Vol. 48 No. 4 IPSJ Journal Apr. 2007

Regular Paper

Duality between Call-by-value Reductions and Call-by-name Reductions

Daisuke Kimura†

Wadler proposed the dual calculus, which corresponds to classical sequent calculus LK, and studied the relationship between the $\lambda\mu$-calculus and the dual calculus as equational systems to explain the duality between call-by-value and call-by-name in a purely syntactical way. Wadler left an open question whether one can obtain similar results by replacing the equations with reductions. This paper gives one answer to his question. We first refine the $\lambda\mu$-calculus as reduction systems by reformulating sum types and omitting problematic reduction rules that are not simulated by reductions of the dual calculus. Secondly, we give translations between the call-by-name $\lambda\mu$-calculus and the call-by-name dual calculus, and show that they preserve the call-by-name reductions. We also show that the compositions of these translations become identity maps up to the call-by-name reductions. We also give translations for the call-by-value systems, and show that they satisfy properties similar to the call-by-name translations. Thirdly, we introduce translations between the call-by-value $\lambda\mu$-calculus and the call-by-name one by composing the above translations with duality on the dual calculus. We finally obtain results corresponding to Wadler's, but our results are based on reductions.

1. Introduction

The Curry-Howard correspondence for classical logic

In the last twenty years, a lot of work has been done to extend the Curry-Howard correspondence to classical logic. Felleisen [3] introduced the $\mathcal{C}$ operator to model `call/cc`, Griffin [5] observed that the type of `call/cc` corresponds to Peirce's Law and extended the Curry-Howard correspondence to classical logic. In this line, the $\lambda\mu$-calculus introduced by Parigot [7] is well known. This calculus corresponds to classical natural deduction and has a simple structure, sufficient expressive power, and nice computational properties such as confluency and strong normalization. Later, a call-by-value (CBV) variant of the $\lambda\mu$-calculus was proposed by Ong and Stewart [6].

Duality

The call-by-name and call-by-value strategies have been well studied as evaluation strategies of programming languages. Filinski [4] suggested that duality between call-by-name and call-by-value is clarified by two notions of programs and continuations. Selinger [9] gave categorical semantics of the call-by-name and call-by-value $\lambda\mu$-calculi and explained Filinski's duality in terms of categorical duality.

Wadler's dual calculus and open question

Wadler [10),11] proposed *the dual calculus*, which corresponds to Gentzen's classical sequent calculus LK. LK is an appropriate formulation of classical logic that clearly expresses the duality that exists inside classical logic. The main feature of the dual calculus is that it has both terms and continuations as primitives. The computational meaning of the duality of classical logic is expressed in the dual calculus by the duality of terms and continuations. In the dual calculus, call-by-name and call-by-value strategies become dual strategies.

Wadler [11] introduced the translation from the $\lambda\mu$-calculus into the dual calculus, and its inverse translation from the dual calculus into the $\lambda\mu$-calculus. He showed that these translations form an equational correspondence, as defined by Sabry and Felleisen [8]. Moreover, he gave the translation from the $\lambda\mu$-calculus into itself by composing the above translations with duality on the dual calculus. This translation satisfies the following properties.

- It takes call-by-value equalities into call-by-value equalities, and vice versa.
- It is an involution up to call-by-value/call-by-name equality.

In other words, he explained Filinski's duality in a purely syntactical way. However, the $\lambda\mu$-calculus and the dual calculus adopted in his

† Department of Informatics, The Graduate University for Advanced Studies/National Institute of Informatics

paper were equational systems, and his results are based on equalities. This is because some rules of the $\lambda\mu$-calculus are not simulated by reductions of the dual calculus, and it is also problematic to introduce some rules, such as (η)-rules, as reductions. But when we discuss whether duality between call-by-value and call-by-name also holds as a computational procedure, we should consider reductions. In fact, Wadler noted an open question in his paper whether one can replace the equations of his paper with reductions, and extend the properties with equations to properties with reductions.

Our purpose, problems, and solutions Our purpose in this paper is to answer his question. We encounter problems when we try to obtain refined results by replacing the equations of his paper with reductions. These problems are grouped in the following three cases.

Problem 1 (ζ)-*rules*
To simulate (ζ)-rule under Wadler's translation $(-)^*$, we need (β_R)-reductions of the dual calculus in both directions. We give a typical example of this problem.

$$\begin{aligned}
&((\mu\alpha.[\gamma]\lambda x.[\alpha]y)z)^* \\
&\quad\equiv \big(([x.(y\bullet\alpha)]\mathrm{not}\bullet\gamma).\alpha\bullet(z@\beta)\big) \\
&\quad=^n_{(\beta_R)} \big([x.((y\bullet(z@\beta)))]\mathrm{not}\bullet\gamma\big).\beta \\
&\quad=^n_{(\beta_R)} \big([x.((y\bullet(z@\delta)).\delta\bullet\beta)]\mathrm{not}\bullet\gamma\big).\beta \\
&\quad\equiv (\mu\beta.[\gamma]\lambda x.[\beta](yz))^*
\end{aligned}$$

Problem 2 $(\eta_\vee)$-*rule:* $M =_n \mu(\alpha,\beta).[\alpha,\beta]M$
To simulate $(\eta_\vee)$-rule under Wadler's translation $(-)^*$, we need both of $(\eta_\vee)$-*reduction* and $(\eta_\vee)$-*expansion* of the dual calculus.

$$\begin{aligned}
&(\mu(\alpha,\beta).[\alpha,\beta]M)^* \\
&\quad\equiv \big(\langle(\langle((M^*\bullet[\alpha,\beta]).\alpha\rangle\mathrm{inl}\bullet\gamma).\beta\rangle\mathrm{inr}\bullet\gamma\big).\gamma \\
&\quad=^n_{(\eta_\vee)} \big(\langle(\langle((M^*\bullet[\alpha,\beta]).\alpha\rangle\mathrm{inl}\bullet\gamma).\beta\rangle\mathrm{inr} \\
&\qquad\bullet\widehat{r}\big).\gamma \\
&\quad=^n_{(\beta_\vee)} \big((\langle(M^*\bullet[\alpha,\beta]).\alpha\rangle\mathrm{inl}\bullet\gamma).\beta \\
&\qquad\bullet y.(\langle y\rangle\mathrm{inr}\bullet\gamma)\big).\gamma \\
&\quad=^n_{(\beta_R)} \big(\langle(M^*\bullet[\alpha,y.(\langle y\rangle\mathrm{inr}\bullet\gamma)]).\alpha\rangle\mathrm{inl} \\
&\qquad\bullet\gamma\big).\gamma \\
&\quad=^n_{(\eta_\vee)} \big(\langle(M^*\bullet[\alpha,y.(\langle y\rangle\mathrm{inr}\bullet\gamma)]).\alpha\rangle\mathrm{inl} \\
&\qquad\bullet\widehat{r}\big).\gamma \\
&\quad=^n_{(\beta_\vee)} \big((M^*\bullet[\alpha,y.(\langle y\rangle\mathrm{inr}\bullet\gamma)]).\alpha \\
&\qquad\bullet x.(\langle x\rangle\mathrm{inl}\bullet\gamma)\big).\gamma \\
&\quad=^n_{(\beta_R)} \big(M^*\bullet[x.(\langle x\rangle\mathrm{inl}\bullet\gamma), \\
&\qquad y.(\langle y\rangle\mathrm{inr}\bullet\gamma)]\big).\gamma \\
&\quad=^n_{(\eta_\vee)} \big(M^*\bullet\gamma\big).\gamma \\
&\quad=^n_{(\eta_R)} M^*
\end{aligned}$$

where $\widehat{r}$ is $[x.(\langle x\rangle\mathrm{inl}\bullet\gamma), y.(\langle y\rangle\mathrm{inr}\bullet\gamma)]$. However, if we simply omit $(\eta_\vee)$-rule to avoid this problem, then we meet another problem. If we want to obtain the equational correspondence formed by $(-)^*$ and $(-)_*$, which is one of Wadler's main results, then we should show $((\mu(\alpha,\beta).S)^*)_* =_n \mu(\alpha,\beta).(S^*)_*$. We need $(\eta_\vee)$-rule to show this claim. (We abbreviate $\mu(\alpha_2,\beta_2).[\alpha_2]\mu\alpha.(S^*)_*$ in M.)

$$\begin{aligned}
&((\mu(\alpha,\beta).S)^*)_* \\
&\quad\equiv \big(\langle(\langle(\langle S^*.\alpha\rangle\mathrm{inl}\bullet\gamma).\beta\rangle\mathrm{inr}\bullet\gamma).\gamma\big)_* \\
&\quad\equiv \mu\gamma.[\gamma]\mu(\alpha_1,\beta_1).[\beta_1]\mu\beta.[\gamma]\mu(\alpha_2,\beta_2) \\
&\qquad.[\alpha_2]\mu\alpha.(S^*)_* \\
&\quad\equiv \mu\gamma.[\gamma]\mu(\alpha_1,\beta_1).[\beta_1]\mu\beta.[\gamma]M \\
&\quad=^{(\eta_\vee)}_n \mu(\alpha_3,\beta_3).[\alpha_3,\beta_3]\mu\gamma.[\gamma]\mu(\alpha_1,\beta_1) \\
&\qquad.[\beta_1]\mu\beta.[\gamma]M \\
&\quad=^{(\zeta_\vee)}_n \mu(\alpha_3,\beta_3).[\alpha_3,\beta_3]\mu(\alpha_1,\beta_1).[\beta_1]\mu\beta \\
&\qquad.[\alpha_3,\beta_3]M \\
&\quad=^{(\beta_\vee)}_n \mu(\alpha_3,\beta_3).[\beta_3]\mu\beta.[\alpha_3,\beta_3]M \\
&\quad\equiv \mu(\alpha_3,\beta_3).[\beta_3]\mu\beta.[\alpha_3,\beta_3]\mu(\alpha_2,\beta_2) \\
&\qquad.[\alpha_2]\mu\alpha.(S^*)_* \\
&\quad=^{(\beta_\vee)}_n \mu(\alpha_3,\beta_3).[\beta_3]\mu\beta.[\alpha_3]\mu\alpha.(S^*)_* \\
&\quad=^{(\beta_\mu)}_n \mu(\alpha_3,\beta_3).(S^*)_*[{}^{\alpha_3}/_{\alpha},{}^{\beta_3}/_{\beta}] \\
&\quad\equiv \mu(\alpha,\beta).(S^*)_*
\end{aligned}$$

Problem 3 $(\eta_\neg)$-*rule:* $M =_n \lambda x.Mx$
To simulate $(\eta_\neg)$-rule under Wadler's translation $(-)^*$, we need both $(\eta_\neg)$-*reduction* and $(\eta_\neg)$-*expansion* of the dual calculus.

$$\begin{aligned}
&(\lambda x.Mx)^* \equiv [\,x.(M^*\bullet\mathrm{not}\langle x\rangle)\,]\mathrm{not} \\
&\quad=^n_{(\eta_R)} ([\,x.(M^*\bullet\mathrm{not}\langle x\rangle)\,]\mathrm{not}\bullet\gamma).\gamma \\
&\quad=^n_{(\eta_\neg)\,exp} ([\,x.(M^*\bullet\mathrm{not}\langle x\rangle)\,]\mathrm{not} \\
&\qquad\bullet\mathrm{not}\langle\,([\alpha]\mathrm{not}\bullet\gamma).\alpha\,\rangle).\gamma \\
&\quad=^n_{(\beta_\neg)} ((([\alpha]\mathrm{not}\bullet\gamma).\alpha\bullet x.(M^*\bullet\mathrm{not}\langle x\rangle)).\gamma \\
&\quad=^n_{(\beta_L)} (M^*\bullet\mathrm{not}\langle\,([\alpha]\mathrm{not}\bullet\gamma).\alpha\,\rangle).\gamma \\
&\quad=^n_{(\eta_\neg)} (M^*\bullet\gamma).\gamma \\
&\quad=^n_{(\eta_R)} M^*
\end{aligned}$$

However, we need $(\eta_\neg)$-rule to simulate $x(\mu\alpha.S) =_v S[{}^{x\{-\}}/_{[\alpha]\{-\}}]$, which is defined in the call-by-value $\lambda\mu$-calculus as a part of (ζ)-rule. For example,

$$\begin{aligned}
&(x(\mu\alpha.[\beta]\lambda z.[\alpha]y))^* \\
&\quad\equiv x\bullet\mathrm{not}\big\langle([z.(y\bullet\alpha)]\mathrm{not}\bullet\beta).\alpha\big\rangle \\
&\quad=^v_{(\eta_\neg)} [y'.(x\bullet\mathrm{not}\langle y'\rangle)]\mathrm{not} \\
&\qquad\bullet\mathrm{not}\big\langle([z.(y\bullet\alpha)]\mathrm{not}\bullet\beta).\alpha\big\rangle
\end{aligned}$$

$$
\begin{aligned}
&=^{v}_{(\beta_\neg)} ([z.(y \bullet \alpha)]\mathrm{not} \bullet \beta).\alpha \\
&\quad \bullet y'.(x \bullet \mathrm{not}\langle y' \rangle) \\
&=^{v}_{(\beta_L)} [z.(y \bullet y'.(x \bullet \mathrm{not}\langle y' \rangle))]\mathrm{not} \bullet \beta \\
&=^{v}_{(\beta_R)} [z.(x \bullet \mathrm{not}\langle y \rangle)]\mathrm{not} \bullet \beta \\
&\equiv ([\beta]\lambda z.(xy))^*
\end{aligned}
$$

For $(\eta_\supset)$-rule of the $\lambda\mu$-calculus, we also encounter a problem similar to this one.

Problem 1 is due to the so-called administrative redexes, and can be solved by modifying Wadler's translations. The idea of this modification is similar to the modified CPS translation introduced by de Groote [1),2)]. However, we need different modifications for call-by-value and call-by-name calculi.

Problem 2 is caused by the difference in how sums are formulated in the $\lambda\mu$-calculus and the dual calculus. Wadler added sum types to the $\lambda\mu$-calculus following Selinger [9)]. This formulation is based on multiple-concluded sequents as follows.

$$\frac{\Gamma \mid S \vdash_{\lambda\mu} \Delta, \alpha : A, \beta : B}{\Gamma \vdash_{\lambda\mu} \Delta \mid \mu(\alpha,\beta).S : A \vee B}$$

$$\frac{\Gamma \vdash_{\lambda\mu} \Delta \mid M : A \vee B}{\Gamma \mid [\alpha,\beta]M \vdash_{\lambda\mu} \Delta, \alpha : A, \beta : B}$$

The formulation of sums in the dual calculus, on the other hand, is based on single-concluded sequents:

$$\frac{\Gamma \vdash_{dc} \Delta \mid M : A}{\Gamma \vdash_{dc} \Delta \mid \langle M \rangle \mathrm{inl} : A \vee B}$$

$$\frac{\Gamma \vdash_{dc} \Delta \mid N : B}{\Gamma \vdash_{dc} \Delta \mid \langle N \rangle \mathrm{inr} : A \vee B}$$

$$\frac{\Gamma \vdash_{dc} \Delta \mid N : B}{[K,L] : A \vee B \mid \Gamma \vdash_{dc} \Delta \mid \langle N \rangle \mathrm{inr} : A \vee B}$$

Our solution to this problem is to refine the formulation of sums in the $\lambda\mu$-calculus, and omit $(\eta_\vee)$-rule. We introduce sums of the $\lambda\mu$-calculus by using usual injections and case-expressions.

To avoid Problem 3, we remove $(\eta_\neg)$ and $(\eta_\supset)$-rules, and restrict the call-by-value $\lambda\mu$-calculus by omitting some rules that cannot be simulated without $(\eta_\neg)$ and $(\eta_\supset)$-rules.

We also encounter problems when we consider the inverse translation from the dual calculus into the $\lambda\mu$-calculus. Since they are similar to the above Problem 1, we can solve them by modifying Wadler's original translation. However, we also need different modifications for call-by-value and call-by-name.

Overview

In Section 2, we present the detailed formulation of our call-by-value and call-by-name $\lambda\mu$-calculi, and compare them with the $\lambda\mu$-calculi given by Wadler (2005). In Section 3, we present the dual calculus as a reduction system. In Section 4, we define the call-by-name translation from the call-by-name $\lambda\mu$-calculus into the call-by-name dual calculus, and show that this translation preserve call-by-name reductions (Theorem 16). We also define the call-by-value translation, and show that it also preserves call-by-value reductions. In Section 5, we give the inverse translations from the dual calculus into the $\lambda\mu$-calculus to both call-by-name and call-by-value, and show that they preserves reductions (Theorem 28, 34). We also show that the compositions of the call-by-name translations become identity maps up to the call-by-name reductions, and show the similar property for the call-by-value translations (Proposition 35, 36). In Section 6, we introduce translations between the call-by-value and call-by-name $\lambda\mu$-calculi by composing the above translations with duality on the dual calculus. We finally obtain results corresponding to Wadler's (Theorem 40), but our results are based on reductions.

2. The $\lambda\mu$-calculus

In this paper, we consider the two variants of the $\lambda\mu$-calculus, call-by-value and call-by-name, as reduction systems.

The types of the $\lambda\mu$-calculus in this paper follow Wadler, i.e., let A and B range over types, then a type is atomic X, a conjunction $A \& B$, a disjunction $A \vee B$, a negation $\neg A$, or an implication $A \supset B$.

Definition (Types of the $\lambda\mu$-calculus)
$A, B ::= X \mid A \& B \mid A \vee B \mid A \supset B \mid \neg A$

Two disjoint countable sets of variables for the $\lambda\mu$-calculus are given, one is called *variables* (denoted by $x, y, z, \ldots$) and the other is called *covariables* (denoted by $\alpha, \beta, \gamma, \ldots$). We distinguish two notions of *terms* (denoted by $M, N, \ldots$) and *statements* (denoted by $S, T, \ldots$) as the expression of the $\lambda\mu$-calculus following Wadler. A term is a variable x, a λ-abstraction $\lambda x.M$ or $\lambda x.S$, an implication application OM (where $O : A \supset B$), a projection $\mathrm{fst}(M)$ or $\mathrm{snd}(M)$, a pairing $\langle M, N \rangle$, a μ-abstraction $\mu\alpha.S$, or a term for sums. A statement is a covariable application $[\alpha]M$, a negation appli-

$$\frac{}{\Gamma, x : A \vdash_{\lambda\mu} \Delta \mid x : A}\ \mathrm{Ax}$$

$$\frac{\Gamma, x : A \vdash_{\lambda\mu} \Delta \mid M : B}{\Gamma \vdash_{\lambda\mu} \Delta \mid \lambda x.M : A \supset B}\ \supset\mathrm{I} \qquad \frac{\Gamma \vdash_{\lambda\mu} \Delta \mid M : A \supset B \quad \Gamma \vdash_{\lambda\mu} \Delta \mid N : A}{\Gamma \vdash_{\lambda\mu} \Delta \mid MN : B}\ \supset\mathrm{E}$$

$$\frac{\Gamma \vdash_{\lambda\mu} \Delta \mid M : A\&B}{\Gamma \vdash_{\lambda\mu} \Delta \mid \mathrm{fst}(M) : A}\ \&\mathrm{E}_1 \qquad \frac{\Gamma \vdash_{\lambda\mu} \Delta \mid M : A\&B}{\Gamma \vdash_{\lambda\mu} \Delta \mid \mathrm{snd}(M) : B}\ \&\mathrm{E}_2$$

$$\frac{\Gamma \vdash_{\lambda\mu} \Delta \mid M : A}{\Gamma \vdash_{\lambda\mu} \Delta \mid \mathrm{inl}(M) : A \vee B}\ \vee\mathrm{I}_1 \qquad \frac{\Gamma \vdash_{\lambda\mu} \Delta \mid M : A}{\Gamma \vdash_{\lambda\mu} \Delta \mid \mathrm{inr}(M) : A \vee B}\ \vee\mathrm{I}_2$$

$$\frac{\Gamma \vdash_{\lambda\mu} \Delta \mid M : A \quad \Gamma \vdash_{\lambda\mu} \Delta \mid N : B}{\Gamma \vdash_{\lambda\mu} \Delta \mid \langle M, N \rangle : A\&B}\ \&\mathrm{I}$$

$$\frac{\Gamma \vdash_{\lambda\mu} \Delta \mid O : A \vee B \quad \Gamma, x : A \vdash_{\lambda\mu} \Delta \mid M : C \quad \Gamma, y : B \vdash_{\lambda\mu} \Delta \mid N : C}{\Gamma \vdash_{\lambda\mu} \Delta \mid \delta(O, x.M, y.N) : C}\ \vee\mathrm{E}$$

$$\frac{\Gamma \vdash_{\lambda\mu} \Delta \mid O : A \vee B \quad \Gamma, x : A \mid S \vdash_{\lambda\mu} \Delta \quad \Gamma, y : B \mid T \vdash_{\lambda\mu} \Delta}{\Gamma \mid \delta(O, x.S, y.T) \vdash_{\lambda\mu} \Delta}\ \vee\mathrm{E}$$

$$\frac{\Gamma, x : A \mid S \vdash_{\lambda\mu} \Delta}{\Gamma \vdash_{\lambda\mu} \Delta \mid \lambda x.S : \neg A}\ \neg\mathrm{I} \qquad \frac{\Gamma \vdash_{\lambda\mu} \Delta \mid M : \neg A \quad \Gamma \vdash_{\lambda\mu} \Delta \mid N : A}{\Gamma \mid MN \vdash_{\lambda\mu} \Delta}\ \neg\mathrm{E}$$

$$\frac{\Gamma \mid S \vdash_{\lambda\mu} \Delta, \alpha : A}{\Gamma \vdash_{\lambda\mu} \Delta \mid \mu\alpha.S : A}\ \mathrm{Act} \qquad \frac{\Gamma \vdash_{\lambda\mu} \Delta \mid M : A}{\Gamma \mid [\alpha]M \vdash_{\lambda\mu} \Delta, \alpha : A}\ \mathrm{Pass}$$

Fig. 1 Typing rules of the $\lambda\mu$-calculus.

cation OM (where $O : \neg A$), or a statement for sums. Any free occurrence of x in M and S is bound in the terms $\lambda x.M$ and $\lambda x.S$ respectively. Any free occurrence of α in S is bound in the term $\mu\alpha.S$.

For sums, we give another formulation without following Selinger [9]. A term for sums is a left injection $\mathrm{inl}(M)$, a right injection $\mathrm{inr}(M)$, and case $\delta(O, x.M, y.N)$, and a statement for sums is a case $\delta(O, x.S, y.T)$. Any free occurrences of x in M and y in N are bound in the term $\delta(O, x.M, y.N)$. Similarly, any free occurrences of x in S and y in T are bound in the term $\delta(O, x.S, y.T)$.

Definition (Terms and Statements of $\lambda\mu$)

$$\begin{aligned} M, N, O ::= & \, x \mid \lambda x.M \mid \lambda x.S \mid MN \mid \mu\alpha.S \\ & \mid \mathrm{fst}(M) \mid \mathrm{snd}(N) \mid \langle M, N \rangle \\ & \mid \mathrm{inl}(M) \mid \mathrm{inr}(N) \mid \delta(O, x.M, y.N) \\ S, T ::= & \, [\alpha]M \mid MN \mid \delta(O, x.S, y.T) \end{aligned}$$

We consider the term modulo α-conversion of variables and covariables. The sets of *free variables* of M and S (denoted by $\mathrm{FV}(M)$ and $\mathrm{FV}(S)$), and the sets of *free covariables* of M and S (denoted by $\mathrm{FCV}(M)$ and $\mathrm{FCV}(S)$) are defined as usual. A *typing judgment* of the $\lambda\mu$-calculus takes the form $\Gamma \vdash_{\lambda\mu} \Delta \mid M : A$ or $\Gamma \mid S \vdash_{\lambda\mu} \Delta$, where Γ denotes a *λ-context*, i.e., $x_1 : A_1, \ldots, x_n : A_n$, and Δ denotes a *μ-context*, i.e., $\alpha_1 : B_1, \ldots, \alpha_m : B_m$. We note that $\vdash_{\lambda\mu}$ is sometimes written as $\vdash$. The *typing rules* for the $\lambda\mu$-calculus are defined in **Fig. 1**.

