Strong Normalization of Second Order Symmetric Lambda-mu Calculus

Yoriyuki Yamagata

Department of Mathematical Science, University of Tokyo
yoriyuki@ms.u-tokyo.ac.jp

Abstract. Parigot suggested symmetric structural reduction rules for application to μ-abstraction in [9] to ensure unique representation of data type. We prove strong normalization of second order $\lambda\mu$-calculus with these rules.

1 Introduction

Originally, $\lambda\mu$-calculus was defined to clarify correspondence between classical logic and control operators in functional programming languages. In this respect, $\lambda\mu$-calculus seems quite successful [5] [6] [7] [12]. In fact, $\lambda\mu$-calculus can be seen as an extension of λ-calculus equipped with variables, binding construct and application for continuation [12]. This makes $\lambda\mu$-calculus suitable for the study of programming languages.

In addition, Parigot was also motivated in [8] by possibility of witness extraction from classical proofs of Σ^0_1-sentences. Unfortunately, reduction rules of $\lambda\mu$-calculus seems not sufficient for this purpose. For example, let $A(x)$ be an atomic formula of arithmetic and $A'(x)$ be its code in second order predicate logic. We represent $\exists x.A(x)$ as $\forall X.\forall x(A(x) \to X) \to X$ in second order language, where X is a variable over propositions. We expect that a closed normal deduction of $\exists x.A'(x)$ somehow contains a unique first order term t such that $A(t)$ holds. However, consider the following situation. Suppose that $A(t)$ holds but $A(u)$ does not hold. Let M be a deduction of $A'(t)$ represented as $\lambda\mu$-terms. $\Lambda X.\lambda\alpha.\mu\beta.[\beta]\alpha u(\mu\gamma.[\beta]\alpha tM)$ is a closed and normal deduction of $\exists x A'(x)$ but apparently contains two terms t, u. Moreover, u is not a witness of $\exists x A(x)$. This suggests that we need additional reduction to extract the witness. In fact, Parigot proposed addition of new reduction rules (symmetric structural reduction) $M(\mu\alpha.N) \Rightarrow \mu\beta.N[M^*/\alpha]$ to solve similar problem on normal forms of the natural number type. $N[M^*/\alpha]$ is defined by inductively replacing all occurrence of $[\alpha]L$ in N to $[\alpha]M(L[M^*/\alpha])$. We will present a new calculus from which rules above can be derivable, and prove that it suffices to ensure that a closed normal term of type $\exists x A(x)$ for an atomic $A(x)$ contains one and only one first order term t and $A(t)$ holds. While numerous works on computational interpretation of classical proof are done, properties of normal form does not seem so well understood. Barbanera and Berardi shows that in symmetric lambda calculus for first order Peano arithmetic, closed normal forms of this calculus consist of introduction rules alone [3]. In addition to this work, we have to mention Parigot's work on output operators, which extract church numeral from classical proof of inhabitance of natural number type [9]. Danos et al. applied this technique to second order sequent calculus and show that it can be used for witness extraction from proofs of Σ^0_1-formulae [4]. Our work could be seen a sequel to these studies.

Obviously, to use such calculus for witness extraction, we need normalization property of it. In addition, if we expect that reduction rules fully specify extraction algorithm of witness, strong normalization is desirable. However, symmetric nature of reduction of application to μ-abstraction seems to prevent direct adaption of the proof of strong normalization of original $\lambda\mu$-calculus [10]. Luke Ong and Charles Stewart addressed strong normalization of a calculus with call-by-value restriction of symmetric structural reduction rules [7]. Their calculus $\lambda\mu_v$ is confluent, hence useful as a programming language, in contrast to our calculus. However, imposing reduction strategy seems to be an alien idea in a logical calculus. Non-confluency is come from unrestricted use of symmetric structural rules, hence essential feature of such calculus.

