall X.A'_{m+1} \equiv A_m$ and $A'_{m+1}[X := A'_m] \equiv A_{m+1}$

$$\frac{z_m : \forall X.A'_{m+1}}{z_m A'_m : A'_{m+1}[X := A'_m]} \ (\forall E)$$

iv. Subcase of $N^*_1 \equiv x_1$:
From the $\lambda^{\exists}_U$ deduction where $A^*_1 \equiv (\neg B^*_2 \wedge A^*_2)$

$$\frac{x_1 : \neg B^*_2 \quad \begin{array}{c} a : A^*_{m+1} \\ \vdots \\ \langle N^*_2, \ldots, N^*_m, a\rangle : A^*_2 \end{array}}{\langle x_1, N^*_2, \ldots, N^*_m, a\rangle : A^*_1} \ (\wedge I_{var})$$

we have the $\lambda 2$ deduction with $(B_2 \Rightarrow A_2) \equiv A_1$

$$\frac{z_1 : B_2 \Rightarrow A_2 \quad x_1 : B_2}{z_1 x_1 : A_2} \ (\Rightarrow E) \quad \text{and} \quad \begin{array}{c} z_2 : A_2 \\ \vdots \\ z_2 N_2 \ldots N_m : A_{m+1} \end{array}$$

and hence,

$$\begin{array}{c} \dfrac{z_1 : B_2 \Rightarrow A_2 \quad x_1 : B_2}{z_1 x_1 : A_2} \ (\Rightarrow E) \\ \vdots \\ (z_1 x_1) N_2 \ldots N_m : A_{m+1} \end{array}$$

v. Subcase of $N^*_1 \equiv \lambda a.P_1$:
From the $\lambda^{\exists}_U$ deduction where $A^*_1 \equiv (\neg B^*_2 \wedge A^*_2)$

$$\frac{\begin{array}{c} a : B^*_2 \\ \vdots \\ P_1 : \bot \end{array} \quad \begin{array}{c} a : A^*_{m+1} \\ \vdots \\ \langle N^*_2, \ldots, N^*_m, a\rangle : A^*_2 \end{array}}{\langle \lambda a.P_1, N^*_2, \ldots, N^*_m, a\rangle : A^*_1} \ (\wedge I_{\lambda})$$

we have the $\lambda 2$ deduction with $(B_2 \Rightarrow A_2) \equiv A_1$

$$\frac{z_1 : B_2 \Rightarrow A_2 \quad P^{\sharp}_1 : B_2}{z_1 P^{\sharp}_1 : A_{m+1}} \ (\Rightarrow E) \quad \text{and} \quad \begin{array}{c} z_2 : A_2 \\ \vdots \\ z_2 N_2 \ldots N_m : A_{m+1} \end{array}$$

and hence,

$$\begin{array}{c} \dfrac{z_1 : B_2 \Rightarrow A_2 \quad P^{\sharp}_1 : B_2}{z_1 P^{\sharp}_1 : A_2} \ (\Rightarrow E) \\ \vdots \\ (z_1 P^{\sharp}_1) N_2 \ldots N_m : A_{m+1} \end{array}$$

vi. Subcase of $N_1^* \equiv A_1'^*$:
From the $\lambda_U^{\exists}$ deduction where $A_2^* \equiv A_2'^*[X := A_1'^*]$ and $A_1^* \equiv \exists X.A_2'^*$

$$\dfrac{\begin{matrix} a : A_{m+1}^* \\ \vdots \\ \langle N_2^*, \ldots, N_m^*, a\rangle : A_2'^*[X := A_1'^*] \end{matrix}}{\langle A_1'^*, N_2^*, \ldots, N_m^*, a\rangle : \exists X.A_2'^*} \ (\exists I)$$

we have the $\lambda 2$ deduction with $\forall X.A_2' \equiv A_1$ and $A_2'[X := A_1'] \equiv A_2$

$$\dfrac{z_1 : \forall X.A_2'}{z_1 A_1' : A_2'[X := A_1']} \ (\forall E) \quad \text{and} \quad \begin{matrix} z_2 : A_2 \\ \vdots \\ z_2 N_2 \ldots N_m : A_{m+1} \end{matrix}$$

and hence,

$$\begin{matrix} \dfrac{z_1 : \forall X.A_2'}{z_1 A_1' : A_2'[X := A_1']} \ (\forall E) \\ \vdots \\ (z_1 A_1') N_2 \ldots N_m : A_{m+1} \end{matrix}$$

