学位申請論文

中澤 巧爾

主論文 1

Confluency and Strong Normalizability of Call-by-Value $\lambda\mu$-Calculus

（値呼び $\lambda\mu$ 計算の合流性と強正規化性）

中澤 巧爾

Confluency and Strong Normalizability of Call-by-Value $\lambda\mu$-Calculus

Koji Nakazawa

Department of Mathematics, Kyoto University
Kyoto 606-8502, JAPAN
e-mail: nakazawa@kusm.kyoto-u.ac.jp

Abstract

This paper proves the confluency and the strong normalizability of the call-by-value $\lambda\mu$-calculus with the domain-free style. The confluency of the system is proved by improving the parallel reduction method of Baba, Hirokawa and Fujita. The strong normalizability is proved by using the modified CPS-translation, which preserves the typability and the reduction relation. This paper defines the class of the reductions whose strictness is preserved by the modified CPS-translation to prove the strong normalizability.

Keywords: call-by-value $\lambda\mu$-calculus, domain-free type system, confluency, strong normalizability, CPS-translation.

1 Introduction

The $\lambda\mu$-calculus, which was introduced by Parigot in [14], is a formal system of calculus which corresponds to the classical logic by the Curry-Howard isomorphism. The $\lambda\mu$-calculus enables us to analyze proofs of the classical logic by studying the terms of the calculus. In particular, the confluency and the strong normalizability of proofs in the classical logic can be proved by investigating the property of $\lambda\mu$-terms. For example, in [15], the strong normalizability of proofs in the second-order classical natural deduction was proved by showing the strong normalizability of corresponding typed $\lambda\mu$-terms.

The $\lambda\mu$-calculus also clarifies the algorithmic aspect of the classical logic. The algorithmic aspect of classical logic is characterized by the control operation. μ-operations express the mechanism of control operation. By this, the $\lambda\mu$-calculus enables us to assign types to programs including control operators. Furthermore, the $\lambda\mu$-calculus enables us also to construct programs with control operators from proofs of the classical logic.

In this sense, it is important to study the call-by-value variants of $\lambda\mu$-calculus. As the programming languages ML and Lisp were developed from the λ-calculus, it is significant to design the programming languages from the $\lambda\mu$-calculus. The call-by-value systems with control operations have been widely studied: the theory of sequential control [8], the calculus of exception handling $\lambda^{\to\vee}_{cxn}$ in [7], the call-by-value $\lambda\mu$-calculus [9], [10], [13], and so on. For example, in [13], Ong and Stewart constructs a deterministic call-by-value programming language $\mu\mathrm{PCF_V}$ from the call-by-value $\lambda\mu$-calculus $\lambda\mu_V$. They also showed that $\mu\mathrm{PCF_V}$ is sufficiently strong to express the various control constructs, such as the ML-style `raise`, `handle`-mechanism and the first-class continuations `callcc`, `throw` and `abort`.

In this paper, we prove the confluency and the strong normalizability of the domain-free call-by-value $\lambda\mu$-calculus for polymorphic types, which was introduced by Fujita in [9]. The results of this paper are applied to the Church-style calculus in a straightforward way, since the domain-free style may be considered as shorthand for the second-order Church-style.

On the simple $\lambda\mu$-calculus, which is the system considered in [14], the proof of confluency was presented by Parigot in [14]. However, later in [1], Baba, Hirokawa and Fujita found an error in this proof. They showed that if the system includes the renaming rule, the straightforward parallel reduction method does not work. They proved the confluency of the simple $\lambda\mu$-calculus by improving the parallel reduction. They also showed that the improved method can be used for the proof of the confluency of the call-by-value $\lambda\mu$-calculus without the $\mu\eta$-rule, that is $\mu\alpha.[\alpha]M \rhd M$ (M does not contain free α). In [9] and [10], the confluency of the call-by-value $\lambda\mu$-calculus with the $\mu\eta$-rule was proved for only typable terms by assuming strong normalizability. In this paper, we show the confluency of untyped terms of the call-by-value $\lambda\mu$-calculus including the $\mu\eta$-rule. To prove this, we improve the parallel reduction method of [1]. However, the straightforward extension of the proof of [1] does not work, since the addition of the $\mu\eta$-rule gives more complicated situations in the proof of the diamond property, which is the main lemma to prove the confluency. This paper solves this problem.

In [5], it is described that Py proved the confluency of the simple $\lambda\mu$-calculus with $\mu\eta$-rule in his thesis. Our work in this paper is independent of Py's work and our method is different from his method.

In [15], the strong normalizability of the simple $\lambda\mu$-calculus was proved in two ways, one was the reducibility method and the other used the CPS-translation. The CPS-translation is a map from the $\lambda\mu$-calculus to the λ-calculus such that the reduction relations are preserved. By this, we can prove the strong normalizability of the $\lambda\mu$-calculus from that of the λ-calculus.

The strong normalizability of the call-by-value $\lambda\mu$-calculus cannot be directly concluded from that of the simple $\lambda\mu$-calculus, since we considered that the call-by-value system contains the symmetric structural reduction, which is not included in the simple $\lambda\mu$-calculus. Ong and Stewart mentioned in [13] that the strong normalizability of the call-by-value $\lambda\mu$-calculus was proved by the reducibility method, but their proof was not published yet.

The ordinary CPS-translation is not adapted to prove the strong normalizability of the call-by-value system with the symmetric structural reduction, since it preserves only the convertibility relations, not the reduction relations. In [9], Fujita gave the sketch of the proof by using the ordinary CPS-translation. However, the proof was not finished, since it contains a deep and difficult gap in the proof of the lemma 6 of [9], which was the most important lemma for the strong normalization.

This paper uses a variant of the translation, which is called the modified CPS-translation and was presented in [6], [7], [10], [15] and so on. It was proved in [10] that the modified CPS-translation preserves the reduction relation of the call-by-value $\lambda\mu$-calculus. However, even if we use it, the strong normalizability cannot be proved in a straightforward way. One of the causes of the difficulties is that the modified CPS-translation does not always preserve the strictness of reductions. That is, even if M reduces to N with more than one steps, $\overline{\overline{M}}$ and $\overline{\overline{N}}$ may be the same terms, where $\overline{\overline{M}}$ and $\overline{\overline{N}}$ are the translations of M and N respectively by the modified CPS-translation. So we must clarify when $\overline{\overline{M}} \rhd^+ \overline{\overline{N}}$ holds in λ-calculus for $\lambda\mu$-terms M and N such that $M \rhd N$ in $\lambda\mu$-calculus. One of the new and important results of this paper is that it precisely clarifies the class of reductions whose strictness is preserved by the modified CPS-translation, and this paper proves the strong normalizability by using the result.

The proof of the strong normalizability of the call-by-value $\lambda\mu$-calculus in [10] used the modified CPS-translation, but the proof of the strong normalizability of [10] was not finished yet. The proof of the lemma 3.14 of [10], which is needed for the proof of strong normalizability, was not finished in the following reason. In the case 3 of the proof of the lemma 3.14 of [10], it was claimed that if we assume that M has no vacuous μ-abstraction and $M \rhd^+_{st} M_1 \rhd^+_{\beta\eta r} M_2 \rhd^+_{st} M_3 \rhd^+_{\beta\eta r} M_4 \rhd^+_{st} \ldots$ holds, then $\overline{\overline{M}} \equiv \overline{\overline{M_1}} \rhd^+ \overline{\overline{M_2}} \equiv \overline{\overline{M_3}} \rhd^+ \overline{\overline{M_4}} \equiv \ldots$ holds. However, $\overline{\overline{M_3}} \rhd^+ \overline{\overline{M_4}}$ does not necessarily hold, since M_3 may have vacuous μ-abstractions even if M does not. For example, if we let $M \equiv (\mu\alpha.[\beta](\lambda x.y)([\alpha]z))uv((\lambda x.x)w)$, $M_1 \equiv (\mu\alpha.[\beta](\lambda x.y)([\alpha]zu))v((\lambda x.x)w)$, $M_2 \equiv (\mu\alpha.[\beta]y)v((\lambda x.x)w)$, $M_3 \equiv (\mu\alpha.[\beta]y)((\lambda x.x)w)$ and $M_4 \equiv (\mu\alpha.[\beta]y)w$, then $M \rhd_{st} M_1 \rhd_{\beta\eta r} M_2 \rhd_{st} M_3 \rhd_{\beta\eta r} M_4$ holds, but we have $\overline{\overline{M_3}} \equiv \overline{\overline{M_4}}$.

2 Call-by-value $\lambda\mu$-calculus $\lambda_V\mu$ in domain-free style

In this section, we give the definition of the domain-free system of the call-by-value $\lambda\mu$-calculus $\lambda_V\mu$ for polymorphic types, which was presented in [9].

Firstly we define the types, the terms and the substitutions for $\lambda_V\mu$. The types of $\lambda_V\mu$ are defined from type variables and a type constant $\perp$. We abbreviate $\sigma\to\perp$ as $\neg\sigma$. For the definition of $\lambda\mu$-terms, we prepare two sorts of variables: ordinary variables, which are called λ-variables, and names, which are called μ-variables.

Definition 2.1. (Types and terms)

Variables, types, terms and values of $\lambda_V\mu$ are defined in a syntactic way as follows.

(1) Variables

(i) Type variables $t_0, t_1\ldots$ (denoted by $s, t\ldots$).

(ii) λ-variables $x_0, x_1\ldots$ (denoted by $x, y\ldots$).

(iii) μ-variables $\alpha_0, \alpha_1\ldots$ (denoted by $\alpha, \beta\ldots$).

(2) Types (denoted by $\sigma, \tau\ldots$)

$\sigma\ ::=\ t \mid \perp \mid \sigma\to\sigma \mid \forall t.\sigma.$

(3) Terms (denoted by $M, N\ldots$ or $P, Q\ldots$)

$M\ ::=\ x \mid \lambda x.M \mid \Lambda t.M \mid \mu\alpha.M \mid MM \mid M\sigma \mid [\alpha]M.$

We call $\lambda x.M$ a λ-abstraction, $\Lambda t.M$ a Λ-abstraction, $\mu\alpha.M$ a μ-abstraction, M_1M_2 a term application, $M\sigma$ a type application and $[\alpha]M$ an α-named term.

(4) Values (denoted by $U, V, W\ldots$)

$V\ ::=\ x \mid \lambda x.M \mid \Lambda t.M \mid [\alpha]M.$

Notation 2.2.

(1) Free variables and bound variables of types and terms are defined as usual. We write $FV(\sigma)$ and $FV(M)$ for the sets of free variables of σ and M respectively.

(2) $M\equiv N$ denotes that N is obtained from M by renaming bound variables. The expression $\sigma\equiv\tau$ is similar.

(3) The subterms of a term are defined as usual. $N\subset M$ denotes that N is a subterm occurrence of M.

(4) We use the following abbreviations,

$\lambda x_1x_2\ldots x_n.M\equiv(\lambda x_1.(\lambda x_2.\ldots(\lambda x_n.M)\ldots)),$

$M_1M_2M_3\ldots M_n\equiv(\ldots((M_1M_2)M_3)\ldots M_n).$

(5) We write $\vec{M}$ for a finite sequence of terms. We also use $\vec{V}$ and $\vec{\sigma}$ for expressing a finite sequence of values and types respectively. When $\vec{M}$ is a sequence $M_1M_2\ldots M_n$, $N\vec{M}$ denotes the term $NM_1M_2\ldots M_n$. If $\vec{M}$ is an empty sequence, $N\vec{M}\equiv N$.

Definition 2.3. (Substitutions)

The substitutions of $\lambda_V\mu$ are defined as follows.

(1) For types σ, τ, terms M, N, type variable t and λ-variable x, $\sigma[t:=\tau]$, $M[x:=N]$ and $M[t:=\tau]$ are defined as usual.

(2) For terms M, N, a type σ and a μ-variable α, $M[\alpha\Leftarrow N]$, $M[\alpha\Leftarrow\sigma]$ and $M[N\Rightarrow\alpha]$ are defined as follows.

(i) $x\theta\equiv x$.

(ii) $(\lambda x.M)\theta\equiv\lambda x.M\theta$.

(iii) $(\Lambda t.M)\theta\equiv\Lambda t.M\theta$.

(iv) $(\mu\beta.M)\theta\equiv\mu\beta.M\theta$.

(v) $(M\tau)\theta\equiv(M\theta)\tau$.

(vi) $(M_1M_2)\theta\equiv(M_1\theta)(M_2\theta)$.

(vii) $([\beta]M)\theta\equiv[\beta](M\theta)$ (if $\alpha\not\equiv\beta$).

(viii-r) $([\alpha]M)[\alpha\Leftarrow N]\equiv[\alpha]((M[\alpha\Leftarrow N])N)$.

(viii-t) $([\alpha]M)[\alpha\Leftarrow\sigma]\equiv[\alpha]((M[\alpha\Leftarrow\sigma])\sigma)$.

(viii-l) $([\alpha]M)[N\Rightarrow\alpha]\equiv[\alpha](N(M[N\Rightarrow\alpha]))$,

where θ is either $[\alpha\Leftarrow N]$, $[\alpha\Leftarrow\sigma]$ or $[N\Rightarrow\alpha]$ and we suppose $x\notin FV(N)$ in (ii), $t\notin FV(N)$ or $t\notin FV(\sigma)$ in (iii), $\beta\not\equiv\alpha$ and $\beta\notin FV(N)$ in (iv) by renaming bound variables.

The substitution lemmas hold in the following form.

Lemma 2.4. (Substitution lemmas)

(1) $M[x := P][y := Q] \equiv M[y := Q][x := P[y := Q]]$, *if* $x \not\equiv y$ *and* $x \notin FV(Q)$.
(2) $M[\alpha \Leftarrow A][\beta \Leftarrow B] \equiv M[\beta \Leftarrow B][\alpha \Leftarrow A[\beta \Leftarrow B]]$, *if* $\alpha \not\equiv \beta$ *and* $\alpha \notin FV(B)$.
(3) $M[x := P][\alpha \Leftarrow A] \equiv M[\alpha \Leftarrow A][x := P[\alpha \Leftarrow A]]$, *if* $x \notin FV(A)$.
(4) $M[\alpha \Leftarrow A][x := P] \equiv M[x := P][\alpha \Leftarrow A[x := P]]$, *if* $\alpha \notin FV(P)$.

Proof. These are proved by the induction on M in a straightforward way. □

We define the type assignment system for $\lambda_\forall\mu$. This system corresponds to the second-order classical natural deduction by the Curry-Howard isomorphism. As there are two sorts of variables, we prepare two sorts of contexts, one for λ-variables and one for μ-variables.

Definition 2.5.

(1) The λ-context is a finite set Γ of pairs $(x : \sigma)$ of a λ-variable x and a type σ such that for any x, y, σ and τ, if both $(x : \sigma)$ and $(y : \tau)$ are elements of Γ then either $x \not\equiv y$ or $\sigma \equiv \tau$. We use the symbols $\Gamma, \Gamma' \ldots$ for λ-contexts. When $(x : \sigma) \in \Gamma$, we define $\Gamma(x) \equiv \sigma$. $FV(\Gamma)$ is defined as follows.

(i) $FV(\phi) = \phi$.

(ii) $FV(\Gamma \cup \{(x : \sigma)\}) = FV(\Gamma) \cup \{x\} \cup FV(\sigma)$.

(2) The μ-context is a finite set Δ of indexed types σ^α for a μ-variable α and a type σ such that for any α, β, σ and τ, if both σ^α and τ^β are elements of Δ then either $\alpha \not\equiv \beta$ or $\sigma \equiv \tau$. We use the symbols $\Delta, \Delta' \ldots$ for μ-contexts. $FV(\Delta)$ and $\Delta(\alpha)$ are defined similarly to (1).

Definition 2.6. (Typing rules)

The axioms and rules of the type assignment of $\lambda_\forall\mu$ are the following.

$$\Gamma; \Delta \vdash x : \Gamma(x) \quad (ass)$$

$$\frac{\Gamma \cup \{x : \sigma\}; \Delta \vdash M : \tau}{\Gamma; \Delta \vdash \lambda x.M : \sigma \to \tau} (\to I) \qquad \frac{\Gamma; \Delta \vdash M : \sigma \to \tau \quad \Gamma; \Delta \vdash N : \sigma}{\Gamma; \Delta \vdash MN : \tau} (\to E)$$

$$\frac{\Gamma; \Delta \vdash M : \sigma}{\Gamma; \Delta \vdash \Lambda t.M : \forall t.\sigma} (\forall I) \qquad \frac{\Gamma; \Delta \vdash M : \forall t.\sigma}{\Gamma; \Delta \vdash M\tau : \sigma[t := \tau]} (\forall E)$$

$$\frac{\Gamma; \Delta \vdash M : \sigma}{\Gamma; \Delta \cup \{\sigma^\alpha\} \vdash [\alpha]M : \perp} (\perp I) \qquad \frac{\Gamma; \Delta \cup \{\sigma^\alpha\} \vdash M : \perp}{\Gamma; \Delta \vdash \mu\alpha.M : \sigma} (\perp E)$$

In the rule $(\forall I)$, neither $FV(\Gamma)$ nor $FV(\Delta)$ contains t. In the rule $(\perp I)$, if $\alpha \in FV(\Delta)$, then $\Delta(\alpha) \equiv \sigma$.