Barbanera and Berardi proved strong normalization of a non-deterministic calculus for propositional classical logic using fixed point construction for reducibility candidates [2]. We will prove strong normalization of second order $\lambda\mu$-calculus with the rules above based on this method.

2 Symmetric $\lambda\mu$-calculus

Our formalization is a second order extension of symmetric $\lambda\mu$-calculus in [11]. Usually, a term of $\lambda\mu$-calculus is understood as a proof with multiple conclusions. On the contrary, we consider a $\lambda\mu$-term as a proof with a single conclusion but two kinds of hypothesis, ordinary hypothesis and denials of propositions, which correspond conclusions other than a principal formula in usual $\lambda\mu$-calculus. Moreover, we do not distinguish λ-variables and μ-variables. $x, y, x_1, \cdots$ and $t, u, t_1, \cdots$ stand for first order variables and terms. X^n, Y^n, X^n_i and denotes n-ary predicate variables.

Definition 1. *The set of* first order term *consists of a constant* 0*, unary function* S*, and function symbols* f *for all primitive recursive function on natural numbers. A* proposition *is that of second order predicate logic built up by equality* $=$*, predicate variables* X^n_i *and logical connectives* $\to$*,* $\forall$*. Formally,*

$$A ::= t_1 = t_2 \mid X^n_i t_1 \cdots t_n \mid A \to A \mid \forall x_i A \mid \forall X^n_i A.$$

A formula *is a proposition* A *or a denial* $\bullet A$ *of proposition or contradiction* $\bot$*. Note that* $\bot$ *is not counted as a proposition. Other connectives are defined by using second order construct. For example,* $\exists x.A(x)$ *is defined as* $\forall X^0.\forall x(A(x) \to X^0) \to X^0$ *and* $A \wedge B$ *as* $\forall X^0.(A \to B \to X) \to X$*.*

Definition 2. *An* abstraction term T *is a form* $\lambda x_1 \cdots x_n.A$ *for a proposition* A*. Substitution* $B[T/X^n]$ *of* T *for a predicate variable* X^n *in* B *is defined by replacing each occurrences of* $X^n t_1 \cdots t_n$ *whose* X^n *is a free variable in* B*, to* $A[t_1, \cdots, t_n/x_1, \cdots, x_n]$*.*

Definition 3. *The set of* axioms *consists of equality axioms, defining axioms for primitive recursive functions and the proposition* $S0 = 0 \to \forall X.X$*. We note that equality axioms and defining axioms can be formulated as atomic rules, that is the set of formulae of forms* $A_1 \to A_2 \to \cdots \to A_n$ *with atomic formula* A_i*. This constraint is relevant to the fact that a closed term of type of atomic formula without* μ *consists of axiom alone.*

Definition 4. $\lambda\mu$-terms of type A *are given by followings rules. (In the figure,* $t : A$ *means* t *have the type* A*.) For each formula* A*,* $\lambda\mu$-terms of type A *are defined inductively as follows. We denote variable by Greek letters* $\alpha, \beta, \cdots$*.*

$$\frac{}{\mathrm{axiom}_i : A} \qquad\qquad \frac{[\alpha^C]}{\alpha : C}$$

$$\frac{M : \bullet A \quad N : A}{[M]N : \bot}\,[] \qquad \frac{\begin{matrix}[\alpha^A]\\ \vdots \\ M : \bot\end{matrix}}{\mu\alpha.M : \bullet A}\,\mu \qquad \frac{\begin{matrix}[\alpha^{\bullet A}]\\ \vdots \\ M : \bot\end{matrix}}{\mu\alpha.M : A}\,\mu$$

$$\frac{M : A \to B \quad N : A}{MN : B}\,\text{app.} \qquad \frac{\begin{matrix}[\alpha^A]\\ \vdots \\ M : B\end{matrix}}{\lambda\alpha.M : A \to B}\,\lambda$$

$$\frac{M : \forall x A}{Mt : A[t/x]}\,app.^1 \qquad \frac{M : A}{\lambda x.M : \forall x A}\,\lambda^1$$

$$\frac{M : \forall X A}{MT : A[T/X]}\,app.^2 \qquad \frac{M : A}{\lambda X.M : \forall x A}\,\lambda^2$$

In the above rules λ^1, λ^2*, the derivation of* M *does not contains* x *or* X *as free variables.*

Remark 1. Well typed terms of Parigot's $\lambda\mu$-calculus can be translated to the calculus above, by replacing μ-variables of type A to variables of type $\bullet A$.