(b) Case of minimum segment:
From the $\lambda_U^{\exists}$ deduction

$$\dfrac{x : \neg A_1^* \quad \begin{matrix} a : A_{m+1}^* \\ \vdots \\ \langle N_1^*, \ldots, N_m^*, a\rangle : A_1^* \end{matrix}}{x\langle N_1^*, \ldots, N_m^*, a\rangle : \bot} \ (\neg E)$$

we have the $\lambda 2$ deduction

$$\begin{matrix} x : A_1 \\ \vdots \\ x N_1 \ldots N_m : A_{m+1} \end{matrix}$$

(c) Case of E-part (A_j with $m+2 \leq j \leq m+n+1$):
In this case, we have either $A_j^* \equiv \neg B_{j-1}^* \wedge A_{j-1}^*$ or $A_j^* \equiv \exists X.A_{j-1}^*$ from the deduction of $M_{nf}^* : \bot$.

i. Case of $A_{m+2}^* \equiv \neg B_{m+1}^* \wedge A_{m+1}^*$:
From the $\lambda_U^{\exists}$ deduction

$$\dfrac{a : A_{m+2}^* \quad \begin{matrix} a : A_{m+1}^* \\ \vdots \\ x\langle N_1^*, \ldots, N_m^*, a\rangle : \bot \end{matrix}}{\texttt{let}\ \langle x_n, a\rangle = a\ \texttt{in}\ x\langle N_1^*, \ldots, N_m^*, a\rangle : \bot} \ (\wedge E)$$

we have the $\lambda 2$ deduction with $B_{m+1} \Rightarrow A_{m+1} \equiv A_{m+2}$

$$\dfrac{x N_1 \ldots N_m : A_{m+1}}{\lambda x_n{:}B_n.x N_1 \ldots N_m : B_{m+1} \Rightarrow A_{m+1}} \ (\Rightarrow I)$$

ii. Case of $A^*_{m+2} \equiv \exists X_n.A^*_{m+1}$:
From the $\lambda^{\exists}_{U}$ deduction

$$\dfrac{a : A^*_{m+2} \quad \begin{array}{c} a : A^*_{m+1} \\ \vdots \\ x\langle N^*_1, \ldots, N^*_m, a\rangle : \bot \end{array}}{\mathtt{let}\ \langle X_n, a\rangle = a\ \mathtt{in}\ x\langle N^*_1, \ldots, N^*_m, a\rangle : \bot}\ (\exists E)$$

we have the $\lambda 2$ deduction with $\forall X_n.A_{m+1} \equiv A_{m+2}$

$$\dfrac{xN_1 \ldots N_m : A_{m+1}}{\lambda X_n.xN_1 \ldots N_m : \forall X_n.A_{m+1}}\ (\forall I)$$

iii. Other cases are confirmed similarly. Finally, we have the stated property for the deduction of $M_{nf} \equiv \lambda\chi_1. \ldots \lambda\chi_n.xN_1 \ldots N_m : A_{m+n+1}$.