M is called a typable term if there exist contexts Γ, Δ and a type σ such that $\Gamma; \Delta \vdash M : \sigma$ is provable by the axioms and the rules above.

If we consider types as logical formulas and read each judgement

$\{x_1 : \sigma_1, \ldots, x_n : \sigma_n\};\ \{\tau_1^{\alpha_1}, \ldots, \tau_m^{\alpha_m}\} \vdash M : \sigma$

as

$\sigma_1, \ldots, \sigma_n, \neg\tau_1, \ldots, \neg\tau_m \vdash \sigma$,

the typing system defined above corresponds to the natural deduction system of second-order classical logic.

We define the reduction relations of $\lambda_\forall\mu$.

Definition 2.7. (Reductions)

(1) The axiom schemes of $\rhd_\lambda$, $\rhd_\mu$ and $\rhd_s$ are the following respectively.

$$\rhd_\lambda \begin{cases} (\beta_v) & (\lambda x.M)V \rhd M[x := V] \\ (\beta_t) & (\Lambda t.M)\sigma \rhd M[t := \sigma] \\ (\eta_v) & \lambda x.Vx \rhd V \quad (\text{if } x \notin FV(V)) \end{cases}$$

$$\triangleright_\mu \begin{cases} (\mu_r) & (\mu\alpha.M)N \triangleright \mu\alpha.M[\alpha \Leftarrow N] \\ (\mu_l) & V(\mu\alpha.M) \triangleright \mu\alpha.M[V \Rightarrow \alpha] \\ (\mu_t) & (\mu\alpha.M)\sigma \triangleright \mu\alpha.M[\alpha \Leftarrow \sigma] \end{cases}$$

$$\triangleright_s \begin{cases} (\mu\eta) & \mu\alpha.[\alpha]M \triangleright M \quad (\text{if } \alpha \notin FV(M)) \\ (rn) & [\alpha](\mu\beta.V) \triangleright V[\beta := \alpha] \end{cases}$$

(2) The one-step reduction relation $M \triangleright_\lambda N$ is defined as follows.

(i) If $M \triangleright N$ is an axiom of $\triangleright_\lambda$, then $M \triangleright_\lambda N$.

(ii) If $M \triangleright_\lambda N$, then $\lambda x.M \triangleright_\lambda \lambda x.N$.

(iii) If $M \triangleright_\lambda N$, then $\Lambda t.M \triangleright_\lambda \Lambda t.N$.

(iv) If $M \triangleright_\lambda N$, then $\mu\alpha.M \triangleright_\lambda \mu\alpha.N$.

(v) If $M \triangleright_\lambda N$, then $MP \triangleright_\lambda NP$.

(vi) If $M \triangleright_\lambda N$, then $PM \triangleright_\lambda PN$.

(vii) If $M \triangleright_\lambda N$, then $M\sigma \triangleright_\lambda N\sigma$.

(viii) If $M \triangleright_\lambda N$, then $[\alpha]M \triangleright_\lambda [\alpha]N$.

The one-step reductions $\triangleright_\mu$ and $\triangleright_s$ are similarly defined from the axioms of $\triangleright_\mu$ and $\triangleright_s$ respectively. $\triangleright_\eta$ denotes the one-step reduction relation defined by the rule (η_v) above. $\triangleright_{\lambda\mu}$ denotes the union of $\triangleright_\lambda$ and $\triangleright_\mu$. Similarly, $\triangleright_{s\lambda}$ denotes the union of $\triangleright_s$ and $\triangleright_\lambda$, and $\triangleright_{s\eta}$ denotes the union of $\triangleright_s$ and $\triangleright_\eta$. $\triangleright_\mu$ is called the one-step μ-reduction or the structural reduction. The rule (rn) is called the renaming rule.

(3) The one-step reduction $\triangleright$ of $\lambda_V\mu$ is defined as the union of $\triangleright_\lambda$, $\triangleright_\mu$ and $\triangleright_s$.

(4) $\triangleright^+$ is the transitive closure of $\triangleright$, and $\triangleright^*$ is the transitive and reflexive closure of $\triangleright$. Similarly, for any symbol $a \equiv \lambda$, μ, s, η, $\lambda\mu$, $s\lambda$ or $s\eta$, we define $\triangleright_a^+$ and $\triangleright_a^*$.

Notation 2.8.

For convenience, we write $M\underline{N}$ for the application NM, and use $A, B, \ldots$ for either ordinary terms M, types σ or underlined values $\underline{V}$. We call $A, B, \ldots$ extended arguments. Also when we use extended arguments, applications are left-associated. For example, $(\mu\alpha.M)\underline{V}P\underline{U} \equiv U(V(\mu\alpha.M)P)$. If $\vec{A}$ is a sequence of extended arguments $A_1A_2\ldots A_n$, $M\vec{A}$ denotes the term $MA_1A_2\ldots A_n$. For any extended argument A, we use the expression $M[\alpha \Leftarrow A]$ for either $M[\alpha \Leftarrow N]$ (if $A \equiv N$), $M[\alpha \Leftarrow \sigma]$ (if $A \equiv \sigma$) or $M[V \Rightarrow \alpha]$ (if $A \equiv \underline{V}$). Then we have $([\alpha]M)[\alpha \Leftarrow A] \equiv [\alpha]M[\alpha \Leftarrow A]A$ and the μ-reduction is defined by the one rule,

(μ) $\quad (\mu\alpha.M)A \triangleright \mu\alpha.M[\alpha \Leftarrow A]$.

If $\vec{A}$ is a sequence of extended arguments, $M[\alpha \Leftarrow \vec{A}]$ denotes $M[\alpha \Leftarrow A_1][\alpha \Leftarrow A_2]\ldots[\alpha \Leftarrow A_n]$. Then we have $(\mu\alpha.M)\vec{A} \triangleright_\mu^* \mu\alpha.M[\alpha \Leftarrow \vec{A}]$.

It should be noted that the class of values is closed under substitutions induced by reductions $(\beta_v), (\beta_t)$ or (μ), that is, if V and U are values, $V[x := U], V[t := \sigma]$ and $V[\alpha \Leftarrow A]$ are values. Furthermore if V is a value and $V \triangleright M$ holds, then M is also a value.

Then we verify the following basic property about the extended arguments.

Lemma 2.9.

Every $\lambda\mu$-term has just one of the following forms:

(1) V

(2) $(\mu\alpha.N)\vec{A}$,

(3) $(VU)\vec{A}$,

(4) $(V\sigma)\vec{A}$,

where V and U are values, $\vec{A}$ is a sequence of extended arguments and it may be an empty sequence.

Proof. This is proved by induction on the term M. When M is an application, M has the form of either $(\mu\alpha.N)\vec{A}$ or $(VB)\vec{A}$, where V is a value, $\vec{A}$ is a sequence of terms or types and B is a term or type. If $M \equiv (\mu\alpha.N)\vec{A}$, then M has the form of (1). If $M \equiv (VB)\vec{A}$ and B is a type, then M has the form of (4). If $M \equiv (VB)\vec{A}$ and B is a term, then, by the induction hypothesis, B has one of the four forms. When B is a value, M has the form of (3). When $B \equiv (\mu\alpha.N)\vec{C}$, $M \equiv (\mu\alpha.N)\vec{C}\underline{V}\vec{A}$ has the form of (2). Other cases are similarly proved. □

3 Confluency of $\lambda_V\mu$

In this section, we prove the confluency of $\lambda_V\mu$ by using the parallel reduction. In the definition of the parallel reduction, we extend the method of [1].

The main result of this section is the following theorem.

Theorem 3.1. (Confluency of $\lambda_V\mu$)
For any terms M, M_1 and M_2 of $\lambda_V\mu$, if $M \rhd^* M_1$ and $M \rhd^* M_2$, there exists a term M_3 such that $M_1 \rhd^* M_3$ and $M_2 \rhd^* M_3$.

Definition 3.2. (Parallel reduction)
The parallel reduction $\succ$ is defined by the following rules.

(P1) $x \succ x$.
(P2) If $M \succ M'$, then $\lambda x.M \succ \lambda x.M'$
(P3) If $M \succ M'$, then $\Lambda t.M \succ \Lambda t.M'$
(P4) If $M \succ M'$, then $\mu\alpha.M \succ \mu\alpha.M'$
(P5) If $M \succ M'$ and $N \succ N'$, then $MN \succ M'N'$
(P6) If $M \succ M'$, then $M\sigma \succ M'\sigma$.
(P7) If $M \succ M'$, then $[\alpha]M \succ [\alpha]M'$
(P8) If $M \succ M'$ and $V \succ V'$, then $(\lambda x.M)V \succ M'[x := V']$
(P9) If $V \succ V'$ and $x \notin FV(V)$, then $\lambda x.Vx \succ V'$
(P10) If $M \succ M'$, then $(\Lambda t.M)\sigma \succ M'[t := \sigma]$.
(P11) If $M \succ M'$ and $\alpha \notin FV(M)$, then $\mu\alpha.[\alpha]M \succ M'$
(P12) If $M \succ M'$ and $\vec{A} \succ \vec{A}'$, then $(\mu\alpha.M)\vec{A} \succ \mu\alpha.M'[\alpha \Leftarrow \vec{A}']$.
(P13) If $V \succ V'$ and $\vec{A} \succ \vec{A}'$, then $[\alpha]((\mu\beta.V)\vec{A}) \succ V'[\beta \Leftarrow \vec{A}][\beta := \alpha]$.

$\vec{A} \succ \vec{A}'$ denotes that $A_i \succ A'_i$ for any $i = 1, \ldots, n$, where $\vec{A} \equiv A_1 \ldots A_n$ and $\vec{A}' \equiv A'_1 \ldots A'_n$. If A is a type, the notion $A \succ A'$ is defined by $A \equiv A'$.

Note that, it is easy to see that $M \succ M$ holds for any term M and that if $M \succ M'$ then $FV(M') \subset FV(M)$.

In [1], Baba, Hirokawa and Fujita proved the confluency of the call-by-value $\lambda\mu$-calculus which does not include polymorphic types and the rules (η_v) and $(\mu\eta)$. The parallel reduction they used is defined by (P1), (P2), (P4), (P5), (P7), (P8), (P13) and

(P12') If $M \succ M'$ and $A \succ A'$ then $(\mu\alpha.M)A \succ \mu\alpha.M'[\alpha \Leftarrow A']$.

It is the point of their parallel reduction that consecutive structural reductions and one-step renaming are considered as one-step parallel reduction by (P13).

If the system includes $(\mu\eta)$ as the reduction rule, we must define the parallel reduction by (P12), not (P12'). If we define the parallel reduction by (P12'), the diamond property, which is the main lemma to prove confluency, does not hold. The diamond property claims that if $M \succ M_1$ and $M \succ M_2$ then there is a M_3 such that $M_1 \succ M_3$ and $M_2 \succ M_3$. For example, if we take $M \equiv \mu\alpha.[\alpha](\mu\beta.x)\vec{A}$, where $\alpha \notin FV((\mu\beta.x)\vec{A})$, then we have

$M \succ \mu\alpha.x$ (by (P13)),
$M \succ (\mu\beta.x)\vec{A}$ (by (P11)).

But these are not always confluent by one-step parallel reduction if we define it by (P12').

Notation 3.3.
(1) Let $\vec{A}$ be a sequence $A_1A_2 \ldots A_n$ and M be a term $\equiv (\mu\alpha.N)\vec{A}$. For example, the parallel reduction can apply to any initial sequence of $\vec{A}$ in the term M, that is, if $N \succ N'$ and $\vec{A} \succ \vec{A}'$, then all of the following hold.

$M \succ (\mu\alpha.N')A'_1A'_2 \ldots A'_n$.
$M \succ (\mu\alpha.N'[\alpha \Leftarrow A'_1])A'_2 \ldots A'_n$.
$M \succ (\mu\alpha.N'[\alpha \Leftarrow A'_1][\alpha \Leftarrow A'_2])A'_3 \ldots A'_n$.
$\vdots$
$M \succ \mu\alpha.N'[\alpha \Leftarrow A'_1][\alpha \Leftarrow A'_2] \ldots [\alpha \Leftarrow A'_n]$.

So we write

$(\mu\alpha.N)\vec{A} \succ (\mu\alpha.N'[\alpha \Leftarrow \vec{A}'_l])\vec{A}'_r$

for representing all of those situations, where $\vec{A_l} = A_1 \ldots A_i$ and $\vec{A_r} = A_{i+1} \ldots A_n$ for some i. Note that $\vec{A_l}$ and $\vec{A_r}$ may be empty.

(2) For any natural number i, we define the i-step parallel reduction $\overset{i}{\succ}$ as follows.

(i) $M \overset{0}{\succ} M$.

(ii) If $M \overset{i}{\succ} P$ and $P \succ N$, then $M \overset{i+1}{\succ} N$.

Firstly, we show the next lemma to prove the diamond property.

Lemma 3.4.

(1) If $M \succ M'$, then $M[t := \sigma] \succ M'[t := \sigma]$.

(2) If $M \succ M'$ and $V \succ V'$, then $M[x := V] \succ M'[x := V']$.

(3) If $M \succ M'$ and $A \succ A'$, then $M[\alpha \Leftarrow A] \succ M'[\alpha \Leftarrow A']$.

By (3) of this lemma, it immediately follows that if $M \succ M'$ and $\vec{A} \succ \vec{A'}$ hold, then $M[\alpha \Leftarrow \vec{A}] \succ M'[\alpha \Leftarrow \vec{A'}]$, since $M[\alpha \Leftarrow \vec{A}] \equiv M[\alpha \Leftarrow A_1][\alpha \Leftarrow A_2] \ldots [\alpha \Leftarrow A_n]$.

Proof. These are proved by induction on $M \succ M'$ Cases are classified by the last rule of the derivation of $M \succ M'$

(3) Case (P12) $(\mu\beta.M)\vec{B} \succ \mu\beta.M'[\beta \Leftarrow \vec{B'}]$. We have $(\mu\beta.M[\alpha \Leftarrow A])\vec{B}[\alpha \Leftarrow A] \succ \mu\beta.M'[\alpha \Leftarrow A'][\beta \Leftarrow \vec{B'}[\alpha \Leftarrow A']]$, by IH. Furthermore, we have $\mu\beta.M'[\alpha \Leftarrow A'][\beta \Leftarrow \vec{B'}[\alpha \Leftarrow A']] \equiv$RHS, from the substitution lemma.

Case (P13) $[\gamma]((\mu\beta.V)\vec{B}) \succ V'[\beta \Leftarrow \vec{B}][\beta := \gamma]$.

Case (P13).1. $\gamma \equiv \alpha$. From IH, we have,

LHS $\equiv [\alpha](\mu\beta.V[\alpha \Leftarrow A])\vec{B}[\alpha \Leftarrow A]A$

$\succ V'[\alpha \Leftarrow A'][\beta \Leftarrow \vec{B'}[\alpha \Leftarrow A']][\beta \Leftarrow A'][\beta := \alpha]$,

$\equiv V'[\beta \Leftarrow \vec{B'}][\alpha \Leftarrow A'][\beta \Leftarrow A'][\beta := \alpha]$.

Since $\alpha \not\equiv \beta$, we have further

$\equiv V'[\beta \Leftarrow \vec{B'}][\beta := \alpha][\alpha \Leftarrow A']$

$\equiv$ RHS.

Case (P13).2. $\gamma \not\equiv \alpha$. This case is simpler than case (P13).1.

Other cases are proved from IH and the substitution lemmas in a straightforward way. □

Then we prove the diamond property. Note that, by the addition of the rule $(\mu\eta)$, much more complicated cases than the proof for the system without the rule $(\mu\eta)$ in [1] arise in the following proof. One of such cases is, for example, the case 2.1.

Lemma 3.5.

If $M \succ M_1$ and $M \succ M_2$, there is a term M_3 such that $M_1 \succ M_3$ and $M_2 \succ M_3$.

Proof. This is proved by induction on the term M.

(Case 1) $M \equiv \lambda x.M'$ The reduction $\lambda x.M' \succ M_i$ is derived from either (P2) or (P9).

(Case 1.1) Both M_1 and M_2 are obtained from (P2). The forms of M_1 and M_2 are $\lambda x.M_1'$ and $\lambda x.M_2'$ respectively, where $M' \succ M_1'$ and $M' \succ M_2'$. From IH for M', there is M_3' such that $M_1', M_2' \succ M_3'$. Hence we can take $\lambda x.M_3'$ as M_3.

(Case 1.2) M_1 is obtained from (P2) and M_2 is from (P9). In this case, we may suppose that $M \equiv \lambda x.Vx$, $M_1 \equiv \lambda x.M_1'$ and $M_2 \equiv V_2$, where $Vx \succ M_1'$ and $V \succ V_2$. The form of $Vx \succ M_1'$ is either $Vx \succ V_1x$ or $(\lambda y.M'')x \succ M_1''[y := x]$.