Remark 2. The reason of "Church style formulation", that is, incorporating typing information as a part of term, is that in the proof of strong normalization, we seems to need the fact that each term has a unique type.

Definition 5. Reduction rules *are the followings. Let* β, γ, δ *be fresh variables.*

$$
\begin{array}{llll}
(\lambda_1) & (\lambda\alpha.M)N & \Rightarrow M[N/\alpha] & \\
(\lambda_2) & (\lambda x.M)t & \Rightarrow M[t/x] & \\
(\lambda_3) & (\lambda X^n.M)T & \Rightarrow M[T/X^n] & \\
(\mu) & [M]\mu\alpha.N & \Rightarrow N[M/\alpha] & [\mu\alpha.M]N \Rightarrow M[N/\alpha] \\
(\zeta_1) & (\mu\alpha.M)N & \Rightarrow \mu\beta.M[\mu\gamma.[\beta](\gamma N)/\alpha] & M(\mu\alpha.N) \Rightarrow \mu\beta.N[\mu\gamma.[\beta](M\gamma)/\alpha] \\
(\zeta_2) & (\mu\alpha.M)t & \Rightarrow \mu\beta.M[\mu\gamma.[\beta](\gamma t)/\alpha] & \\
(\zeta_3) & (\mu\alpha.M)T & \Rightarrow \mu\beta.M[\mu\gamma.[\beta](\gamma T)/\alpha] &
\end{array}
$$

As usual, compatible closure of the rules above is called *one-step reduction relation* (denoted $\Rightarrow_1$) and reflexive and transitive closure of one-step reduction is called *reduction relation* (denoted $\Rightarrow$). $w(M)$ is the maximal length of sequences $M \Rightarrow_1 M_1 \cdots \Rightarrow_1 M_n$ if the maximum exists. Otherwise $w(M)$ is undefined. M is *strongly normalizable* if and only if $w(M)$ is defined.

Using μ and ζ-rules, Parigot's structural reduction [8] and symmetric one [9] mentioned in Section 1 can be derived.

$$(\mu\alpha.\ ...[\alpha]N...)L \Rightarrow_\zeta \mu\beta.\ ...[\mu\gamma.[\beta](\gamma L)]N... \Rightarrow_\mu \mu\beta.\ ...[\beta](NL)...$$

$$N(\mu\alpha.\ ...[\alpha]L...) \Rightarrow_\zeta \mu\beta.\ ...[\mu\gamma.[\beta](N\gamma)]L... \Rightarrow_\mu \mu\beta.\ ...[\beta](NL)...$$

If we understand $\bullet$ as the usual negation symbol, our ζ-rules resemble to Andou's reduction for $\bot_c$ [1].

3 Extraction of witnesses from Σ^0_1-formulae

Let $N(x)$ be the formula $\forall X^1.X^1 0 \to \forall x(X^1 x \to X^1 Sx) \to X^1 x$. It is well known that we can encode the second order Peano arithmetic into second order predicate logic as presented above. Σ^0_1-sentences are represented as $\exists x N(x) \wedge A(x)$. Since $\exists x A(x)$ is derivable from such formula, we extract a witness from the proof of the formula $\exists x A(x)$.