We use two kinds of substitution for the $\lambda\mu$-calculus. The first $M[{}^{N}/{}_{x}]$ and $S[{}^{N}/{}_{x}]$ are the usual substitutions of a term N for all free occurrences of the variable x in M and S. The second $M[{}^{\mathcal{T}\{-\}}/{}_{[\alpha]\{-\}}]$ and $S[{}^{\mathcal{T}\{-\}}/{}_{[\alpha]\{-\}}]$ are substitutions of a statement context $\mathcal{T}\{-\}$ (i.e., a statement with a single hole accepting a term) for a covariable α. This second substitution is defined by induction on M and S using the following clause.

$$([\alpha]M)[{}^{\mathcal{T}\{-\}}/{}_{[\alpha]\{-\}}] \equiv \mathcal{T}\{\, M[{}^{\mathcal{T}\{-\}}/{}_{[\alpha]\{-\}}]\,\}$$

The other clauses are defined homomorphically like

$$\begin{aligned} &(MN)[{}^{\mathcal{T}(-)}/{}_{[\alpha](-)}] \\ &\quad \equiv M[{}^{\mathcal{T}(-)}/{}_{[\alpha](-)}]N[{}^{\mathcal{T}(-)}/{}_{[\alpha](-)}]. \end{aligned}$$

Note that $M[{}^{[\beta](-)}/{}_{[\alpha](-)}]$ and $S[{}^{[\beta](-)}/{}_{[\alpha](-)}]$ are sometimes written as $M[{}^{\beta}/{}_{\alpha}]$ and $S[{}^{\beta}/{}_{\alpha}]$ respectively.

2.1 The Call-by-name $\lambda\mu$-calculus

We need a notion of call-by-name evaluation and statement contexts to introduce the call-by-name $\lambda\mu$-calculus, which is equivalent as an equational theory to the one given by Wadler.

$(\beta_{\supset})$	$(\lambda x.M)N \longrightarrow_n M[^N/_x]$	
$(\beta_{\&})$	$\mathrm{fst}\langle M,N\rangle \longrightarrow_n M$	
	$\mathrm{snd}\langle M,N\rangle \longrightarrow_n N$	
$(\beta_{\vee})$	$\delta(\mathrm{inl}(O), x.E_n\{x\}, y.E'_n\{y\}) \longrightarrow_n E_n\{O\}$	
	$\delta(\mathrm{inr}(O), x.E_n\{x\}, y.E'_n\{y\}) \longrightarrow_n E'_n\{O\}$	
	$\delta(\mathrm{inl}(O), x.D_n\{x\}, y.D'_n\{y\}) \longrightarrow_n D_n\{O\}$	
	$\delta(\mathrm{inr}(O), x.D_n\{x\}, y.D'_n\{y\}) \longrightarrow_n D'_n\{O\}$	
$(\beta_{\neg})$	$(\lambda x.S)N \longrightarrow_n S[^N/_x]$	
(ζ)	$E_n\{\mu\alpha.S\} \longrightarrow_n \mu\beta.S[^{[\beta]E_n\{-\}}/_{[\alpha]\{-\}}]$	(where E_n is not $\{-\}$)
	$D_n\{\mu\alpha.S\} \longrightarrow_n S[^{D_n\{-\}}/_{[\alpha]\{-\}}]$	
(η_{μ})	$M \longrightarrow_n \mu\alpha.[\alpha]M$	(where $\alpha \notin \mathrm{FCV}(M)$)
(π)	$E_n\{\delta(O, x.M, y.N)\} \longrightarrow_n \delta(O, x.E_n\{M\}, y.E_n\{N\})$	(where E_n is not $\{-\}$)
	$D_n\{\delta(O, x.M, y.N)\} \longrightarrow_n \delta(O, x.D_n\{M\}, y.D_n\{N\})$	
(ν)	$\delta(O, x.S, y.T) \longrightarrow_n (\lambda y.T)\,\mu\beta.\delta(O, x.S, y.[\beta]y)$	(if T is not a simple form w.r.t. y)
	$\delta(O, x.S, y.D_n\{y\}) \longrightarrow_n (\lambda x.S)\,\mu\alpha.\delta(O, x.[\alpha]x, y.D_n\{y\})$	(if S is not a simple form w.r.t. x)

Fig. 2 Reduction rules of the call-by-name $\lambda\mu$-calculus.

A *call-by-name evaluation term context* (denoted by $E_n, E'_n, \ldots$) is a term context with a hole, and a *call-by-name evaluation statement context* (denoted by $D_n, D'_n, \ldots$) is a statement context with a hole. We write $\{-\}$ for a hole, and the results of filling a term M in an evaluation context E_n and a statement context D_n are written $E_n\{M\}$ and $D_n\{M\}$, respectively.

Definition (CBN Term and Statement Contexts)

$$E_n, E'_n, E''_n ::= \{-\} \mid E_n M \mid \mathrm{fst}(E_n) \mid \mathrm{snd}(E_n) \mid \delta(E_n, x.E'_n\{x\}, y.E''_n\{y\})$$

$$D_n, D'_n ::= [\alpha]E_n \mid E_n M \mid \delta(E_n, x.D_n\{x\}, y.D'_n\{y\})$$

In the following, we say a term M is a *simple form with respect to* x if there is a call-by-name evaluation term context E such that $M \equiv E\{x\}$ and x is not free in E, and a statement S is a *simple form with respect to* x if there is a call-by-name evaluation statement context D such that $S \equiv D\{x\}$ and x is not free in D.

The one-step *call-by-name reduction* relation for the $\lambda\mu$-calculus, denoted by $\longrightarrow_n$, is defined as the compatible closure of the rules in **Fig. 2**. We write ${\longrightarrow_n}^*$ for the reflexive transitive closure of $\longrightarrow_n$. Similarly, we write ${\longrightarrow_n}^+$ and $=_n$ for the transitive closure and the reflexive symmetric transitive closure of $\longrightarrow_n$ respectively.

In the following, when expressions X and Y are in a relation $\mathcal{R}$ of system S, we write $S \vdash X\,\mathcal{R}\,Y$. For example, we write $\lambda\mu \vdash M \longrightarrow_n N$ if a term M of the $\lambda\mu$-calculus reduces a term N by the one-step call-by-name reduction of the $\lambda\mu$-calculus.

(β)-rules reduce a deconstructor applied to a constructor. Note that $(\beta_{\vee})$-rule has an unusual and a restricted form using the call-by-name evaluation and statement contexts. This restriction is needed to obtain sums equivalent to ones formulated in Wadler's call-by-name system, (ζ)-rules substitute an evaluation context and a statement context for a covalue, and (η_{μ})-rule introduces a μ-abstraction applied to a covariable application. (π)-rules correspond to the permutative conversions, and (ν)-rules expand a case statement $\delta(M, x.S, y.T)$ when S or T is not a simple form, and introduce new bindings. These rules are also needed to obtain sums equivalent to sums formulated in Wadler's system.

We write the $\lambda\mu$-calculus given by Wadler by $\lambda\mu^{wad}$ and write the call-by-name and call-by-value variants of $\lambda\mu^{wad}$ by $\lambda\mu^{wad}_n$ and $\lambda\mu^{wad}_v$ respectively. Detailed definitions of these systems can be found in Wadler (2005). We compare our call-by-name system with the $\lambda\mu^{wad}_n$-calculus. The differences between them are summarized in the following three points:

- Our system is based on reduction relations while $\lambda\mu^{wad}_n$ is based on equations,
- formulates sums differently, and
- does not have (η)-rules related to implications, negations, pairs, and sums while his system does have them.

We give two translations, $[\![-]\!]$ and $\langle\!\langle - \rangle\!\rangle$, between our $\lambda\mu$ and $\lambda\mu^{wad}$ that interpret sums as follows.

$$[\![\mu(\alpha,\beta).S]\!] \equiv \mu\gamma.[\![S]\!][^{[\gamma]\mathrm{inl}\{-\}}/_{[\alpha]\{-\}}, {}^{[\gamma]\mathrm{inr}\{-\}}/_{[\beta]\{-\}}]$$

$$[\![[\alpha,\beta]M]\!] \equiv \delta([\![M]\!], x.[\alpha]x, y.[\beta]y)$$

The other clauses are defined homomorphically like $[\![MN]\!] \equiv [\![M]\!][\![N]\!]$.

$\langle\!\langle \mathrm{inl}(M) \rangle\!\rangle \equiv \mu(\alpha,\beta).[\alpha]\langle\!\langle M \rangle\!\rangle$

$\langle\!\langle \mathrm{inr}(N) \rangle\!\rangle \equiv \mu(\alpha,\beta).[\beta]\langle\!\langle N \rangle\!\rangle$

$\langle\!\langle \delta(O, x.M, y.N) \rangle\!\rangle$
$\quad \equiv \mu\gamma.\big(\lambda y.[\gamma]\langle\!\langle N \rangle\!\rangle\big)\big(\mu\beta.(\lambda x.[\gamma]\langle\!\langle M \rangle\!\rangle)$
$\quad\quad (\mu\alpha.[\alpha,\beta]\langle\!\langle O \rangle\!\rangle)\big)$

$\langle\!\langle \delta(O, x.S, y.T) \rangle\!\rangle$
$\quad \equiv \big(\lambda y.\langle\!\langle T \rangle\!\rangle\big)\big(\mu\beta.(\lambda x.\langle\!\langle S \rangle\!\rangle)(\mu\alpha.[\alpha,\beta]\langle\!\langle O \rangle\!\rangle)\big)$

The other clauses are defined homomorphically like $\langle\!\langle MN \rangle\!\rangle \equiv \langle\!\langle M \rangle\!\rangle\langle\!\langle N \rangle\!\rangle$.

We define the call-by-name system $\lambda\mu_n^\eta$ as the one generated by the rules in Fig. 2 and the following (η)-rules,

$(\eta_\supset)$ $M \longrightarrow_n \lambda x.Mx$ (where $M : A \supset B$)

$(\eta_\&)$ $M \longrightarrow_n \langle \mathrm{fst}(M), \mathrm{snd}(M) \rangle$ (where $M : A\&B$)

$(\eta_\vee)$ $M \longrightarrow_n \delta(M, x.\mathrm{inl}(x), y.\mathrm{inr}(y))$ (where $M : A \vee B$)

$(\eta_\neg)$ $M \longrightarrow_n \lambda x.Mx$ (where $M : \neg A$)

$\lambda\mu_n^\eta$ is equivalent to Wadler's call-by-name system $\lambda\mu_n^{wad}$ as an equational system. To show this, we need some preparations. Let E_n and D_n be the call-by-name evaluation contexts of the $\lambda\mu_n^\eta$-calculus. Then, we define the call-by-name contexts $\overline{E_n}$ and $\overline{D_n}$ of the $\lambda\mu_n^{wad}$-calculus as follows.

$\overline{\{-\}} \equiv \{-\}$ $\quad$ $\overline{E_n N} \equiv \overline{E_n}\langle\!\langle N \rangle\!\rangle$

$\overline{\mathrm{fst}(E_n)} \equiv \mathrm{fst}(\overline{E_n})$ $\quad$ $\overline{\mathrm{snd}(E_n)} \equiv \mathrm{snd}(\overline{E_n})$

$\overline{\delta(E_n, x.E_n'\{x\}, y.E_n''\{y\})}$
$\quad \equiv \mu\gamma.[\gamma]\overline{E_n''}\{\mu\beta.[\gamma]\overline{E_n'}\{\mu\alpha.[\alpha,\beta]\overline{E_n}\}\}$

$\overline{[\alpha]E_n} \equiv [\alpha]\overline{E_n}$ $\quad$ $\overline{E_n N} \equiv \overline{E_n}\langle\!\langle N \rangle\!\rangle$

$\overline{\delta(E_n, x.D_n\{x\}, y.D_n'\{y\})}$
$\quad \equiv \overline{D_n'}\{\mu\beta.\overline{D_n}\{\mu\alpha.[\alpha,\beta]\overline{E_n}\}\}$

For the call-by-name contexts $\overline{E_n}$ and $\overline{D_n}$ defined above, the following lemma holds.

Lemma 1

(1) $\lambda\mu_n^{wad} \vdash \langle\!\langle E_n\{M\} \rangle\!\rangle =_n \overline{E_n}\{\langle\!\langle M \rangle\!\rangle\}$ and $\lambda\mu_n^{wad} \vdash \langle\!\langle D_n\{M\} \rangle\!\rangle =_n \overline{D_n}\{\langle\!\langle M \rangle\!\rangle\}$ hold for any term M of our $\lambda\mu$-calculus.

(2) $\lambda\mu_n^{wad} \vdash \langle\!\langle M \rangle\!\rangle[\overline{D_n}\{-\}/[\alpha]\{-\}] =_n \langle\!\langle M[D_n\{-\}/[\alpha]\{-\}] \rangle\!\rangle$ and $\lambda\mu_n^{wad} \vdash \langle\!\langle S \rangle\!\rangle[\overline{D_n}\{-\}/[\alpha]\{-\}] =_n \langle\!\langle S[D_n\{-\}/[\alpha]\{-\}] \rangle\!\rangle$.

(3) $\lambda\mu_n^{wad} \vdash \overline{E_n}\{\mu\alpha.S\} =_n \mu\beta.S[[\beta]\overline{E_n}\{-\}/[\alpha]\{-\}]$ and $\lambda\mu_n^{wad} \vdash \overline{D_n}\{\mu\alpha.S\} =_n S[\overline{D_n}\{-\}/[\alpha]\{-\}]$.

Proof. See Appendix. □

Proposition 2

(1) If $\lambda\mu_n^\eta \vdash M =_n N$, then $\lambda\mu_n^{wad} \vdash \langle\!\langle M \rangle\!\rangle =_n \langle\!\langle N \rangle\!\rangle$, and if $\lambda\mu_n^\eta \vdash S =_n T$, then $\lambda\mu_n^{wad} \vdash \langle\!\langle S \rangle\!\rangle =_n \langle\!\langle T \rangle\!\rangle$.

(2) If $\lambda\mu_n^{wad} \vdash M =_n N$, then $\lambda\mu_n^\eta \vdash [\![M]\!] =_n [\![N]\!]$, and if $\lambda\mu_n^{wad} \vdash S =_n T$, then $\lambda\mu_n^\eta \vdash [\![S]\!] =_n [\![T]\!]$.

(3) $\lambda\mu_n^\eta \vdash [\![\langle\!\langle M \rangle\!\rangle]\!] =_n M$ and $\lambda\mu_n^\eta \vdash [\![\langle\!\langle S \rangle\!\rangle]\!] =_n S$.

(4) $\lambda\mu_n^{wad} \vdash \langle\!\langle [\![M]\!] \rangle\!\rangle =_n M$ and $\lambda\mu_n^{wad} \vdash \langle\!\langle [\![S]\!] \rangle\!\rangle =_n S$.

Proof. See Appendix. □

2.2 The Call-by-value $\boldsymbol{\lambda\mu}$-calculus

For the call-by-value calculus, we need a notion of values. A *values* (denoted by V, W,…) is a variable, a λ-abstraction, a pair of values, or an injection of a value.

Definition (Values of the Call-by-value $\boldsymbol{\lambda\mu}$)

$V, W ::= x \mid \lambda x.M \mid \langle V, W \rangle \mid \mathrm{inl}(V) \mid \mathrm{inr}(W)$
$\quad\quad \mid \lambda x.S$

We also use a notion of the call-by-value evaluation term and statement contexts to introduce the call-by-value calculus. However, in this case, it is useful to give the evaluation contexts as singular contexts. The call-by-value evaluation singular contexts (denoted by $E_v, E_v', \ldots$) are grouped into the *elimination contexts* (denoted by $E_e, E_e', \ldots$), which are obtained from an elimination rule. The *introduction contexts* (denoted by $E_i, E_i', \ldots$), which are constructed by an introduction rule, and the contexts which have a hole as the argument of a lambda abstraction, i.e., $(\lambda x.M)\{-\}$. The call-by-value evaluation singular statement contexts (denoted by $D_v, D_v', \ldots$) are grouped into the *elimination contexts* (denoted by $D_e, D_e', \ldots$) and the contexts which have a hole as the argument of a lambda abstraction, i.e., $(\lambda x.S)\{-\}$.

Definition (CBV Evaluation Singular Contexts)

$E_v ::= (\lambda x.M)\{-\} \mid E_e \mid E_i$
$E_e ::= \{-\}M \mid \mathrm{fst}(\{-\}) \mid \mathrm{snd}(\{-\})$
$\quad\quad \mid \delta(\{-\}, x.M, y.N)$
$E_i ::= \mathrm{inl}(\{-\}) \mid \mathrm{inr}(\{-\}) \mid \langle \{-\}, M \rangle$
$\quad\quad \mid \langle V, \{-\} \rangle$
$D_v ::= (\lambda x.S)\{-\} \mid D_e$
$D_e ::= [\alpha]\{-\} \mid \{-\}M \mid \delta(\{-\}, x.S, y.T)$

The one-step *call-by-value reduction* relation for the $\lambda\mu$-calculus, denoted by $\longrightarrow_v$, is defined as the compatible closure of the rules in

$(\beta_\supset)$	$(\lambda x.M)V \longrightarrow_v M[^V/_x]$	
$(\beta_\&)$	$\mathrm{fst}\langle V, W\rangle \longrightarrow_v V$	
	$\mathrm{snd}\langle V, W\rangle \longrightarrow_v W$	
$(\beta_\vee)$	$\delta(\mathrm{inl}(V), x.M, y.N) \longrightarrow_v M[^V/_x]$	
	$\delta(\mathrm{inr}(V), x.M, y.N) \longrightarrow_v N[^V/_x]$	
	$\delta(\mathrm{inl}(V), x.S, y.T) \longrightarrow_v S[^V/_x]$	
	$\delta(\mathrm{inr}(V), x.S, y.T) \longrightarrow_v T[^V/_x]$	
$(\beta_\neg)$	$(\lambda x.S)V \longrightarrow_v S[^V/_x]$	
(ζ)	$E_{e\lambda}\{\mu\alpha.S\} \longrightarrow_v \mu\beta.S[^{[\beta]E_{e\lambda}\{-\}}/_{[\alpha]\{-\}}]$	(where $E_{e\lambda}$ is E_e or $(\lambda x.M)\{-\}$)
	$D_v\{\mu\alpha.S\} \longrightarrow_v S[^{D_v\{-\}}/_{[\alpha]\{-\}}]$	
$(comp)$	$E_{e\lambda}\{(\lambda x.M)N\} \longrightarrow_v (\lambda x.E_{e\lambda}\{M\})N$	
	$D_v\{(\lambda x.M)N\} \longrightarrow_v (\lambda x.D_v\{M\})N$	
(π)	$E_{e\lambda}\{\delta(O, x.M, y.N)\} \longrightarrow_v \delta(O, x.E_{e\lambda}\{M\}, y.E_{e\lambda}\{N\})$	
	$D_v\{\delta(O, x.M, y.N)\} \longrightarrow_v \delta(O, x.D_v\{M\}, y.D_v\{N\})$	
(η_μ)	$M \longrightarrow_v \mu\alpha.[\alpha]M$	(where $\alpha \notin \mathrm{FCV}(M)$)
$(name)$	$E_{ie}\{O\} \longrightarrow_v (\lambda x.E_{ie}\{x\})O$	(where O is not a value, E_{ie} is E_i or E_e)
	$D_e\{O\} \longrightarrow_v (\lambda x.D_e\{x\})O$	(where O is not a value)

Fig. 3 Reduction rules of the call-by-value $\lambda\mu$-calculus.

Fig. 3. We write $\longrightarrow_v{}^*$ and $\longrightarrow_v{}^+$ for the reflexive transitive closure and the transitive closure of $\longrightarrow_v$ respectively. (β)-rules reduce the deconstructor applied to a constructor with call-by-value restrictions, (ζ)-rules substitute a call-by-value evaluation context and a statement context for a covalue, (η_μ)-rule introduces a μ-abstraction applied to a covariable application, and (π)-rules correspond to the permutative conversions. The $(name)$-rules push the next term to be evaluated out as an argument of the function. These rules correspond to the $(name)$-rule of the $\lambda\mu_v^{wad}$-calculus. The $(comp)$-rules are associativity rules, which correspond to the $(comp)$-rule of the $\lambda\mu_v^{wad}$-calculus.

We now compare our call-by-value system with Wadler's call-by-value system. The differences between them are summarized in the following four points:

- Our system is based on reduction relations while his system is based on equations,
- formulated sums differently,
- defines values differently: a projection from a value is not a value in our system, and
- does not have (η)-rules related to implications, negations, pairs, and sums while his system does have them.

We introduce the call-by-value calculus $\lambda\mu_v^\eta$ as the system generated by the rules in Fig. 3 and the following (η)-rules.

$(\eta_\supset)$ $V \longrightarrow_n \lambda x.Vx$ $(V : A \supset B)$

$(\eta_\&)$ $V \longrightarrow_n \langle\mathrm{fst}(V), \mathrm{snd}(V)\rangle$ $(V : A\&B)$

$(\eta_\vee)$ $M \longrightarrow_n \delta(M, x.\mathrm{inl}(x), y.\mathrm{inr}(y))$ $(M : A \vee B)$

$(\eta_\neg)$ $V \longrightarrow_n \lambda x.Vx$ $(V : \neg A)$

We define the $\lambda\mu_v^{wad-}$-calculus as the restricted system of the $\lambda\mu_v^{wad}$-calculus obtained by excluding a projection from a value from definition of values. We again consider translations $\langle\!\langle - \rangle\!\rangle$ and $[\![-]\!]$ given in the previous subsection. Our call-by-value system $\lambda\mu_v^\eta$ with (η)-rules is equivalent to the $\lambda\mu_v^{wad-}$-calculus as the equational systems.

Proposition 3
(1) If $\lambda\mu_v^\eta \vdash M =_v N$, then $\lambda\mu_v^{wad-} \vdash \langle\!\langle M\rangle\!\rangle =_v \langle\!\langle N\rangle\!\rangle$, and if $\lambda\mu_v^\eta \vdash S =_n T$, then $\lambda\mu_v^{wad-} \vdash \langle\!\langle S\rangle\!\rangle =_v \langle\!\langle T\rangle\!\rangle$.
(2) If $\lambda\mu_v^{wad-} \vdash M =_v N$, then $\lambda\mu_v^\eta \vdash [\![M]\!] =_v [\![N]\!]$, and if $\lambda\mu_v^{wad-} \vdash S =_v T$, then $\lambda\mu_v^\eta \vdash [\![S]\!] =_v [\![T]\!]$.
(3) $\lambda\mu_v^\eta \vdash [\![\langle\!\langle M\rangle\!\rangle]\!] =_v M$ and $\lambda\mu_v^\eta \vdash [\![\langle\!\langle S\rangle\!\rangle]\!] =_v S$.
(4) $\lambda\mu_v^{wad-} \vdash \langle\!\langle[\![M]\!]\rangle\!\rangle =_v M$ and $\lambda\mu_v^{wad-} \vdash \langle\!\langle[\![S]\!]\rangle\!\rangle =_v S$.

Proof. See Appendix. □

We mention some basic properties of the $\lambda\mu$-calculus at the end of this section.

Lemma 4 (Substitution Lemma for $\lambda\mu$)
Let M and N be terms, and S and T be statements of the $\lambda\mu$-calculus.
(1) Suppose $\Gamma \vdash_{\lambda\mu} \Delta \mid M : A$. If $\Gamma, x : A \vdash_{\lambda\mu}$

$\Delta \mid N : B$, then $\Gamma \vdash_{\lambda\mu} \Delta \mid N[^M/_x] : B$, and if $\Gamma, x : A \mid S \vdash_{\lambda\mu} \Delta$, then $\Gamma \mid S[^M/_x] \vdash_{\lambda\mu} \Delta$.