(Case 1.2.1) $M_1' \equiv V_1x$. From IH for V, there is V_3 such that $V_1, V_2 \succ V_3$. Since $x \notin FV(V)$ and $V \succ V_1$, we have $x \notin FV(V_1)$, therefore, $M_1 \equiv \lambda x.V_1x \succ V_3$ holds. Hence we can take V_3 as M_3.

(Case 1.2.2) $M_1' \equiv M_1''[y := x]$. We may suppose that $M \equiv \lambda x.(\lambda y.M'')x$, $M_1 \equiv \lambda x.M_1''[y := x]$ and $M_2 \equiv V_2$, where $M'' \succ M_1''$ and $\lambda y.M'' \succ V_2$. Then, from IH for $\lambda y.M''$, we can find V_3 such that $\lambda y.M_1'', V_2 \succ V_3$. Furthermore we have $\lambda x.M_1''[y := x] \equiv \lambda y.M_1''$, since $x \notin FV(M'') \supseteq FV(M_1'')$. Therefore, $M_1 \equiv \lambda y.M_1'' \succ V_3$ holds.

(Case 1.3) M_1 is obtained from (P9) and M_2 is from (P2). This case is similar to the case 1.2.

(Case 1.4) Both M_1 and M_2 are obtained from (P9). This case is similar to the case 1.1.

(Case 2) $M \equiv M'\sigma$. The reduction $M'\sigma \succ M_i$ is derived from either (P6), (P10) or (P12).

(Case 2.1) M_1 is from (P6) and M_2 is from (P12). In this case, we may suppose that $M \equiv (\mu\alpha.M')\vec{A}\sigma$, $M_1 \equiv N_1\sigma$ and $M_2 \equiv \mu\alpha.M_2'[\alpha \Leftarrow \vec{A}_2\sigma]$, where $(\mu\alpha.M')\vec{A} \succ N_1$. The form of $(\mu\alpha.M')\vec{A} \succ N_1$ is either $(\mu\alpha.M')\vec{A} \succ (\mu\alpha.M_1'[\alpha \Leftarrow \vec{A}_{1,l}])\vec{A}_{1,r}$ or $(\mu\alpha.[\alpha]M'')\vec{A} \succ M_1''\vec{A}_1$.

(Case 2.1.1) $N_1 \equiv (\mu\alpha.M_1'[\alpha \Leftarrow \vec{A}_{1,l}])\vec{A}_{1,r}$. This case is proved by IH and the lemma 3.4 (3).

(Case 2.1.2) $N_1 \equiv M_1''\vec{A}_1$. By specifying the consecutive applications of (P11), the form of $\mu\alpha.[\alpha]M'' \succ M_1''$ may be expressed by

$$\mu\alpha_0.[\alpha_0](\mu\alpha_1.[\alpha_1]\dots(\mu\alpha_n.[\alpha_n]P)\vec{A}^{(n)}\dots)\vec{A}^{(1)} \succ P_1\vec{A}_1^{(n)}\dots\vec{A}_1^{(1)},$$

where $\vec{A}^{(i)} \succ \vec{A}_1^{(i)}$ for any i, and $P \succ P_1$ does not have the form of $(\mu\beta.[\beta]P')\vec{B} \succ P_1'\vec{B}_1$. If $[\alpha_0](\mu\alpha_1.\dots)\vec{A}^{(1)} \succ M_2'$ is obtained from (P7), the claim is proved easily.

So, in the following, we consider the case that it is obtained from (P13). If we specify the consecutive applications of (P13), the form of $\mu\alpha_0.[\alpha_0](\mu\alpha_1.\dots)\vec{A}^{(1)} \succ \mu\alpha_0.M_2'$ may be expressed by

$$\begin{aligned}&\mu\alpha_0.[\alpha_0](\mu\alpha_1.[\alpha_1]\dots(\mu\alpha_m.Q)\vec{A}^{(m)}\dots)\vec{A}^{(1)}\\ &\quad\succ \mu\alpha_0.Q_2[\alpha_m \Leftarrow \vec{A}_2^{(m)}][\alpha_m := \alpha_{m-1}]\dots[\alpha_1 \Leftarrow \vec{A}_2^{(1)}][\alpha_1 := \alpha_0],\end{aligned}$$

where $\vec{A}^{(i)} \succ \vec{A}_1^{(i)}$ for any i, and $Q \succ Q_2$ does not have the form of $[\alpha_m](\mu\beta.Q')\vec{B} \succ Q_1'[\beta \Leftarrow \vec{B}_1][\beta := \alpha_m]$.

(Case 2.1.2.1) $n \geq m$. In this case, we have

$$Q \equiv [\alpha_m](\mu\alpha_{m+1}.[\alpha_{m+1}]\dots(\mu\alpha_n.[\alpha_n]P)\vec{A}^{(n)}\dots\vec{A}^{(m+1)}) \equiv [\alpha_m]Q',$$

and we may suppose that $Q_2 \equiv [\alpha_m]Q_2'$. Note that $Q \equiv [\alpha_m]P$ if $n = m$. Then $\mu\alpha.[\alpha]M'' \succ M_1''$ may be expressed by

$$\mu\alpha_0.[\alpha_0](\dots(\mu\alpha_m.[\alpha_m]Q')\vec{A}^{(m)}\dots)\vec{A}^{(1)} \succ Q_1'\vec{A}_1^{(m)}\dots\vec{A}_1^{(1)},$$

where $Q_1' \equiv P_1\vec{A}_1^{(n)}\dots\vec{A}_1^{(m+1)}$ Furthermore, in this case, since neither $FV(\vec{A}^{(i+1)}),\dots,FV(\vec{A}^{(m)})$ nor $FV(Q')$ does not contain α_i for any i, we have

$$\begin{aligned}&Q_2[\alpha_m \Leftarrow \vec{A}_2^{(m)}][\alpha_m := \alpha_{m-1}]\dots[\alpha_1 \Leftarrow \vec{A}_2^{(1)}][\alpha_1 := \alpha_0]\\ &\equiv ([\alpha_m]Q_2')[\alpha_m \Leftarrow \vec{A}_2^{(m)}\dots\vec{A}_2^{(1)}][\alpha_m := \alpha_0]\\ &\equiv [\alpha_0]Q_2'\vec{A}_2^{(m)}\dots\vec{A}_2^{(1)}\end{aligned}$$

Therefore, we have

$$\begin{aligned}&M_1 \equiv Q_1'\vec{A}_1^{(m)}\dots\vec{A}_1^{(1)}\vec{A}_1\sigma,\\ &M_2 \equiv \mu\alpha_0.([\alpha_0]Q_2'\vec{A}_2^{(m)}\dots\vec{A}_2^{(1)})[\alpha_0 \Leftarrow \vec{A}_1\sigma] \equiv \mu\alpha_0.[\alpha_0]Q_2'\vec{A}_2^{(m)}\dots\vec{A}_2^{(1)}\vec{A}_1\sigma.\end{aligned}$$

Hence we can find $M_3 \equiv Q_3'\vec{A}_3^{(m)}\dots\vec{A}_3^{(1)}\vec{A}_3\sigma$ from IH.

(Case 2.1.2.2) $n < m$. In this case, we have

$$P \equiv (\mu\alpha_{n+1}.[\alpha_{n+1}]\dots(\mu\alpha_m.[\alpha_m]Q)\vec{A}^{(m)}\dots)\vec{A}^{(n+1)} \equiv (\mu\alpha_{n+1}.P')\vec{A}^{(n+1)},$$

and we may suppose that $P_1 \equiv (\mu\alpha_{n+1}.P_1'[\alpha_{n+1} \Leftarrow \vec{A}_{1,l}^{(n+1)}])\vec{A}_{1,r}^{(n+1)}$ Then the form of $\mu\alpha_0.[\alpha_0](\mu\alpha_1.\dots)\vec{A}^{(1)} \succ \mu\alpha_0.M_2'$ may be supposed to be

$$\begin{aligned}&\mu\alpha_0.[\alpha_0](\mu\alpha_1.[\alpha_1]\dots(\mu\alpha_{n+1}.[\alpha_{n+1}]P')\vec{A}^{(n+1)}\dots)\vec{A}^{(1)}\\ &\quad\succ \mu\alpha_0.P_2'[\alpha_{n+1} \Leftarrow \vec{A}_2^{(n+1)}][\alpha_{n+1} := \alpha_n]\dots[\alpha_1 \Leftarrow \vec{A}_2^{(1)}][\alpha_1 := \alpha_0],\end{aligned}$$

and similarly to the case 2.1.2.2, we have further

$$\equiv \mu\alpha_0.P_2'[\alpha_{n+1} \Leftarrow \vec{A}_2^{(n+1)}\dots\vec{A}_2^{(1)}][\alpha_{n+1} := \alpha_0].$$

Therefore, we have

$$\begin{aligned}M_1 &\equiv P_1\vec{A}_1^{(n)}\dots\vec{A}_1^{(1)}\vec{A}_1\sigma\\ &\equiv (\mu\alpha_{n+1}.P_1'[\alpha_{n+1} \Leftarrow \vec{A}_{1,l}^{(n+1)}])\vec{A}_{1,r}^{(n+1)}\vec{A}_1^{(n)}\dots\vec{A}_1^{(1)}\vec{A}_1\sigma,\\ M_2 &\equiv \mu\alpha_0.P_2'[\alpha_{n+1} \Leftarrow \vec{A}_2^{(n+1)}\dots\vec{A}_2^{(1)}][\alpha_{n+1} := \alpha_0][\alpha_0 \Leftarrow \vec{A}_2\sigma]\\ &\equiv \mu\alpha_{n+1}.P_2'[\alpha_{n+1} \Leftarrow \vec{A}_2^{(n+1)}\dots\vec{A}_2^{(1)}\vec{A}_2\sigma].\end{aligned}$$

Hence we can take $M \equiv \mu\alpha_{n+1}.P_3'[\alpha_{n+1} \Leftarrow \vec{A}_3^{(n+1)}\dots\vec{A}_3^{(1)}\vec{A}_3\sigma]$ from IH.

Other cases are proved similarly to the above cases. □

The properties we need to prove the confluency are the following two lemmas.

Lemma 3.6.
If $M \overset{m}{\succ} M_1$ and $M \overset{n}{\succ} M_2$, then there is M_3 such that $M_1 \overset{n}{\succ} M_3$ and $M_2 \overset{m}{\succ} M_3$.

Lemma 3.7.
(1) If $M \rhd^* N$, then $M \overset{n}{\succ} N$ for some n.
(2) If $M \overset{n}{\succ} N$ for some n, then $M \rhd^* N$.

The lemma 3.6 can be directly concluded from the lemma 3.5, and the lemma 3.7 can be verified in a straightforward way.

The confluency of the call-by-value $\lambda\mu$-calculus is proved from the lemmas 3.7 and 3.6 as follows.

Proof of the theorem 3.1. By the lemma 3.7 (1), $M \overset{n}{\succ} M_1$ and $M \overset{m}{\succ} M_2$ for some n and m, therefore, by the lemma 3.6, we can find M_3 such that $M_1 \overset{m}{\succ} M_3$ and $M_2 \overset{n}{\succ} M_3$. Hence we have $M_1 \rhd^* M_3$ and $M_2 \rhd^* M_3$ by the lemma 3.7 (2). □

4 Modified CPS-translation

In this section, to prove the strong normalizability of $\lambda_V\mu$, we give the definition of the modified CPS-translation and prove that it preserves the typability of terms.

The modified CPS-translation, which was presented in [6], [7], [10], [15] and so on, is an interpretation from the $\lambda\mu$-calculus to the λ-calculus. From a logical point of view, it can be considered that the translation from the classical logic to the intuitionistic logic. Note that this translation preserves the typability and the reduction relation.

Firstly, we define the domain-free system of the polymorphic typed λ-calculus. This system is a domain-free variant of the Girard's system F.

Definition 4.1. (Domain-free polymorphic typed λ-calculus)
The domain-free polymorphic typed λ-calculus is defined as follows. In this system, both λ-variables $x, y, \ldots$ and μ-variables $\alpha, \beta, \ldots$ are treated as the same sort of variables. The types of the domain-free polymorphic typed λ-calculus are the same as those of $\lambda_V\mu$.

(1) Terms (denoted by $K, L \ldots$)
$K ::= x \mid \lambda x.K \mid \Lambda t.K \mid KK \mid K\sigma$.

(2) The reduction relation $\rhd_{\beta\eta}$ is defined from the following rules.
(β) $(\lambda x.L)K \rhd_{\beta\eta} L[x := K]$,
(β_t) $(\Lambda t.L)\sigma \rhd_{\beta\eta} L[t := \sigma]$,
(η) $\lambda x.Kx \rhd_{\beta\eta} K$ (if $x \notin FV(K)$),
where K is not necessarily a value. We call the reduction relation $\rhd_{\beta\eta}$ the one-step $\beta\eta$-reduction.

(3) The typing axioms and rules of the domain-free polymorphic typed λ-calculus are the following.

$$\Gamma \vdash x : \Gamma(x) \quad (ass)$$

$$\frac{\Gamma \cup \{x : \sigma\} \vdash K : \tau}{\Gamma \vdash \lambda x.K : \sigma \to \tau}\ (\to I) \qquad \frac{\Gamma \vdash K : \sigma \to \tau \quad \Gamma \vdash L : \sigma}{\Gamma \vdash KL : \tau}\ (\to E)$$

$$\frac{\Gamma \vdash K : \sigma}{\Gamma \vdash \Lambda t.K : \forall t.\sigma}\ (\forall I) \qquad \frac{\Gamma \vdash K : \forall t.\sigma}{\Gamma \vdash K\tau : \sigma[t := \tau]}\ (\forall E)$$

In the rule $(\forall I)$, $FV(\Gamma)$ does not contain t.

Theorem 4.2. (Strong normalizability of polymorphic typed λ-calculus)
Every typable term of the domain-free polymorphic typed λ-calculus is strongly normalizable.

The strong normalizability of F was proved by Girard, and his proof in English is found, for example, in [12]. For variants of F, the proofs of the strong normalizability were given. The strong normalizability of the domain-free polymorphic typed λ-calculus is easily proved from that of the Curry-style polymorphic typed λ-calculus by considering the map translating both $\Lambda t.M$ and $M\sigma$ to M. The proof of the strong normalizability of the Curry-style polymorphic typed λ-calculus is found, for example, in [3].

Definition 4.3. (Modified CPS-translation)

The modified CPS-translation, which is a map from a term of $\lambda_V\mu$ to a term of the domain-free polymorphic λ-calculus, is defined as follows. We define the modified CPS-translation $\overline{\overline{M}}$ for a $\lambda\mu$-term M, the map $M : K$ for a $\lambda\mu$-term M and a λ-term K, the map $\Phi(V)$ for a value V and the map σ^q for a type σ simultaneously.

(1) $\overline{\overline{M}} \equiv \lambda k.(M : k)$ (k is a fresh λ-variable).

(2) $V : K \equiv K\Phi(V)$ (V is a value),
$\mu\alpha.M : K \equiv (M : I)[\alpha := K]$,
$VU : K \equiv \Phi(V)\Phi(U)K$ (V and U are values),
$MU : K \equiv M : \lambda m.m\Phi(U)K$ (M is not a value and U is a value),
$VN : K \equiv N : \lambda n.\Phi(V)nK$ (V is a value and N is not a value),
$MN : K \equiv M : \lambda m.(N : \lambda n.mnK)$ (Neither M nor N is a value),
$V\sigma : K \equiv \Phi(V)\sigma^q K$ (V is a value),
$M\sigma : K \equiv M : \lambda m.m\sigma^q K$ (M is not a value),

where m, n are fresh λ-variables and I is the λ-term $\lambda x.x$.

(3) $\Phi(x) \equiv x$,
$\Phi(\lambda x.M) \equiv \lambda x.\overline{\overline{M}}$,
$\Phi(\Lambda t.M) \equiv \Lambda t.\overline{\overline{M}}$,
$\Phi([\alpha]M) \equiv M : \alpha$.

(4) $t^q \equiv t$,
$\bot^q \equiv \bot$,
$(\sigma \to \tau)^q \equiv \sigma^q \to \neg\neg\tau^q$,
$(\forall t.\sigma)^q \equiv \forall t.\neg\neg\sigma^q$

Notation 4.4.

(1) For contexts, we define the translation Γ^q and $\neg\Delta^q$ as follows.
(i) If $\Gamma = \{(x_1 : \sigma_1), \ldots, (x_n : \sigma_n)\}$, then $\Gamma^q = \{(x_1 : \sigma_1^q), \ldots, (x_n : \sigma_n^q)\}$.
(ii) If $\Delta = \{\alpha_1^{\sigma_1}, \ldots, \alpha_n^{\sigma_n}\}$, then $\neg\Delta^q = \{(\alpha_1 : \neg\sigma_1^q), \ldots, (\alpha_n : \neg\sigma_n^q)\}$.