Definition 6. *I-context $E_I[]$ is a context defined by the following grammar.*

$$I[] ::= \lambda\alpha.E_I[] \mid \lambda x.E_I[] \mid \lambda X.E_I[]$$

$$E_I[] ::= [] \mid I[] \mid \mu\alpha.[\beta]I[]$$

Definition 7. *For a proposition A, $\alpha(A)$, $\beta(A)$ and $\gamma(A)$ are defined as follows. $\alpha(A)$ is the set of formulae $\{A\}$ for atomic A and $\{A\} \cup \alpha(A_2)$ for the case $A \equiv A_1 \to A_2, \forall x A_2, \forall X A_2$. $\beta(A)$ is the set of formulae $\{\bullet A\}$ for atomic A, $\{\bullet A\} \cup \beta(A_1)$ for $A \equiv \forall x A_1, \forall X A_1$ and $\{A_1\} \cup \{\bullet A\} \cup \beta(A_2)$ for $A \equiv A_1 \to A_2$. $\gamma(A)$ is the set of variables $\emptyset$ for atomic A, $\{x\} \cup \gamma(A_1)$ for $A \equiv \forall x A_1$, $\{X\} \cup \gamma(A_1)$ for $A \equiv \forall X A_1$, $\gamma(A_2)$ for $A \equiv A_1 \to A_2$. For a set S of formulae, $\alpha(S) = \bigcup_{A \in S} \alpha(A)$. $\beta(S), \gamma(S)$ are defined similarly. Note that $S \subset \alpha(S)$ and if $B \in \alpha(A)$ then $\alpha(B) \subset \alpha(A)$.*

Lemma 1. *Let $E_I[M]$ be a term of type A with free variables of type $A_1, \cdots, A_m$ (usual propositions) and $\bullet B_1, \cdots, \bullet B_n$ (denials). Then, the type of M is an element of $\alpha(\{A, B_1, \cdots, B_n\})$, types of free variables are contained in the set $\{A_1, \cdots, A_m\} \cup \beta(\{A, B_1, \cdots, B_n\})$. Free first order and predicate variable contained in M are those of $E_I[M]$ or elements of $\gamma(A)$.*

Proof. By induction on construction of $E_I[M]$.

Lemma 2. *All normal forms of $\lambda\mu$-term have forms $E_I[\alpha M_1 \cdots M_n]$.*

Proof. By induction on construction of a term.

Proposition 1. *Let $A(x)$ be an atomic formula and M be a normal closed term of type $\exists x A(x)$. M contains one and only one first order term t and $A(t)$ holds.*

Proof. By Lemma 2 and considering $\beta(\exists x A(x))$, together with consistency of the calculus, we see that M has a form $E_I[\alpha t K]$ where α has type $\forall x(A(x) \to X)$, t is a first order term and K is a term of type $A(t)$. Since K can not begin with μ, and whose type is atomic, $K \equiv K_1 K_2 \cdots K_m$. K_1 is either a variable of type $\forall x(A(x) \to X)$ or axioms, but $\forall x(A(x) \to X)$ is impossible since $A(x)$ does not contain X as a free variable. Hence K_1 is an axiom and types of all of $K_2, \cdots, K_m$ are atomic. By similar argument to K, $K_2, \cdots, K_m$ have a form of application to axioms. Repeatedly applying this argument, we can conclude that K consists of axioms alone.

4 Strong normalization

This section is devoted to the proof of strong normalization theorem.

Definition 8. *First we prepare several notations.*

1. *A term beginning with μ is called a μ-form.*
2. *For a set S of terms of type C, $Cl(S)$ is defined as the smallest set which satisfies clauses*
   (a) *$S \subset Cl(S)$ and contains all variables of type C.*
   (b) *$MN \in Cl(S)$ if $L \in Cl(S)$ for all L such that $MN \Rightarrow_1 L$.*
   (c) *$Mt \in Cl(S)$ if $L \in Cl(S)$ for all L such that $Mt \Rightarrow_1 L$ for a first order term t.*
   (d) *$MT \in Cl(S)$ if $L \in Cl(S)$ for all L such that $MT \Rightarrow_1 L$ for an abstraction term T.*
3. *The set of strongly normalizable terms of type $\bot$ is also denoted $\bot$.*
4. *For a set S of terms of type $C \neq \bot$,*

$$\bullet S := \{\mu\alpha.M | \forall N \in S, M[N/\alpha] \in \bot\}$$

   where α is a variable of type C and M has a type $\bot$.