(2) Suppose $\Gamma, x : A \mid \mathcal{T}\{x\} \vdash_{\lambda\mu} \Delta$. If $\Gamma \vdash_{\lambda\mu} \Delta, \alpha : A \mid N : B$, then $\Gamma \vdash_{\lambda\mu} \Delta \mid N[^{\mathcal{T}\{-\}}/_{[\alpha]\{-\}}] : B$, and if $\Gamma, \mid S \vdash_{\lambda\mu} \Delta, \alpha : A$, then $\Gamma \mid S[^{\mathcal{T}\{-\}}/_{[\alpha]\{-\}}] \vdash_{\lambda\mu} \Delta$

where $\mathcal{T}\{-\}$ is a statement context, that is, a statement with a single hole.

Proof. (1) is shown by a straightforward induction on N and S. (2) can be shown by an induction on M and S using (1). The key case of (2) is $S \equiv [\alpha]M$. Suppose $\Gamma \mid [\alpha]M \vdash_{\lambda\mu} \Delta, \alpha : A$ is derived. Since the last rule to obtain this sequent is (*pass*), we obtain $\Gamma \vdash_{\lambda\mu} \Delta, \alpha : A \mid M : A$. Hence we have $\Gamma \vdash_{\lambda\mu} \Delta \mid M[^{\mathcal{T}\{-\}}/_{[\alpha]\{-\}}] : A$ by the induction hypothesis, and then $\Gamma \mid \mathcal{T}\{M[^{\mathcal{T}\{-\}}/_{[\alpha]\{-\}}]\} \vdash_{\lambda\mu} \Delta$ by (1). This means $\Gamma \mid ([\alpha]M)[^{\mathcal{T}\{-\}}/_{[\alpha]\{-\}}] \vdash_{\lambda\mu} \Delta$. □

Proposition 5 (Subject Reduction for $\boldsymbol{\lambda\mu}$)

[Subject Reduction for $\boldsymbol{\lambda\mu}$] Let M and N be terms, and S and T be statements of the $\lambda\mu$-calculus.

(1) If $\Gamma \vdash_{\lambda\mu} \Delta \mid M : A$ and $\lambda\mu \vdash M \longrightarrow_n N$, then $\Gamma \vdash_{\lambda\mu} \Delta \mid N : A$. If $\Gamma \mid S \vdash_{\lambda\mu} \Delta$ and $\lambda\mu \vdash S \longrightarrow_n T$, then $\Gamma \mid T \vdash_{\lambda\mu} \Delta$.

(2) If $\Gamma \vdash_{\lambda\mu} \Delta \mid M : A$ and $\lambda\mu \vdash M \longrightarrow_v N$, then $\Gamma \vdash_{\lambda\mu} \Delta \mid N : A$. If $\Gamma \mid S \vdash_{\lambda\mu} \Delta$ and $\lambda\mu \vdash S \longrightarrow_v T$, then $\Gamma \mid T \vdash_{\lambda\mu} \Delta$.

Proof. Using the substitution lemma, (1) and (2) are shown by an induction on $\longrightarrow_n$ and $\longrightarrow_v$ respectively. □

The call-by-name and call-by-value $\lambda\mu$-calculi given in this paper are confluent. This is proved as a corollary of the results in Section 4 and 5 (see Proposition 37).

3. The Dual Calculus

The dual calculus was proposed by Wadler [10),11)] as a term calculus that corresponds to the classical sequent calculus LK. Wadler [10)] first gave the dual calculus as a reduction system, and introduced it as an equation system in his later paper [11)]. Since we want to consider the system based on reduction relations, we will give the reduction system of the dual calculus referring to the system in his first paper.

Types, variables, and covariables of the dual calculus are the same as those of the $\lambda\mu$-calculus.

Definition (Types of the Dual Calculus)

$A, B ::= X \mid A\&B \mid A \vee B \mid A \supset B \mid \neg A$

where X is an atomic type.

The expressions of the dual calculus consist of *terms* (denoted by $M, N, \ldots$), *coterms* (denoted by $K, L, \ldots$), and *statements* (denoted by $S, T, \ldots$). A term is either a variable x, a pair $\langle M, N\rangle$, a left injection $\langle M\rangle$inl or a right injection $\langle N\rangle$inr, a complement of a coterm $[K]$not, a function abstraction $\lambda x.M$, with x bound in M, or a covariable abstraction $S.\alpha$ with α bound in S. A coterm is either a covariable α, a case $[K, L]$; a projection from the left of a product fst$[K]$ or a projection from the right of a product snd$[L]$, a complement of a term not$\langle M\rangle$, a function application $M@K$, or a variable abstraction $x.S$ with x bound in S. A statement is a cut of a term against a coterm $M \bullet K$.

Definition (Expressions of the Dual Calculus)

$$\begin{aligned} M,\, N &::= x \mid \langle M, N\rangle \mid \langle M\rangle\mathrm{inl} \mid \langle M\rangle\mathrm{inr} \mid [K]\mathrm{not} \\ &\quad\mid \lambda x.M \mid S.\alpha \qquad (\textit{terms}) \\ K,\, L &::= \alpha \mid [K, L] \mid \mathrm{fst}[K] \mid \mathrm{snd}[L] \mid \mathrm{not}\langle M\rangle \\ &\quad\mid M@K \mid x.S \qquad (\textit{coterms}) \\ S,\, T &::= M \bullet K \qquad (\textit{statements}) \end{aligned}$$

The set of free variables and covariables occurring in M, K, and S are denoted by FV(M), FV(K), and FV(S) respectively. We identify the two expressions in the α-equivalence relation and will use $\equiv$ for the syntactic identity on the expressions. The expressions $M[^N/_x]$, $K[^N/_x]$, and $S[^N/_x]$ denote the expressions obtained by substituting N for every free occurrence of the variable x in M, K, and S. The expressions $M[^L/_\alpha]$, $K[^L/_\alpha]$ and $S[^L/_\alpha]$ are similarly defined.

A *context* of the dual calculus (denoted by Γ, Σ, ...) is a finite set of term variables annotated with types (denoted by $x_1 : A_1, \ldots x_m : A_m$), in which each variable occurs once at the most. Similarly, a *cocontext* of the dual calculus (denoted by Δ, Λ, ...) is defined as a finite set of covariables with types (denoted by $\alpha_1 : B_1, \ldots \alpha_m : B_m$). A *typing judgment* of the dual calculus takes either the form $\Gamma \vdash_{dc} \Delta \mid M : A$, the form $K : A \mid \Gamma \vdash_{dc} \Delta$, or the form $\Gamma \mid S \vdash_{dc} \Delta$. We note that $\vdash_{dc}$ is sometimes written as $\vdash$. The typing rules of the dual calculus are shown in **Fig. 4**. These rules are the same as those in Wadler's later paper [11)].

A *value* of the dual calculus, denoted by V, W ..., is a variable x, a pair of values $\langle V, W\rangle$, an

$$\frac{}{\Gamma, x : A \vdash_{dc} \Delta \mid x : A}\ \mathrm{AxR} \qquad \frac{}{\alpha : A \mid \Gamma \vdash_{dc} \Delta, \alpha : A}\ \mathrm{AxL}$$

$$\frac{\Gamma \vdash_{dc} \Delta \mid M : A \quad K : A \mid \Gamma \vdash_{dc} \Delta}{\Gamma \mid M \bullet K \vdash_{dc} \Delta}\ \mathrm{Cut}$$

$$\frac{\Gamma \vdash_{dc} \Delta \mid M : A \quad \Gamma \vdash_{dc} \Delta \mid N : B}{\Gamma \vdash_{dc} \Delta \mid \langle M, N\rangle : A \& B}\ \&R$$

$$\frac{K : A \mid \Gamma \vdash_{dc} \Delta}{\mathrm{fst}[K] : A\&B \mid \Gamma \vdash_{dc} \Delta}\ \&L \qquad \frac{L : B \mid \Gamma \vdash_{dc} \Delta}{\mathrm{snd}[L] : A\&B \mid \Gamma \vdash_{dc} \Delta}\ \&L$$

$$\frac{\Gamma \vdash_{dc} \Delta \mid M : A}{\Gamma \vdash_{dc} \Delta \mid \langle M\rangle\mathrm{inl} : A \vee B}\ \vee R \qquad \frac{\Gamma \vdash_{dc} \Delta \mid N : B}{\Gamma \vdash_{dc} \Delta \mid \langle N\rangle\mathrm{inr} : A \vee B}\ \vee R$$

$$\frac{K : A \mid \Gamma \vdash_{dc} \Delta \quad L : B \mid \Gamma \vdash_{dc} \Delta}{[K, L] : A \vee B \mid \Gamma \vdash_{dc} \Delta}\ \vee L$$

$$\frac{\Gamma \vdash_{dc} \Delta \mid M : A}{\mathrm{not}\langle M\rangle : \neg A \mid \Gamma \vdash_{dc} \Delta}\ \neg L \qquad \frac{K : A \mid \Gamma \vdash_{dc} \Delta}{\Gamma \vdash_{dc} \Delta \mid [K]\mathrm{not} : \neg A}\ \neg R$$

$$\frac{\Gamma \vdash_{dc} \Delta \mid M : A \quad K : B \mid \Gamma \vdash_{dc} \Delta}{M@K : A \supset B \mid \Gamma \vdash_{dc} \Delta}\ \supset L \qquad \frac{\Gamma, x : A \vdash_{dc} \Delta \mid M : B}{\Gamma \vdash_{dc} \Delta \mid \lambda x.M : A \supset B}\ \supset R$$

$$\frac{x : A, \Gamma \mid S \vdash_{dc} \Delta}{x.S : A \mid \Gamma \vdash_{dc} \Delta}\ \mathrm{LI} \qquad \frac{\Gamma \mid S \vdash_{dc} \Delta, \alpha : A}{\Gamma \vdash_{dc} \Delta \mid S.\alpha : A}\ \mathrm{RI}$$

Fig. 4 Typing rules of the dual calculus.

injection of a value $\langle V\rangle$inl or $\langle W\rangle$inr, a complement of a coterm $[K]$not, or a function $\lambda x.M$. A *covalue* of the dual calculus is denoted by P, Q .... A covalue is a covariable α, a case over a pair of covalues $[K, L]$, a projection of a covalue fst$[P]$ or snd$[Q]$, a complement of a term not$\langle M\rangle$, or a function application over a covalue $M@Q$.

Definition (Values and Covalues)

$V, W ::= x \mid \langle V, W\rangle \mid \langle V\rangle\mathrm{inl} \mid \langle W\rangle\mathrm{inr} \mid [K]\mathrm{not} \mid \lambda x.M$

$P, Q ::= \alpha \mid [P, Q] \mid \mathrm{fst}[P] \mid \mathrm{snd}[Q] \mid \mathrm{not}\langle M\rangle \mid M@Q$

These definitions of the values and covalues are same as those in Wadler (2003) but different from those in Wadler (2005). Note that if we adopt the definitions in Wadler (2005), then terms containing beta redexes at the top level may also be values. For example, a term $(\langle x, y\rangle \bullet \mathrm{fst}[\alpha]).\alpha$ includes a beta redex at the top level even though it is a value according to the definition in Wadler (2005).

A *term context* for the dual calculus (denoted by E) is a term that contains a hole that accepts a term, and a *coterm context* (denoted by F) is a coterm that contains a hole that accepts a coterm. The hole is written $\{-\}$, and the result of filling the hole in the term context E with a term M is written $E\{M\}$. Similarly, the result of filling the hole in the coterm context F with a coterm K is written $F\{K\}$.

Definition (Term Contexts and Coterm Contexts)

$E ::= \langle\{-\}, N\rangle \mid \langle V, \{-\}\rangle \mid \langle\{-\}\rangle\mathrm{inl} \mid \langle\{-\}\rangle\mathrm{inr}$

$F ::= [K, \{-\}] \mid [\{-\}, P] \mid \mathrm{fst}[\{-\}] \mid \mathrm{snd}[\{-\}] \mid M@\{-\}$

Note that the context of the form of $M@\{-\}$ is defined as a coterm context in this paper, though it was not defined in Wadler (2003). This means the reduction rule

$N \bullet (M@K) \longrightarrow^n (N \bullet (M@\alpha)).\alpha \bullet K$

is permitted as (*name*)-rule in our call-by-name calculus. This seems to have added a new rule to Wadler's original system. However, this rule is not an essentially new rule, because this rule is justified when implication is defined in terms of conjunction, disjunction and negation (see Proposition 7).

The *call-by-name reduction relation* $\longrightarrow^n$ and the *call-by-value reduction relation* $\longrightarrow^v$ of the dual calculus are defined to be the compatible closure of rules in **Fig. 5**. In the sequel, we use $\longrightarrow^{n*}$, $\longrightarrow^{n+}$, and $=^n$ as the reflexive transitive closure, the transitive closure, and the symmetric reflexive transitive closure of $\longrightarrow^n$ respectively. $\longrightarrow^{v*}$, $\longrightarrow^{v+}$, and $=^v$ are defined similarly.

Some of our reduction rules are slightly different from those in Wadler (2003), but the differences are not essential. $(\beta_\supset)$-rules given

	Call-by-name reduction	Call-by-value reduction
$(\beta_{\&})$	$\langle M,N\rangle \bullet \mathrm{fst}[P] \longrightarrow^n M\bullet P$	$\langle V,W\rangle \bullet \mathrm{fst}[K] \longrightarrow^v V\bullet K$
	$\langle M,N\rangle \bullet \mathrm{snd}[Q] \longrightarrow^n N\bullet Q$	$\langle V,W\rangle \bullet \mathrm{snd}[L] \longrightarrow^v W\bullet L$
$(\beta_{\vee})$	$\langle M\rangle\mathrm{inl} \bullet [P,Q] \longrightarrow^n M\bullet P$	$\langle V\rangle\mathrm{inl} \bullet [K,L] \longrightarrow^v V\bullet K$
	$\langle N\rangle\mathrm{inr} \bullet [P,Q] \longrightarrow^n N\bullet Q$	$\langle W\rangle\mathrm{inr} \bullet [K,L] \longrightarrow^v W\bullet L$
$(\beta_{\neg})$	$[K]\mathrm{not} \bullet \mathrm{not}\langle M\rangle \longrightarrow^n M\bullet K$	$[K]\mathrm{not} \bullet \mathrm{not}\langle M\rangle \longrightarrow^v M\bullet K$
$(\beta_{\supset})$	$\lambda x.M \bullet (N@P) \longrightarrow^n M[^N/_x]\bullet P$	$\lambda x.M \bullet (N@K) \longrightarrow^v N\bullet x.(M\bullet K)$
(β_L)	$M\bullet x.S \longrightarrow^n S[^M/_x]$	$V\bullet x.S \longrightarrow^v S[^V/_x]$
(β_R)	$S.\alpha\bullet P \longrightarrow^n S[^P/_\alpha]$	$S.\alpha\bullet K \longrightarrow^v S[^K/_\alpha]$
(η_L)	$K \longrightarrow^n x.(x\bullet K)$	$K \longrightarrow^v x.(x\bullet K)$
(η_R)	$M \longrightarrow^n (M\bullet\alpha).\alpha$	$M \longrightarrow^v (M\bullet\alpha).\alpha$
$(name)$	$M\bullet F\{K\} \longrightarrow^n (M\bullet F\{\alpha\}).\alpha\bullet K$	$E\{M\}\bullet K \longrightarrow^v M\bullet x.(E\{x\}\bullet K)$

Fig. 5 Reduction rules of the call-by-value and call-by-name dual calculus.

here are justified in Proposition 6. $(name)$-rules correspond to (ς)-rules of the dual calculus in Wadler (2003), though these rules are not included in his system. Indeed, we can easily show $(name)$-rules from (ς)-rules using (β_L) and (β_R)-rules in both the call-by-name and call-by-value systems. Conversely, we can obtain (ς)-rules from $(name)$-rules using (η_L) and (η_R)-rules.

When a term M of the dual calculus reduces a term N by the one-step call-by-name reduction, we write $DC \vdash M \longrightarrow^n N$. We also write $DC \vdash K \longrightarrow^n L$, $DC \vdash S \longrightarrow^n T$ etc. For call-by-value calculus, we also define these notations similarly.

As in Wadler's original dual calculus, implication can be defined in terms of the other connectives, i.e., the following propositions hold.

Proposition 6
Under call-by-value, an implication can be defined by

$A \supset B \equiv \neg(A\&\neg B)$
$\lambda x.M \equiv [z.(z\bullet \mathrm{fst}[x.(z\bullet \mathrm{snd}[\mathrm{not}\langle M\rangle])])]\mathrm{not}$
$N@K \equiv \mathrm{not}\langle\ \langle N,[K]\mathrm{not}\rangle\ \rangle.$

The translation of a function abstraction is a value, and the typing and reduction rules for implication can be derived from the typing rules for the other connectives.

Proof. The call-by-value $(\beta_\supset)$-rule is validated as follows.

(a) If N is a value V, then

$$
\begin{aligned}
&(\lambda x.M)\bullet(V@K)\\
&\quad\equiv [z.(z\bullet \mathrm{fst}[x.(z\bullet \mathrm{snd}[\mathrm{not}\langle M\rangle])])]\mathrm{not}\\
&\qquad \bullet\mathrm{not}\langle\ \langle V,[K]\mathrm{not}\rangle\ \rangle\\
&\quad\longrightarrow^v_{(\beta_\neg)} \langle V,[K]\mathrm{not}\rangle\bullet z.(z\bullet\mathrm{fst}[x.(z\\
&\qquad \bullet\mathrm{snd}[\mathrm{not}\langle M\rangle])])\\
&\quad\longrightarrow^v_{(\beta_L)} \langle V,[K]\mathrm{not}\rangle\bullet\mathrm{fst}[x.(\langle V,[K]\mathrm{not}\rangle\\
&\qquad \bullet\mathrm{snd}[\mathrm{not}\langle M\rangle])]\\
&\quad\longrightarrow^{v*}_{(\beta_\&)} V\bullet x.([K]\mathrm{not}\bullet\mathrm{not}\langle M\rangle)\\
&\quad\longrightarrow^v_{(\beta_\neg)} V\bullet x.(M\bullet K).
\end{aligned}
$$

(b) If N is not a value, we need $(name)$-rule:

$$
\begin{aligned}
&(\lambda x.M)\bullet(N@K)\\
&\quad\equiv [z.(z\bullet \mathrm{fst}[x.(z\bullet \mathrm{snd}[\mathrm{not}\langle M\rangle])])]\mathrm{not}\\
&\qquad \bullet\mathrm{not}\langle\ \langle N,[K]\mathrm{not}\rangle\ \rangle\\
&\quad\longrightarrow^v_{(\beta_\neg)} \langle N,[K]\mathrm{not}\rangle\\
&\qquad \bullet z.(z\bullet\mathrm{fst}[x.(z\bullet\mathrm{snd}[\mathrm{not}\langle M\rangle])])\\
&\quad\longrightarrow^v_{(name)} N\bullet y.(\langle y,[K]\mathrm{not}\rangle\\
&\qquad \bullet z.(z\bullet\mathrm{fst}[x.(z\bullet\mathrm{snd}[\mathrm{not}\langle M\rangle])]))\\
&\quad\longrightarrow^v_{(\beta_L)} N\bullet y.(\langle y,[K]\mathrm{not}\rangle\\
&\qquad \bullet\mathrm{fst}[x.(\langle y,[K]\mathrm{not}\rangle\bullet\mathrm{snd}[\mathrm{not}\langle M\rangle])])\\
&\quad\longrightarrow^v_{(\beta_\&)} N\bullet y.(y\bullet x.([K]\mathrm{not}\bullet\mathrm{not}\langle M\rangle))\\
&\quad\longrightarrow^v_{(\beta_L)} N\bullet x.([K]\mathrm{not}\bullet\mathrm{not}\langle M\rangle)\\
&\quad\longrightarrow^v_{(\beta_\neg)} N\bullet x.(M\bullet K).
\end{aligned}
$$
□

Proposition 7
Under call-by-name, an implication can be defined by

$A \supset B \equiv \neg A \vee B$
$\lambda x.M \equiv (\langle[x.(\langle M\rangle\mathrm{inr}\bullet\gamma)]\mathrm{not}\rangle\mathrm{inl}\bullet\gamma).\gamma$
$N@K \equiv [\mathrm{not}\langle N\rangle, L].$

The translation of a function application with covalue is a covalue, and the typing and reduction rules for implication can be derived from the typing rules for the other connectives.

Proof. The call-by-name $(\beta_\supset)$ and $(name)$-rules are validated as follows.

$$
\begin{aligned}
&(\lambda x.M)\bullet(N@P)\\
&\quad\equiv (\langle[x.(\langle M\rangle\mathrm{inr}\bullet\gamma)]\mathrm{not}\rangle\mathrm{inl}\bullet\gamma).\gamma\\
&\qquad \bullet[\mathrm{not}\langle N\rangle, P]\\
&\quad\longrightarrow^n_{(\beta_R)} (\langle[x.(\langle M\rangle\mathrm{inr}\\
&\qquad \bullet[\mathrm{not}\langle N\rangle,P])]\mathrm{not}\rangle\mathrm{inl}\bullet[\mathrm{not}\langle N\rangle,P]
\end{aligned}
$$

$\longrightarrow^{n*}_{\beta_\vee} [x.(M \bullet P)]\text{not} \bullet \text{not}\langle N \rangle$
$\longrightarrow^{n}_{(\beta_\neg)} N \bullet x.(M \bullet P)$
$\longrightarrow^{n}_{(\beta_L)} M[^N/_x] \bullet P$
$N \bullet (M@K) \equiv N \bullet [\text{not}\langle M \rangle, K]$
$\longrightarrow^{n}_{(name)} (N \bullet [\text{not}\langle M \rangle, \alpha]).\alpha \bullet K$
$\equiv (N \bullet (M@\alpha)).\alpha \bullet K$ □

We now mention some basic properties of the dual calculus.

Lemma 8 (Substitution Lemma for DC) Let M and N be terms, K and L be coterms, and S and T be statements of the dual calculus.
(1) Suppose $\Gamma \vdash \Delta \mid M : A$. If $\Gamma, x : A \vdash \Delta \mid N : B$, then $\Gamma \vdash \Delta \mid N[^M/_x] : B$. If $L : B \mid \Gamma, x : A \vdash \Delta$, then $L[^M/_x] : B \mid \Gamma \vdash \Delta$. If $\Gamma, x : A \mid S \vdash \Delta$, then $\Gamma \mid S[^M/_x] \vdash \Delta$.
(2) Suppose $K : A \mid \Gamma \vdash \Delta$. If $\Gamma \vdash \Delta, \alpha : A \mid N : B$, then $\Gamma \vdash \Delta \mid N[^K/_\alpha] : B$. If $L : B \mid \Gamma \vdash \Delta, \alpha : A$, then $L[^K/_\alpha] : B \mid \Gamma \vdash \Delta$. If $\Gamma \mid S \vdash \Delta, \alpha : A$, then $\Gamma \mid S[^K/_\alpha] \vdash \Delta$.
Proof. (1) and (2) are shown by a straightforward induction on M, K, and S. □

Proposition 9 (Subject Reduction for DC) Let M and N be terms, K and L be coterms, and S and T be statements of the dual calculus.
(1) If $\Gamma \vdash \Delta \mid M : A$ and $DC \vdash M \longrightarrow^n N$, then $\Gamma \vdash \Delta \mid N : A$. If $K : A \mid \Gamma \vdash \Delta$ and $DC \vdash K \longrightarrow^n L$, then $L : A \mid \Gamma \vdash \Delta$. If $\Gamma \mid S \vdash \Delta$ and $DC \vdash S \longrightarrow^n T$, then $\Gamma \mid T \vdash \Delta$.
(2) If $\Gamma \vdash \Delta \mid M : A$ and $DC \vdash M \longrightarrow^v N$, then $\Gamma \vdash \Delta \mid N : A$, If $K : A \mid \Gamma \vdash \Delta$ and $DC \vdash K \longrightarrow^v L$, then $L : A \mid \Gamma \vdash \Delta$, If $\Gamma \mid S \vdash \Delta$ and $DC \vdash S \longrightarrow^v T$, then $\Gamma \mid T \vdash \Delta$.
Proof. Using the substitution lemma, (1) and (2) are shown by an induction on $\longrightarrow_n$ and $\longrightarrow_v$ respectively. □

As Wadler mentioned in his paper, the reductions of his dual calculus are confluent. Moreover, if (η_L), (η_R), and (ς)-rules are omitted, then the remaining reductions are strongly normalizing for typed terms. Our systems enjoy similar properties. However, since (η_L) and (η_R)-rules are expansions, the full reductions are not strongly normalizing. Moreover, the full reductions of our systems, like Wadler's original systems, include looping terms even if for typed terms. For example, $\langle x, y \rangle \bullet \alpha$ is a typable statement, and this statement loops in the call-by-value calculus.