(2) For any term M which is not a value, and any extended argument A, the term $MA : K$ has the form of $M : L$. So we write $\phi(A, K)$ for this L. The map ϕ is syntactically defined as follows.
(i) $\phi(V, K) \equiv \lambda m.m\Phi(V)K$.
(ii) $\phi(N, K) \equiv \lambda m.(N : \lambda n.mnK)$.
(iii) $\phi(\sigma, K) \equiv \lambda m.m\sigma^q K$.
(iv) $\phi(\underline{V}, K) \equiv \lambda n.\Phi(V)nK$.

Then the map $M : K$ is defined as follows:
$V : K \equiv K\Phi(V)$, $\mu\alpha.M : K \equiv (M : I)[\alpha := K]$,
$VU : K \equiv \Phi(V)\Phi(U)K$, $V\sigma : K \equiv \Phi(V)\sigma^q K$,
$MA : K \equiv M : \phi(A, K)$.

We prepare the following lemma to prove the properties in this and the following sections.

Lemma 4.5.

(1) $(\sigma[t := \tau])^q \equiv \sigma^q[t := \tau^q]$.
(2) $FV(\sigma) = FV(\sigma^q)$, $FV(\Gamma) = FV(\Gamma^q)$ *and* $FV(\Delta) = FV(\neg\Delta^q)$.
(3) If $FV(K) \subseteq FV(L)$, $FV(M : K) \subseteq FV(M : L)$.
(4) For any term M, *the following hold.*
(i) $FV(\Phi(M)) \subseteq FV(M)$.
(ii) $FV(M : K) \subseteq FV(M) \cup FV(K)$.
(iii) $FV(\overline{\overline{M}}) \subseteq FV(M)$.

(5) If $x \notin FV(M)$, $(M : K)[x := L] \equiv M : K[x := L]$.
(6) $FV(\phi(A, K)) \subseteq FV(A) \cup FV(K)$.
(7) $M : \phi(A,K) \begin{cases} \equiv (MA : K) & (M \text{ not a value}), \\ \rhd_{\beta\eta} (MA : K) & (M \text{ a value}). \end{cases}$

Proof. (1), (2), (3), (4) and (5) are proved by induction in a straightforward way and (6) is proved from (4). (7) is also easily proved. If M is not a value, the assertion is clear from the definition of ϕ. In the case M is a value, we have LHS$\equiv \phi(A,K)\Phi(M)$. Therefore, if $A \equiv N$ (N is not a value), LHS$\equiv (\lambda m.(N : \lambda n.mnK))\Phi(M) \rhd (N : \lambda n.mnK)[m := \Phi(M)]$. Hence we have LHS$\equiv N : \lambda n.\Phi(M)nK \equiv$RHS from (6). Other cases are similarly proved. □

In the following, we show that the modified CPS-translation preserves the typability of terms.

Lemma 4.6.
In the domain-free polymorphic λ-calculus, if we have $\Gamma \cup \{x : \tau\} \vdash K : \sigma$ and $\Gamma \vdash L : \tau$, then it follows that $\Gamma \vdash K[x := L] : \sigma$.

Proof. This lemma is proved by induction on the proof of $\Gamma \cup \{x : \tau\} \vdash K : \sigma$. □

Theorem 4.7.
For any term M, type σ, and contexts Γ, Δ, if $\Gamma; \Delta \vdash M : \sigma$ holds in $\lambda_\vee\mu$, then the following hold in the λ-calculus.
(1) $\Gamma^q \cup \neg\Delta^q \vdash \overline{\overline{M}} : \neg\neg\sigma^q$
(2) $\Gamma^q \cup \neg\Delta^q \cup \{(k : \neg\sigma^q)\} \vdash (M : k) : \bot$
(3) $\Gamma^q \cup \neg\Delta^q \vdash \Phi(M) : \sigma^q$ (M is a value).

Proof. This theorem is proved by simultaneous induction on the proof of $\Gamma; \Delta \vdash_{\lambda_\vee\mu} M : \sigma$. When M is a value, we prove only (3), since (1) and (2) follows from (3) immediately. When M is not a value, we prove only (2), since (1) follows from (2) immediately.

(Case 1) (*ass*). If the proof is $\Gamma \cup \{x : \sigma\}; \Delta \vdash x : \sigma$, we have to show $\Gamma^q \cup \{(x : \sigma^q)\} \cup \neg\Delta^q \vdash x : \sigma^q$, which trivially holds.

(Case 2) ($\to E$). Suppose that the proof ends with $\dfrac{\Gamma; \Delta \vdash M : \sigma \to \tau \quad \Gamma; \Delta \vdash N : \sigma}{\Gamma; \Delta \vdash MN : \tau}$. Note that we don't have to consider (3) in this case, since MN is not a value.

(Case 2.1) If both M and N are values, $MN : k \equiv \Phi(M)\Phi(N)k$ holds. From IH (3), we have $\Gamma^q \cup \neg\Delta^q \vdash \Phi(M) : \sigma^q \to \neg\neg\tau^q$ and $\Gamma^q \cup \neg\Delta^q \vdash \Phi(N) : \sigma^q$. Therefore, we have $\Gamma^q \cup \neg\Delta^q \vdash \Phi(M)\Phi(N) : \neg\neg\tau^q$ Hence we have $\Gamma^q \cup \neg\Delta^q \cup \{(k : \neg\tau^q)\} \vdash \Phi(M)\Phi(N)k : \bot$.

(Case 2.2) If M is not a value and N is a value, $MN : k \equiv M : \lambda m.m\Phi(N)k$ holds. We have $\Gamma^q \cup \neg\Delta^q \vdash \Phi(N) : \sigma^q$ from IH (3), therefore, $\Gamma^q \cup \neg\Delta^q \cup \{(k : \neg\tau^q)\} \vdash \lambda m.m\Phi(N)k : \neg(\sigma^q \to \neg\neg\tau^q)$ is provable. Note that $\neg(\sigma^q \to \neg\neg\tau^q) \equiv \neg(\sigma \to \tau)^q$ On the other hand, from IH (2), we have $\Gamma^q \cup \neg\Delta^q \cup \{(l : \neg(\sigma \to \tau)^q)\} \vdash (M : l) : \bot$. Therefore, from the lemma 4.6, we have $\Gamma^q \cup \neg\Delta^q \vdash (M : l)[l := \lambda m.m\Phi(N)k] : \bot$, where $(M : l)[l := \lambda m.m\Phi(N)k] \equiv M : \lambda m.m\Phi(N)k$ from the lemma 4.5 (5).

Other cases are similarly proved. □

5 Soundness of the modified CPS-translation

In this section, we define the class of the reductions of $\lambda_\vee\mu$ whose strictness is preserved by the modified CPS-translation.

It was proved in [10] that the modified CPS-translation preserves the reduction relation $\rhd^*$ By this, we can reduce the proof of the strong normalizability of $\lambda_\vee\mu$ to the strong normalizability of the λ-calculus. However, even if we use this idea, the proof of the strong normalizability of $\lambda_\vee\mu$ is not simple, since the modified CPS-translation does not necessarily preserve the strictness of the reduction, that is, there are $\lambda\mu$-terms M and N such that $M \rhd^+ N$ and $\overline{\overline{M}} \equiv \overline{\overline{N}}$ hold. This fact is one of the obstacles to the proof of the strong normalizability of $\lambda_\vee\mu$, since that suggests the possibility of existence of an infinite reduction sequence of $\lambda\mu$-terms $M_1 \rhd M_2 \rhd \dots$ such that $\overline{\overline{M_1}} \rhd^* \overline{\overline{M_2}} \rhd^* \dots$ is not infinite in λ-calculus. So, in this section, we clarify the class of the reductions whose strictness are preserved by the modified CPS-translation, and by using this result, we prove the strong normalizability in the following sections.

The reason why the modified CPS-translation does not necessarily preserve the strictness is that it eliminates the information of "redundant" parts of $\lambda\mu$-terms. For example, if we take $P \equiv \mu\alpha.x$, then for any term N, $\overline{\overline{PN}} \equiv \lambda k.(Ix)[\alpha := \phi(N,k)] \equiv \lambda k.Ix$ does not contain any information of N. So if we have $N \rhd N'$, then $PN \rhd PN'$ holds, but $\overline{\overline{PN}}$ and $\overline{\overline{PN'}}$ are the same term $\lambda k.Ix$. We introduce the following new notions to clarify such a situation. An eliminator is the term M such that $M : K$ does not have the information of K. An inessential subterm occurrence is the subterm occurrence N of a term M whose information does not remain after translating M to $\overline{\overline{M}}$. In the above example, P is an eliminator, and N is an inessential subterm occurrence in PN. These notions are formally defined as follows.

Definition 5.1. (Eliminators and inessential subterm occurrences)

We simultaneously define eliminators, the relation $\subset_i$ between a term and its subterm occurrence, and the relation $\in_i$ between a μ-variable occurrence and a term as follows. We call N an inessential subterm occurrence of M if $N \subset_i M$, and we call α an inessential variable occurrence of M if $\alpha \in_i M$.

(1) Eliminators

(i) If $\alpha \in_i M$ holds for any occurrence of α in M, then $\mu\alpha.M$ is an eliminator. Note that this condition includes the case of $\alpha \notin FV(M)$, that is, if α does not occur in M then $\mu\alpha.M$ is an eliminator.

(ii) If M is an eliminator, $MN, NM, M\sigma$ are eliminators. Note that, even if N is not a value, NM is an eliminator when M is an eliminator.

(2) Inessential subterm occurrences

(i) $N \subset_i \lambda x.M$ if $N \subset_i M$.

(ii) $N \subset_i \Lambda t.M$ if $N \subset_i M$.

(iii) $N \subset_i \mu\alpha.M$ if $N \subset_i M$.

(iv) $N \subset_i [\alpha]M$ if $N \subset_i M$.

(vi) $N \subset_i M\sigma$ if $N \subset_i M$.

(vii) When M_1 is an eliminator, $N \subset_i M_1M_2$ if either $N \subset_i M_1$ or $N \subset M_2$.

(viii) When M_1 is a value and M_2 is an eliminator, $N \subset_i M_1M_2$ if either $N \subset M_1$ or $N \subset_i M_2$.

(ix) When M_1 is not an eliminator and either M_1 is not a value or M_2 is not an eliminator, $N \subset_i M_1M_2$ if either $N \subset_i M_1$ or $N \subset_i M_2$.

(3) $\alpha \in_i M$ if α is a free μ-variable occurrence in M and, for the subterm occurrence $[\alpha]N$ in M which named with this α, either $[\alpha]N \subset_i M$ holds or N is an eliminator.

If $N \subset M$ and $N \not\subset_i M$, we call N an essential subterm occurrence of M. $N \subset_e M$ denotes that N is an essential subterm occurrence of M. If α is a μ-variable occurrence in M and $\alpha \notin_i M$, we call α an essential μ-variable occurrence of M. Note that, a free μ-variable occurrence α in M is essential iff the subterm occurrence $[\alpha]N$ in M which named with this α occurs essentially in M and N is not an eliminator. $\alpha \in_e M$ denotes that α is an essential μ-variable occurrence of M.

The notion of eliminators is characterized in the following lemma 5.6. Note that if M includes no α, then $\mu\alpha.M$ is an eliminator, that is, in term of [10], vacuous μ-abstractions are eliminators.

The reason why we separate the definition of essentiality of subterms and μ-variables is that even if M has $[\alpha]N$ as its essential subterm occurrence, this α is inessential occurrence in M when N is an eliminator. For example, $M \equiv [\alpha](\mu\beta.x)$ is essential occurrence in M itself, but $\overline{\overline{M}} \equiv \lambda k.k(\mu\beta.x : \alpha) \equiv \lambda k.k(Ix)[\beta := \alpha]$ does not contain α, so the occurrence of α in M is not essential.

We classify the reductions of $\lambda_V\mu$ as follows.

Definition 5.2.

(1) Suppose that $M \rhd N$. $M \rhd^e N$ denotes that the redex is an essential subterm occurrence in M. $M \rhd^i N$ denotes that the redex is an inessential subterm occurrence in M. Similarly, for any symbol $a \equiv \lambda$, μ, s, η, $\lambda\mu$, $s\lambda$ or $s\eta$, we define $\rhd^e_a$ and $\rhd^i_a$.

(2) Suppose that $M \rhd_{\mu} N$ and its redex is $(\mu\alpha.P)Q$. $M \rhd_{\mu^+} N$ denotes that P has an α-named value $[\alpha]V$ as its subterm. $M \rhd_{\mu^-} N$ denotes that P has no α-named value $[\alpha]V$ as its subterm.

(3) Suppose that $M \rhd_{\mu^+} N$ and its redex is $(\mu\alpha.P)Q$. $M \rhd_{\mu^+_e} N$ denotes that there is an α-named value $[\alpha]V$ which is an essential subterm occurrence in P. $M \rhd_{\mu^+_i} N$ denotes that any occurrence of α-named value $[\alpha]V$ in P is inessential.

Note that if the redex $(\mu\alpha.P)A$ of a reduction does not have any α-named subterm in P, the reduction is μ^--reduction.

The main result of this section is the following theorem.

Theorem 5.3.

If either $M \rhd^e_{s\lambda} N$ or $M \rhd^e_{\mu^+_e} N$ then $\overline{\overline{M}} \rhd^+_{\beta\eta} \overline{\overline{N}}$. If either $M \rhd^e_{\mu^+_i} N$, $M \rhd^e_{\mu^-} N$ or $M \rhd^i N$ then $\overline{\overline{M}} \equiv \overline{\overline{N}}$.

The soundness of the modified CPS-translation follows immediately from the theorem 5.3.

Theorem 5.4. (Soundness of the modified CPS-translation)

If $M \rhd N$ in $\lambda\nu\mu$, then $\overline{\overline{M}} \rhd^*_{\beta\eta} \overline{\overline{N}}$ in the domain-free polymorphic λ-calculus.

The soundness of the modified CPS-translation has been already proved by Fujita in [10]. However, the class of reductions of $\lambda\mu$-calculus whose strictness is preserved by the modified CPS-translation was not precisely defined. So, in the following, we prove the theorem 5.3.

Firstly, we show the next lemma.

Lemma 5.5.

(1) $FV(M : K) = \begin{cases} FV(M : I) & \text{(if } M \text{ is an eliminator),} \\ FV(M : I) \cup FV(K) & \text{(otherwise).} \end{cases}$

(2) Suppose $\alpha \notin FV(K)$. If there is an essential occurrence of α in M, then $\alpha \in FV(\Phi(M)) \cap FV(M : K) \cap FV(\overline{\overline{M}})$. Otherwise $\alpha \notin FV(\Phi(M)) \cup FV(M : K) \cup FV(\overline{\overline{M}})$.

Proof. These are proved by induction on M simultaneously.

(1) Suppose that $M \equiv \mu\alpha.M_1$. Note that $\mu\alpha.M_1 : K \equiv (M_1 : I)[\alpha := K]$ from the definition.

If $\mu\alpha.M_1$ is an eliminator, then M_1 has no essential occurrence of α, therefore, $\alpha \notin FV(M_1 : I)$ from IH (2). Hence we have $(M_1 : I)[\alpha := K] \equiv (M_1 : I)[\alpha := I] \equiv (\mu\alpha.M_1 : I)$.

If $\mu\alpha.M_1$ is not an eliminator, then there is an essential occurrence of α in M_1, therefore, $\alpha \in FV(M_1 : I)$ from IH (2). Then we have $FV((M_1 : I)[\alpha := K]) = (FV(M_1 : I) - \{\alpha\}) \cup FV(K)$ and $FV(\mu\alpha.M_1 : I) = FV((M_1 : I)[\alpha := I]) = FV(M_1 : I) - \{\alpha\}$. Hence we have $FV((M_1 : I)[\alpha := K]) = (FV(M_1 : I) - \{\alpha\}) \cup FV(K) = (FV(\mu\alpha.M_1 : I)) \cup FV(K)$.

Other cases are proved in a straightforward way.

(2) (Case 1) $M \equiv [\alpha]M_1$.

(Case 1.1) There is an essential occurrence of α in $[\alpha]M_1$. Note that $\alpha \in FV([\alpha]M_1 : K) \cap FV(\overline{\overline{[\alpha]M_1}})$ is immediately proved from $\alpha \in FV(\Phi([\alpha]M_1))$, so we show $\alpha \in FV(\Phi([\alpha]M_1))$ in the following. In this case, either M_1 is not an eliminator or an α occurs essentially in M_1. If M_1 is not an eliminator, we have $\alpha \in FV(M_1 : \alpha) = FV(\Phi([\alpha]M_1))$ from IH (1). If there is an essential occurrence of α in M_1, we have $\alpha \in FV(M_1 : I)$ from IH (2). Since $FV(M_1 : I) \subset FV(M_1 : \alpha)$ from IH (1), we have $\alpha \in FV(M_1 : \alpha) = FV(\Phi([\alpha]M_1))$.