5. *the operator* $D(\mathcal{X})$ *is defined as* $Cl(\mathcal{X} \cup \bullet\bullet\mathcal{X})$. *Note that* $\bullet\bullet$ *and hence* D *are monotone operators. For ordinal* κ,

$$D^{\kappa}(\mathcal{X}) := D(\bigcup_{\tau<\kappa} D^{\tau}(\mathcal{X})).$$

Definition 9 (Reducibility candidates). *Let* ω_1 *be the first uncountable ordinal and* A *be a proposition. Let* S *be a set of strongly normalizable terms of type* A. *Suppose* S *does not contain a* μ*-form and* S *is closed under reduction relation. Then, a set* $D^{\omega_1}(S)$ *is called a* reducibility candidate *of the proposition* A. *Note that from monotonicity of* D, *a reducibility candidate is a fixed point of* D. *The set of candidates of the proposition* A *is denoted by* $\mathbf{R}_A$. $\mathbf{R}$ *is the union of all* $\mathbf{R}_A$.

Lemma 3. *Let* $\mathcal{R}$ *be a reducibility candidate. Then the followings hold.*

1. *All terms in* $\mathcal{R}$ *are strongly normalizable.*
2. $\mathcal{R} = Cl(S \cup \bullet\bullet\mathcal{R})$.
3. *For* $M \in \bullet\mathcal{R}$ *and* $N \in \mathcal{R}$, $[M]N \in \bot$

Proof. The clause 1 follows from induction on ω_1.

Since $\mathcal{R}$ is a fixed point of D, we have $\mathcal{R} = Cl(\mathcal{R} \cup \bullet\bullet\mathcal{R}) \supset Cl(S \cup \bullet\bullet\mathcal{R})$, while $D^{\kappa}(S) \subset Cl(S \cup \bullet\bullet\mathcal{R})$. We have the clause 2.

To prove the clause 3, it suffices to prove that all L such that $[M]N \Rightarrow_1 L$ are strongly normalizable. The proof is induction on $w(M)+w(N)$. We consider each possibility of the reduction of $[M]N$.

The case where L has the form $[M']N'$ and $M \Rightarrow M'$ and $N \Rightarrow N'$. The thesis follows from induction hypothesis on $w(M)+w(N)$.

The case where $M \equiv \mu\alpha.M_1$ and $L \equiv M_1[N/\alpha]$. By the hypothesis $M \in \bullet\mathcal{R}$.

The case where $N \equiv \mu\alpha.N_1$ and $L \equiv N_1[M/\alpha]$. By Lemma 3, N should be an element of $\bullet\bullet\mathcal{R}$. We have the thesis.

Definition 10. *Let* $\mathcal{A} \in \mathbf{R}_A$ *and* $\mathcal{B} \in \mathbf{R}_B$. *Assume that* $(t_i)_{i\in I}$ *is a non-empty family of first order terms and* $(T_j)_{j\in J}$ *is a non-empty family of abstraction terms. Further,* $\mathcal{A}_i$ *is a candidate of the proposition* $A[t_i/x]$ *for each* $i \in I$ *and* $\mathcal{A}_j$ *is a candidate of the proposition* $A[T_j/X]$ *for each* $j \in J$. *Candidates* $\mathcal{A} \to \mathcal{B}$ $\bigwedge^1_{i\in I}\mathcal{A}_i$, $\bigwedge^2_{j\in J}\mathcal{A}_j$ *are defined by the following steps.*

$$L(\mathcal{A},\mathcal{B}) := \{\lambda\alpha^{A}.M \mid \forall N \in \mathcal{A}, M[N/\alpha^{A}] \in \mathcal{B}\} \tag{1}$$