$\langle x, y \rangle \bullet \alpha \longrightarrow^{v}_{(\eta_R)} \langle (x \bullet \beta).\beta, y \rangle \bullet \alpha$
$\longrightarrow^{v}_{(name)} (x \bullet \beta).\beta \bullet z.(\langle z, y \rangle \bullet \alpha)$
$\longrightarrow^{v}_{(\beta_R)} x \bullet z.(\langle z, y \rangle \bullet \alpha)$
$\longrightarrow^{v}_{(\beta_L)} \langle x, y \rangle \bullet \alpha$

We can give a similar example for the call-by-name calculus.

We now consider the two versions of the dual calculus; one given in Wadler (2003) and the other given in Wadler (2005). For the latter version, we write $DC^{\eta=}_n$ as the call-by-name system and $DC^{\eta=}_v$ as the call-by-value system. The differences between the two versions of the dual calculus are summarized in the following three points:

- The first version is based on reduction relations while the second one is based on equations,
- the first version does not have (η)-rules related to implications, negations, pairs and sums while the second one does contain them, and
- the second version contains terms of the forms $(V \bullet \text{fst}[\alpha]).\alpha$ and $(V \bullet \text{snd}[\beta]).\beta$ as values and coterms of the forms $x.(\langle x \rangle\text{inl} \bullet P)$ and $y.(\langle y \rangle\text{inr} \bullet Q)$ as covalues.

4. Translations from the $\lambda\mu$-calculus into the Dual Calculus

In this section, we give the translations from the $\lambda\mu$-calculus into the dual calculus. We consequently introduce two different translations for the call-by-name and call-by-value calculi, and show that these translations preserve typing and reductions.

4.1 The Naive Translation

In this subsection, we give the naive translation from the $\lambda\mu$-calculus into the dual calculus. This translation preserves equalities, but does not preserves reductions.

Definition (The Naive Translation $(-)^{\circledast}$) The naive translation $(-)^{\circledast}$ from the $\lambda\mu$-calculus into the dual calculus is defined as follows. This translation maps a term M and a statement S of our $\lambda\mu$-calculus to a term $M^{\circledast}$ and a statement $S^{\circledast}$ of the dual calculus respectively.

$(x)^{\circledast} \equiv x$
$(\langle M, N \rangle)^{\circledast} \equiv \langle M^{\circledast}, N^{\circledast} \rangle$
$(\delta(O, x.M, y.N))^{\circledast}$
$\quad \equiv (O^{\circledast} \bullet [x.(M^{\circledast} \bullet \alpha), y.(N^{\circledast} \bullet \alpha)]).\alpha$
$(\delta(O, x.S, y.T))^{\circledast} \equiv O^{\circledast} \bullet [x.S^{\circledast}, y.T^{\circledast}]$
$(\text{fst}(O))^{\circledast} \equiv (O^{\circledast} \bullet \text{fst}[\alpha]).\alpha$
$(\text{inl}(O))^{\circledast} \equiv \langle O^{\circledast} \rangle\text{inl}$

$$
\begin{array}{rll}
(M)^\sharp \equiv (M :_n \alpha).\alpha & & \text{(where } \alpha \text{ is a fresh covariable)}\\
([\alpha]M)^\sharp \equiv M :_n \alpha & & \\
(OM)^\sharp \equiv O :_n \mathrm{not}\langle M^\sharp\rangle & & \text{(where } O \text{ is not a } \lambda\text{-abstraction)}\\
((\lambda x.S)N)^\sharp \equiv N^\sharp \bullet x.S^\sharp & & \\
(\delta(O, x.S, y.T))^\sharp \equiv O^\sharp \bullet [x.S^\sharp, y.T^\sharp] & & \\
x :_n K \equiv x \bullet K & \langle M, N\rangle :_n K \equiv \langle M^\sharp, N^\sharp\rangle \bullet K & \\
\mathrm{fst}(M) :_n K \equiv M :_n \mathrm{fst}[K] & \mathrm{inl}(M) :_n K \equiv \langle M^\sharp\rangle\mathrm{inl} \bullet K & \\
\mathrm{snd}(M) :_n K \equiv M :_n \mathrm{snd}[K] & \mathrm{inr}(M) :_n K \equiv \langle M^\sharp\rangle\mathrm{inr} \bullet K & \\
(\lambda x.S) :_n K \equiv [x.S^\sharp]\mathrm{not} \bullet K & \mu\alpha.S :_n K \equiv S^\sharp.\overline{\alpha} \bullet K & \\
(\lambda x.M) :_n K \equiv (\lambda x.M^\sharp) \bullet K & & \\
(OM) :_n K \equiv O :_n (M^\sharp @ K) & & \text{(where } O \text{ is not a } \lambda\text{-abstraction)}\\
((\lambda x.M)N) :_n K \equiv (N^\sharp \bullet x.(M :_n \alpha)).\overline{\alpha} \bullet K & & \\
\delta(O, x.M, y.N) :_n K \equiv \big(O^\sharp \bullet [x.(M :_n \alpha), y.(N :_n \alpha)]\big).\overline{\alpha} \bullet K & &
\end{array}
$$

Fig. 6 Translation $(-)^\sharp$: CBN $\lambda\mu$-calculus $\longrightarrow$ CBN dual calculus.

$(\mathrm{snd}(O))^\circledast \equiv (O^\circledast \bullet \mathrm{snd}[\alpha]).\alpha$
$(\mathrm{inr}(O))^\circledast \equiv \langle O^\circledast\rangle\mathrm{inr}$
$(OM)^\circledast \equiv O^\circledast \bullet \mathrm{not}\langle M^\circledast\rangle$
$(\lambda x.S)^\circledast \equiv [x.S^\circledast]\mathrm{not}$
$(\mu\alpha.S)^\circledast \equiv S^\circledast.\alpha$
$([\alpha]M)^\circledast \equiv M^\circledast \bullet \alpha$
$(OM)^\circledast \equiv (O^\circledast \bullet (M^\circledast @\alpha)).\alpha$
$(\lambda x.M)^\circledast \equiv \lambda x.M^\circledast$

This naive translation is defined by changing the part of sums of the original translation $(-)^*$ given in Wadler (2005). The naive translation is consistent with Wadler's translation in the sense of the following lemma.

Lemma 10
Let M be a term, and S be a statement of our $\lambda\mu$-calculus, then
(1) $DC^{\eta=} \vdash \langle\!\langle M\rangle\!\rangle^* =^n M^\circledast$ and $DC^{\eta=} \vdash \langle\!\langle S\rangle\!\rangle^* =^n S^\circledast$,
(2) $DC^{\eta=} \vdash \langle\!\langle M\rangle\!\rangle^* =^v M^\circledast$ and $DC^{\eta=} \vdash \langle\!\langle S\rangle\!\rangle^* =^v S^\circledast$.

Proof. See Appendix. □

The naive translation preserves typing rules and equalities.

Proposition 11
(1) If $\Gamma \vdash_{\lambda\mu} \Delta \mid M : A$, then $\Gamma \vdash \Delta \mid M^\circledast : A$. If $\Gamma \mid S \vdash_{\lambda\mu} \Delta$, then $\Gamma \mid S^\circledast \vdash \Delta$.
(2) If $\lambda\mu \vdash M =_n N$, then $DC \vdash M^\circledast =^n N^\circledast$. If $\lambda\mu \vdash S =_n T$, then $DC \vdash S^\circledast =^n T^\circledast$.
(3) If $\lambda\mu \vdash M =_v N$, then $DC \vdash M^\circledast =^v N^\circledast$. If $\lambda\mu \vdash S =_v T$, then $DC \vdash S^\circledast =^v T^\circledast$.

Proof. (1) We can prove this claim by a straight forward induction on $\vdash_{\lambda\mu}$.
(2) This claim can be shown directly by an induction on $=_n$. Even if we do not adopt this approach, we can show this claim as a corollary of Theorem 16 using Lemma 12.
(3) As with (2), we can show this claim directly, or as a corollary of Theorem 21 using Lemma 17. □

In general, this naive translation does not preserve reductions as well as Wadler's translation. This is because of the so-called administrative redexes. A typical example is (ζ)-reduction : $(\mu\alpha.[\beta]\lambda x.[\alpha]x)y \longrightarrow_n \mu\gamma.[\beta]\lambda x.[\gamma](xy)$

$$
\begin{aligned}
&\big((\mu\alpha.[\beta]\lambda x.[\alpha]x)y\big)^\circledast\\
&\quad\equiv \big(([x.(x \bullet \alpha)]\mathrm{not} \bullet \beta).\alpha \bullet (y@\gamma)\big).\gamma\\
&\quad\longrightarrow^n_{(\beta_R)} \big([x.(x \bullet (y@\gamma))]\mathrm{not} \bullet \beta\big).\gamma\\
&\quad\longleftarrow^n_{(\beta_R)} \big([x.((x \bullet (y@\gamma')).\gamma' \bullet \gamma)]\mathrm{not} \bullet \beta\big).\gamma\\
&\quad\equiv \big(\mu\gamma.[\beta]\lambda x.[\gamma](xy)\big)^\circledast
\end{aligned}
$$

To solve this problem, we modify the naive translation. The idea of modification is similar to the modified CPS translation by de Groote [1),2)]. In the following two subsections, we give different translations for the call-by-name and call-by-value calculi. This is because the administrative redexes of these two calculi are slightly different.

4.2 The Translation from CBN $\lambda\mu$-calculus into CBN Dual Calculus

The call-by-name translation consists of the following two translations.

- $(-)^\sharp$ maps any term M and statement S of the $\lambda\mu$-calculus to a term $M^\sharp$ and statement $S^\sharp$ of the dual calculus, respectively.
- $((-) :_n K)$ is a translation given by coterm K and maps any term M of the $\lambda\mu$-calculus to a statement $M :_n K$ of the dual calculus.

The definition of the call-by-name translation $(-)^\sharp$ is given in **Fig. 6**. In this figure, $S.\overline{\alpha} \bullet K$ means $S[{}^K/_\alpha]$ if K is a covalue, otherwise it means $S.\alpha \bullet K$.

This translation is consistent with the naive translation, that is, the following lemma holds.

Lemma 12

Let M and S be a term and a statement of the $\lambda\mu$-calculus, then

(1) $DC \vdash M^{\circledast} \bullet P \longrightarrow^{n*} M :_n P$ for any covalue P,

(2) $DC \vdash M^{\circledast} \longrightarrow^{n*} M^{\sharp}$, and

(3) $DC \vdash S^{\circledast} \longrightarrow^{n*} S^{\sharp}$.

Proof. See Appendix. □

This translation preserves the typing derivation, that is, the following proposition holds.

Proposition 13

(1) If $\Gamma \vdash_{\lambda\mu} \Delta \mid M : A$, then $\Gamma \vdash_{dc} \Delta \mid M^{\sharp} : A$.

(2) If $\Gamma \mid S \vdash_{\lambda\mu} \Delta$, then $\Gamma \mid S^{\sharp} \vdash_{dc} \Delta$.

Proof. This proposition can be shown by the subject reduction property for the dual calculus using Proposition 11 and Lemma 12. □

Lemma 14

(1) Let M and N be terms, and S be a statement of the $\lambda\mu$-calculus, then $DC \vdash (M :_n P)[^{N^{\sharp}}/_x] \longrightarrow^{n*} M[^N/_x] :_n P[^{N^{\sharp}}/_x]$, and $DC \vdash S^{\sharp}[^{N^{\sharp}}/_x] \longrightarrow^{n*} (S[^N/_x])^{\sharp}$ hold for any covalue P.

(2) Let P and Q be covalues, and M be a term of the dual calculus, then $DC \vdash M \bullet [x.(x \bullet P), y.(y \bullet Q)] \longrightarrow^{n*} M \bullet [P, Q]$

(3) If K is not a covalue, then $DC \vdash O :_n K \longrightarrow^{n*} O^{\sharp} \bullet K$ for any term O of the $\lambda\mu$-calculus.

(4) $DC \vdash (M :_n N^{\sharp} @ P) \longrightarrow^{n*} (MN :_n P)$ and $DC \vdash (M :_n \mathrm{not}\langle N^{\sharp}\rangle) \longrightarrow^{n*} (MN :_n P)$ for any terms M and N of the $\lambda\mu$-calculus, and covalue P.

Proof. See Appendix. □

Let E_n and D_n be a call-by-name evaluation term context and statement context of the $\lambda\mu$-calculus, and P be a covalue of the dual calculus, then we define covalues $\Phi(E_n, P)$ and $\Phi(D_n)$ as follows:

$$\Phi(\{-\}, P) \equiv P$$
$$\Phi(E_n N, P) \equiv \Phi(E_n, N^{\sharp} @ P)$$
$$\Phi(\mathrm{fst}(E_n), P) \equiv \Phi(E_n, \mathrm{fst}[P])$$
$$\Phi(\mathrm{snd}(E_n), P) \equiv \Phi(E_n, \mathrm{snd}[P])$$
$$\Phi(\delta(E_n, x.E'_n\{x\}, y.E''_n\{y\}), P) \equiv \Phi(E_n, [\Phi(E'_n, P), \Phi(E''_n, P)])$$
$$\Phi([\alpha]E_n) \equiv \Phi(E_n, \alpha)$$
$$\Phi(E_n N) \equiv \Phi(E_n, \mathrm{not}\langle N^{\sharp}\rangle)$$
$$\Phi(\delta(E_n, x.D_n\{x\}, y.D'_n\{y\}), P) \equiv \Phi(E_n, [\Phi(D_n), \Phi(D'_n)])$$

Then the following properties hold.

Lemma 15

(1) If M is not a λ-abstraction, then $DC \vdash (E_n\{M\} :_n P) \longrightarrow^{n*} (M :_n \Phi(E_n, P))$ and $DC \vdash (D_n\{M\})^{\sharp} \longrightarrow^{n*} (M :_n \Phi(D_n))$ hold.

(2) For any term M of the $\lambda\mu$-calculus, $DC \vdash (M :_n \Phi(E_n, P)) \longrightarrow^{n*} (E_n\{M\} :_n P)$ and $DC \vdash (M :_n \Phi(D_n)) \longrightarrow^{n*} (D_n\{M\})^{\sharp}$ hold.

(3) Let M be a term and S be a statement of the $\lambda\mu$-calculus, then $DC \vdash (M :_n P)[^{\Phi(D_n)}/_{\alpha}] \longrightarrow^{n*} M[^{D_n\{-\}}/_{[\alpha]\{-\}}] :_n P[^{\Phi(D_n)}/_{\alpha}]$, and $DC \vdash S^{\sharp}[^{\Phi(D_n)}/_{\alpha}] \longrightarrow^{n*} (S[^{D_n\{-\}}/_{[\alpha]\{-\}}])^{\sharp}$ hold for any covalue P.

Proof. See Appendix. □

Then, we prove that the call-by-name modified translation $(-)^{\sharp}$ preserves reductions.

Theorem 16 (Soundness of $(-)^{\sharp}$)

(1) If $\lambda\mu \vdash M \longrightarrow_n N$, then $DC \vdash (M :_n P) \longrightarrow^{n*} (N :_n P)$ for any covalue P, especially $DC \vdash M^{\sharp} \longrightarrow^{n*} N^{\sharp}$.

(2) If $\lambda\mu \vdash S \longrightarrow_n T$, then $DC \vdash S^{\sharp} \longrightarrow^{n*} T^{\sharp}$. Moreover, if $\longrightarrow_n$ is $(\beta_{\supset})$, $(\beta_{\&})$, $(\beta_{\vee})$ or $(\beta_{\neg})$, then $\longrightarrow^{n*}$ can be replaced by $\longrightarrow^{n+}$.

Proof. See Appendix. □

4.3 The Translation from CBV $\lambda\mu$-calculus into CBV Dual Calculus

In this subsection, we introduce the call-by-value translation from the $\lambda\mu$-calculus into the dual calculus by modifying the naive translation $(-)^{\circledast}$. The call-by-name translation also consists of the two translations: $(-)^{\dagger}$ and $(-) :_v K$. $(-)^{\dagger}$ maps any term M and statement S of the $\lambda\mu$-calculus to a term $M^{\dagger}$ and statement $S^{\dagger}$ of the dual calculus respectively. The infix operator "$:_v$" translates a pair of terms M of the $\lambda\mu$-calculus and a coterm K of the dual calculus into a statement $M :_v K$ of the dual calculus. The definition of the call-by-value translation $(-)^{\dagger}$ is given in **Fig. 7**.

When we compare the CBV translation given here with the CBN translation, the definitions of $(\mu\alpha.S :_v K)$, $((\lambda x.M)N :_v K)$, $(\delta(O, x.M, y.N) :_v K)$, $((\lambda x.S)N)^{\dagger}$, and $\delta(O, x.S, y.T)^{\dagger}$ are different. This is because the administrative redexes differ according to the difference of (ζ)-rules of the call-by-name and call-by-value systems.

Like the call-by-name translation, the call-by-value translation is also consistent with the naive translation in the sense of the following lemma.

Lemma 17

Let M and S be a term and a statement of the $\lambda\mu$-calculus, then

$$
\begin{aligned}
(M)^\dagger &\equiv (M :_v \alpha).\alpha && \text{(where } \alpha \text{ is a fresh covariable)}\\
([\alpha]M)^\dagger &\equiv M :_v \alpha\\
(OM)^\dagger &\equiv O :_v \mathrm{not}\langle M^\dagger\rangle && \text{(where } O \text{ is not a } \lambda\text{-abstraction)}\\
((\lambda x.S)N)^\dagger &\equiv N :_v x.S^\dagger\\
(\delta(O,x.S,y.T))^\dagger &\equiv M :_v [x.S^\dagger, y.T^\dagger]\\
x :_v K &\equiv x \bullet K & \langle M,N\rangle :_v K &\equiv \langle M^\dagger, N^\dagger\rangle \bullet K\\
\mathrm{fst}(M) :_v K &\equiv M :_v \mathrm{fst}[K] & \mathrm{inl}(M) :_v K &\equiv \langle M^\dagger\rangle \mathrm{inl} \bullet K\\
\mathrm{snd}(M) :_v K &\equiv M :_v \mathrm{snd}[K] & \mathrm{inr}(M) :_v K &\equiv \langle M^\dagger\rangle \mathrm{inr} \bullet K\\
(\lambda x.S) :_v K &\equiv [x.S^\dagger]\mathrm{not} \bullet K & \mu\alpha.S :_v K &\equiv S^\dagger[{}^K/_\alpha]\\
(\lambda x.M) :_v K &\equiv (\lambda x.M^\dagger) \bullet K\\
(OM) :_v K &\equiv O :_v (M^\dagger @ K) && \text{(where } O \text{ is not a } \lambda\text{-abstraction)}\\
((\lambda x.M)N) :_v K &\equiv N :_v x.(M :_v K)\\
\delta(O,x.M,y.N) :_v K &\equiv O :_v [x.(M :_v K), y.(N :_v K)]
\end{aligned}
$$

Fig. 7 Translation $(-)^\dagger$: CBV $\lambda\mu$-calculus $\longrightarrow$ CBV dual calculus.

(1) $DC \vdash M^{\circledast} \bullet K \longrightarrow^{v*} M :_v K$ for any coterm K,
(2) $DC \vdash M^{\circledast} \longrightarrow^{v*} M^\dagger$, and
(3) $DC \vdash S^{\circledast} \longrightarrow^{v*} S^\dagger$.

Proof. We prove (1), (2), and (3) by a simultaneous induction on M and S. Most parts of this proof are similar to the one in Lemma 12. For example,
Case of $(\lambda x.M)N$:

$$
\begin{aligned}
&((\lambda x.M)N)^{\circledast} \bullet K\\
&\quad\equiv ((\lambda x.M^{\circledast}) \bullet (N^{\circledast} @ \alpha)).\alpha \bullet K\\
&\quad\longrightarrow^v_{(\beta_L)} (\lambda x.M^{\circledast}) \bullet (N^{\circledast} @ K)\\
&\quad\longrightarrow^v_{(\beta_\supset)} N^{\circledast} \bullet x.(M^{\circledast} \bullet K)\\
&\quad\overset{I.H.(1)}{\longrightarrow^{v*}} N :_v x.(M :_v K)\\
&\quad\equiv ((\lambda x.M)N) :_v K \qquad \square
\end{aligned}
$$

This call-by-value translation also preserves the typing derivation.

Proposition 18
(1) If $\Gamma \vdash_{\lambda\mu} \Delta \,|\, M : A$, then $\Gamma \vdash_{dc} \Delta \,|\, M^\dagger : A$.
(2) If $\Gamma \mid S \vdash_{\lambda\mu} \Delta$, then $\Gamma \mid S^\dagger \vdash_{dc} \Delta$.

Proof. This proposition can be shown by the subject reduction property for the dual calculus using Proposition 11 and Lemma 17. $\square$

Definition
For each value V in the $\lambda\mu$-calculus, we define a value $(V)^v$ in the dual calculus as follows.

$$
\begin{aligned}
(x)^v &\equiv x, & (\langle V,W\rangle)^v &\equiv \langle V^v, W^v\rangle\\
(\mathrm{inl}(V))^v &\equiv \langle V^v\rangle\mathrm{inl} & (\lambda x.M)^v &\equiv \lambda x.M^\dagger\\
(\mathrm{inr}(W))^v &\equiv \langle W^v\rangle\mathrm{inr} & (\lambda x.S)^v &\equiv [x.S^\dagger]\mathrm{not}
\end{aligned}
$$

Using this notation, we can show the following lemma.

Lemma 19
(1) $DC \vdash V^v \bullet K \longrightarrow^{v*} (V :_v K)$ for any coterm K, especially $V^v \longrightarrow^{v*} V^\dagger$.
(2) $DC \vdash (V :_v K) \longrightarrow^{v*} V^v \bullet K$ for any coterm K. That is, the statement $(V :_v K)$ loops in the call-by-value dual calculus.
(3) Let M be a term, S be a statement, and V be a value of the $\lambda\mu$-calculus, then $DC \vdash (M :_v K)[{}^{V^v}/_x] \longrightarrow^{v*} (M[{}^V/_x] :_v K[{}^{V^v}/_x])$, and $DC \vdash S^\dagger[{}^{V^v}/_x] \longrightarrow^{v*} (S[{}^V/_x])^\dagger$ for any coterm K.