(Case 1.2) There is no essential occurrence of α in $[\alpha]M_1$. Similarly to the above case, we show only $\alpha \notin FV(\Phi([\alpha]M_1))$. In this case, M_1 is an eliminator and there is no essential occurrence of α in M_1. Since M_1 is an eliminator, we have $FV(M_1 : \alpha) = FV(M_1 : I)$ from IH (1). Since M_1 has no essential α, we have $\alpha \notin FV(M_1 : I)$ from IH (2). Hence we have $\alpha \notin FV(M_1 : \alpha) = FV(\Phi([\alpha]M_1))$.

(Case 2) $M \equiv M_1M_2$. We show only the case where neither M_1 nor M_2 is a value, since other cases are similarly proved.

(Case 2.1) M_1 is an eliminator. In this case, any subterm occurrence in M_2 is inessential in M_1M_2, so there is an $\alpha \in_e M_1M_2$ iff there is an $\alpha \in_e M_1$. If there is an $\alpha \in_e M_1$, from IH (2), we have $\alpha \in FV(M_1 : I)$, which is a subset of $FV(M_1M_2 : K)$ from the definition and IH (1). If there is no $\alpha \in_e M_1$, from IH (2), we have $\alpha \notin FV(M_1 : I)$. Since M_1 is an eliminator, we have $FV(M_1 : I) = FV(M_1M_2 : K)$ from IH (1).

(Case 2.2) M_1 is not an eliminator. From the definition and IH (1), we have $FV(M_1M_2 : K) = FV(M_1 : \lambda m.(M_2 : \lambda n.mnK)) = FV(M_1 : I) \cup FV(\lambda m.(M_2 : \lambda n.mnK))$. When there is an $\alpha \in_e M_1M_2$, α occurs essentially in either M_1 or M_2, then we have either $\alpha \in FV(M_1 : I)$ or $\alpha \in FV(M_2 : \lambda n.mnK)$ from IH (2). Therefore, we have $\alpha \in FV(M_1M_2 : K)$. When there is no $\alpha \in_e M_1M_2$, α occurs essentially in neither M_1 nor M_2, then $\alpha \notin FV(M_1 : I) \cup FV(M_2 : \lambda n.mnK)$ from IH (2). Hence we have $\alpha \notin FV(M_1M_2 : K)$.

Other cases are similarly proved. □

By this lemma, we characterize the notion of the eliminators as follows.

Lemma 5.6.

(1) If M is an eliminator, $M : K \equiv M : L$ for any λ-terms K, L.

(2) If M is not an eliminator and $K \rhd^+_{\beta\eta} L$ in the λ-calculus, then $M : K \rhd^+_{\beta\eta} M : L$.

Proof. Let x be a fresh variable. From the lemma 4.5 (5), $M : K \equiv (M : x)[x := K]$ for any λ-term K. By the lemma 5.5 (1), if M is an eliminator, $x \notin FV(M : x)$. Then $(M : x)[x := K] \equiv (M : x)[x := L]$, so (1) is proved. If M is not an eliminator, $x \in FV(M : x)$. Hence we have $(M : x)[x := K] \rhd^+_{\beta\eta} (M : x)[x := L]$, so (2) is proved. □

Furthermore we show the next lemma.

Lemma 5.7.

For any M in which there is no α-named value as its essential subterm occurrence and any extended argument A, M is an eliminator iff $M[\alpha \Leftarrow A]$ is an eliminator.

Proof. For any M in which there is no α-named value as its essential subterm occurrence, we prove the following two claims by induction on M simultaneously.

(1) For any subterm occurrence N in M, $N \subset_e M$ iff $N[\alpha \Leftarrow A] \subset_e M[\alpha \Leftarrow A]$,

(2) M is an eliminator iff $M[\alpha \Leftarrow A]$ is an eliminator.

Note that we use the notation $N \subset M$ to express occurrences of subterms, so $N[\alpha \Leftarrow A] \subset_e M[\alpha \Leftarrow A]$ in (1) denotes the subterm occurrence $N[\alpha \Leftarrow A]$ in $M[\alpha \Leftarrow A]$ corresponding to the subterm occurrence N in M.

(1) (Case 1) $M \equiv [\alpha]M_1$. By the assumption, M_1 is not a value. If $N \subset_i [\alpha]M_1$, then $N \subset_i M_1$ holds. Therefore, we have $N[\alpha \Leftarrow A] \subset_i M_1[\alpha \Leftarrow A]$ from IH (1). Hence we have $N[\alpha \Leftarrow A] \subset_i [\alpha]M_1[\alpha \Leftarrow A]A$. If $N \subset_e [\alpha]M_1$, then either $N \equiv [\alpha]M_1$ or $N \subset_e M_1$. When $N \equiv [\alpha]M_1$, we have to show $N[\alpha \Leftarrow A] \subset_e N[\alpha \Leftarrow A]$, which trivially holds. When $N \subset_e M_1$, we have $N[\alpha \Leftarrow A] \subset_e M_1[\alpha \Leftarrow A]$ from IH (1). Since M_1 is not a value, we have $N[\alpha \Leftarrow A] \subset_e M_1[\alpha \Leftarrow A]A$, therefore, $N[\alpha \Leftarrow A] \subset_e [\alpha]M_1[\alpha \Leftarrow A]A$.

(Case 2) $M \equiv M_1M_2$ and $N \subset_i M_1M_2$.

(Case 2.1) $N \subset M_1$. In this case, either $N \subset_i M_1$ or both M_1 is a value and M_2 is an eliminator. If $N \subset_i M_1$, the claim is proved from IH (1). Otherwise, $M_2[\alpha \Leftarrow A]$ is an eliminator from IH (2), and $M_1[\alpha \Leftarrow A]$ is a value. Hence we have $N[\alpha \Leftarrow A] \subset_i M_1[\alpha \Leftarrow A]M_2[\alpha \Leftarrow A]$.

(Case 2.2) $N \subset M_2$. In this case, either $N \subset_i M_2$ or M_1 is an eliminator. If $N \subset_i M_2$, the claim is proved from IH (1). Otherwise, $M_1[\alpha \Leftarrow A]$ is an eliminator from IH (2), therefore, the claim is proved.

Other cases are similarly proved.

(2) (Case 1) $M \equiv \mu\beta.M_1$.

(Case 1.1) $\mu\beta.M_1$ is an eliminator. Note that, we may suppose that $\beta \notin FV(A)$ by renaming bound variables. In this case, we have to show that any occurrence of β in $M_1[\alpha \Leftarrow A]$ is inessential. Suppose that $[\beta]Q$ is an arbitrary β-named subterm occurrence in $M_1[\alpha \Leftarrow A]$. Then $[\beta]Q$ has the form of $[\beta]P[\alpha \Leftarrow A]$ for some β-named subterm occurrence $[\beta]P \subset M_1$. Since $\mu\beta.M_1$ is an eliminator, we have either $[\beta]P \subset_i M_1$ or P is an eliminator. If $[\beta]P \subset_i M_1$, then we have $[\beta]Q \subset_i M_1[\alpha \Leftarrow A]$ from IH (1). If P is an eliminator, then Q is an eliminator from IH (2) since $Q \equiv P[\alpha \Leftarrow A]$. Therefore, any occurrence of β in $M_1[\alpha \Leftarrow A]$ is inessential.

(Case 1.2) $\mu\beta.M_1$ is not an eliminator. In this case, there is an essential occurrence of β in M_1, that is, there is a β-named subterm occurrence $[\beta]P \subset_e M_1$ such that P is not an eliminator. From IH (1) and (2), we have that $[\beta]P[\alpha \Leftarrow A] \subset_e M_1[\alpha \Leftarrow A]$ and $P[\alpha \Leftarrow A]$ is not an eliminator. Hence $\mu\beta.M_1[\alpha \Leftarrow A]$ is not an eliminator.

Other cases are similarly proved. □

By the lemmas 5.6 and 5.7, we prove the following property.

Proposition 5.8.

(1) For any $\lambda\mu$-term M, value V, and λ-term K, the following hold.

(i) $\Phi(M)[x := \Phi(V)] \equiv \Phi(M[x := V])$ (if M is a value).

(ii) $(M : K)[x := \Phi(V)] \equiv M[x := V] : K[x := \Phi(V)]$.

(iii) $\overline{\overline{M}}[x := \Phi(V)] \equiv \overline{\overline{M[x := V]}}$.

(2) For any $\lambda\mu$-term M, type σ, and λ-term K, the following hold.

(i) $\Phi(M)[t := \sigma^q] \equiv \Phi(M[t := \sigma])$ (if M is a value).

(ii) $(M : K)[t := \sigma^q] \equiv M[t := \sigma] : K[x := \sigma^q]$.

(iii) $\overline{\overline{M}}[t := \sigma^q] \equiv \overline{\overline{M[t := \sigma]}}$.

(3) For any $\lambda\mu$-term M, type σ, μ-variables α, β and λ-term K, the following hold.

(i) $\Phi(M)[\alpha := \beta] \equiv \Phi(M[\alpha := \beta])$ (if M is a value).

(ii) $(M : K)[\alpha := \beta] \equiv M[\alpha := \beta] : K[\alpha := \beta]$.

(iii) $\overline{\overline{M}}[\alpha := \beta] \equiv \overline{\overline{M[\alpha := \beta]}}$.

(4) For any $\lambda\mu$-term M, extended argument A, and λ-term K, if $\alpha \notin FV(A) \cup FV(K)$, then the following hold.

(i) $\Phi(M)[\alpha := \phi(A, K)] \trianglerighteq_{\alpha,M} \Phi(M[\alpha \Leftarrow A])[\alpha := K]$ (if M is a value).

(ii) $(M : L)[\alpha := \phi(A, K)] \trianglerighteq_{\alpha,M} (M[\alpha \Leftarrow A] : L[\alpha := \phi(A, K)])[\alpha := K]$.

(iii) $\overline{\overline{M}}[\alpha := \phi(A, K)] \trianglerighteq_{\alpha,M} \overline{\overline{M[\alpha \Leftarrow A]}}[\alpha := K]$.

$\trianglerighteq_{\alpha,M}$ denotes $\equiv$ if M has no α-named value as its essential subterm, and denotes $\triangleright^+_{\beta\eta}$ otherwise.

Proof. We show only (4), since (1), (2) and (3) are more simply proved in a similar way. In the proof of (4), we must be careful about whether $\trianglerighteq_{\alpha,M}$ is $\triangleright^+_{\beta\eta}$ or $\equiv$.

(i) (Case 1) x. It is clear that x has no any α-named value as its essential subterm, therefore, what we have to show is $\Phi(x)[\alpha := \phi(A, K)] \equiv \Phi(x[\alpha \Leftarrow A])[\alpha := K]$, which means $x \equiv x$.

(Case 2) $M \equiv \lambda x.M_1$. Note that $\lambda x.M_1$ has no α-named value as its essential subterm iff M_1 has no α-named value as its essential subterm, therefore, $\trianglerighteq_{\alpha,\lambda x.M_1} \equiv \trianglerighteq_{\alpha,M_1}$. From IH (iii), we have $\overline{\overline{M_1}}[\alpha := \phi(A, K)] \trianglerighteq_{\alpha,M_1} \overline{\overline{M_1[\alpha \Leftarrow A]}}[\alpha := K]$, hence $(\lambda x.\overline{\overline{M_1}})[\alpha := \phi(A, K)] \trianglerighteq_{\alpha,M_1} (\lambda x.\overline{\overline{M_1[\alpha \Leftarrow A]}})[\alpha := K]$.

(Case 3) $M \equiv [\alpha]M_1$. In this case, $\trianglerighteq_{\alpha,[\alpha]M_1}$ is $\triangleright^+_{\beta\eta}$ if M_1 is a value, and $\trianglerighteq_{\alpha,[\alpha]M_1}$ is $\trianglerighteq_{\alpha,M_1}$ otherwise. This case is proved as follows.

$$
\begin{aligned}
\text{LHS} &\equiv (M_1 : \alpha)[\alpha := \phi(A, K)] \\
&\trianglerighteq_{\alpha,M_1} (M_1[\alpha \Leftarrow A] : \phi(A, K))[\alpha := K] \quad \text{(from IH (ii))} \\
&\triangleright^*_{\beta\eta} (M_1[\alpha \Leftarrow A]A : K)[\alpha := K] \\
&\equiv (M_1[\alpha \Leftarrow A]A : \alpha)[\alpha := K] \quad (\text{since } \alpha \notin FV(K)) \\
&\equiv \Phi(([\alpha]M_1)[\alpha \Leftarrow A])[\alpha := K] \\
&\equiv \text{RHS}.
\end{aligned}
$$

In the third line, from the lemma 4.5 (7), we have that $\triangleright^*_{\beta\eta}$ is $\triangleright_{\beta\eta}$ if M_1 is a value, and that it is $\equiv$ otherwise.

(Case 4) $M \equiv \Lambda t.M_1$. This case is similarly proved.

(ii) (Case 1) M is a value. This case is proved as follows.

$$
\begin{aligned}
\text{LHS} &\equiv (L\Phi(M))[\alpha := \phi(A, K)] \\
&\trianglerighteq_{\alpha,M} L[\alpha := \phi(A, K)]\Phi(M[\alpha \Leftarrow A])[\alpha := K] \quad \text{(from (i))} \\
&\equiv (M[\alpha \Leftarrow A] : L[\alpha := \phi(A, K)])[\alpha := K] \quad (\text{since } \alpha \notin FV(L[\alpha := \phi(A, K)])) \\
&\equiv \text{RHS}.
\end{aligned}
$$

(Case 2) $M \equiv \mu\beta.M_1$. In this case, $\trianglerighteq_{\alpha,\mu\beta.M_1}$ is $\trianglerighteq_{\alpha,M_1}$. This case is proved as follows.

$$\text{LHS} \equiv (M_1 : I)[\beta := L][\alpha := \phi(A, K)]$$

$\equiv (M_1 : I)[\alpha := \phi(A,K)][\beta := L[\alpha := \phi(A,K)]]$ (since $\alpha \not\equiv \beta$ and $\beta \notin FV(\phi(A,K))$)
$\trianglerighteq_{\alpha,M_1} (M_1[\alpha \Leftarrow A] : I)[\alpha := K][\beta := L[\alpha := \phi(A,K)]]$ (from IH (ii))
$\equiv (M_1[\alpha \Leftarrow A] : I)[\beta := L[\alpha := \phi(A,K)]][\alpha := K]$ (since $\alpha \notin FV(L[\alpha := \phi(A,K)])$)
$\equiv$ RHS.

(Case 3) $M_1 M_2$.

(Case 3.1) M_1 is not a value and M_2 is a value. In this case, we have the following.

LHS $\equiv (M_1 : \lambda m.m\Phi(M_2)L)[\alpha := \phi(A,K)]$
$\trianglerighteq_{\alpha,M_1} (M_1[\alpha \Leftarrow A] : \lambda m.m\Phi(M_2)[\alpha := \phi(A,K)]L)[\alpha := K]$ (from IH (ii))
$\trianglerighteq' (M_1[\alpha \Leftarrow A] : \lambda m.m\Phi(M_2[\alpha \Leftarrow A])L)[\alpha := K]$ (from IH (i))
$\equiv$ RHS,

where, by the lemma 5.6, $\trianglerighteq'$ in the third line is $\trianglerighteq_{\alpha,M_2}$ if $M_1[\alpha \Leftarrow A]$ is not an eliminator, and it is $\equiv$ otherwise. We prove LHS$\trianglerighteq_{\alpha,M_1 M_2}$RHS as follows.

(Case 3.1.1) M_1 has no α-named value as its essential subterm. In this case, since $\trianglerighteq_{\alpha,M_1}$ is $\equiv$, we have LHS$\trianglerighteq'$RHS.

(Case 3.1.1.1) M_1 is an eliminator. In this case, since any subterm occurrence in M_2 is inessential, $M_1 M_2$ has no α-named value as its essential subterm. Therefore, we have to show LHS$\equiv$RHS. Since, from the lemma 5.7, $M_1[\alpha \Leftarrow A]$ is an eliminator, $\trianglerighteq'$ is $\equiv$. Hence we have LHS$\equiv$RHS.

(Case 3.1.1.2) M_1 is not an eliminator. In this case, $M_1 M_2$ has no α-named value as its essential subterm iff M_2 has no α-named value as its essential subterm, so we have to show LHS$\trianglerighteq_{\alpha,M_2}$RHS. Since, from the lemma 5.7, $M_1[\alpha \Leftarrow A]$ is not an eliminator, $\trianglerighteq'$ is $\trianglerighteq_{\alpha,M_2}$, so this case is proved.

(Case 3.1.2) M_1 includes an α-named value as its essential subterm. In this case, since $M_1 M_2$ includes an α-named value as its essential subterm, we have to show LHS$\triangleright^+_{\beta\eta}$RHS. Since $\trianglerighteq_{\alpha,M_1}$ is $\triangleright^+_{\beta\eta}$, this case is proved.