$$\Pi^1_{i\in I}\mathcal{A}_i := \{\lambda x.M \mid \forall i \in I, M[t_i/x] \in \mathcal{A}_i\} \tag{2}$$

$$\Pi^2_{j\in J}\mathcal{A}_j := \{\lambda X.M \mid \forall j \in J, M[T_j/X] \in \mathcal{A}_j\} \tag{3}$$

$$\mathcal{A} \to \mathcal{B} := D^{\omega_1}(L(\mathcal{A},\mathcal{B})) \tag{4}$$

$$\bigwedge^{1}_{i\in I}\mathcal{A}_i := D^{\omega_1}(\Pi^1_{i\in I}\mathcal{A}_i) \tag{5}$$

$$\bigwedge^{2}_{j\in J}\mathcal{A}_i := D^{\omega_1}(\Pi^2_{j\in J}\mathcal{A}_i) \tag{6}$$

Lemma 4. *Let $\mathcal{A} \in \mathbf{R}_A$ and $\mathcal{B} \in \mathbf{R}_B$. If $M \in \mathcal{A} \to \mathcal{B}$ and $N \in \mathcal{A}$, $MN \in \mathcal{B}$.*

Proof. Let $\mathcal{A} = D^{\omega_1}(S)$. Assume that κ is the least ordinal such that $M \in D^{\kappa}(L(\mathcal{A},\mathcal{B}))$ and τ is the least ordinal such that $N \in D^{\tau}(S)$. By induction on the natural sum $\kappa \oplus \tau$ and $w(M) + w(N)$, we will prove that if $MN \Rightarrow_1 L$, $L \in \mathcal{B}$, which is the exact condition of $MN \in \mathcal{B}$.

The case $L \equiv M'N'$ and either $M \Rightarrow_1 M'$ and $N \equiv N'$ or $M \equiv M'$ and $N \Rightarrow_1 N'$. The thesis follows from induction hypothesis on $w(M) + w(N)$.

The case $M \equiv \lambda\alpha.M_1$ and $L \equiv M_1[N/\alpha]$. Since $M \in L(\mathcal{A},\mathcal{B})$, we have the thesis.

The case where M has a form $\mu\alpha.M_1$ and L is obtained from reduction of the outermost redex. Then, L has a form $\mu\beta.M_1[\mu\gamma.[\beta](\gamma N)/\alpha]$. Let $J \in \bullet\mathcal{B}$, $K \in D^{\kappa_1}(L(\mathcal{A},\mathcal{B}))$ for $\kappa_1 < \kappa$. We can assume that $kappa_1$ is smallest one such that $D^{\kappa_1}(L(\mathcal{A},\mathcal{B}))$ contains K. By induction hypothesis on κ_1, we have $KN \in \mathcal{B}$. It follows $[J](KN) \in \perp$. From arbitrariness of K and κ_1, $\mu\gamma.[J](\gamma N) \in \bullet\bigcup_{\kappa_1<\kappa} D^{\kappa_1}(L(\mathcal{A},\mathcal{B}))$ follows. Since M is a μ-form, $M \in \bullet\bullet\bigcup_{\kappa_1<\kappa} D^{\kappa_1}(L(\mathcal{A},\mathcal{B}))$. We can infer $M_1[\mu\gamma.[J](\gamma N)/\alpha] \in \perp$. Since $J \in \bullet\mathcal{B}$, we have $L \in \bullet\bullet\mathcal{B}$. Now, from $\bullet\bullet\mathcal{B} \subset \mathcal{B}$, the thesis follows.