Proof. (1) is proved by a straightforward induction on V. For example, we consider the case of $\langle V,W\rangle$:

$$
\begin{aligned}
&(\langle V,W\rangle^v \bullet K) \equiv \langle V^v, W^v\rangle \bullet K\\
&\quad\overset{I.H.}{\longrightarrow^{v*}} \langle V^\dagger, W^\dagger\rangle \bullet K \equiv \langle V,W\rangle^\dagger \bullet K
\end{aligned}
$$

(2) is also proved by an induction on V. We consider the key cases.
Case of $\langle V,W\rangle$:

$$
\begin{aligned}
&(\langle V,W\rangle :_v K) \equiv \langle V^\dagger, W^\dagger\rangle \bullet K\\
&\quad\longrightarrow^v_{(name)} V^\dagger \bullet x.(\langle x, W^\dagger\rangle \bullet K)\\
&\quad\longrightarrow^v_{(\beta_R)} V :_v x.(\langle x, W^\dagger\rangle \bullet K)\\
&\quad\overset{I.H.}{\longrightarrow^{v*}} V^v \bullet x.(\langle x, W^\dagger\rangle \bullet K)\\
&\quad\longrightarrow^v_{(\beta_L)} \langle V^v, W^\dagger\rangle \bullet K\\
&\quad\longrightarrow^v_{(name)} W^\dagger \bullet y.(\langle V^v, y\rangle \bullet K)\\
&\quad\longrightarrow^v_{(\beta_R)} W :_v y.(\langle V^v, y\rangle \bullet K)\\
&\quad\overset{I.H.}{\longrightarrow^{v*}} W^v \bullet y.(\langle V^v, y\rangle \bullet K)\\
&\quad\longrightarrow^v_{(\beta_L)} \langle V^v, W^v\rangle \bullet K \equiv \langle V,W\rangle^v \bullet K
\end{aligned}
$$

Cases of $\langle V\rangle$inl and $\langle W\rangle$inr: this case can be proved in a way similar to the above case using the induction hypothesis and (*name*)-rule.
(3) is proved by a straightforward induction on M and S. The key case is:

$(x :_v K)[^{V^v}/_x] \equiv (x \bullet K)[^{V^v}/_x]$

$\equiv V^v \bullet K[^{V^v}/_x] \overset{(1)}{\longrightarrow^{v*}} (V :_v K[^{V^v}/_x])$ □

Let E_v be a call-by-value evaluation singular term context, and K be a coterm of the dual calculus. Then we define coterm $\Psi(E_v, K)$ as follows.

$\Psi(\{-\}N, K) \equiv N^\dagger @K$
$\Psi((\lambda x.M)\{-\}, K) \equiv x.(M :_v K)$
$\Psi(\langle V, \{-\}\rangle, K) \equiv y.(\langle V^v, y\rangle \bullet K)$
$\Psi(\langle \{-\}, M\rangle, K) \equiv x.(\langle x, M^\dagger\rangle \bullet K)$
$\Psi(\delta(\{-\}, x.M, y.N), K)$
$\quad \equiv [x.(M :_v K), y.(N :_v K)]$
$\Psi(\mathrm{fst}(-), K) \equiv \mathrm{fst}[K]$
$\Psi(\mathrm{inl}(-), K) \equiv x.(\langle x\rangle\mathrm{inl} \bullet K)$
$\Psi(\mathrm{snd}\{-\}, K) \equiv \mathrm{snd}[K]$
$\Psi(\mathrm{inr}(-), K) \equiv y.(\langle y\rangle\mathrm{inr} \bullet K)$

and for every singular statement context D_v, we define coterm $\Psi(D_v)$ as follows.

$\Psi([\alpha]\{-\}) \equiv \alpha \qquad \Psi(\{-\}N) \equiv \mathrm{not}\langle N^\dagger\rangle$
$\Psi((\lambda x.S)\{-\}) \equiv x.S^\dagger$
$\Psi(\delta(\{-\}, x.S, y.T)) \equiv [x.S^\dagger, y.T^\dagger]$

About this notation, the following properties hold.

Lemma 20
(1) Let M not be a λ-abstraction, then $DC \vdash (E_v\{M\} :_v K) \longrightarrow^{v*} (M :_v \Psi(E_v, K))$ and $DC \vdash (D_v\{M\})^\dagger \longrightarrow^{v*} (M :_v \Psi(D_v))$ hold.
(2) Let E be an elimination context or $(\lambda x.M)\{-\}$, and D_v be an evaluation singular statement context. Then $DC \vdash (M :_v \Psi(E, K)) \longrightarrow^{v*} (E\{M\} :_v K)$ and $DC \vdash (M :_v \Psi(D_v)) \longrightarrow^{v*} (D_v\{M\})^\dagger$ hold for any term M of the $\lambda\mu$-calculus.
(3) Let E_i be an introduction context of the $\lambda\mu$-calculus. Then $DC \vdash (M :_v \Psi(E_i, K)) \longrightarrow^{v*} ((\lambda x.E_i\{x\})M :_v K)$ for any term M.
(4) Let E be an elimination context or $(\lambda x.M)\{-\}$ of the call-by-value $\lambda\mu$-calculus. Then $DC \vdash (M :_v K)[^{\Psi(E,\beta)}/_\alpha] \longrightarrow^{v*} M[^{[\beta]E\{-\}}/_{[\alpha]\{-\}}] :_v K[^{\Psi(E,\beta)}/_\alpha]$, and $DC \vdash S^\dagger[^{\Psi(E,\beta)}/_\alpha] \longrightarrow^{v*} (S[^{[\beta]E\{-\}}/_{[\alpha]\{-\}}])^\dagger$ hold for any coterm K.
(5) Let D_v be an evaluation singular statement context of the call-by-value $\lambda\mu$-calculus. Then $DC \vdash (M :_v K)[^{\Psi(D_v)}/_\alpha] \longrightarrow^{v*} M[^{D_v\{-\}}/_{[\alpha]\{-\}}] :_v K[^{\Psi(D_v)}/_\alpha]$, and $DC \vdash S^\dagger[^{\Psi(D_v)}/_\alpha] \longrightarrow^{v*} (S[^{D_v\{-\}}/_{[\alpha]\{-\}}])^\dagger$ hold for any coterm K.

Proof. See Appendix. □

Then we prove that the call-by-value modified translation $(-)^\dagger$ preserves reductions.

Theorem 21 (Soundness of $(-)^\dagger$)
(1) If $\lambda\mu \vdash M \longrightarrow_v N$, then $DC \vdash (M :_v K) \longrightarrow^{v*} (N :_v K)$ for any coterm K, especially $DC \vdash M^\dagger \longrightarrow^{v*} N^\dagger$.
(2) If $\lambda\mu \vdash S \longrightarrow_v T$, then $DC \vdash S^\dagger \longrightarrow^{v*} T^\dagger$. Moreover, if $\longrightarrow_v$ is $(\beta_\supset)$, $(\beta_\&)$, $(\beta_\vee)$ or $(\beta_\neg)$, then $\longrightarrow^{v*}$ can be replaced by $\longrightarrow^{v+}$.

Proof. See Appendix. □

5. Translation from the Dual Calculus into the $\boldsymbol{\lambda\mu}$-calculus

In this section, we introduce the translations from the dual calculus into the $\lambda\mu$-calculus. As in the previous section, we give two different translations for the call-by-name and call-by-value calculi.

5.1 The Naive Translation

In this subsection, we give the naive translation from the dual calculus into the $\lambda\mu$-calculus

This translation preserves equalities but does not preserve reductions.

Definition (The Naive Translation $(-)_\circledast$)
The naive translation from the dual calculus into the $\lambda\mu$-calculus is defined as follows. This translation $(-)_\circledast$ maps a term M and a statement S of the dual calculus to a term $M_\circledast$ and a statement $S_\circledast$ of the $\lambda\mu$-calculus respectively, and maps a coterm K with a term O of the $\lambda\mu$-calculus to a statement $K_\circledast\{O\}$ of the $\lambda\mu$-calculus.

$(x)_\circledast \equiv x \qquad \alpha_\circledast\{O\} \equiv [\alpha]O$
$(\langle M, N\rangle)_\circledast \equiv \langle M_\circledast, N_\circledast\rangle$
$[K, L]_\circledast\{O\} \equiv \delta(O, x.K_\circledast\{x\}, y.L_\circledast\{y\})$
$(\langle M\rangle\mathrm{inl})_\circledast \equiv \mathrm{inl}(M_\circledast)$
$(\mathrm{fst}[K])_\circledast\{O\} \equiv K_\circledast\{\mathrm{fst}(O)\}$
$(\langle N\rangle\mathrm{inr})_\circledast \equiv \mathrm{inr}(N_\circledast)$
$(\mathrm{snd}[L])_\circledast\{O\} \equiv L_\circledast\{\mathrm{snd}(O)\}$
$([K]\mathrm{not})_\circledast \equiv \lambda x.K_\circledast\{x\}$
$\mathrm{not}\langle M\rangle_\circledast\{O\} \equiv OM_\circledast$
$(\lambda x.M)_\circledast \equiv \lambda x.M_\circledast$
$(M@K)_\circledast\{O\} \equiv K_\circledast\{OM_\circledast\}$
$(S.\alpha)_\circledast \equiv \mu\alpha.S_\circledast$
$(x.S)_\circledast\{O\} \equiv (\lambda x.S_\circledast)O$
$(M \bullet K)_\circledast \equiv K_\circledast\{M_\circledast\}$

This naive translation is given by changing the part of sums of the original translation $(-)_*$ given by Wadler (2005). The naive translation is consistent with Wadler's translation in the sense of the following lemma.

Lemma 22

Let M be a term, K be a coterm, S be a statement of the dual calculus, and O be a term of the $\lambda\mu^{wad}$-calculus. Then

(1) $\lambda\mu \vdash [\![M_*]\!] =^n M_\circledast$, $\lambda\mu \vdash [\![K_*\{O\}]\!] =^n K_\circledast\{[\![O]\!]\}$, and $\lambda\mu \vdash [\![S_*]\!] =^n S_\circledast$.

(2) $\lambda\mu \vdash [\![M_*]\!] =^v M_\circledast$, $\lambda\mu \vdash [\![K_*\{O\}]\!] =^v K_\circledast\{[\![O]\!]\}$, and $\lambda\mu \vdash [\![S_*]\!] =^v S_\circledast$.

Proof. See Appendix. □

The naive translation preserves typing rules and equalities.

Proposition 23

(1) If $\Gamma \vdash \Delta \mid M : A$, then $\Gamma \vdash_{\lambda\mu} \Delta \mid M_\circledast : A$.
If $K : A \mid \Gamma \vdash \Delta$ and $\Gamma \vdash_{\lambda\mu} \Delta \mid O : A$, then $\Gamma \mid K_\circledast\{O\} \vdash_{\lambda\mu} \Delta$.
If $\Gamma \mid S \vdash \Delta$, then $\Gamma \mid S_\circledast \vdash_{\lambda\mu} \Delta$.

(2) If $DC \vdash M =^n N$, then $\lambda\mu \vdash M_\circledast =_n N_\circledast$.
If $DC \vdash K =^n L$, then $\lambda\mu \vdash K_\circledast\{O\} =_n L_\circledast\{O\}$.
If $DC \vdash S =^n T$, then $\lambda\mu \vdash S_\circledast =_n T_\circledast$.

(3) If $DC \vdash M =^v N$, then $\lambda\mu \vdash M_\circledast =_v N_\circledast$.
If $DC \vdash K =^v L$, then $\lambda\mu \vdash K_\circledast\{O\} =_v L_\circledast\{O\}$.
If $DC \vdash S =^v T$, then $\lambda\mu \vdash S_\circledast =_v T_\circledast$.

Proof. (1) We can prove this claim by a straightforward induction on $\vdash$.
(2) This claim can be shown directly by an induction on $=^n$. Even if we do not adopt this approach, we can show this claim as a corollary of Theorem 28 using Lemma 24.
(3) As in (2), we can show this claim directly, or as a corollary of Theorem 34 using Lemma 30. □

In general, this naive translation does not preserve reductions as well as Wadler's translation. (η_L)-rule is a counter-example for the call-by-name system:

$$\begin{aligned}\alpha_\circledast\{O\} &\equiv [\alpha]O \longleftarrow_n (\lambda x.[\alpha]x)O \\ &\equiv (x.(x \bullet \alpha))_\circledast\{O\}\end{aligned}$$

On the other hand, $(\beta_\neg)$-rule is a counter-example for the call-by-value system:

$$\begin{aligned}&([\alpha]\mathrm{not} \bullet \mathrm{not}\langle (x \bullet \beta).\gamma\rangle)_\circledast \\ &\quad\equiv (\mathrm{not}\langle (x \bullet \beta).\gamma\rangle)_\circledast\{[\alpha]\mathrm{not}_\circledast\} \\ &\quad\equiv (\lambda x.[\alpha]x)\mu\gamma.[\beta]x \longleftarrow_{(name)} [\alpha]\mu\gamma.[\beta]x \\ &\quad\equiv \alpha_\circledast\{((x \bullet \beta).\gamma)\circledast\} \equiv ((x \bullet \beta).\gamma \bullet \alpha)_\circledast\end{aligned}$$

We also need to modify this naive translation. In the following two subsections, we give different translations for the call-by-name and call-by-value calculi.

5.2 The Translation from CBN Dual Calculus into CBN $\lambda\mu$-calculus

To solve the problem for the call-by-name calculus displayed at the end of the previous subsection, we need to modify the translation of the coterm $x.S$.

Definition ($(-)_\sharp$: CBN DC $\longrightarrow$ CBN $\lambda\mu$)
We introduce the new translation $(-)_\sharp$ by modifying the definition of $(-)_\circledast$ as

$(x.S)_\sharp\{O\} \equiv S_\sharp[{}^O/_x]$.

The other clauses are similar to the naive translation.
The full definition of $(-)_\sharp$ is displayed in **Fig. 8** (see Appendix).

The following lemma means that the call-by-name translation $(-)_\sharp$ is consistent with the naive translation.

Lemma 24

Let M be a term, K be a coterm, and S be a statement of the dual calculus. Then

(1) $\lambda\mu \vdash M_\circledast \longrightarrow_n^* M_\sharp$,
(2) $\lambda\mu \vdash K_\circledast\{O\} \longrightarrow_n^* K_\sharp\{O\}$ for any term O of the $\lambda\mu$-calculus, and
(3) $\lambda\mu \vdash S_\circledast \longrightarrow_n^* S_\sharp$.

Proposition 25

(1) If $\Gamma \vdash_{dc} \Delta \mid M : A$, then $\Gamma \vdash_{\lambda\mu} \Delta \mid M_\sharp : A$.
(2) If $K : A \mid \Gamma \vdash_{dc} \Delta$ and $\Gamma \vdash_{\lambda\mu} \Delta \mid O : A$, then $\Gamma \vdash_{\lambda\mu} \Delta \mid K_\sharp\{O\}$.
(3) If $\Gamma \mid S \vdash_{dc} \Delta$, then $\Gamma \vdash_{\lambda\mu} \Delta \mid S_\sharp$.

Proof. These claims can be shown by the subject reduction property for the $\lambda\mu$-calculus using Proposition 23 (1) and Lemma 24. □

Lemma 26

(1) If $\lambda\mu \vdash O \longrightarrow_n O'$, then $\lambda\mu \vdash K_\sharp\{O\} \longrightarrow_n^* K_\sharp\{O'\}$ for any coterm K.
(2) $M_\sharp[{}^{N_\sharp}/_x] \equiv (M[{}^N/_x])_\sharp$, $K_\sharp\{O\}[{}^{N_\sharp}/_x] \equiv (K[{}^N/_x])_\sharp\{O[{}^{N_\sharp}/_x]\}$ and $S_\sharp[{}^{N_\sharp}/_x] \equiv (S[{}^N/_x])_\sharp$.
(3) If P is a covalue, then $P_\sharp\{-\}$ is a call-by-name evaluation context of the $\lambda\mu$-calculus.
(4) $M_\sharp[{}^{L_\sharp\{-\}}/_{[\alpha]\{-\}}] \equiv (M[{}^L/_\alpha])_\sharp$, $(K_\sharp\{O\})[{}^{L_\sharp\{-\}}/_{[\alpha]\{-\}}] \equiv (K[{}^L/_\alpha])_\sharp\{O[{}^{L_\sharp\{-\}}/_{[\alpha]\{-\}}]\}$, and $S_\sharp[{}^{L_\sharp\{-\}}/_{[\alpha]\{-\}}] \equiv (S[{}^L/_\alpha])_\sharp$ for any coterm L.

Proof. (1) The claim is proved by induction on K.
(2) The claim is proved by induction on M, K, and S.
(3) The claim is proved by induction on P.
(4) The claim is proved by induction on M, K, and S. We give the key case:

$$(\alpha_\sharp\{O\})[{}^{L_\sharp\{-\}}/_{[\alpha]\{-\}}] \equiv ([\alpha]O)[{}^{L_\sharp\{-\}}/_{[\alpha]\{-\}}] \equiv L_\sharp\{O[{}^{L_\sharp\{-\}}/_{[\alpha]\{-\}}]\}$$

□

Let F be a coterm context of the dual calculus, and O be a term of the $\lambda\mu$-calculus. Then we define term $F_\sharp\{O\}$ of the $\lambda\mu$-calculus as follows:

$(\{-\})_\sharp\{O\} \equiv O \quad (M@\{-\})_\sharp\{O\} \equiv OM_\sharp$
$(\mathrm{fst}[-])_\sharp\{O\} \equiv \mathrm{fst}(O)$
$(\mathrm{snd}[-])_\sharp\{O\} \equiv \mathrm{snd}(O)$
$([-,P])_\sharp\{O\} \equiv \mu\alpha.\delta(O, x.[\alpha]x, y.P_\sharp\{y\})$
$([K,-])_\sharp\{O\} \equiv \mu\beta.\delta(O, x.K_\sharp\{x\}, y.[\beta]y)$

This notation satisfies the following property.

Lemma 27
Let coterm L not be a covalue, and P be a covalue. Then
(1) $\lambda\mu \vdash F\{L\}_\sharp\{O\} \longrightarrow_n^* L_\sharp\{F_\sharp\{O\}\}$, and
(2) $\lambda\mu \vdash P_\sharp\{F_\sharp\{O\}\} \longrightarrow_n^* F\{P\}_\sharp\{O\}$
for any term O of the $\lambda\mu$-calculus.

Proof. (1) is proved by a case analysis of F. We give the key cases.
Case of $[-,P]$:

$$\begin{aligned}[L,P]_\sharp\{O\} &\equiv \delta(O, x.L_\sharp\{x\}, y.P_\sharp\{y\})\\ &\longrightarrow_{(\nu)} (\lambda x.L_\sharp\{x\})\mu\alpha.\delta(O, x.[\alpha]x, y.P_\sharp\{y\})\\ &\longrightarrow_{(\beta_\neg)} L_\sharp\{\,\mu\alpha.\delta(O, x.[\alpha]x, y.P_\sharp\{y\})\,\}\\ &\equiv L_\sharp\{\,[-,P]_\sharp\{O\}\,\}\end{aligned}$$

Case of $[K,-]$:

$$\begin{aligned}[K,L]_\sharp\{O\} &\equiv \delta(O, x.K_\sharp\{x\}, y.L_\sharp\{y\})\\ &\longrightarrow_{(\nu)} (\lambda y.L_\sharp\{y\})\mu\beta.\delta(O, x.K_\sharp\{x\}, y.[\beta]y)\\ &\longrightarrow_{(\beta_\neg)} L_\sharp\{\,\mu\beta.\delta(O, x.K_\sharp\{x\}, y.[\beta]y)\,\}\\ &\equiv L_\sharp\{\,[K,-]_\sharp\{O\}\,\}\end{aligned}$$

(2) is also proved by a case analysis of F. We give the key cases.
Case of $[-,Q]$:

$$\begin{aligned}&P_\sharp\{\,[-,Q]_\sharp\{O\}\,\}\\ &\equiv P_\sharp\{\,\mu\alpha.\delta(O, x.[\alpha]x, y.Q_\sharp\{y\})\,\}\\ &\longrightarrow_{(\zeta)} \delta(O, x.P_\sharp\{x\}, y.Q_\sharp\{y\})\\ &\equiv [P,Q]_\sharp\{O\}\end{aligned}$$

Case of $[K,-]$:

$$\begin{aligned}&P_\sharp\{\,[K,-]_\sharp\{O\}\,\}\\ &\equiv P_\sharp\{\,\mu\beta.\delta(O, x.K_\sharp\{x\}, y.[\beta]y)\,\}\\ &\longrightarrow_{(\zeta)} \delta(O, x.K_\sharp\{x\}, y.P_\sharp\{y\})\\ &\equiv [K,P]_\sharp\{O\}\end{aligned}$$

□

We now prove that the call-by-name translation $(-)_\sharp$ preserves reductions.

Theorem 28 (Soundness of $(-)_\sharp$)
(1) If $DC \vdash M \longrightarrow^n N$, then $\lambda\mu \vdash M_\sharp \longrightarrow_n^* N_\sharp$.
(2) If $DC \vdash K \longrightarrow^n L$, then $\lambda\mu \vdash K_\sharp\{O\} \longrightarrow_n^* L_\sharp\{O\}$.
(3) If $DC \vdash S \longrightarrow^n T$, then $\lambda\mu \vdash S_\sharp \longrightarrow_n^* T_\sharp$.

Moreover, in (1), (2) and (3), if $\longrightarrow^n$ is $(\beta_\supset)$, $(\beta_\&)$, $(\beta_\vee)$ or $(\beta_\neg)$, then $\longrightarrow_n^*$ can be replaced by $\longrightarrow_n^+$.

Proof. See Appendix. □

5.3 The Translation from CBV Dual Calculus into CBV $\lambda\mu$-calculus

Now we introduce the modified translation $(-)_\dagger$ for the call-by-value calculus. The problematic cases of the naive translation were (η_L) and $(\beta_\neg)$-rules. To solve these problems, we introduce a notation: $(\overline{\lambda}_v x.S)O$. This means $S[{}^O/_x]$ if O is a value, otherwise $(\lambda x.S)O$. Our idea for solving the latter case is to modify the definition of $(M \bullet K)_\circledast$ to $(\overline{\lambda}_v x.K_\circledast\{x\})M_\circledast$.

Definition ($(-)_\dagger$: CBV DC $\longrightarrow$ CBV $\lambda\mu$)
We define the translation $(-)_\dagger$ as follows:

$K_\dagger[O] \equiv (\overline{\lambda}_v x.K_\dagger\{x\})O$
$\mathrm{fst}[K]_\dagger\{O\} \equiv K_\dagger[\mathrm{fst}(O)]$
$\mathrm{snd}[K]_\dagger\{O\} \equiv K_\dagger[\mathrm{snd}(O)]$
$(M@K)_\dagger\{O\} \equiv K_\dagger[OM_\dagger]$
$(x.S)_\dagger\{O\} \equiv (\overline{\lambda}_v x.S_\dagger)O$
$(M \bullet K)_\dagger \equiv K_\dagger[M_\dagger]$

The other clauses are similar to the naive translation. The full definition of $(-)_\dagger$ is displayed in **Fig. 9** (see Appendix).

For the call-by-value translation, we use the two notations $K_\dagger\{O\}$ and $K_\dagger[O]$. The relation between these two notations is as follows.

Lemma 29
Let O be a term of the $\lambda\mu$-calculus, and K be a coterm, then
(1) $K_\dagger\{V\} \equiv K_\dagger[V]$ for any value V; and
(2) $\lambda\mu \vdash K_\dagger\{O\} \longrightarrow_v^* K_\dagger[O]$.

Proof. See Appendix. □

The call-by-value translation, $(-)_\dagger$, is consistent with the naive translation as well as the call-by-name translation.

Lemma 30
Let M be a term, K be a coterm, and S be a statement of the dual calculus, then
(1) $\lambda\mu \vdash M_\circledast \longrightarrow_v^* M_\dagger$,
(2) $\lambda\mu \vdash K_\circledast\{O\} \longrightarrow_v^* K_\dagger\{O\}$ for any term O of the $\lambda\mu$-calculus , and
(3) $\lambda\mu \vdash S_\circledast \longrightarrow_v^* S_\dagger$.

Proof. See Appendix. □

The translation $(-)_\dagger$ is compatible with the type system.

Proposition 31
(1) If $\Gamma \vdash_{dc} \Delta \mid M : A$, then $\Gamma \vdash_{\lambda\mu} \Delta \mid M_\dagger : A$.