Other cases are similarly proved. □

The theorem 5.3 is proved from the proposition 5.8 as follows.

Proof of the theorem 5.3 We prove the proposition by showing the following by induction on $M \triangleright N$ simultaneously: if $M \triangleright N$, then

(i) $\Phi(M) \triangleright^*_{\beta\eta} \Phi(N)$ (if M is a value),
(ii) $M : K \triangleright^*_{\beta\eta} B : K$ (for arbitrary λ-term K),
(iii) $\overline{\overline{M}} \triangleright^*_{\beta\eta} \overline{\overline{N}}$,

where $\triangleright^*_{\beta\eta}$ is $\triangleright^+_{\beta\eta}$ if either $M \triangleright^e_{s\lambda} N$ or $M \triangleright^{\iota}_{\mu^+_t}$, and it is $\equiv$ otherwise.

At first, note that, (iii) is easily proved from (ii) by taking variable k as K in (ii), and if M is a value then (ii) is easily proved from (i). So we prove only (i) if M is a value, and otherwise we prove only (ii).

(Case 1) M is a redex. It should be noted that, in this case, the redex is always essential in M.

(Case 1.1) $(\beta_v) : (\lambda x.M_1)V \triangleright M_1[x := V]$ We show (ii), since $(\lambda x.M_1)V$ is not a value. Since this reduction is $\triangleright^e_{s\lambda}$, what we have to show is $((\lambda x.M_1)V : K) \triangleright^+_{\beta\eta} (M_1[x := V] : K)$ for any λ-term K. It is proved as follows.

$(\lambda x.M_1)V : K \equiv \Phi(\lambda x.M_1)\Phi(V)K$
$\equiv (\lambda x.\overline{\overline{M_1}})\Phi(V)K$
$\triangleright_{\beta\eta} \overline{\overline{M_1}}[x := \Phi(V)]K$
$\equiv \overline{\overline{M_1[x := V]}}K$ (by the proposition 5.8 (1))
$\equiv (\lambda k.(M_1[x := V] : k))K$
$\triangleright_{\beta\eta} M_1[x := V] : K$ (from the lemma 4.5 (5), since $k \notin FV(M_1[x := V])$).

(Case 1.2) $(\eta_v) : \lambda x.Vx \triangleright V$ What we have to show is $\Phi(\lambda x.Vx) \triangleright^+_{\beta\eta} \Phi(V)$. Since $x \notin FV(V)$, we have $x \notin FV(\Phi(V))$, therefore, $\Phi(\lambda x.Vx) \equiv \lambda xk.\Phi(V)xk \triangleright_{\beta\eta} \lambda x.\Phi(V)x \triangleright_{\beta\eta} \Phi(V)$.

(Case 1.3) $(\beta_t) : (\Lambda t.M_1)\sigma \triangleright M_1[t := \sigma]$. This case is similarly proved by the proposition 5.8 (2).

(Case 1.4) $(rn) : [\alpha](\mu\beta.V) \triangleright V[\beta := \alpha]$. This is proved by the proposition 5.8 (3) as follows.

$\Phi([\alpha](\mu\beta.V)) \equiv (V : I)[\beta := \alpha]$
$\equiv (V[\beta := \alpha] : I)$ (by the proposition 5.8 (3))
$\equiv I\Phi(V[\beta := \alpha])$

$\triangleright_{\beta\eta}\Phi(V[\beta := \alpha])$.

(Case 1.5) $(\mu\eta) : \mu\alpha.[\alpha]M_1 \triangleright M_1$. This case is proved as follows.

$\mu\alpha.[\alpha]M_1 \equiv I(M_1 : \alpha)[\alpha := K]$

$\equiv I(M_1 : K)$ (since $\alpha \notin FV(M_1)$)

$\triangleright_{\beta\eta} M_1 : K$.

(Case 1.6) $(\mu) : (\mu\alpha.M_1)A \triangleright \mu\alpha.M_1[\alpha \Leftarrow A]$. Since $(\mu\alpha.M_1)A \triangleright^{e}_{\mu^{+}_{s}} \mu\alpha.M_1[\alpha \Leftarrow A]$ holds iff M_1 contains an α-named value as its essential subterm, we have to show $((\mu\alpha.M_1)A : K) \trianglerighteq_{\alpha,M_1} (\mu\alpha.M_1[\alpha \Leftarrow A] : K)$, where the symbol $\trianglerighteq_{\alpha,M_1}$ is that of the proposition 5.8. This case is proved as follows.

$(\mu\alpha.M_1)A : K \equiv (M_1 : I)[\alpha := \phi(A, K)]$

$\trianglerighteq_{\alpha,M_1} (M_1[\alpha \Leftarrow A] : I)[\alpha := K]$ (by the proposition 5.8 (4))

$\equiv \mu\alpha.M_1[\alpha \Leftarrow A] : K$.

(Case 2) $M_1M_2 \triangleright N_1M_2$. From IH, we have $\Phi(M_1) \trianglerighteq' \Phi(N_1)$ if M_1 is a value, and $M_1 : K \trianglerighteq' N_1 : K$, where $\trianglerighteq'$ is $\triangleright^{+}_{\beta\eta}$ if $M_1 \triangleright^{e}_{s\lambda} N_1$ or $M_1 \triangleright^{e}_{\mu^{+}_{s}} N_1$, and it is $\equiv$ otherwise. Note that, if M_1 is a value and M_2 is an eliminator, then we have to show $M_1M_2 : K \equiv N_1M_2 : K$ since the redex is inessential in M_1M_2, otherwise we have to show $M_1M_2 : K \trianglerighteq' N_1M_2 : K$.

(Case 2.1) Both M_1 and M_2 are values. We have $M_1M_2 : K \equiv \Phi(M_1)\Phi(M_2)K \trianglerighteq' \Phi(N_1)\Phi(M_2)K$ from IH.

(Case 2.2) M_1 is a value and M_2 is not a value. In this case, we have $M_1M_2 : K \equiv M_2 : \lambda n.\Phi(M_1)nK$. From the lemma 5.6, we have $M_2 : \lambda n.\Phi(M_1)nK \equiv M_2 : \lambda n.\Phi(N_1)nK$ if M_2 is an eliminator, otherwise $M_2 : \lambda n.\Phi(M_1)nK \trianglerighteq' M_2 : \lambda n.\Phi(N_1)nK$ from IH.

(Case 2.3) M_1 is not a value. From IH, we have $M_1M_2 : K \equiv M_1 : \phi(M_2, K) \trianglerighteq' N_1 : \phi(M_2, K) \equiv N_1M_2 : K$.

Other cases are similarly proved from IH (ii) or (iii). □

6 Strong normalizability of $\triangleright_{\mu^-}$

In this section, we prove the strong normalizability of $\triangleright_{\mu^-}$ for untyped terms.

Proposition 6.1. (Strong normalizability of $\triangleright_{\mu^-}$)

There is no infinite sequence of terms $M_0, M_1, \ldots$ such that $M_i \triangleright_{\mu^-} M_{i+1}$ for any i.

The strong normalizability of μ-reduction is very complicated to prove. For example, let M_1 be $(\mu\alpha.\ldots[\alpha]V\ldots)(\mu\beta.N)$, then M_1 reduces to $M_2 \equiv \mu\alpha.\ldots[\alpha]V(\mu\beta.N)\ldots$ by μ^+-reduction. Then the subterm $\mu\beta.N$ is an "argument" of the μ-redex in M_1, and it is also a "function" of the μ-redex in M_2, so it can be considered that μ^+-reduction produces a new "function" That makes the proof of the strong normalizability of $\triangleright_\mu$ difficult. On the other hand, the μ^--reduction does not increase such new "functions", so the strong normalizability of μ^--reduction can be proved more easily than that of μ-reduction.

In fact, by the result of the previous section, the strong normalizability of $\triangleright_{\mu^-}$ is sufficient to prove the strong normalizability of $\triangleright$. That is proved in the next section.

Definition 6.2.

Firstly, we define the maps π and $|\cdot|$ simultaneously, then we define the map $\#$.

(1) For a term M and an occurrence of subterm N in M, the natural number $\pi(N, M)$ is defined as follows.

(i) $\pi(M, M) = 1$.

(ii) If $N \subset M$, then $\pi(N, \lambda x.M) = \pi(N, M)$.

(iii) If $N \subset M$, then $\pi(N, \Lambda t.M) = \pi(N, M)$.

(iv) If $N \subset M$, then $\pi(N, M\sigma) = \pi(N, M)$.

(v) If $N \subset M$, then $\pi(N, \mu\alpha.M) = \pi(N, M)$.

(vi) If $N \subset M$, then $\pi(N, [\alpha]M) = \pi(N, M)$.

(vii) If $N \subset M$, then $\pi(N, LM) = |L| \cdot \pi(N, M)$.

(viii) If $N \subset M$ and M is a value, then $\pi(N, ML) = |L| \cdot \pi(N, M)$.

(ix) If $N \subset M$ and M is not a value, then $\pi(N, ML) = \pi(N, M)$.

(2) For a term M, the natural number $|M|$ is defined as follows.

(i) If M has an α-named subterm, $|(\mu\alpha.M)\vec{A}| = \sum_{[\alpha]P\subset M} \pi([\alpha]P, M)\ \ |P|$, which is the sum for all α-named subterm occurrences in M.

(ii) If M has no α-named subterm, $|(\mu\alpha.M)\vec{A}| = 1$.

(iii) If M does not have the form of $(\mu\alpha.N)\vec{A}$, $|M| = 1$.

(3) $\#M$ is defined as follows.

(i) If there is a μ-abstraction as subterm of M, $\#M = \sum_{\mu\alpha.P\subset M} \pi(\mu\alpha.P, M)$, which is the sum for all subterm occurrences in M that are μ-abstractions.

(ii) Otherwise $\#M = 0$.

Firstly we show some properties of the functions defined above. Then we show that $M \rhd_{\mu^-} N$ implies $\#M \geq \#N$.

Lemma 6.3.

Suppose that A is an arbitrary extended argument.

(1) If M is not a value and $N \subset M$, then we have

(i) $|MA| = |M|$,

(ii) $\pi(N, MA) = \pi(N, M)$.

(2) If $N \subset M$ and M does not include any α-named value as its subterm, then we have

(i) $|M[\alpha \Leftarrow A]| = |M|$,

(ii) $\pi(N[\alpha \Leftarrow A], M[\alpha \Leftarrow A]) = \pi(N, M)$.

(3) If $M_3 \subset M_2 \subset M_1$, then $\pi(M_3, M_1) = \pi(M_3, M_2) \cdot \pi(M_2, M_1)$.

(4) If both N and N' are values, or neither N nor N' is a value, $\pi(N, M[N]) = \pi(N', M[N'])$ for any context $M[\ ]$. The context $M[\ ]$ is defined as follows.

$M[\ ]\ ::=\ [\ ] \mid \lambda x.(M[\ ]) \mid \Lambda t.(M[\ ]) \mid \mu\alpha.(M[\ ]) \mid (M[\ ])N \mid N(M[\ ]) \mid (M[\ ])\sigma \mid [\alpha](M[\ ])$.

Proof. (1) (i) By the lemma 2.9, if M is not a value, M is either $(\mu\alpha.N)\vec{B}$, $(UV)\vec{B}$ or $(V\sigma)\vec{B}$. If $M \equiv (\mu\alpha.N)\vec{B}$, we have $|(\mu\alpha.N)\vec{B}A| = |\mu\alpha.N| = |(\mu\alpha.N)\vec{B}|$ by the definition.

(ii) If A is an underlined value $\underline{V}$, we have LHS= $\pi(N, VM) = |V|\ \ \pi(N, M)$ =RHS since $|V| = 1$, and otherwise $\pi(N, MA) = \pi(N, M)$ is as the definition.

(2) (i) and (ii) are proved by induction on M simultaneously.

(i) We show only the non-trivial case, where $M \equiv (\mu\beta.M')\vec{B}$ and $\beta \in FV(M')$. Suppose that $[\beta]P_1, \ldots, [\beta]P_n$ are all of the β-named subterm occurrences in M' Then $[\beta]P_1[\alpha \Leftarrow A], \ldots, [\beta]P_n[\alpha \Leftarrow A]$ are all of the β-named subterm occurrences in $M'[\alpha \Leftarrow A]$. So we have

$$\text{RHS} = |(\mu\beta.M')\vec{B}| = \sum_{i=1}^{n} \pi([\beta]P_i, M') \cdot |P_i|,$$

$$\text{LHS} = |(\mu\beta.M'[\alpha \Leftarrow A])\vec{B}[\alpha \Leftarrow A]| = \sum_{i=1}^{n} \pi([\beta]P_i[\alpha \Leftarrow A], M'[\alpha \Leftarrow A]) \cdot |P_i[\alpha \Leftarrow A]|.$$

For each $1 \leq i \leq n$, we have $|P_i| = |P_i[\alpha \Leftarrow A]|$ from IH (i) and $\pi([\beta]P_i, M') = \pi([\beta]P_i[\alpha \Leftarrow A], M'[\alpha \Leftarrow A])$ from IH (ii). Hence we have LHS=RHS.

(ii) We show only the non-trivial case, where $M \equiv [\alpha]M'$ and $N \subset M'$ Note that M' is not a value from the assumption. This case is proved as follows.

RHS $= \pi(N, [\alpha]M') = \pi(N, M')$

$= \pi(N[\alpha \Leftarrow A], M'[\alpha \Leftarrow A])$ (from IH (ii))

$= \pi(N[\alpha \Leftarrow A], M'[\alpha \Leftarrow A]A)$ (from (1))

$= \pi(N[\alpha \Leftarrow A], [\alpha]M'[\alpha \Leftarrow A]A)$

$=$ RHS.

(3) is proved by induction on M_1 in a straightforward way.

(4) is proved by induction on $M[\ ]$ Note that $M[N]$ is a value iff $M[N']$ is a value. □

The next definitions are only supplementary notions to make explicit of the subterm occurrences we consider.

Definition 6.4.

(1) The $*$-marked terms are defined as follows.

(i) If M is a term, M and M^* are $*$-marked terms.

(ii) If M is a $*$-marked term and M does not have the form of N^*, then M^* is a $*$-marked term.

(iii) If M and N are $*$-marked terms, $\lambda x.M$, $\Lambda t.M$, $\mu\alpha.M$, MN, $M\sigma$ and $[\alpha]M$ are $*$-marked terms.

(2) For a $*$-marked term M, $E(M)$ denotes the term obtained by eliminating all $*$'s from M.

(3) A $*$-marked term M is a value iff $E(M)$ is a value. Extended arguments for $*$-marked terms are defined in a similar way to those for terms.

(4) The μ-reduction for $*$-marked terms are defined from the following rules.

(μ) $(\mu\alpha.M)A \rhd \mu\alpha.M[\alpha \Leftarrow A]$.

(μ^*) $(\mu\alpha.M)^*A \rhd (\mu\alpha.M[\alpha \Leftarrow A])^*$,

where the substitution $[\alpha \Leftarrow A]$ is defined in a similar way to that for ordinary terms with the additional definition

$M^*[\alpha \Leftarrow A] \equiv (M[\alpha \Leftarrow A])^*$

The notions of the μ_c^+-reduction, the μ_t^+-reduction and the μ^--reduction are similarly defined for $*$-marked terms.

(6) For $*$-marked term M and its subterm occurrence P, the map π is defined as $\pi(P, M) = \pi(E(P), E(M))$. The other maps $|\cdot|$ and $\#$ are defined in a similar way.

Any $*$-marked term is obtained from a term by marking some subterm occurrences of the term with $*$. It is easily shown that if $M \rhd_\mu N$ and M' is the $*$-marked subterm such that $E(M') \equiv M$, then there is a unique $*$-marked term N' such that $M' \rhd_\mu N'$ and $E(N') \equiv N$.

Lemma 6.5.

If M is a $*$-marked term which has only one $*$, and P is the subterm occurrence in M which is marked with $*$, then $\pi(P^*, M) \geq \sum_{Q^* \subset N} \pi(Q^*, N)$, where RHS is the sum for all $*$-marked subterm occurrences in N, and RHS$= 0$ if there is no $*$ in N.

Proof. We prove the following two claims for $*$-marked terms M and N: if we have $M \rhd_{\mu^-} N$, the following hold.

(1) $|M| \geq |N|$.

(2) If M is a $*$-marked term which has only one $*$, and P is the subterm occurrence in M which is marked with $*$, then $\pi(P^*, M) \geq \sum_{Q^* \subset N} \pi(Q^*, N)$.

These are proved by induction on $M \rhd_{\mu^-} N$ simultaneously.

(1) We may suppose that M contains no $*$ since we define $|M| = |E(M)|$ for $*$-marked terms. We consider only non-trivial cases, where M has the form of $(\mu\alpha.M_1)\vec{A}$, since $|M| = |N| = 1$ otherwise.