The case where N has a form $\mu\alpha.N_1$ and L is obtained from reduction of the outermost redex. L has a form $\mu\beta.N_1[\mu\gamma.[\beta](M\gamma)/\alpha]$. Let $J \in \mathcal{B}$ and $K \in D^{\tau_1}(S)$ for $\tau_1 < \tau$. (as the above, we chose the smallest one.) From induction hypothesis on τ_1, we have $MK \in \mathcal{B}$. Similarly as above, it follows $\mu\gamma.[J](M\gamma) \in \bullet\bigcup_{\tau_1<\tau} D^{\tau_1}(S)$. Since N has a μ-form, $N \in \bullet\bullet\bigcup_{\tau_1<\tau} D^{\tau_1}(S)$. We have $N_1[\mu\gamma.[\beta](M\gamma)/\alpha] \in \perp$ and hence, $L \in \mathcal{B}$.

Lemma 5. *Assume that $(t_i)_{i\in I}, (\mathcal{A}_i)_{i\in I}$ is defined as Definition 10. If $M \in \bigwedge^1_{i\in I} \mathcal{A}_i$, $Mt_i \in \mathcal{A}_i$. Similarly, for $(T_j)_{j\in J}$ and $(\mathcal{A}_j)_{j\in J}$ defined as Definition 10, if $M \in \bigwedge^2_{j\in J} \mathcal{A}_j$, $MT_j \in \mathcal{A}_j$.*

Proof. The proof goes on the same line of that of Lemma 4. We concentrate the second order case. Let $D^{\omega_1}(S) = \bigwedge_{i\in I} \mathcal{A}_i$. Assume that κ is the least ordinal such that $t \in D^{\kappa}(S)$. We will prove that for all L such that $MT_j \Rightarrow_1 L$, $L \in \mathcal{A}_j$ holds, by induction on κ and $w(M)$.

The case where $L \equiv M'T_j$ and $M \Rightarrow_1 M'$. From induction hypothesis of $w(M')$, the thesis follows.

The case where $M \equiv \lambda X.M_1$ and $L \equiv M_1[T_j/X]$. Since $M \in \Pi^2_{j\in J}\mathcal{A}_j$, we have the thesis.

The case where $M \equiv \mu\alpha.M_1$ and $L \equiv \mu\beta.M_1[\mu\gamma.[\beta](\gamma T_i)/\alpha]$. Let $J \in \bullet\mathcal{A}_i$ and $K \in D^{\kappa_1}(S)$. (as Lemma 4, we choose the smallest one.) By induction hypothesis on κ_1, we have $KT_i \in \mathcal{A}_i$. From arbitrariness of K and κ_1, it follows

$$\mu\gamma.[J](\gamma T_i) \in \bullet \bigcup_{\kappa_1<\kappa} D^{\kappa_1}(S).$$

Since M has a μ-form, $M \in \bullet\bullet\bigcup_{\kappa_1<\kappa} D^{\kappa_1}(S)$. We can infer $M_1[\mu\gamma.[J](\gamma T_i)/\alpha] \in \perp$. Hence we have $L \in \bullet\bullet\mathcal{A}_i$.

The rest of the proof runs similarly to the usual method of reducibility candidates. Let $\mathcal{T}$ be the set of all first order terms. $\mathcal{F}^n$ denotes the set of all functions from $\mathcal{T}^n$ to $\mathbf{R}$. Suppose that ξ is a map sending first order variables to first order terms, a predicate variable X^n to n-ary function from the set of first order terms to $\mathbf{R}$. We extend ξ to be a map on the whole types using $\xi(\bot) = \bot$ and the following clauses.

$$\xi(\bullet A) = \bullet \xi(A) \tag{7}$$
$$\xi(A \to B) = \xi(A) \to \xi(B) \tag{8}$$
$$\xi(\forall x A) = \bigwedge_{t \in \mathcal{T}}^{1} \xi[t/x](A) \tag{9}$$
$$\xi(\forall X^n A) = \bigwedge_{f \in \mathcal{F}^n}^{2} \xi[f/X^n](A) \tag{10}$$

where $\xi[a/b]$ is defined as a map $\xi[a/b](b) = a$ and for $c \neq b$, $\xi[a/b](c) = \xi(c)$.