(2) If $K : A \mid \Gamma \vdash_{dc} \Delta$ and $\Gamma \vdash_{\lambda\mu} \Delta \mid O : A$, then $\Gamma \vdash_{\lambda\mu} \Delta \mid K_{\dagger}\{O\}$.
(3) If $\Gamma \mid S \vdash_{dc} \Delta$, then $\Gamma \vdash_{\lambda\mu} \Delta \mid S_{\dagger}$.

Proof. These claims can be shown by the subject reduction property for the $\lambda\mu$-calculus using Proposition 23 (1) and Lemma 30. □

Lemma 32
Let O and O' be terms of the $\lambda\mu$-calculus, M and N be terms, V be a value, K and L be coterms, and S be a statement of the dual calculus.

(1) If $\lambda\mu \vdash O \longrightarrow_v O'$, then $\lambda\mu \vdash K_{\dagger}\{O\} \longrightarrow_v^* K_{\dagger}\{O'\}$ and $\lambda\mu \vdash K_{\dagger}[O] \longrightarrow_v^* K_{\dagger}[O']$.
(2) $M_{\dagger}[{}^{V_{\dagger}}/_{x}] \equiv (M[{}^{V}/_{x}])_{\dagger}$,
$(K_{\dagger}\{O\})[{}^{V_{\dagger}}/_{x}] \equiv (K[{}^{V}/_{x}])_{\dagger}\{O[{}^{V_{\dagger}}/_{x}]\}$,
$(K_{\dagger}[O])[{}^{V_{\dagger}}/_{x}] \equiv (K[{}^{V}/_{x}])_{\dagger}[\,O[{}^{V_{\dagger}}/_{x}]\,]$, and
$S_{\dagger}[{}^{V_{\dagger}}/_{x}] \equiv (S[{}^{V}/_{x}])_{\dagger}$.

Proof. See Appendix. □

Lemma 33
Let O be a term of the $\lambda\mu$-calculus, M be a term, K and L be coterms, and S be a statement of the dual calculus. Then

$$\lambda\mu \vdash M_{\dagger}[{}^{(\lambda y.L_{\dagger}\{y\})\{-\}}/_{[\alpha]\{-\}}] \longrightarrow_v^* (M[{}^{L}/_{\alpha}])_{\dagger},$$
$$\lambda\mu \vdash (K_{\dagger}\{O\})[{}^{(\lambda y.L_{\dagger}\{y\})\{-\}}/_{[\alpha]\{-\}}] \longrightarrow_v^* (K[{}^{L}/_{\alpha}])_{\dagger}\left[O[{}^{(\lambda y.L_{\dagger}\{y\})\{-\}}/_{[\alpha]\{-\}}]\right],$$
$$\lambda\mu \vdash (K_{\dagger}[O])[{}^{(\lambda y.L_{\dagger}\{y\})\{-\}}/_{[\alpha]\{-\}}] \longrightarrow_v^* (K[{}^{L}/_{\alpha}])_{\dagger}\left[\,O[{}^{(\lambda y.L_{\dagger}\{y\})\{-\}}/_{[\alpha]\{-\}}]\,\right], \text{and}$$
$$\lambda\mu \vdash S_{\dagger}[{}^{(\lambda y.L_{\dagger}\{y\})\{-\}}/_{[\alpha]\{-\}}] \longrightarrow_v^* (S[{}^{L}/_{\alpha}])_{\dagger}.$$

Proof. See Appendix. □

We now prove that the call-by-name translation $(-)_{\dagger}$ preserves reductions.

Theorem 34 (Soundness of $(-)_{\dagger}$)
(1) If $DC \vdash M \longrightarrow^v N$, then $\lambda\mu \vdash M_{\dagger} \longrightarrow_v^* N_{\dagger}$.
(2) If $DC \vdash K \longrightarrow^v L$, then $\lambda\mu \vdash K_{\dagger}\{O\} \longrightarrow_v^* L_{\dagger}\{O\}$ and $\lambda\mu \vdash K_{\dagger}[O] \longrightarrow_v^* L_{\dagger}[O]$.
(3) If $DC \vdash S \longrightarrow^v T$, then $\lambda\mu \vdash S_{\dagger} \longrightarrow_v^* T_{\dagger}$.

Moreover, in (1), (2), and (3), if $\longrightarrow^v$ is $(\beta_{\supset})$, $(\beta_{\&})$, $(\beta_{\vee})$ or $(\beta_{\neg})$, then $\longrightarrow_v^*$ can be replaced by $\longrightarrow_v^+$.

Proof. See Appendix. □

5.4 Reloading Property

Wadler (2005) showed that the compositions of his translations $\lambda\mu \to dual \to \lambda\mu$ and $dual \to \lambda\mu \to dual$ reload into the $\lambda\mu$-calculus and the dual calculus respectively. That is, they become identity maps up to the call-by-name/call-by-value equalities. When we consider about the composition of our translations, we can obtain corresponding results as follows:

- a reloaded term by the call-by-name modified translations reduced from the original term by the call-by-name reductions (Proposition 35 (1), 36 (1)), and
- a reloaded term by the call-by-value modified translations reduced from the original term by the call-by-value reductions (Proposition 35 (2), 36 (2)).

Proposition 35 (Reloading Property (1))
Let O be a term and S be a statement of the $\lambda\mu$-calculus. Then

(1) $\lambda\mu \vdash S \longrightarrow_n^* (S^{\sharp})_{\sharp}$,
$\lambda\mu \vdash P_{\sharp}\{O\} \longrightarrow_n^* (O :_n P)_{\sharp}$ for any covalue P, especially $\lambda\mu \vdash O \longrightarrow_n^* (O^{\sharp})_{\sharp}$;
(2) $\lambda\mu \vdash S \longrightarrow_v^* (S^{\dagger})_{\dagger}$,
$\lambda\mu \vdash K_{\dagger}[O] \longrightarrow_v^* (O :_v K)_{\dagger}$ for any coterm K, especially $\lambda\mu \vdash O \longrightarrow_v^* (O^{\dagger})_{\dagger}$.

Proof. See Appendix. □

Proposition 36 (Reloading Property (2))
Let M be a term, K be a coterm, and S be a statement of the dual calculus. Then

(1) $DC \vdash M \longrightarrow^{n*} (M_{\sharp})^{\sharp}$,
$DC \vdash O^{\sharp} \bullet K \longrightarrow^{n*} (K_{\sharp}\{O\})^{\sharp}$ for any term O of the $\lambda\mu$-calculus, and
$DC \vdash S \longrightarrow^{n*} (S_{\sharp})^{\sharp}$;
(2) $DC \vdash M \longrightarrow^{v*} (M_{\dagger})^{\dagger}$,
$DC \vdash O^{\dagger} \bullet K \longrightarrow^{v*} (K_{\dagger}\{O\})^{\dagger}$ for any term O of the $\lambda\mu$-calculus, and
$DC \vdash S \longrightarrow^{v*} (S_{\dagger})^{\dagger}$.

Proof. See Appendix. □

We can obtain the Church-Rosser property for the $\lambda\mu$-calculus by using the Church-Rosser property for the dual calculus and the results in Section 4 and 5.

Proposition 37 (Church-Rosser Property for $\boldsymbol{\lambda\mu}$)

(1) If $\lambda\mu \vdash O \longrightarrow_n^* M$ and $\lambda\mu \vdash O \longrightarrow_n^* M'$, then there exists a term O' such that $\lambda\mu \vdash M \longrightarrow_n^* O'$ and $\lambda\mu \vdash M' \longrightarrow_n^* O'$.
If $\lambda\mu \vdash S \longrightarrow_n^* T$ and $\lambda\mu \vdash S \longrightarrow_n^* T'$, then there exists a statement S' such that $\lambda\mu \vdash T \longrightarrow_n^* S'$ and $\lambda\mu \vdash T' \longrightarrow_n^* S'$.
(2) If $\lambda\mu \vdash O \longrightarrow_v^* M$ and $\lambda\mu \vdash O \longrightarrow_v^* M'$, then there exists a term O' such that $\lambda\mu \vdash M \longrightarrow_v^* O'$ and $\lambda\mu \vdash M' \longrightarrow_v^* O'$.
If $\lambda\mu \vdash S \longrightarrow_v^* T$ and $\lambda\mu \vdash S \longrightarrow_v^* T'$, then there exists a statement S' such that $\lambda\mu \vdash T \longrightarrow_v^* S'$ and $\lambda\mu \vdash T' \longrightarrow_v^* S'$.

Proof. See Appendix. □

6. Duality between Call-by-name and Call-by-value

Duality is the essential feature of the dual calculus. The dual calculus corresponds to Gentzen's sequent calculus and has explicit duality of classical logic at each level.

- Types: disjunction is dual to conjunction, and negation is self-dual,
- Expressions: terms are dual to coterms, and statements are self-dual,
- Typing rules: right rules are dual to left rules, and cut is self-dual, and
- Evaluation strategies: call-by-value is dual to call-by-name.

In this section, following Wadler's approach, we discuss the systems that do not involve implication, since duality is not defined for implication.

The duality translation from the dual calculus to itself is given as follows.

Duality for the dual calculus

$(X)^\circ \equiv X$
$(A\&B)^\circ \equiv B^\circ \vee A^\circ$
$(A \vee B)^\circ \equiv B^\circ \& A^\circ$
$(\neg A)^\circ \equiv \neg A^\circ$
$(x)^\circ \equiv x \quad (\alpha)^\circ \equiv \alpha$
$(\langle M, N\rangle)^\circ \equiv [N^\circ, M^\circ] \quad ([K, L])^\circ \equiv \langle L^\circ, K^\circ\rangle$
$(\langle M\rangle\mathrm{inl})^\circ \equiv \mathrm{snd}[M^\circ] \quad (\mathrm{fst}[K])^\circ \equiv \langle K^\circ\rangle\mathrm{inr}$
$(\langle N\rangle\mathrm{inr})^\circ \equiv \mathrm{fst}[N^\circ] \quad (\mathrm{snd}[L])^\circ \equiv \langle L^\circ\rangle\mathrm{inl}$
$(S.\alpha)^\circ \equiv x.S^\circ \quad (x.S)^\circ \equiv S^\circ.x$

Proposition 38 (Duality for the Dual Calculus)

(*Involution*) Duality is an involution, that is, $A^{\circ\circ} \equiv A$, $M^{\circ\circ} \equiv M$, $K^{\circ\circ} \equiv K$, and $S^{\circ\circ} \equiv S$.

(*Expressions and typing rules*)

(a) For any term M of the dual calculus, M° is a coterm.
If M has type A, then M° also has type A, i.e.,
$\Gamma \mid\!- \Delta \,|\, M : A$ implies $M^\circ : A \,|\, \Delta^\circ \mid\!- \Gamma^\circ$.
where Γ° is $x_m : A_m^\circ, \ldots, x_1 : A_1^\circ$ for $\Gamma \equiv x_1 : A_1, \ldots, x_m : A_m$, and Δ° is $\alpha_n : B_n^\circ, \ldots, \alpha_1 : B_1^\circ$ for $\Delta \equiv \alpha_1 : B_1, \ldots, \alpha_n : B_n$.

(b) For any coterm K of the dual calculus, K° is a term.
If K has type A, then K° also has type A, i.e.,
$K : A \,|\, \Gamma \mid\!- \Delta$ implies $\Delta^\circ \mid\!- \Gamma^\circ \,|\, K^\circ : A$.

(c) For any statement S of the dual calculus, S° is also a statement, and
$\Gamma \mid S \mid\!- \Delta$ implies $\Delta^\circ \mid S^\circ \mid\!- \Gamma^\circ$.

(*Evaluation strategies*)

(a) If $DC \vdash M \longrightarrow^n N$, then $DC \vdash M^\circ \longrightarrow^v N^\circ$.
If $DC \vdash K \longrightarrow^n L$, then $DC \vdash K^\circ \longrightarrow^v L^\circ$.
If $DC \vdash S \longrightarrow^n T$, then $DC \vdash S^\circ \longrightarrow^v T^\circ$.

(b) If $DC \vdash M \longrightarrow^v N$, then $DC \vdash M^\circ \longrightarrow^n N^\circ$.
If $DC \vdash K \longrightarrow^v L$, then $DC \vdash K^\circ \longrightarrow^n L^\circ$.
If $DC \vdash S \longrightarrow^v T$, then $DC \vdash S^\circ \longrightarrow^n T^\circ$.

Wadler (2005) gave a translation between the call-by-name and call-by-value $\lambda\mu$-calculi by composing the translation, $(-)^\circ$, and his translations between the dual calculus and the $\lambda\mu$-calculus. He explained duality between the call-by-name $\lambda\mu$-calculus and the call-by-value $\lambda\mu$-calculus by purely syntactic techniques. We follow his approach. Since we gave the different translations for call-by-name and call-by-value in the previous sections, we introduce two distinct translations between the call-by-name and call-by-value $\lambda\mu$-calculi.

Definition ($(-)_\circ$: CBN $\boldsymbol{\lambda\mu}$ → CBV $\boldsymbol{\lambda\mu}$)

Let A be a type, M and O be terms, and S be a statement of the $\lambda\mu$-calculus. Then we define the translation $(-)_\circ$ as follows.

$(A)_\circ \equiv A^\circ$
$M_\circ\{O\} \equiv ((M^\sharp)^\circ)_\dagger\{O\}$
$S_\circ \equiv ((S^\sharp)^\circ)_\dagger$

Definition ($(-)_\bullet$: CBV $\boldsymbol{\lambda\mu}$ → CBN $\boldsymbol{\lambda\mu}$)

Let A be a type, M and O be terms, and S be a statement of the $\lambda\mu$-calculus. Then we define the translation $(-)_\bullet$ as follows.

$(A)_\bullet \equiv A^\circ$
$M_\bullet\{O\} \equiv ((M^\dagger)^\circ)_\sharp\{O\}$
$S_\bullet \equiv ((S^\dagger)^\circ)_\sharp$

The following properties of these translations are easily shown.

Proposition 39

(1) For any term M of the $\lambda\mu$-calculus, $M_\circ\{O\}$ and $M_\bullet\{O\}$ are statements of the $\lambda\mu$-calculus. For any statement S of the $\lambda\mu$-calculus, $S_\circ$ and $S_\bullet$ are statements of the $\lambda\mu$-calculus.

(2) If $\Gamma \mid\!-_{\lambda\mu} \Delta \mid M : A$ and $\Delta_\circ \mid\!-_{\lambda\mu} \Gamma_\circ \mid O : A_\circ$, then $\Gamma \mid M_\circ\{O\} \mid\!-_{\lambda\mu} \Delta$.
If $\Gamma \mid S \mid\!-_{\lambda\mu} \Delta$, then $\Gamma \mid S_\circ \mid\!-_{\lambda\mu} \Delta$.

(3) If $\Gamma \mid\!-_{\lambda\mu} \Delta \mid M : A$ and $\Delta_\bullet \mid\!-_{\lambda\mu} \Gamma_\bullet \mid O : A_\bullet$, then $\Gamma \mid M_\bullet\{O\} \mid\!-_{\lambda\mu} \Delta$.
If $\Gamma \mid S \mid\!-_{\lambda\mu} \Delta$, then $\Gamma \mid S_\bullet \mid\!-_{\lambda\mu} \Delta$.

Then, we obtain our final results.

Theorem 40

Let M, N, and O be terms, and S and T be statements. Then the following hold.

(1) The translation $(-)_\circ$ preserves reductions.
$\lambda\mu \vdash M \longrightarrow_n N$ implies $\lambda\mu \vdash M_\circ\{O\} \longrightarrow_v N_\circ\{O\}$
$\lambda\mu \vdash S \longrightarrow_n T$ implies $\lambda\mu \vdash S_\circ \longrightarrow_v T_\circ$

(2) The translation $(-)_\bullet$ preserves reductions.

$\lambda\mu \vdash M \longrightarrow_v N$ implies $\lambda\mu \vdash M_\bullet\{O\} \longrightarrow_n N_\bullet\{O\}$

$\lambda\mu \vdash S \longrightarrow_v T$ implies $\lambda\mu \vdash S_\bullet \longrightarrow_n T_\bullet$

(3) The composition of translations obtained by applying $(-)_\bullet$ after $(-)_\circ$ is identity up to the call-by-name reductions.

$\lambda\mu \vdash M \longrightarrow_n \mu\alpha.(M_\circ\{\alpha\})_\bullet$

$\lambda\mu \vdash O_\bullet\{M\} \longrightarrow_n (M_\circ\{O\})_\bullet$

$\lambda\mu \vdash S \longrightarrow_n (S_\circ)_\bullet$

(4) The composition of translations obtained by applying $(-)_\circ$ after $(-)_\bullet$ is identity up to the call-by-value reductions.

$\lambda\mu \vdash M \longrightarrow_v \mu\alpha.(M_\bullet\{\alpha\})_\circ$

$\lambda\mu \vdash O_\circ\{M\} \longrightarrow_v (M_\bullet\{O\})_\circ$

$\lambda\mu \vdash S \longrightarrow_v (S_\bullet)_\circ$

Proof. See Appendix. □

Although Wadler gave the same translation which goes back and forth between the call-by-name and call-by-value $\lambda\mu$-calculi, we needed two different translations. However, although Wadler's translation preserved only *equations*, our translations preserve *reductions*. This is the greatest advantage of our results.

Acknowledgments I thank Philip Wadler for the very fruitful discussions. I also thank Makoto Tatsuta, Kazushige Terui, and Makoto Kanazawa for very helpful comments and discussions on this work.

References

1) de Groote, P.: A CPS-translation of the $\lambda\mu$-calculus, *Proc. 19th Intl. Coll. on Trees in Algebra and Programming, CAAP'94*, Edinburgh, UK, 11–13 Apr.1994, Vol.787, pp.85–99, Berlin, Springer-Verlag (1994).
2) de Groote, P.: Strong normalization of classical natural deduction with disjunction, *5th International Conference on Typed Lambda Calculi and Applications, TLCA'01*, *Lecture Notes in Computer Science*, Vol.2044, pp.182–196, Springer-Verlag (2001).
3) Felleisen, M., Friedman, D.P., Kohlbecker, E. and Duba, B.: A syntactic theory of sequential control, *Th. Comp. Sci.*, Vol.52, No.3, pp.205–237 (1987).
4) Filinski, A.: Declarative continuations and categorical duality, Master's thesis, Univ. of Copenhagen (1989).
5) Griffin, T.G.: A formulae-as-types notion of control, *Proc. 1990 Principles of Programming Languages Conference*, pp.47–58, IEEE Computer Society Press (1990).
6) Ong, C.-H.L. and Stewart, C.A.: A curry-howard foundation for functional computation with control, *Proc. Symposium on Principles of Programming Languages*, pp.215–227 (1997).
7) Parigot, M.: $\lambda\mu$-calculus: An algorithmic interpretation of classical natural deduction, *Proc. International Conference on Logic Programming and Automated Deduction*, *Lecture Notes in Computer Science*, Vol.624, pp.190–201, Springer-Verlag (1992).
8) Sabry, A. and Felleisen, M.: Reasoning about programs in continuation-passing style, *Lisp and Symbolic Computation*, Vol.6, No.3/4, pp.289–360 (1993).
9) Selinger, P.: Control categories and duality: on the categorical semantics of the lambda-mu calculus, *Mathematical Structures in Computer Science*, pp.207–260 (2001).
10) Wadler, P.: Call-by-Value is Dual to Call-by-Name, *International Conference on Functional Programming*, Uppsala, Sweden, pp.25–29 (2003).
11) Wadler, P.: Call-by-Value is Dual to Call-by-Name — Reloaded, *Rewriting Techniques and Applications*, Nara, Japan, pp.185–203, Springer (2005).

Appendix

A.1 The Full Definitions of Translations $(-)_\sharp$ and $(-)_\dagger$

The full definition of the translation $(-)_\sharp$ from the call-by-name dual calculus into the call-by-name $\lambda\mu$-calculus is displayed in Fig. 8, and the full definition of the translation $(-)_\dagger$ from the call-by-value dual calculus into the call-by-value $\lambda\mu$-calculus is displayed in Fig. 9.

A.2 Proofs

Proof of Lemma 1 (1) is easily shown by induction on E_n and D_n. (2) is shown by induction on M and S using (1). We give the key case.

$$\begin{aligned}
&\langle\!\langle [\alpha]M \rangle\!\rangle[{}^{\overline{D_n}\{-\}}/_{[\alpha]\{-\}}] \\
&\quad\equiv ([\alpha]\langle\!\langle M \rangle\!\rangle)[{}^{\overline{D_n}\{-\}}/_{[\alpha]\{-\}}] \\
&\quad\equiv \overline{D_n}\{\, \langle\!\langle M \rangle\!\rangle[{}^{\overline{D_n}\{-\}}/_{[\alpha]\{-\}}]\,\} \\
&\quad\overset{I.H.}{=}_n \overline{D_n}\{\langle\!\langle M[{}^{D_n\{-\}}/_{[\alpha]\{-\}}] \rangle\!\rangle\} \\
&\quad\overset{(1)}{=}_n \langle\!\langle D_n\{M[{}^{D_n\{-\}}/_{[\alpha]\{-\}}]\} \rangle\!\rangle \\
&\quad\equiv \langle\!\langle ([\alpha]M)[{}^{D_n\{-\}}/_{[\alpha]\{-\}}] \rangle\!\rangle
\end{aligned}$$

(3) is also shown by induction on E_n and D_n. For example, the case of $\delta(E_n, x.D_n\{x\}, y.D'_n\{y\})$ can be proved as follows.

$$
\begin{aligned}
(x)_\sharp &\equiv x & \alpha_\sharp\{O\} &\equiv [\alpha]O\\
(\langle M,N\rangle)_\sharp &\equiv \langle M_\sharp, N_\sharp\rangle & [K,L]_\sharp\{O\} &\equiv \delta(O, x.K_\sharp\{x\}, y.L_\sharp\{y\})\\
(\langle M\rangle\mathrm{inl})_\sharp &\equiv \mathrm{inl}(M_\sharp) & (\mathrm{fst}[K])_\sharp\{O\} &\equiv K_\sharp\{\mathrm{fst}(O)\}\\
(\langle N\rangle\mathrm{inr})_\sharp &\equiv \mathrm{inr}(N_\sharp) & (\mathrm{snd}[L])_\sharp\{O\} &\equiv L_\sharp\{\mathrm{snd}(O)\}\\
([K]\mathrm{not})_\sharp &\equiv \lambda x.K_\sharp\{x\} & \mathrm{not}\langle M\rangle_\sharp\{O\} &\equiv OM_\sharp\\
(\lambda x.M)_\sharp &\equiv \lambda x.M_\sharp & (M@K)_\sharp\{O\} &\equiv K_\sharp\{OM_\sharp\}\\
(S.\alpha)_\sharp &\equiv \mu\alpha.S_\sharp & (x.S)_\sharp\{O\} &\equiv S_\sharp[{}^{O}/{}_{x}]
\end{aligned}
$$
$$(M\bullet K)_\sharp \equiv K_\sharp\{M_\sharp\}$$

Fig. 8 Translation $(-)_\sharp$: CBN dual calculus$\longrightarrow$ CBN $\lambda\mu$-calculus.

$$
\begin{aligned}
(x)_\dagger &\equiv x & \alpha_\dagger\{O\} &\equiv [\alpha]O\\
(\langle M,N\rangle)_\dagger &\equiv \langle M_\dagger, N_\dagger\rangle & [K,L]_\dagger\{O\} &\equiv \delta(O, x.K_\dagger\{x\}, y.L_\dagger\{y\})\\
(\langle M\rangle\mathrm{inl})_\dagger &\equiv \mathrm{inl}(M_\dagger) & (\mathrm{fst}[K])_\dagger\{O\} &\equiv K_\dagger[\mathrm{fst}(O)]\\
(\langle N\rangle\mathrm{inr})_\dagger &\equiv \mathrm{inr}(N_\dagger) & (\mathrm{snd}[L])_\dagger\{O\} &\equiv L_\dagger[\mathrm{snd}(O)]\\
([K]\mathrm{not})_\dagger &\equiv \lambda x.K_\dagger\{x\} & \mathrm{not}\langle M\rangle_\dagger\{O\} &\equiv OM_\dagger\\
(\lambda x.M)_\dagger &\equiv \lambda x.M_\dagger & (M@K)_\dagger\{O\} &\equiv K_\dagger[OM_\dagger]\\
(S.\alpha)_\dagger &\equiv \mu\alpha.S_\dagger & (x.S)_\dagger\{O\} &\equiv (\overline{\lambda}_v x.S_\dagger)O\\
(M\bullet K)_\dagger &\equiv K_\dagger[M_\dagger] & K_\dagger[O] &\equiv (\overline{\lambda}_v x.K_\dagger\{x\})O
\end{aligned}
$$

Fig. 9 Translation $(-)_\dagger$: CBV dual calculus $\longrightarrow$ CBV $\lambda\mu$-calculus.

$$
\begin{aligned}
&\overline{\delta(E_n, x.D_n\{x\}, y.D'_n\{y\})}\{\mu\alpha.S\}\\
&\quad\equiv \overline{D'_n}\{\mu\beta.\overline{D_n}\{\mu\alpha.[\alpha,\beta]\overline{E_n}\{\mu\alpha.S\}\}\}\\
&\quad\overset{I.H.}{=_n} \overline{D'_n}\{\mu\beta.\overline{D_n}\{\mu\alpha.[\alpha,\beta]\mu\alpha'\\
&\qquad .S[{}^{[\alpha']\overline{E_n}\{-\}}/{}_{[\alpha]\{-\}}]\}\}\\
&\quad=_{\zeta_\vee} \overline{D'_n}\{\mu\beta.\overline{D_n}\{\mu\alpha.S[{}^{[\alpha,\beta]\overline{E_n}\{-\}}/{}_{[\alpha]\{-\}}]\}\}\\
&\quad\overset{I.H.}{=_n} \overline{D'_n}\{\mu\beta.S[{}^{[\alpha,\beta]\overline{E_n}\{-\}}/{}_{[\alpha]\{-\}}]\\
&\qquad [{}^{\overline{D_n}\{-\}}/{}_{[\alpha]\{-\}}]\}\\
&\quad\overset{I.H.}{=_n} S[{}^{[\alpha,\beta]\overline{E_n}\{-\}}/{}_{[\alpha]\{-\}}][{}^{\overline{D_n}\{-\}}/{}_{[\alpha]\{-\}}]\\
&\qquad [{}^{\overline{D'_n}\{-\}}/{}_{[\beta]\{-\}}]\\
&\quad\equiv S[{}^{([\alpha,\beta]\overline{E_n}\{-\})}\\
&\qquad {}^{.[{}^{\overline{D_n}\{-\}}/{}_{[\alpha]\{-\}}][{}^{\overline{D'_n}\{-\}}/{}_{[\beta]\{-\}}]}/{}_{[\alpha]\{-\}}]\\
&\quad\overset{(*)}{\equiv} S[{}^{\overline{D'_n}\{\mu\beta.\overline{D_n}\{\mu\alpha.[\alpha,\beta]\overline{E_n}\{-\}\}\}}/{}_{[\alpha]\{-\}}]\\
&\quad\equiv S[{}^{\overline{\delta(E_n, x.D_n\{x\}, y.D'_n\{y\})}\{-\}}/{}_{[\alpha]\{-\}}]
\end{aligned}
$$

where I.H. means the induction hypothesis, and $(*)$ is by the definition of the substitution for a covariable. □

(1) We can show this by induction on the call-by-name equation of the $\lambda\mu^\eta_n$-calculus. We consider only the rules about sums, i.e., $(\beta_\vee)$, (ζ), (π), (ν), and $(\eta_\vee)$-rules.