(Case 1) $(\mu\alpha.M_1)A\vec{B} \rhd_{\mu^-} (\mu\alpha.M_1[\alpha \Leftarrow A])\vec{B}$. Since $|(\mu\alpha.M_1)A\vec{B}| = |\mu\alpha.M_1|$ and $|(\mu\alpha.M_1[\alpha \Leftarrow A])\vec{B}| = |\mu\alpha.M_1[\alpha \Leftarrow A]|$, we ignore $\vec{B}$ in this case. If $\alpha \notin FV(M_1)$ then LHS=RHS=1, so we suppose $\alpha \in FV(M_1)$. Suppose that $[\alpha]P_1, \ldots, [\alpha]P_n$ are all of the α-named subterm occurrences in M_1. Then all of the α-named subterm occurrences in $M_1[\alpha \Leftarrow A]$ are $[\alpha]P_1[\alpha \Leftarrow A]A, \ldots, [\alpha]P_n[\alpha \Leftarrow A]A$, so, in this case, we have to show

$$\sum_{i=1}^{n} \pi([\alpha]P_i, M_1) \cdot |P_i| \geq \sum_{i=1}^{n} \pi([\alpha]P_i[\alpha \Leftarrow A]A, M_1[\alpha \Leftarrow A]) \cdot |P_i[\alpha \Leftarrow A]A|.$$

Note that, since we consider the μ^--reduction, any P_i is not a value and any P_i has no α-named value as its subterm. For each $1 \leq i \leq n$, we have that $|P_i| = |P_i[\alpha \Leftarrow A]A|$ from the lemma 6.3 (1) (i) and (2) (i), and that $\pi([\alpha]P_i, M_1) = \pi([\alpha]P_i[\alpha \Leftarrow A]A, M_1[\alpha \Leftarrow A])$ from the lemma 6.3 (2) (ii). Hence we have LHS=RHS.

(Case 2) $(\mu\alpha.M_1)\vec{A} \rhd_{\mu^-} (\mu\alpha.N_1)\vec{A}$. Similarly we can ignore $\vec{A}$, and we suppose $\alpha \in FV(M_1)$. Suppose that the redex in M_1 is $(\mu\beta.M_2)B$, and that $[\alpha]P_1, \ldots, [\alpha]P_n$ are all of the α-named subterm occurrences in M_1. For each $1 \le i \le n$, if $M_{1,i}$ denotes the $*$-marked term obtained from M_1 by marking $[\alpha]P_i$ with $*$, then we can find a unique $N_{1,i}$ such that $M_{1,i} \rhd_{\mu^-} N_{1,i}$ and $E(N_{1,i}) = N_1$. We suppose that $([\alpha]P_{i,1})^*, \ldots, ([\alpha]P_{i,m_i})^*$ are all of the subterm occurrences in $N_{1,i}$ that are marked with $*$. Then, from IH (2), we have

$$\pi(([\alpha]P_i)^*, M_{1,i}) \ge \sum_{j=1}^{m_i} \pi(([\alpha]P_{i,j})^*, N_{1,i})$$

for each i. If we ignore all $*$'s, we have

$$\pi([\alpha]P_i, M_1) \ge \sum_{j=1}^{m_i} \pi([\alpha]P_{i,j}, N_1).$$

Furthermore, we can show $|P_i| \ge |P_{i,j}|$ for any i and j as follows. If $[\alpha]P_i \subset M_2$, then $m_i = 1$ and $P_{i,1}$ has the form of $P_i[\beta \Leftarrow B]$, and we have $|P_{i,1}| = |P_i[\beta \Leftarrow B]| = |P_i|$ from the lemma 6.3 (2) (i), since the subterm P_i of M_2 has no β-named value. If $(\mu\beta.M_2)B \subset [\alpha]P_i$, then we have $m_i = 1$ and $P_i \rhd_{\mu^-} P_{i,1}$, so we have $|P_i| \ge |P_{i,1}|$ from IH (1). Otherwise, since we have $P_{i,j} \equiv P_i$ for any j, we have $|P_i| = |P_{i,1}|$. Therefore, we have

$$\pi([\alpha]P_i, M_1) \cdot |P_i| \ge \sum_{j=1}^{m_i} \pi([\alpha]P_{i,j}, N_1) \cdot |P_{i,j}|$$

for each $1 \le i \le n$. Note that, if there is no such $P_{i,j}$, then we consider $m_i = 0$ and RHS of this inequality is 0. Hence, we have

$$\text{LHS} = |\mu\alpha.M_1| = \sum_{i=1}^{n} \pi([\alpha]P_i, M_1) \cdot |P_i| \ge \sum_{i=1}^{n}\sum_{j=1}^{m_i} \pi([\alpha]P_{i,j}, N_1) \cdot |P_{i,j}| = |\mu\alpha.N_1| = \text{RHS},$$

since $[\alpha]P_{1,1}, \ldots, [\alpha]P_{n,m_n}$ are all of the α-named subterm occurrences in N_1.

(Case 3) $(\mu\alpha.M_1)\vec{A} \rhd_{\mu^-} (\mu\alpha.M_1)\vec{B}$. We have $|(\mu\alpha.M_1)\vec{A}| = |\mu\alpha.M_1| = |(\mu\alpha.M_1)\vec{B}|$.

(2) We show only non-trivial cases, where M is the redex $(\mu\alpha.M_1)A$, since other cases are easily proved IH (1) and IH (2). Note that there is no α-named value in M_1.

(Case 1) $((\mu\alpha.M_1)A)^* \rhd_{\mu^-} (\mu\alpha.M_1[\alpha \Leftarrow A])^*$ In this case, what we have to show is $\pi((\mu\alpha.M_1)A, (\mu\alpha.M_1)A) \ge \pi(\mu\alpha.M_1[\alpha \Leftarrow A], \mu\alpha.M_1[\alpha \Leftarrow A])$, which holds since LHS=RHS= 1.

(Case 2) $(\mu\alpha.M_1)^*A \rhd_{\mu^-} (\mu\alpha.M_1[\alpha \Leftarrow A])^*$ Since $\pi((\mu\alpha.M_1)^*, (\mu\alpha.M_1)^*A) = \pi(\mu\alpha.M_1, \mu\alpha.M_1) = 1$ by the lemma 6.3 (1) (ii), this case is proved.

(Case 3) $P^* \subset M_1$. In this case, $N \equiv \mu\alpha.M_1[\alpha \Leftarrow A]$ has the only one subterm occurrence marked with $*$, which is $(P[\alpha \Leftarrow A])^* \subset N$. So we have to show $\pi(P^*, (\mu\alpha.M_1)A) \ge \pi((P[\alpha \Leftarrow A])^*, \mu\alpha.M_1[\alpha \Leftarrow A])$, but we can show LHS= $\pi(P, M_1)$ =RHS from the lemma 6.3 (1) (ii) and (2) (ii).

(Case 4) $P^* \subset A$. If $\alpha \notin FV(M_1)$ then $N \equiv \mu\alpha.M_1[\alpha \Leftarrow A]$ has no $*$, and the proof is finished since RHS= 0. So we consider the case where $\alpha \in FV(M_1)$ in the following. Suppose that $[\alpha]Q_1, \ldots, [\alpha]Q_n$ are all of the α-named subterm occurrences in M_1, and that, for each $1 \le i \le n$, $A_i \subset M_1[\alpha \Leftarrow A]$ denotes the occurrence of A applied to $Q_i[\alpha \Leftarrow A]$ in $M_1[\alpha \Leftarrow A]$. Then any subterm marked with $*$ in N has the same form with P^* and occurs in A_i for some i, so we suppose that P_i^* denotes the subterm occurrence marked with $*$ in A_i for each i. Then what we have to show is $\pi(P^*, (\mu\alpha.M_1)A) \ge \sum_{i=1}^{n} \pi(P_i^*, \mu\alpha.M_1[\alpha \Leftarrow A])$, but we can show LHS=RHS as follows. By the definition, we have

$$\text{LHS} = |\mu\alpha.M_1| \cdot \pi(P^*, A) = \sum_{i=1}^{n} \pi([\alpha]Q_i, M_1) \cdot |Q_i| \cdot \pi(P^*, A).$$

On the other hand, for each i, since $P_i^* \subset A_i \subset [\alpha]Q_i[\alpha \Leftarrow A]A_i \subset \mu\alpha.M_1[\alpha \Leftarrow A]$, we have $\pi(P_i^*, \mu\alpha.M_1[\alpha \Leftarrow A]) = \pi(P_i^*, A_i) \cdot \pi(A_i, [\alpha]Q_i[\alpha \Leftarrow A]A_i) \cdot \pi([\alpha]Q_i[\alpha \Leftarrow A]A_i, \mu\alpha.M_1[\alpha \Leftarrow A])$ from the lemma 6.3 (3). Furthermore, we have

$$\pi(A_i, [\alpha]Q_i[\alpha \Leftarrow A]A_i) = |Q_i[\alpha \Leftarrow A]| \quad \pi(A_i, A_i) = |Q_i|,$$

$$\pi([\alpha]Q_i[\alpha \Leftarrow A]A_i, \mu\alpha.M_1[\alpha \Leftarrow A]) = \pi(([\alpha]Q_i)[\alpha \Leftarrow A], M_1[\alpha \Leftarrow A]) = \pi([\alpha]Q_i, M_1)$$

from the lemma 6.3 (2). Since $\pi(P_i^*, A_i) = \pi(P^*, A)$, we have

$$\pi(P_i^*, \mu\alpha.M_1[\alpha \Leftarrow A]) = \pi(P^*, A) \quad |Q_i| \cdot \pi([\alpha]Q_i, M_1)$$

for each $1 \leq i \leq n$. Therefore, we have

$$\text{RHS} = \sum_{i=1}^{n} \pi(P_i^*, \mu\alpha.M_1[\alpha \Leftarrow A]) = \sum_{i=1}^{n} \pi(P^*, A) \cdot |Q_i| \cdot \pi([\alpha]Q_i, M_1)$$

Hence, LHS=RHS is proved. □

From the previous lemma, we can show the following property.

Lemma 6.6.

If $M \triangleright_{\mu^-} N$, then $\#M \geq \#N$

Proof. Suppose that $P_1, \ldots, P_n$ are all of the subterm occurrences of M that are μ-abstractions. For each $1 \leq i \leq n$, if M_i is the $*$-marked term obtained from M by marking the subterm P_i with $*$, then we can find a unique N_i such that $M_i \triangleright_{\mu^-} N_i$ and $E(N_i) \equiv N$. We suppose that $P^*_{i,1}, \ldots, P^*_{i,m_i}$ are all of subterm occurrence in N_i that are marked with $*$. If there is no such $*$-marked subterm occurrence in N_i, we define $m_i = 0$. Note that, since P_i^* in M_i is a μ-abstraction, any $P^*_{i,j}$ in N_i is a μ-abstraction, and $P_{1,1}, \ldots, P_{n,m_n}$ are all of the subterm occurrences in N that are μ-abstractions. Then we have

$$\#M = \sum_{i=1}^{n} \pi(P_i, M), \text{ and } \#N = \sum_{i=1}^{n}\sum_{j=1}^{m_i} \pi(P_{i,j}, N),$$

where, if $m_i = 0$ then we consider $\sum_{j=1}^{m_i} \pi(P_{i,j}, N) = 0$. From the lemma 6.5, we have,

$$\pi(P_i^*, M_i) \geq \sum_{j=1}^{m_i} \pi(P^*_{i,j}, N_i)$$

for each i. If we ignore all $*$'s, we have

$$\pi(P_i, M) \geq \sum_{j=1}^{m_i} \pi(P_{i,j}, N).$$

This inequality also holds for i such that $m_i = 0$, since RHS$= 0$. Hence we have $\#M \geq \#N$. □

Furthermore, we need to define another map $\|\cdot\|$ from a type or a term to a finite sequence of natural number. Finite sequences are defined as maps $\overline{a}$ from natural numbers to natural numbers such that there exists a number n and $\overline{a}(i) = 0$ holds for any $i \geq n$.

Definition 6.7.

(1) $\overline{a}, \overline{b}, \overline{c}, \ldots$ denote infinite sequences of natural numbers. $\overline{a}(i)$ denotes the i-th element of $\overline{a}$ for any natural number $i \geq 0$. $\overline{0}$ denotes the sequence such that $\overline{0}(i) = 0$ for any i. If there is n such that $\overline{a}(i) = 0$ holds for any $i \geq n$, $\overline{a}$ is called a finite sequence. For any finite sequence $\overline{a}$, we define the length $l(\overline{a})$ of $\overline{a}$ as the maximum natural number n such that $\overline{a}(n-1) \neq 0$. We define $l(\overline{0}) = 0$.

(2) For a natural number n and a sequence $\overline{a}$, $n :: \overline{a}$ denotes the sequence such that $(n :: \overline{a})(0) = n$ and $(n :: \overline{a})(i+1) = \overline{a}(i)$ holds for any i. For sequences $\overline{a}$ and $\overline{b}$, $\overline{a} + \overline{b}$ denotes the sequence such that $(\overline{a} + \overline{b})(i) = \overline{a}(i) + \overline{b}(i)$ holds for any i.

(3) $\overline{a} \succ \overline{b}$ iff there is n such that $\overline{a}(n) > \overline{b}(n)$ and $\overline{a}(i) = \overline{b}(i)$ for any $i < n$.

Definition 6.8.

The map $\|\cdot\|$ from a type or a term to an sequence of natural numbers is defined as follows.

(i) $\|\sigma\| = \overline{0}$ and $\|x\| = \overline{0}$.

(ii) $\|\lambda x.M\| = \|M\|$, $\|\Lambda t.M\| = \|M\|$ and $\|[\alpha]M\| = \|M\|$.

(iii) $\|(\mu\alpha.M)\vec{A}\| = (s(\vec{A}) :: \|M\|) + \sum_i \|A_i\|$.

(iv) $\|(UV)\vec{A}\| = \|U\| + \|V\| + \sum_i \|A_i\|$.

(v) $\|(U\sigma)\vec{A}\| = \|U\| + \sum_i \|A_i\|$,

where we define $s(\vec{A}) = n$ for $\vec{A} \equiv (A_1, \ldots, A_n)$.

Note that, if we consider all of finite sequences, $\succ$ is not well-founded, but if the length of finite sequence is bounded by a natural number, $\succ$ is well-founded.

Lemma 6.9.
(1) For any term M, we have $l(\|M\|) \le \#M$.
(2) If $M \rhd_{\mu^-} N$, then $\|M\| \succ \|N\|$.

Proof. (1) We define $n_\mu(M)$ as the number of the symbols μ in M. Then we can prove (i) $l(\|M\|) \le n_\mu(M)$ and (ii) $n_\mu(M) \le \#M$ as follows. (ii) is clear from the definition of $\#$ and (i) is proved by induction on M in a straightforward way. Suppose that $M \equiv (\mu\alpha.N)\vec{A}$ and $s(\vec{B}) = n$. Then we have $l(\|(\mu\alpha.N)\vec{A}\|) \le \max\{l(\|N\|) + 1, l(\|A_1\|), \ldots, l(\|A_n\|)\}$. On the other hand, we have $n_\mu((\mu\alpha.N)\vec{A}) = 1 + n_\mu(N) + n_\mu(A_1) + \cdots + n_\mu(A_n)$. Since, from IH, we have $l(\|N\|) \le n_\mu(N)$ and $l(\|A_i\|) \le n_\mu(A_i)$ for any $1 \le i \le n$, $l(\|(\mu\alpha.N)\vec{A}\|) \le n_\mu((\mu\alpha.N)\vec{A})$ holds.

(2) This is proved by induction on $M \rhd_{\mu^-} N$. Note that if $\bar{a} \succ \bar{b}$ then $n :: \bar{a} \succ n :: \bar{b}$ for any n, and $\bar{a} + \bar{c} \succ \bar{b} + \bar{c}$ for any $\bar{c}$.

In the case where $M \equiv (\mu\alpha.M')A\vec{B}$ and $N \equiv (\mu\alpha.M'[\alpha \Leftarrow A])\vec{B}$, we have

$$\|M\| = \|(\mu\alpha.M')A\vec{B}\| = ((n+1) :: \|M'\|) + \|A\| + \sum_{i=1}^{n} \|B_i\|,$$

$$\|N\| = \|(\mu\alpha.M'[\alpha \Leftarrow A])\vec{B}\| = (n :: \|M'[\alpha \Leftarrow A]\|) + \sum_{i=1}^{n} \|B_i\|,$$

where $n = s(\vec{B})$. Then we have $\|M\|(0) \ge \|N\|(0)$. Hence $\|M\| \succ \|N\|$ holds.

Other cases are proved more simply. □

From the lemmas 6.6 and 6.9, the proposition 6.1 is proved as follows.

Proof of the proposition 6.1 For any $\lambda\mu$-terms M and N, if $M \rhd^*_{\mu^-} N$, we have $l(\|N\|) \le \#M$ from the lemma 6.6 and the lemma 6.9 (1). Therefore, for any N such that $M \rhd^*_{\mu^-} N$, $\|N\|$ is the finite sequence of the length $\le \#M$. From the lemma 6.9 (2), if there is an infinite sequence $M \rhd_{\mu^-} M_1 \rhd_{\mu^-} M_2 \rhd_{\mu^-} \ldots$, there is an infinite decreasing sequence of finite sequences whose length is bounded by $\#M$, but it is contradictory. □

7 Strong normalizability of $\lambda_V\mu$

In this section, we prove the strong normalizability of $\lambda_V\mu$.