5 Concluding Remarks

The target calculus $\lambda^{\exists}$ can be regarded as a subsystem of $\lambda 2$, in the sense of the impredicative encoding of $\bot$, $\wedge$ and $\exists$. Along the line of Theorem 2, we have a correspondence such that $\Gamma \vdash_{\lambda 2} M : A$ for some $M \in \Lambda 2$ if and only if $\neg\Gamma, a\!:\!A^* \vdash_{\lambda^{\exists}} P : \bot$ for $P \in \mathit{Univ}$, which itself does not imply the undecidability of the inhabitation of $\lambda^{\exists}$. We conjecture that the inhabitation problem of $\lambda^{\exists}$ is decidable, which remains open.

In the previous conference version [4], we have provided yet another call-by-name CPS-translation. For the theorem (proof duality), we have introduced the notion of dual paths, where dual paths form a duality with respect to the so-called paths. On the other hand, this paper introduced much simpler CPS-translation for the extensional $\lambda 2$-calculus, and derived a natural form of proof duality. The simple framework can serve as an basis for extensions with control operators, recursions, and so on. For instance, this framework has been applied for analyzing parametricity in the classical system $\lambda\mu$-calculus [9] ($\lambda 2$ plus control operators) in [6]. This CPS-translation is sound and complete with respect to the equational $\beta\eta$-theory of the $\lambda\mu$-calculus, whose syntactic analysis will appear in a forthcoming paper with Masahito Hasegawa.

References

[1] R. Backhouse: Galois Connections and Fixed Point Calculus, Lecture Notes in Computer Science **2297** (2002) 89–150.

[2] O. Danvy and J. L. Lawall: Back to Direct Style II: First-Class Continuations, Proc. of the ACM Conference on Lisp and Functional Programming (1992) 299-310.

[3] K. Fujita: A sound and complete CSP-translation for $\lambda\mu$-Calculus, Lecture Notes in Computer Science **2701** (2003) 120–134.

[4] K. Fujita: Galois embedding from polymorphic types into existential types, Lecture Notes in Computer Science **3461** (2005) 194–208.

[5] K. Fujita: Galois embedding from universal types into existential types –extended abstract–, Kyoto University RIMS Kokyuroku **1503** (2006) 121–128.

[6] M. Hassegawa: Relational parametricity and control (extended abstract), *Proc. Logic in Computer Science* (2005) 72–81.

[7] J. C. Mitchell and G. D. Plotkin: Abstract types have existential type, *Proc. the 12th Annual ACM Symposium on Principles of Programming Languages* (1985) 37–51.

[8] A. Melton, D. A. Schmidt, G. E. Strecker: Galois Connections and Computer Science Applications, Lecture Notes in Computer Science **240** (1986) 299–312.

[9] M. Parigot: $\lambda\mu$-Calculus: An Algorithmic Interpretation of Classical Natural Deduction, Lecture Notes in Computer Science **624** (1992) 190–201.

[10] G. Plotkin: Call-by-Name, Call-by-Value and the λ-Calculus, *Theoretical Computer Science* **1** (1975) 125–159.

[11] D. Prawitz: *NATURAL DEDUCTION, A Proof Theoretical Study*, ALMQVIST&WIKSELL, Stockholm, 1965.

[12] H. A. Priestley: Ordered Sets and Complete Lattices, A Primer for Computer Science, Lecture Notes in Computer Science **2297** (2002) 21–78.

[13] J. C. Reynolds: The discoveries of continuation, *Lisp and Symbolic Computation* **6** (1993) 233–247.

[14] P. Selinger: Control Categories and Duality: on the Categorical Semantics of the Lambda-Mu Calculus, *Math. Struct. in Compu. Science* **11** (2001) 207–260.

[15] A. Sabry and M. Felleisen: Reasoning about Programs in Continuation-Passing Style, *LISP AND SYMBOLIC COMPUTATION: An International Journal* **6** (1993) 289–360.

[16] A. Sabry and Ph. Wadler: A Reflection on Call-by-Value, *ACM Transactions on Programming Languages and Systems* **19-6** (1997) 916–941.

[17] Ph. Wadler: Call-by-value is dual to call-by-name, *International Conference on Functional Programming*, August 25-29, Uppsala, 2003.