Case of $(\beta_\vee)$-rule :

$$
\begin{aligned}
&\langle\!\langle \delta(\mathrm{inl}(O), x.E'_n\{x\}, y.E''_n\{y\})\rangle\!\rangle\\
&\quad\equiv \mu\gamma.M\big(\mu\beta.(\lambda x.[\gamma]\langle\!\langle E'_n\{x\}\rangle\!\rangle)\\
&\qquad (\mu\alpha.[\alpha,\beta]\langle\!\langle \mathrm{inl}(O)\rangle\!\rangle)\big)\\
&\quad\equiv \mu\gamma.M\big(\mu\beta.(\lambda x.[\gamma]\langle\!\langle E'_n\{x\}\rangle\!\rangle)\\
&\qquad (\mu\alpha.[\alpha,\beta]\mu(\alpha',\beta').[\alpha']\langle\!\langle O\rangle\!\rangle)\big)\\
&\quad=_{\beta_\vee} \mu\gamma.M\big(\mu\beta.(\lambda x.[\gamma]\langle\!\langle E'_n\{x\}\rangle\!\rangle)\\
&\qquad (\mu\alpha.[\alpha]\langle\!\langle O\rangle\!\rangle)\big)\\
&\quad=_{\eta_\mu} \mu\gamma.M\big(\mu\beta.(\lambda x.[\gamma]\langle\!\langle E'_n\{x\}\rangle\!\rangle)\langle\!\langle O\rangle\!\rangle\big)\\
&\quad=_n \mu\gamma.\big(\lambda y.[\gamma]\overline{E''_n}\{y\}\big)\\
&\qquad \big(\mu\beta.(\lambda x.[\gamma]\overline{E'_n}\{x\})\langle\!\langle O\rangle\!\rangle\big) && \text{(by Lem 1 (1))}\\
&\quad=_{\beta_\neg} \mu\gamma.[\gamma]\overline{E''_n}\{\mu\beta.[\gamma]\overline{E'_n}\{\langle\!\langle O\rangle\!\rangle\}\}\\
&\quad=_n \mu\gamma.[\gamma]\overline{E'_n}\{\langle\!\langle O\rangle\!\rangle\} && \text{(by Lem 1 (3))}\\
&\quad=_{\eta_\mu} \overline{E'_n}\{\langle\!\langle O\rangle\!\rangle\}\\
&\quad=_n \langle\!\langle E'_n\{O\}\rangle\!\rangle && \text{(by Lem 1 (1))}
\end{aligned}
$$

where M is an abbreviation of $\big(\lambda y.[\gamma]\langle\!\langle E''_n\{y\}\rangle\!\rangle\big)$. The other rules of the $(\beta_\vee)$-rule can also be shown similarly.

Case of (ζ)-rule :

$$
\begin{aligned}
&\langle\!\langle E_n\{\mu\alpha.S\}\rangle\!\rangle =_n \overline{E_n}\{\langle\!\langle \mu\alpha.S\rangle\!\rangle\}\\
&&& \text{(by Lem 1 (1))}\\
&\quad\equiv \overline{E_n}\{\mu\alpha.\langle\!\langle S\rangle\!\rangle\}\\
&\quad=_n \mu\beta.\langle\!\langle S\rangle\!\rangle[{}^{[\beta]\overline{E_n}\{-\}}/{}_{[\alpha]\{-\}}]\\
&&& \text{(by Lem 1 (3))}\\
&\quad=_n \langle\!\langle \mu\beta.S[{}^{[\beta]E_n\{-\}}/{}_{[\alpha]\{-\}}]\rangle\!\rangle\\
&&& \text{(by Lem 1 (2))}\\
&\quad\langle\!\langle D_n\{\mu\alpha.S\}\rangle\!\rangle =_n \overline{D_n}\{\langle\!\langle \mu\alpha.S\rangle\!\rangle\}\\
&\quad=_n \overline{D_n}\{\mu\alpha.\langle\!\langle S\rangle\!\rangle\} =_n S[{}^{\overline{D_n}\{-\}}/{}_{[\alpha]\{-\}}]\\
&\quad=_n \langle\!\langle S[{}^{D_n\{-\}}/{}_{[\alpha]\{-\}}]\rangle\!\rangle
\end{aligned}
$$

Case of (π)-rule :

$$\begin{aligned}
&\langle\!\langle E_n\{\delta(O, x.M, y.N\}\rangle\!\rangle \\
&\quad \overset{Lem\ 1(1)}{=_n} \overline{E_n}\{\langle\!\langle \delta(O, x.M, y.N)\rangle\!\rangle\} \\
&\quad \equiv \overline{E_n}\{\mu\gamma.\big(\lambda y.[\gamma]\langle\!\langle N\rangle\!\rangle\big)\big(\mu\beta.(\lambda x.[\gamma]\langle\!\langle M\rangle\!\rangle) \\
&\qquad (\mu\alpha.[\alpha,\beta]\langle\!\langle O\rangle\!\rangle)\big)\} \\
&\quad \overset{Lem\ 1(3)}{=_n} \mu\gamma'.(\lambda y.[\gamma']\overline{E_n}\{\langle\!\langle N\rangle\!\rangle\}) \\
&\qquad \mu\beta.(\lambda x.[\gamma']\overline{E_n}\{\langle\!\langle M\rangle\!\rangle\})(\mu\alpha.[\alpha,\beta]\langle\!\langle O\rangle\!\rangle) \\
&\quad \overset{Lem\ 1(1)}{=_n} \langle\!\langle \delta(O, x.E_n\{M\}, y.E_n\{N\})\rangle\!\rangle
\end{aligned}$$

The other rule of the (π)-rule can also be shown similarly.

Case of (ν)-rule :

$$\begin{aligned}
&\langle\!\langle \delta(O, x.S, y.T)\rangle\!\rangle \equiv \big(\lambda y.\langle\!\langle T\rangle\!\rangle\big)\big(\mu\beta.(\lambda x.\langle\!\langle S\rangle\!\rangle) \\
&\quad (\mu\alpha.[\alpha,\beta]\langle\!\langle O\rangle\!\rangle)\big) \\
&\quad =_n \big(\lambda y.\langle\!\langle T\rangle\!\rangle\big)\big(\mu\beta.(\lambda y'.[\beta]y')(\mu\beta'.(\lambda x.\langle\!\langle S\rangle\!\rangle) \\
&\qquad (\mu\alpha.[\alpha,\beta']\langle\!\langle O\rangle\!\rangle))\big) \\
&\quad \equiv \big(\lambda y.\langle\!\langle T\rangle\!\rangle\big)\big(\mu\beta.(\lambda y'.\langle\!\langle [\beta]y'\rangle\!\rangle) \\
&\qquad (\mu\beta'.(\lambda x.\langle\!\langle S\rangle\!\rangle)(\mu\alpha.[\alpha,\beta']\langle\!\langle O\rangle\!\rangle))\big) \\
&\quad \equiv \langle\!\langle (\lambda y.T)\,\mu\beta.\delta(O, x.S, y.[\beta]y)\rangle\!\rangle \\
&\langle\!\langle \delta(O, x.S, y.D_n\{y\})\rangle\!\rangle \\
&\quad \equiv \big(\lambda y.\langle\!\langle D_n\{y\}\rangle\!\rangle\big)\big(\mu\beta.(\lambda x.\langle\!\langle S\rangle\!\rangle) \\
&\qquad (\mu\alpha.[\alpha,\beta]\langle\!\langle O\rangle\!\rangle)\big) \\
&\quad \equiv \big(\lambda y.\langle\!\langle D_n\{y\}\rangle\!\rangle\big)\big(\mu\beta.M(\mu\alpha.[\alpha,\beta]\langle\!\langle O\rangle\!\rangle)\big) \\
&\quad =_n \big(\lambda y.\overline{D_n}\{y\}\big)\big(\mu\beta.M(\mu\alpha.[\alpha,\beta]\langle\!\langle O\rangle\!\rangle)\big) \\
&\quad =_n \overline{D_n}\{\mu\beta.M(\mu\alpha.[\alpha,\beta]\langle\!\langle O\rangle\!\rangle)\} \\
&\quad =_n \overline{D_n}\{\mu\beta.M\big(\mu\alpha.(\lambda y.[\beta]y) \\
&\qquad (\mu\beta'.(\lambda x.[\alpha]x)(\mu\alpha'.[\alpha',\beta']\langle\!\langle O\rangle\!\rangle))\big)\} \\
&\quad =_n M\big(\mu\alpha.(\lambda y.\overline{D_n}\{y\}) \\
&\qquad (\mu\beta'.(\lambda x.[\alpha]x)(\mu\alpha'.[\alpha',\beta']\langle\!\langle O\rangle\!\rangle))\big) \\
&\quad =_n M\big(\mu\alpha.(\lambda y.\langle\!\langle D_n\{y\}\rangle\!\rangle) \\
&\qquad (\mu\beta'.(\lambda x.\langle\!\langle [\alpha]x\rangle\!\rangle)(\mu\alpha'.[\alpha',\beta']\langle\!\langle O\rangle\!\rangle))\big) \\
&\quad \equiv M\big(\mu\alpha.\langle\!\langle \delta(O, x.[\alpha]x, y.D_n\{y\})\rangle\!\rangle\big) \\
&\quad \equiv \langle\!\langle (\lambda x.S)\,(\mu\alpha.\delta(O, x.[\alpha]x, y.D_n\{y\}))\rangle\!\rangle
\end{aligned}$$

where M is an abbreviation of $(\lambda x.\langle\!\langle S\rangle\!\rangle)$.

Case of $(\eta_\vee)$-rule :

$$\begin{aligned}
&\langle\!\langle \delta(M, x.\mathrm{inl}(x), y.\mathrm{inr}(y))\rangle\!\rangle \\
&\quad \equiv \mu\gamma.\big(\lambda y.[\gamma]\langle\!\langle \mathrm{inr}(y)\rangle\!\rangle\big)\big(\mu\beta.(\lambda x.[\gamma]\langle\!\langle \mathrm{inr}(x)\rangle\!\rangle) \\
&\qquad (\mu\alpha.[\alpha,\beta]\langle\!\langle M\rangle\!\rangle)\big) \\
&\quad \equiv \mu\gamma.\big(\lambda y.[\gamma]\mu(\alpha'',\beta'').[\beta'']y\big) \\
&\qquad (\mu\beta.(\lambda x.[\gamma]\mu(\alpha',\beta').[\alpha']x) \\
&\qquad (\mu\alpha.[\alpha,\beta]\langle\!\langle M\rangle\!\rangle) \\
&\quad =_{\beta_\neg} \mu\gamma.\big(\lambda y.[\gamma]\mu(\alpha'',\beta'').[\beta'']y\big) \\
&\qquad (\mu\beta.[\gamma]\mu(\alpha',\beta').[\alpha']\mu\alpha.[\alpha,\beta]\langle\!\langle M\rangle\!\rangle) \\
&\quad =_{\beta_\mu} \mu\gamma.(\lambda y.[\gamma]\mu(\alpha'',\beta'').[\beta'']y) \\
&\qquad (\mu\beta.[\gamma]\mu(\alpha',\beta').[\alpha',\beta]\langle\!\langle M\rangle\!\rangle) \\
&\quad =_{\beta_\neg} \mu\gamma.[\gamma]\mu(\alpha'',\beta'').[\beta'']\mu\beta.[\gamma]\mu(\alpha',\beta') \\
&\qquad .[\alpha',\beta]\langle\!\langle M\rangle\!\rangle \\
&\quad =_{\beta_\mu} \mu\gamma.[\gamma]\mu(\alpha'',\beta'').[\gamma]\mu(\alpha',\beta') \\
&\qquad .[\alpha',\beta'']\langle\!\langle M\rangle\!\rangle \\
&\quad =_{\eta_\vee} \mu(\alpha_1,\beta_1).[\alpha_1,\beta_1] \\
&\quad \mu\gamma.[\gamma]\mu(\alpha'',\beta'').[\gamma]\mu(\alpha',\beta').[\alpha',\beta'']\langle\!\langle M\rangle\!\rangle \\
&\quad =_{\zeta_\vee} \mu(\alpha_1,\beta_1).[\alpha_1,\beta_1]\mu(\alpha'',\beta'') \\
&\qquad .[\alpha_1,\beta_1]\mu(\alpha',\beta').[\alpha',\beta'']\langle\!\langle M\rangle\!\rangle \\
&\quad =_{\beta_\vee} \mu(\alpha_1,\beta_1).[\alpha_1,\beta_1]\langle\!\langle M\rangle\!\rangle \\
&\quad =_{\eta_\vee} \langle\!\langle M\rangle\!\rangle
\end{aligned}$$

(2) We can show this by induction on the call-by-name equation of the $\lambda\mu_n^{wad}$-calculus. We consider only the rules about sums, i.e., $(\beta_\vee)$, $(\zeta_\vee)$, and $(\eta_\vee)$-rules.

$(\beta_\vee)$-rule :

$$\begin{aligned}
&[\![[\alpha',\beta']\mu(\alpha,\beta).S]\!] \\
&\quad \equiv \delta\big([\![\mu(\alpha,\beta).S]\!], x.[\alpha']x, y.[\beta']y\big) \\
&\quad \equiv \delta\big(\mu\gamma.[\![S]\!][{}^{[\gamma]\mathrm{inl}\{-\}}/_{[\alpha]\{-\}}, {}^{[\gamma]\mathrm{inr}\{-\}}/_{[\beta]\{-\}}], \\
&\qquad x.[\alpha']x, y.[\beta']y\big) \\
&\quad =_\zeta [\![S]\!][{}^{\delta(\mathrm{inl}\{-\}, x.[\alpha']x, y.[\beta']y)}/_{[\alpha]\{-\}}, \\
&\qquad {}^{\delta(\mathrm{inr}\{-\}, x.[\alpha']x, y.[\beta']y)}/_{[\beta]\{-\}}] \\
&\quad =_{\beta_\vee} [\![S]\!][{}^{[\alpha']\{-\}}/_{[\alpha]\{-\}}, {}^{[\beta']\{-\}}/_{[\beta]\{-\}}] \\
&\qquad \equiv [\![S[{}^{\alpha'}/_{\alpha}, {}^{\beta'}/_{\beta}]]\!]
\end{aligned}$$

$(\zeta_\vee)$-rule :

$$\begin{aligned}
&[\![[\alpha,\beta]\mu\gamma.S]\!] \equiv \delta([\![\mu\gamma.S]\!], x.[\alpha]x, y.[\beta]y) \\
&\quad \equiv \delta(\mu\gamma.[\![S]\!], x.[\alpha]x, y.[\beta]y) \\
&\quad =_\zeta [\![S]\!][{}^{\delta(\{-\}, x.[\alpha]x, y.[\beta]y)}/_{[\gamma]\{-\}}] \\
&\quad \overset{(*)}{\equiv} [\![S[{}^{[\alpha,\beta]\{-\}}/_{[\gamma]\{-\}}]]\!]
\end{aligned}$$

$(*)$ is shown by induction on terms and statements. The key case is as follows.

$$\begin{aligned}
&[\![[\gamma]M]\!][{}^{\delta(\{-\}, x.[\alpha]x, y.[\beta]y)}/_{[\gamma]\{-\}}] \\
&\equiv ([\gamma][\![M]\!])[{}^{\delta(\{-\}, x.[\alpha]x, y.[\beta]y)}/_{[\gamma]\{-\}}] \\
&\equiv \delta\big([\![M]\!][{}^{\delta(\{-\}, x.[\alpha]x, y.[\beta]y)}/_{[\gamma]\{-\}}], \\
&\quad x.[\alpha]x, y.[\beta]y\big) \\
&\overset{I.H.}{\equiv} \delta\big([\![M[{}^{[\alpha,\beta]\{-\}}/_{[\gamma]\{-\}}]]\!], x.[\alpha]x, y.[\beta]y\big) \\
&\equiv [\![[\alpha,\beta]\,(M[{}^{[\alpha,\beta]\{-\}}/_{[\gamma]\{-\}}])]\!] \\
&\equiv [\![([\gamma]M)[{}^{[\alpha,\beta]\{-\}}/_{[\gamma]\{-\}}]]\!]
\end{aligned}$$

$(\eta_\vee)$-rule :

$$\begin{aligned}
&[\![\mu(\alpha,\beta).[\alpha,\beta]M]\!] \\
&\quad \equiv \mu\gamma.[\![[\alpha,\beta]M]\!]
\end{aligned}$$

$$
\begin{aligned}
&\quad [{}^{[\gamma]\mathrm{inl}\{-\}}/_{[\alpha]\{-\}}, {}^{[\gamma]\mathrm{inr}\{-\}}/_{[\beta]\{-\}}] \\
&\equiv \mu\gamma.\delta(\llbracket M \rrbracket, x.[\alpha]x, y.[\beta]y) \\
&\quad [{}^{[\gamma]\mathrm{inl}\{-\}}/_{[\alpha]\{-\}}, {}^{[\gamma]\mathrm{inr}\{-\}}/_{[\beta]\{-\}}] \\
&\equiv \mu\gamma.\delta(\llbracket M \rrbracket, x.[\gamma]\mathrm{inl}(x), y.[\gamma]\mathrm{inr}(y)) \\
&=_{\pi} \mu\gamma.[\gamma]\delta(\llbracket M \rrbracket, x.\mathrm{inl}(x), y.\mathrm{inr}(y)) \\
&=_{\eta_\mu} \delta(\llbracket M \rrbracket, x.\mathrm{inl}(x), y.\mathrm{inr}(y)) \\
&=_{\eta_\vee} \llbracket M \rrbracket
\end{aligned}
$$

(3) We can show this by induction on term M and statement S of the $\lambda\mu_n^\eta$-calculus. We consider the cases of $\mathrm{inl}(M)$, $\mathrm{inr}(M)$, $\delta(O, x.M, y.N)$ and $\delta(O, x.S, y.T)$.

Case of $\mathrm{inl}(M)$:

$$
\begin{aligned}
&\llbracket \langle\!\langle \mathrm{inl}(M) \rangle\!\rangle \rrbracket \equiv \llbracket \mu(\alpha,\beta).[\alpha]\langle\!\langle M \rangle\!\rangle \rrbracket \\
&\quad \equiv \mu\gamma.\big([\alpha]\llbracket \langle\!\langle M \rangle\!\rangle \rrbracket\big) \\
&\quad\quad [{}^{[\gamma]\mathrm{inl}\{-\}}/_{[\alpha]\{-\}}, {}^{[\gamma]\mathrm{inr}\{-\}}/_{[\beta]\{-\}}] \\
&\quad \equiv \mu\gamma.[\gamma]\mathrm{inl}\big(\llbracket \langle\!\langle M \rangle\!\rangle \rrbracket\big) \\
&\quad\quad \overset{I.H.}{=}_n \mu\gamma.[\gamma]\mathrm{inl}(M) =_{\eta_\mu} \mathrm{inl}(M)
\end{aligned}
$$

The case of $\mathrm{inr}(M)$ can also be shown similarly.