We prove the following claims to prove the strong normalizability: (1) $\rhd_{s\eta}$ is strongly normalizable for untyped terms, (2) $\rhd_{s\eta}$ can be postponed if the term is typable, and (3) $\rhd_{\lambda\mu}$ is strongly normalizable for typable terms, and If (1) and (2) hold and we assume that there is an infinite reduction sequence of a typable term in $\lambda_V\mu$, we can find an infinite sequence of $\rhd_{\lambda\mu}$ by postponing $\rhd_{s\eta}$, and that contradicts (3).

Firstly, we prove (1) and (2), then we prove (3) by the results of the previous sections.

Lemma 7.1.
$\rhd_{s\eta}$ is strongly normalizable.

Proof. We define n_M by (the number of symbols λ in M)+(the number of symbols μ in M). Then it is clear that if $M \rhd_{s\eta} N$ then $n_M > n_N$. □

Lemma 7.2.
If M is typable and $M \rhd_{s\eta} \rhd_{\lambda\mu} N$, then $M \rhd^+_{\lambda\mu} \cdot \rhd^*_{s\eta} N$.

Proof. This is proved in a straightforward way except the case of

$([\alpha]\mu\beta.(\lambda y.M))N \rhd_{s\eta} (\lambda y.M[\beta := \alpha])N \rhd_{\lambda\mu} M[\beta := \alpha][y := N]$.

But there is no such case, since if $([\alpha]\mu\beta.(\lambda y.M))N$ is typable then the subterm $\lambda y.M$ must have the type $\bot$, but it is impossible. □

Then we prove the strong normalizability of $\rhd_{\lambda\mu}$. Firstly we define augmentations of terms, which have no inessential subterm occurrences, since they contain no eliminator. Then it is proved that any $\lambda\mu$-reduction sequence of typable terms gives a $\lambda\mu$-reduction sequence of typable augmentations with the same length. Therefore, if we suppose the existence of an infinite $\lambda\mu$-reduction sequence of a typable term, it gives an infinite $\lambda\mu$-reduction sequence of augmentations. However, since any reduction from an augmentation is either $\rhd^{e}_{s\lambda}$, $\rhd^{t}_{\mu^+_e}$ or $\rhd^{e}_{\mu^-}$, that contradicts with the results of the sections 5 and 6 of this paper.

Definition 7.3. (Augmentations)

For a $\lambda\mu$-term M, the augmentations of M are defined inductively as follows. In the following, $\mathrm{Aug}(M)$ denotes the set of augmentations of M.

(1) For any λ-variable x, $\mathrm{Aug}(x) = \{x\}$.

(2) If $M' \in \mathrm{Aug}(M)$, then $\lambda x.M' \in \mathrm{Aug}(\lambda x.M)$.

(3) If $M' \in \mathrm{Aug}(M)$, then $\Lambda t.M' \in \mathrm{Aug}(\Lambda t.M)$.

(4) If $M' \in \mathrm{Aug}(M)$, then $[\alpha]M' \in \mathrm{Aug}([\alpha]M)$.

(5) If $M' \in \mathrm{Aug}(M)$ and $N' \in \mathrm{Aug}(N)$, then $M'N' \in \mathrm{Aug}(MN)$.

(6) If $M' \in \mathrm{Aug}(M)$, then $M'\sigma \in \mathrm{Aug}(M\sigma)$.

(7) If $M' \in \mathrm{Aug}(M)$, P is a term which includes no eliminator as its subterm and z is a fresh variable, then $\mu\alpha.(\lambda z.M')([\alpha]P) \in \mathrm{Aug}(\mu\alpha.M)$.

Lemma 7.4.

(1) If $\Gamma;\Delta \vdash M : \sigma$ holds, then there is an augmentation M' of M such that $\Gamma, c : \forall t.t;\Delta \vdash M' : \sigma$, where c is a variable which does not occur in M.

(2) If $M' \in \mathrm{Aug}(M)$ for some M, then every subterm of M' is essential.

Proof. (1) This claim is proved by induction on the proof of $\Gamma;\Delta \vdash M : \sigma$.

(2) It is easily shown by induction on M that M' includes no eliminator. Then the assertion is immediately proved. □

Lemma 7.5.

Suppose that $M' \in \mathrm{Aug}(M)$, $V' \in \mathrm{Aug}(V)$ and $A' \in \mathrm{Aug}(A)$. Then we have the following.

(1) $M'[x := V']$ is an augmentation of $M[x := V]$.

(2) $M'[t := \sigma]$ is an augmentation of $M[t := \sigma]$.

(3) $M'[\alpha := \beta]$ is an augmentation of $M[\alpha := \beta]$.

(4) $M'[\alpha \Leftarrow A']$ is an augmentation of $M[\alpha \Leftarrow A]$.

Proof. These are proved by induction on M in a straightforward way. □

Lemma 7.6.

Suppose that $M' \in \mathrm{Aug}(M)$ and $M \rhd_{\lambda\mu} N$, then there is an augmentation N' of N such that $M' \rhd_{\lambda\mu} N'$

Proof. By induction on $M \rhd N$.

(Case 1) M is a redex.

(Case 1.1) (β_v) : $(\lambda x.N)V \rhd N[x := V]$. We have $M' \equiv (\lambda x.N')V' \rhd_{\lambda\mu} N'[x := V']$, and we have $N'[x := V'] \in \mathrm{Aug}(N[x := V])$ from the lemma 7.5.

(Case 1.2) (μ) : $(\mu\alpha.M)A \rhd \mu\alpha.M[\alpha \Leftarrow A]$. We have $M' \equiv (\mu\alpha.(\lambda z.N')([\alpha]P))A' \rhd_{\lambda\mu} \mu\alpha.(\lambda z.N'[\alpha \Leftarrow A'])([\alpha]P[\alpha \Leftarrow A']A')$. From the lemma 7.5, $N'[\alpha \Leftarrow A'] \in \mathrm{Aug}(N[\alpha \Leftarrow A])$ holds, therefore, we have that $P[\alpha \Leftarrow A']A'$ includes no eliminator since so do P and A'.

(Case 2) $\mu\alpha.M \rhd \mu\alpha.N$. From IH, we have $M' \rhd N'$ for some $N' \in \mathrm{Aug}(N)$. Hence, we have $\mu\alpha.(\lambda z.M')([\alpha]P) \rhd \mu\alpha.(\lambda z.N')([\alpha]P)$.

Other cases are similarly proved. □

Proposition 7.7. (Strong normalizability of $\rhd_{\lambda\mu}$)

If M is a typable term, there is no infinite sequence $M \equiv M_0, M_1, \ldots$ such that $M_i \rhd_{\lambda\mu} M_{i+1}$ for any i.

Proof. From the lemma 7.4 (1), there is a typable augmentation M' of M. If we suppose that $M \equiv M_0 \rhd_{\lambda\mu} M_1 \rhd_{\lambda\mu} \dots$ is an infinite sequence, then, by the lemma 7.6, we can find an infinite sequence $M'_0, M'_1, \dots$ such that $M'_i \in \mathrm{Aug}(M_i)$ and $M'_i \rhd_{\lambda\mu} M'_{i+1}$ for any i. From the lemma 7.4 (2), the redex of the reduction $M'_i \rhd_{\lambda\mu} M'_{i+1}$ is essential in M'_i and the reduction is not $\rhd_{\mu_1^+}$ for any i, so any reduction $M'_i \rhd_{\lambda\mu} M'_{i+1}$ is $\rhd^e_{s\lambda}$, $\rhd^{\iota}_{\mu_2^+}$ or $\rhd^e_{\mu^-}$. Therefore, from the proposition 5.3, we have

$\overline{\overline{M'}}_i \equiv \overline{\overline{M'}}_{i+1}$ (if $M'_i \rhd_{\mu^-} M'_{i+1}$),
$\overline{\overline{M'}}_i \rhd^+ \overline{\overline{M'}}_{i+1}$ (otherwise).

Since the μ^--reduction is strongly normalizable, the reduction $M'_i \rhd_{\lambda\mu} M'_{i+1}$ is not μ^--reduction for infinitely many i, so we can find an infinite reduction sequence of $\overline{\overline{M'}}_0$ in the domain-free polymorphic λ-calculus. We have also that $\overline{\overline{M'}}_0$ is typable from the proposition 4.7. Therefore, that contradicts the strong normalizability of typable λ-terms. □

Theorem 7.8. (Strong normalizability of $\lambda_V\mu$)
Every typable $\lambda_V\mu$-term is strongly normalizable.

Proof. Suppose that M is a typable term and there is an infinite sequence $M \equiv M_0 \rhd M_1 \rhd M_2 \rhd \dots$. From the lemma 7.1, infinitely many $\rhd$'s in the sequence above are $\rhd_{\lambda\mu}$. Furthermore from the lemma 7.2, we can find an infinite sequence $M \equiv M_0 \rhd_{\lambda\mu} M'_1 \rhd_{\lambda\mu} M'_2 \rhd_{\lambda\mu} \dots$, but this contradicts the proposition 7.7. □

8 Concluding remarks

In this paper, we proved the confluency and the strong normalizability of the call-by-value $\lambda\mu$-calculus which has the domain-free style. However, we can consider variants of $\lambda\mu$-calculus.

For example, as a reduction rule of $\rhd_\lambda$ in $\lambda\mu$-calculus, we can consider the additional rule

(η_t) $\Lambda t.Vt \rhd V$ (if t is not free in V).

In fact, we can prove the confluency and the strong normalizability of the system with the rule (η_t) by using the method in this paper. But we should note that, we must take the domain-free polymorphic typed λ-calculus with the same η_t-rule as the codomain of the modified CPS-translation to prove the soundness of the modified CPS-translation. If we consider λ-calculus with η_t-rule, the reduction relation $\Lambda t.Vt \rhd V$ in $\lambda\mu$-calculus is proved to be preserved by the modified CPS-translation as follows.

$\Phi(\Lambda t.Vt) \equiv \Lambda t.\lambda k.\Phi(V)tk \rhd_{\beta\eta} \Lambda t.\Phi(V)t \rhd_{\beta\eta} \Phi(V)$,

where we use the η_t-rule in λ-calculus at the last step.

Another variant is the Church-style system. The Church-style call-by-value $\lambda\mu$-calculus is defined as follows. The pseudo-terms are defined as

$M ::= x \mid \lambda x:\sigma.M \mid \Lambda t.M \mid \mu\alpha^\sigma.M \mid MM \mid M\sigma \mid [\alpha]M.$

The terms of the Church-style $\lambda\mu$-calculus are defined as the pseudo-terms which are typable by the following axioms and rules.

$$\Gamma;\Delta \vdash x : \Gamma(x) \quad (ass)$$

$$\frac{\Gamma \cup \{x:\sigma\};\Delta \vdash M:\tau}{\Gamma;\Delta \vdash \lambda x:\sigma.M : \sigma \to \tau}\ (\to I) \qquad \frac{\Gamma;\Delta \vdash M:\sigma \to \tau \quad \Gamma;\Delta \vdash N:\sigma}{\Gamma;\Delta \vdash MN:\tau}\ (\to E)$$

$$\frac{\Gamma;\Delta \vdash M:\sigma}{\Gamma;\Delta \vdash \Lambda t.M : \forall t.\sigma}\ (\forall I) \qquad \frac{\Gamma;\Delta \vdash M:\forall t.\sigma}{\Gamma;\Delta \vdash M\tau : \sigma[t:=\tau]}\ (\forall E)$$

$$\frac{\Gamma;\Delta \vdash M:\sigma}{\Gamma;\Delta \cup \{\sigma^\alpha\} \vdash [\alpha]M : \perp}\ (\perp I) \qquad \frac{\Gamma;\Delta \cup \{\sigma^\alpha\} \vdash M:\perp}{\Gamma;\Delta \vdash \mu\alpha^\sigma.M : \sigma}\ (\perp E)$$

And the reduction rules of the Church-style call-by-value $\lambda\mu$-calculus are defined from the following axiom schemes.

(β_v) $(\lambda x : \sigma.M)V^\sigma \rhd M[x := V]$,
(β_t) $(\Lambda t.M)\sigma \rhd M[t := \sigma]$,
(η_v) $\lambda x : \sigma.Vx \rhd V$ (if $x \notin FV(V)$),
(μ_r) $(\mu\alpha^{\sigma\to\tau}.M)N^\sigma \rhd \mu\alpha^\tau.M[\alpha \Leftarrow N]$,
(μ_l) $V^{\sigma\to\tau}(\mu\alpha^\sigma.M) \rhd \mu\alpha^\tau.M[V \Rightarrow \alpha]$,
(μ_t) $(\mu\alpha^{\forall t.\sigma}.M)\tau \rhd \mu\alpha^{\sigma[t:=\tau]}.M[\alpha \Leftarrow \tau]$,
$(\mu\eta)$ $\mu\alpha^\sigma.[\alpha^\sigma]M \rhd M$ (if $\alpha \notin FV(M)$),
(rn) $[\alpha^\sigma](\mu\beta^\sigma.V) \rhd V[\beta := \alpha]$.

The confluency for this system can be proved in the same way by the method in this paper. The strong normalizability can be proved easily from the result of this paper by using the following fact. If we define the map $\lfloor\cdot\rfloor$ from Church-style terms to domain-free-style terms such that

$\lfloor\lambda x : \sigma.M\rfloor \equiv \lambda x.\lfloor M\rfloor$,
$\lfloor\mu\alpha^\sigma.M\rfloor \equiv \mu\alpha.\lfloor M\rfloor$,

then it is clear that $M \rhd N$ iff $\lfloor M\rfloor \rhd \lfloor N\rfloor$ for any Church-style terms M and N.

Acknowledgements I wish to thank Makoto Tatsuta, Ken-etsu Fujita and the referees for their helpful comments and advises.

References

[1] K. Baba, S. Hirokawa and K. Fujita, Parallel reduction in type free $\lambda\mu$-calculus, Electronic Notes in Theoretical Computer Science 42 (2001) 52-66.

[2] F. Barbanera and S. Berardi, A strong normalization result for classical logic, Annals of Pure and Applied Logic 76 (1995) 99-116.

[3] H.P. Barendregt, Lambda calculi with types, in: S. Abramsky, Dov M. Gabbay and T.S.E. Maibaum, eds., Handbook of Logic in Computer Science Vol.II (Oxford University Press 1992).

[4] G. Barthe and M.H. Sørensen, Domain-free pure type systems, in: S. Adian and A. Nerode, eds., Logical Foundations of Computer Science, Lecture Notes in Computer Science 1234 (1997) 9-20.

[5] R. David and W. Py, $\lambda\mu$-calculus and Böhm's theorem, Journal of Symbolic Logic 66 (1) (2001) 407–413.

[6] P. de Groote, A CPS-translation for the $\lambda\mu$-calculus, in: S. Tison, ed., Trees in Algebra and Programming, Lecture Notes in Computer Science 787 (1994) 85-99.

[7] P. de Groote, A simple calculus of exception handling, in: M. Dezani-Ciancaglini and G. Plotkin, eds., Typed Lambda Calculi and Applications, Lecture Notes in Computer Science 902 (1995) 201-215.

[8] M. Felleisen, D.P. Friedman, E. Kohlbecker and B. Duba, A syntactic theory of sequential control, Theoretical Computer Science 52 (1987) 205-237.

[9] K. Fujita, Explicitly typed $\lambda\mu$-calculus for polymorphism and call-by-value, in: J.-Y. Girard, ed., Typed Lambda Calculi and Applications, Lecture Notes in Computer Science 1581 (1999) 162-176.

[10] K. Fujita, Domain-free $\lambda\mu$-calculus, Theoretical Informatics and Applications 34 (2000) 433-466.

[11] J.-Y. Girard, The system F of variable types, fifteen years later, Theoretical Computer Science 45 (1986) 159-192.

[12] J.-Y. Girard, P. Taylor and Y. Lafont, Proofs And Types (Cambridge University Press 1989).

[13] C.-H.L. Ong and C.A. Stewart, A Curry-Howard foundation for functional computation with control, Proc. 24th Annual ACM symposium of Principles of Programming Languages (1997) 215-227.

[14] M. Parigot, $\lambda\mu$-Calculus: An algorithmic interpretation of natural deduction, in: A. Voronkov, ed., Proceedings of the International Conference on Logic Programming and Automated Reasoning, Lecture Notes in Computer Science 624 (1992) 190-201.

[15] M. Parigot, Proofs of strong normalization for second order classical natural deduction, Journal of Symbolic Logic 62 (4) (1997) 1461-1479.

[16] G.D. Plotkin, Call-by-name, call-by-value and the λ-calculus, Theoretical Computer Science 1 (1975) 125-159.