Proposition 2. *Let M be a term of type A. Assume that free first order variables of M are $x_1, \cdots, x_m$, free predicate variables of M are $X_1, \cdots, X_n$ and free variables of M are $\alpha_1^{A_1}, \cdots, \alpha_l^{A_l}$. Suppose that ξ is a map sending first order variables to first order terms, a predicate variable X^k to k-ary function from the set of first order terms to $\mathbf{R}$. For each $1 \leq i \leq n$ and $t_1, \cdots, t_k \in \mathcal{T}$ (k is the arity of $\xi(X_i)$) $\xi(X_i)t_1 \cdots t_n \in \mathbf{R}_{B_i[t_1/x_1, \cdots, t_n/x_k]}$. Let $N_j \in \xi(A_j)$ for $1 \leq j \leq l$. We define $\tilde{M}$ by simultaneous substitution $\xi(x_1), \cdots, \xi(x_m)$ into $x_1, \cdots, x_m$, $B_1, \cdots, B_n$ into $X_1, \cdots, X_n$, $N_1, \cdots, N_l$ into $\alpha_1, \cdots, \alpha_l$ on M. Then we have $\tilde{M} \in \xi(A)$.*

Proof. By induction on the construction of M.

As a special case, $t \in \xi(A)$ holds. From Lemma 3, we have the following theorem.

Theorem 1. *All terms are strongly normalizable.*

Acknowledgement. I am grateful to Ken-etsu Fujita, Ryu Hasegawa and Charles Stewart for their helpful comments and discussion.

References

1. Yuuki Andou. A normalization-procedure for the first order classical natural deduction with full symbols. *Tsukuba Journal of Mathematics*, 19(1):153–162, 1995.
2. F. Barbanera and S. Berardi. A strong normalization result for classical logic. *Ann. Pure Appl. Logic*, 76:99–116, 1995.
3. F. Barbanera and S. Berardi. A symmetric lambda calculus for "classical" program extraction. *Inf. Comput.*, 125(2):103–117, 1996.

4. V. Danos, J. B. Joinet, and H. Schellinx. A new deconstructive logic:linear logic. *J. Symb. Log.*, 62(3):755–807, 1997.
5. Ph. de Groote. A cps-translation of the $\lambda\mu$-calculus. In *Trees in algebra and programming, CAAP '94*, number 787 in Lect. Notes Comput. Sci, pages 85–99. Springer-Verlag, 1994.
6. Ph. de Groote. On the relation between $\lambda\mu$-calculus and the syntactic theory of sequential control. In *Logic programming and automated reasoning*, volume 822 of *Lect. Notes Comput. Sci*, pages 31–43. Springer-Verlag, 1994.
7. C.-H. L. Ong and C. A. Stewart. A curry-howard foundation for functional computation with control. In *Proceedings of the 24th Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*. ACM press, January 1997.
8. M. Parigot. $\lambda\mu$-calculus: an algorithmic interpretation of classical natural deduction. In A. Voronkov, editor, *Logic Programming and Automated Reasoning*, volume 624 of *Lecture Notes in Artificial Intelligence*, pages 190–201. Springer-Verlag, 1992.
9. M. Parigot. Classical proofs as programs. In *Computational logic and proof theory*, volume 713 of *Lect. Notes Comput. Sci*, pages 263–276. Springer-Verlag, 1993.
10. M. Parigot. Strong normalization for second order classical natural deduction. *J. Symb. Log.*, 62(4):1461–1479, 1997.
11. M. Parigot. On the computational interpretation of negation. In P. Clote and H. Schwichtenberg, editors, *Computer Science Logic*, volume 1862 of *Lect. Notes Comput. Sci*, pages 472–484. Springer-Verlag, 2000.
12. Th. Streicher and B. Reus. Classical logic, continuation semantics and abstract machines. *Journal of Functional Programming*, 8(6):543–572, 1998.