Case of $\delta(O, x.M, y.N)$:

$$
\begin{aligned}
&\llbracket \langle\!\langle \delta(O, x.M, y.N) \rangle\!\rangle \rrbracket \\
&\quad \equiv \llbracket \mu\gamma.\big(\lambda y.[\gamma]\langle\!\langle N \rangle\!\rangle\big)\big(\mu\beta.(\lambda x.[\gamma]\langle\!\langle M \rangle\!\rangle) \\
&\quad\quad (\mu\alpha.[\alpha,\beta]\langle\!\langle O \rangle\!\rangle)\big)\rrbracket \\
&\quad \equiv \mu\gamma.\big(\lambda y.[\gamma]\llbracket \langle\!\langle N \rangle\!\rangle \rrbracket\big)\big(\mu\beta.(\lambda x.[\gamma]\llbracket \langle\!\langle M \rangle\!\rangle \rrbracket) \\
&\quad\quad (\mu\alpha.\llbracket\, [\alpha,\beta]\langle\!\langle O \rangle\!\rangle\, \rrbracket)\big) \\
&\quad \overset{I.H.}{=}_n \mu\gamma.\big(\lambda y.[\gamma]N\big)\big(\mu\beta.(\lambda x.[\gamma]M) \\
&\quad\quad (\mu\alpha.\llbracket\, [\alpha,\beta]\langle\!\langle O \rangle\!\rangle\, \rrbracket)\big) \\
&\quad \equiv \mu\gamma.\big(\lambda y.[\gamma]N\big)\big(\mu\beta.(\lambda x.[\gamma]M) \\
&\quad\quad (\mu\alpha.\delta(\llbracket \langle\!\langle O \rangle\!\rangle \rrbracket, x.[\alpha]x, y.[\beta]y))\big) \\
&\quad \overset{I.H.}{=}_n \mu\gamma.\big(\lambda y.[\gamma]N\big)\big(\mu\beta.(\lambda x.[\gamma]M) \\
&\quad\quad (\mu\alpha.\delta(O, x.[\alpha]x, y.[\beta]y))\big) \\
&\quad \overset{(*)}{=}_n \mu\gamma.\delta(O, x.[\gamma]M, y.[\gamma]N) \\
&\quad =_{(\pi)} \mu\gamma.[\gamma]\delta(O, x.M, y.N) \\
&\quad =_{(\eta_\mu)} \delta(O, x.M, y.N)
\end{aligned}
$$

$(*)$ is shown by case analysis of M and N. If M and N are not simple forms w.r.t. x and y respectively, then we have

$$
\begin{aligned}
&\mu\gamma.\delta(O, x.[\gamma]M, y.[\gamma]N) \\
&\quad =_{(\nu)} \mu\gamma.(\lambda y.[\gamma]N)(\mu\beta.\delta(O, x.[\gamma]M, y.[\beta]y)) \\
&\quad =_{(\nu)} \mu\gamma.\big((\lambda y.[\gamma]N)(\mu\beta.(\lambda x.[\gamma]M) \\
&\quad\quad (\mu\alpha.\delta(O, x.[\alpha]x, y.[\beta]y))\big)
\end{aligned}
$$

If M is $E_n\{x\}$ and N is not a simple form w.r.t. y, then we have

$$
\begin{aligned}
&\mu\gamma.\delta(O, x.[\gamma]E_n\{x\}, y.[\gamma]N) \\
&\quad =_{(\nu)} \mu\gamma.(\lambda y.[\gamma]N) \\
&\quad\quad (\mu\beta.\delta(O, x.[\gamma]E_n\{x\}, y.[\beta]y)) \\
&\quad =_{(\zeta)} \mu\gamma.\big((\lambda y.[\gamma]N) \\
&\quad\quad (\mu\beta.[\gamma]E_n\{\mu\alpha.\delta(O, x.[\alpha]x, y.[\beta]y)\})\big) \\
&\quad =_{(\beta_\supset)} \mu\gamma.\big((\lambda y.[\gamma]N)(\mu\beta.(\lambda x.[\gamma]E_n\{x\}) \\
&\quad\quad (\mu\alpha.\delta(O, x.[\alpha]x, y.[\beta]y)))\big)
\end{aligned}
$$

The rest of the cases are shown similarly.

Case of $\delta(O, x.S, y.T)$: this case is similar to the above case.

(4) We can show this by induction on term M and statement S of the $\lambda\mu_n^{wad}$-calculus. We consider only $\mu(\alpha,\beta).S$ and $[\alpha,\beta]M$.

Case of $\mu(\alpha,\beta).S$:

$$
\begin{aligned}
&\langle\!\langle \llbracket \mu(\alpha,\beta).S \rrbracket \rangle\!\rangle \\
&\quad \equiv \langle\!\langle \mu\gamma.\llbracket S \rrbracket [{}^{[\gamma]\mathrm{inl}\{-\}}/_{[\alpha]\{-\}}, {}^{[\gamma]\mathrm{inr}\{-\}}/_{[\beta]\{-\}}] \rangle\!\rangle \\
&\quad \equiv \mu\gamma.\langle\!\langle \llbracket S \rrbracket [{}^{[\gamma]\mathrm{inl}\{-\}}/_{[\alpha]\{-\}}, {}^{[\gamma]\mathrm{inr}\{-\}}/_{[\beta]\{-\}}] \rangle\!\rangle \\
&\quad \equiv \mu\gamma.\langle\!\langle \llbracket S \rrbracket \rangle\!\rangle \\
&\quad\quad [{}^{[\gamma]\langle\!\langle \mathrm{inl}\{-\}\rangle\!\rangle}/_{[\alpha]\{-\}}, {}^{[\gamma]\langle\!\langle \mathrm{inr}\{-\}\rangle\!\rangle}/_{[\beta]\{-\}}] \\
&\quad \equiv \mu\gamma.\langle\!\langle \llbracket S \rrbracket \rangle\!\rangle [{}^{[\gamma]\mu(\alpha',\beta').[\alpha']\{-\}} \\
&\quad\quad /_{[\alpha]\{-\}}, {}^{[\gamma]\mu(\alpha'',\beta'').[\beta'']\{-\}}/_{[\beta]\{-\}}] \\
&\quad =_{\eta_\vee} \mu(\alpha_1,\beta_1).[\alpha_1,\beta_1] \\
&\quad\quad \mu\gamma.\langle\!\langle \llbracket S \rrbracket \rangle\!\rangle [{}^{[\gamma]\mu(\alpha',\beta').[\alpha']\{-\}}/_{[\alpha]\{-\}}, \\
&\quad\quad\quad {}^{[\gamma]\mu(\alpha'',\beta'').[\beta'']\{-\}}/_{[\beta]\{-\}}] \\
&\quad =_{\zeta_\vee} \mu(\alpha_1,\beta_1). \\
&\quad\quad \langle\!\langle \llbracket S \rrbracket \rangle\!\rangle [{}^{[\alpha_1,\beta_1]\mu(\alpha',\beta').[\alpha']\{-\}}/_{[\alpha]\{-\}}, \\
&\quad\quad\quad {}^{[\alpha_1,\beta_1]\mu(\alpha'',\beta'').[\beta'']\{-\}}/_{[\beta]\{-\}}] \\
&\quad =_{\beta_\vee} \mu(\alpha,\beta).\langle\!\langle \llbracket S \rrbracket \rangle\!\rangle \\
&\quad\quad [{}^{[\alpha_1]\{-\}}/_{[\alpha]\{-\}}, {}^{[\beta_1]\{-\}}/_{[\beta]\{-\}}] \\
&\quad \equiv \mu(\alpha,\beta).\langle\!\langle \llbracket S \rrbracket \rangle\!\rangle \\
&\quad \overset{I.H.}{=}_n \mu(\alpha,\beta).S
\end{aligned}
$$

Case of $[\alpha,\beta]M$:

$$
\begin{aligned}
&\langle\!\langle \llbracket [\alpha,\beta]M \rrbracket \rangle\!\rangle \equiv \langle\!\langle\, \delta(\llbracket M \rrbracket, x.[\alpha]x, y.[\beta]y)\, \rangle\!\rangle \\
&\quad \equiv \big((\lambda y.[\beta]y)(\mu\beta'.(\lambda x.[\alpha]x) \\
&\quad\quad (\mu\alpha'.[\alpha',\beta']\langle\!\langle \llbracket M \rrbracket \rangle\!\rangle))\big) \\
&\quad =_{\beta_\supset} [\beta]\mu\beta'.[\alpha]\mu\alpha'.[\alpha',\beta']\langle\!\langle \llbracket M \rrbracket \rangle\!\rangle \\
&\quad =_{\beta_\mu} [\alpha,\beta]\langle\!\langle \llbracket M \rrbracket \rangle\!\rangle \\
&\quad \overset{I.H.}{=}_n [\alpha,\beta]M \qquad \square
\end{aligned}
$$

Proof of Proposition 3 (1) We can show this by induction on the call-by-value equation of the $\lambda\mu_v^\eta$-calculus. We consider only the rules about sums, i.e., $(\beta_\vee)$, (ζ), (π), $(comp)$, $(name)$, and $(\eta_\vee)$-rules.

Case of $(\beta_\vee)$-rule :

$$
\langle\!\langle \delta(\mathrm{inl}(V), x.M, y.N) \rangle\!\rangle
$$

$$\equiv \mu\gamma.\widehat{N}\big(\mu\beta.(\lambda x.[\gamma]\langle\!\langle M\rangle\!\rangle)$$
$$(\mu\alpha.[\alpha,\beta]\langle\!\langle \mathrm{inl}(V)\rangle\!\rangle)\big)$$
$$\equiv \mu\gamma.\widehat{N}\big(\mu\beta.(\lambda x.[\gamma]\langle\!\langle M\rangle\!\rangle)$$
$$(\mu\alpha.[\alpha,\beta]\mu(\alpha',\beta').[\alpha']\langle\!\langle V\rangle\!\rangle)\big)$$
$$=_{(\beta_\vee)} \mu\gamma.\widehat{N}\big(\mu\beta.(\lambda x.[\gamma]\langle\!\langle M\rangle\!\rangle)(\mu\alpha.[\alpha]\langle\!\langle V\rangle\!\rangle)\big)$$
$$=_{(\eta_\mu)} \mu\gamma.\widehat{N}\big(\mu\beta.(\lambda x.[\gamma]\langle\!\langle M\rangle\!\rangle)\langle\!\langle V\rangle\!\rangle\big)$$
$$=_{(\zeta)} \mu\gamma.(\lambda x.[\gamma]\langle\!\langle M\rangle\!\rangle)\langle\!\langle V\rangle\!\rangle$$
$$=_{(\beta_\neg)} \mu\gamma.[\gamma]\langle\!\langle M\rangle\!\rangle[{}^{\langle\langle V\rangle\rangle}/{}_x]$$
$$\overset{(*)}{\equiv} \mu\gamma.[\gamma]\langle\!\langle M[{}^V/{}_x]\rangle\!\rangle$$
$$=_{\eta_\mu} \langle\!\langle M[{}^V/{}_x]\rangle\!\rangle$$

where $\widehat{N}$ is an abbreviation of $(\lambda y.[\gamma]\langle\!\langle N\rangle\!\rangle)$. $(*)$ comes from the claim: $\langle\!\langle M\rangle\!\rangle[{}^{\langle\langle V\rangle\rangle}/{}_x] \equiv \langle\!\langle M[{}^V/{}_x]\rangle\!\rangle$ and $\langle\!\langle S\rangle\!\rangle[{}^{\langle\langle V\rangle\rangle}/{}_x] \equiv \langle\!\langle S[{}^V/{}_x]\rangle\!\rangle$. This claim is shown by a straightforward induction on M and S. The other rules of the $(\beta_\vee)$-rule can also be shown similarly.

Case of (ζ)-rule : This case can be shown by a case analysis of the evaluation contexts. The key case is when E_e is $\delta(\{-\},x.M,y.N)$ and D_e is $\delta(\{-\},x.S,y.T)$.

Subcase of E_e is $\delta(\{-\},x.M,y.N)$:

$$\langle\!\langle \delta(\mu\alpha.S,x.M,y.N)\rangle\!\rangle$$
$$\equiv \mu\gamma.\big(\lambda y.[\gamma]\langle\!\langle N\rangle\!\rangle\big)$$
$$\big(\mu\beta'.(\lambda x.[\gamma]\langle\!\langle M\rangle\!\rangle)\mu\alpha'.[\alpha',\beta']\mu\alpha.\langle\!\langle S\rangle\!\rangle\big)$$
$$\equiv \mu\gamma.\widehat{N}\big(\mu\beta'.\widehat{M}\mu\alpha'.[\alpha',\beta']\mu\alpha.\langle\!\langle S\rangle\!\rangle\big)$$
$$=_\zeta \mu\gamma.\widehat{N}\big(\mu\beta'.\widehat{M}\mu\alpha'.\langle\!\langle S\rangle\!\rangle[{}^{[\alpha',\beta']\{-\}}/{}_{[\alpha]\{-\}}]\big)$$
$$=_\zeta \mu\gamma.\langle\!\langle S\rangle\!\rangle[{}^{[\alpha',\beta']\{-\}}/{}_{[\alpha]\{-\}}]$$
$$[{}^{\widehat{M}\{-\}}/{}_{[\alpha']\{-\}}][{}^{\widehat{N}\{-\}}/{}_{[\beta']\{-\}}]$$
$$\equiv \mu\gamma.\langle\!\langle S\rangle\!\rangle[{}^{([\alpha',\beta']\{-\})[{}^{\widehat{M}\{-\}}/{}_{[\alpha']\{-\}}]\cdot[{}^{\widehat{N}\{-\}}/{}_{[\beta']\{-\}}]}/{}_{[\alpha]\{-\}}]$$
$$\equiv \mu\gamma.\langle\!\langle S\rangle\!\rangle[{}^{\widehat{N}(\mu\beta'.\widehat{M}(\mu\alpha'.[\alpha',\beta']\{-\}))}/{}_{[\alpha]\{-\}}]$$
$$=_{\beta_\mu} \mu\gamma.\langle\!\langle S\rangle\!\rangle[{}^{[\gamma]\mu\gamma.\widehat{N}(\mu\beta'.\widehat{M}(\mu\alpha'.[\alpha',\beta']\{-\}))}/{}_{[\alpha]\{-\}}]$$
$$\overset{(*)}{\equiv} \mu\gamma.\langle\!\langle S[{}^{[\gamma]\delta(\{-\},x.M,y.N)}/{}_{[\alpha]\{-\}}]\rangle\!\rangle$$
$$\equiv \langle\!\langle \mu\gamma.S[{}^{[\gamma]\delta(\{-\},x.M,y.N)}/{}_{[\alpha]\{-\}}]\rangle\!\rangle$$

where $\widehat{M}$ and $\widehat{N}$ are abbreviations of $\lambda x.[\gamma]\langle\!\langle M\rangle\!\rangle$ and $\lambda y.[\gamma]\langle\!\langle N\rangle\!\rangle$ respectively. (*) is shown by a straightforward induction on terms and statements.

Case of (π)-rule : We can obtain $=_{(\pi)}$ from $[\alpha]\delta(O,x.M,y.N) =_v \delta(O,x.[\alpha]M,y.[\alpha]N)$ with (η_μ)-rule and (ζ)-rule in the following way. Let E be E_e or $(\lambda x.M)\{-\}$, then

$$E\{\delta(O,x.M,y.N)\}$$
$$=_{\eta_\mu} E\{\mu\alpha.[\alpha]\delta(O,x.M,y.N)\}$$
$$=_v E\{\mu\alpha.\delta(O,x.[\alpha]M,y.[\alpha]N)\}$$
$$=_\zeta \mu\beta.\delta(O,x.[\beta]E\{M\},y.[\beta]E\{N\})\}$$
$$=_v \mu\beta.[\beta]\delta(O,x.E\{M\},y.E\{N\})\}$$
$$=_{\eta_\mu} \delta(O,x.E\{M\},y.E\{N\})\}.$$

Similarly, we can obtain $D_v\{\delta(O,x.M,y.N)\} =_v \delta(O,x.D_v\{M\},y.D_v\{N\})$. Therefore, it is sufficient to prove $\langle\!\langle[\alpha]\delta(O,x.M,y.N)\rangle\!\rangle =_v \langle\!\langle\delta(O,x.[\alpha]M,y.[\alpha]N)\rangle\!\rangle$.

$$\langle\!\langle[\alpha]\delta(O,x.M,y.N)\rangle\!\rangle$$
$$\equiv [\alpha]\mu\gamma.(\lambda y.[\gamma]\langle\!\langle N\rangle\!\rangle)$$
$$(\mu\beta'.(\lambda x.[\gamma]\langle\!\langle M\rangle\!\rangle)\mu\alpha'.[\alpha',\beta']\langle\!\langle O\rangle\!\rangle)$$
$$=_{\beta_\mu} (\lambda y.[\alpha]\langle\!\langle N\rangle\!\rangle)$$
$$(\mu\beta'.(\lambda x.[\alpha]\langle\!\langle M\rangle\!\rangle)\mu\alpha'.[\alpha',\beta']\langle\!\langle O\rangle\!\rangle)$$
$$\equiv \langle\!\langle\,\delta(O,x.[\alpha]M,y.[\alpha]N)\,\rangle\!\rangle$$

Case of (*comp*)-rule : To show this case, it is sufficient to have $\langle\!\langle[\alpha]((\lambda x.M)N)\rangle\!\rangle =_v \langle\!\langle(\lambda x.[\alpha]M)N\rangle\!\rangle$ by the discussion similar to that for (π)-rule.

$$\langle\!\langle\,[\alpha]((\lambda x.M)N)\,\rangle\!\rangle =_v [\alpha]((\lambda x.\langle\!\langle M\rangle\!\rangle)\langle\!\langle N\rangle\!\rangle$$
$$=_{comp} (\lambda x.[\alpha]\langle\!\langle M\rangle\!\rangle)\langle\!\langle N\rangle\!\rangle$$
$$\equiv \langle\!\langle\,(\lambda x.[\alpha]M)N\,\rangle\!\rangle$$

Case of (*name*)-rule : We can easily show $\lambda\mu_v^{wad-} \vdash \langle\!\langle E_i\{O\}\rangle\!\rangle =_v \langle\!\langle(\lambda x.E_i\{x\})O\rangle\!\rangle$ by a case analysis of E_i. On the other hand, we have $\langle\!\langle[\alpha]O\rangle\!\rangle \equiv [\alpha]\langle\!\langle O\rangle\!\rangle =_{comp} (\lambda x.[\alpha]x)\langle\!\langle O\rangle\!\rangle \equiv \langle\!\langle\,(\lambda x.[\alpha]x)O\,\rangle\!\rangle$. Therefore we obtain $\lambda\mu_v^{wad-} \vdash \langle\!\langle E_e\{O\}\rangle\!\rangle =_v \langle\!\langle(\lambda x.E_e\{x\})O\rangle\!\rangle$, since $\lambda\mu_v \vdash E_e\{O\} =_v (\lambda x.E_e\{x\})O$ can be shown from $\lambda\mu \vdash [\alpha]O =_v (\lambda x.[\alpha]x)O$ with (ζ), (η_μ) and (*comp*)-rules as follows.

$$E_e\{O\} =_{\eta_\mu} E_e\{\mu\alpha.[\alpha]O\}$$
$$=_v E_e\{\mu\alpha.(\lambda x.[\alpha]x)O\}$$
$$=_\zeta \mu\beta.((\lambda x.[\beta]E_e\{x\})O)$$
$$=_{name} \mu\beta.[\beta]((\lambda x.E_e\{x\})O)$$
$$=_{\eta_\mu} (\lambda x.E_e\{x\})O$$

Case of $(\eta_\vee)$-rule :

$$\langle\!\langle\delta(M,x.\mathrm{inl}(x),y.\mathrm{inr}(y))\rangle\!\rangle$$
$$\equiv \mu\gamma.\big(\lambda y.[\gamma]\langle\!\langle\mathrm{inr}(y)\rangle\!\rangle\big)$$
$$\big(\mu\beta.(\lambda x.[\gamma]\langle\!\langle\mathrm{inl}(x)\rangle\!\rangle)(\mu\alpha.[\alpha,\beta]\langle\!\langle M\rangle\!\rangle)\big)$$
$$\equiv \mu\gamma.\big(\lambda y.[\gamma]R_y\big)\big(\mu\beta.(\lambda x.[\gamma]L_x)\widehat{M}\big)$$
$$=_{\eta_\vee} \mu(\alpha_1,\beta_1).[\alpha_1,\beta_1]\mu\gamma.\big(\lambda y.[\gamma]R_y\big)$$
$$\big(\mu\beta.(\lambda x.[\gamma]L_x)\widehat{M}\big)$$
$$=_{\zeta_\vee} \mu(\alpha_1,\beta_1).\big(\lambda y.[\alpha_1,\beta_1]R_y\big)$$

$$
\begin{aligned}
&\big(\mu\beta.(\lambda x.[\alpha_1,\beta_1]L_x)\widehat{M}\big)\\
&=_{\beta_\vee} \mu(\alpha_1,\beta_1).(\lambda y.[\beta_1]y)(\mu\beta.(\lambda x.[\alpha_1]x)\widehat{M})\\
&=_{name} \mu(\alpha_1,\beta_1).[\beta_1]\mu\beta.[\alpha_1]\widehat{M}\\
&\equiv \mu(\alpha_1,\beta_1).[\beta_1]\mu\beta.[\alpha_1]\mu\alpha.[\alpha,\beta]\langle\!\langle M\rangle\!\rangle\\
&=_{\beta_\mu} \mu(\alpha_1,\beta_1).[\alpha_1,\beta_1]\langle\!\langle M\rangle\!\rangle\\
&=_{\eta_\vee} \langle\!\langle M\rangle\!\rangle
\end{aligned}
$$

where $\widehat{M}$, L_x, and R_y are abbreviations of $(\mu\alpha.[\alpha,\beta]\langle\!\langle M\rangle\!\rangle)$, $\langle\!\langle \mathrm{inl}(x)\rangle\!\rangle$ and $\langle\!\langle \mathrm{inr}(y)\rangle\!\rangle$ respectively.

(2) We can show this by induction on the call-by-name equation of the $\lambda\mu_n^{wad}$-calculus. We consider only the rules about sums, i.e., $(\beta_\vee)$, (ζ), $(\eta_\vee)$, (*name*), and (*comp*)-rules.

Case of $(\beta_\vee)$-rule :

$$
\begin{aligned}
&[\![[\alpha',\beta']\mu(\alpha,\beta).S]\!]\\
&\quad\equiv \delta\big([\![\mu(\alpha,\beta).S]\!],x.[\alpha']x,y.[\beta']y\big)\\
&\quad\equiv \delta\big(\mu\gamma.[\![S]\!][{}^{[\gamma]\mathrm{inl}\{-\}}/_{[\alpha]\{-\}},{}^{[\gamma]\mathrm{inr}\{-\}}/_{[\beta]\{-\}}],\\
&\qquad x.[\alpha']x,y.[\beta']y\big)\\
&\quad=_{(\zeta)} [\![S]\!][{}^{\delta(\mathrm{inl}\{-\},x.[\alpha']x,y.[\beta']y)}/_{[\alpha]\{-\}},\\
&\qquad {}^{\delta(\mathrm{inr}\{-\},x.[\alpha']x,y.[\beta']y)}/_{[\beta]\{-\}}]\\
&\quad\overset{(*)}{=}_v [\![S]\!][{}^{[\alpha']\{-\}}/_{[\alpha]\{-\}},{}^{[\beta']\{-\}}/_{[\beta]\{-\}}]\\
&\quad\equiv [\![S[{}^{\alpha'}/_{\alpha},{}^{\beta'}/_{\beta}]]\!]
\end{aligned}
$$

$(*)$ can be shown by the following claim: $\delta(\mathrm{inl}(M),x.[\alpha]x,y.[\beta]y) =_v [\alpha]M$ for any M. We show this. If M is a value, then the claim is obtained by $(\beta_\vee)$-rule. If M is not a value, then

$$
\begin{aligned}
&\delta(\mathrm{inl}(M),x.[\alpha]x,y.[\beta]y)\\
&\quad=_{name} (\lambda z.\delta(z,x.[\alpha]x,y.[\beta]y))\mathrm{inl}(M)\\
&\quad=_{name} \big(\lambda z.\delta(z,x.[\alpha]x,y.[\beta]y)\big)\\
&\qquad((\lambda z'.\mathrm{inl}(z'))M)\\
&\quad=_{comp} \big(\lambda z'.(\lambda z.\delta(z,x.[\alpha]x,\\
&\qquad y.[\beta]y))\mathrm{inl}(z')\big)M\\
&\quad=_{\beta_\supset} (\lambda z'.\delta(\mathrm{inl}(z'),x.[\alpha]x,y.[\beta]y))M\\
&\quad=_{\beta_\vee} (\lambda z'.[\alpha]z')M =_{name} [\alpha]M
\end{aligned}
$$

Case of (ζ)-rule : to show this case, we introduce evaluation singular context E_w and singular statement context D_w of the $\lambda\mu_v^{wad-}$-calculus.

$$
\begin{aligned}
E_w &::= \{-\}N \mid V\{-\} \mid \langle V,\{-\}\rangle \mid \langle\{-\},M\rangle\\
&\qquad \mid \mathrm{fst}(\{-\}) \mid \mathrm{snd}(\{-\})\\
D_w &::= [\alpha]\{-\} \mid [\alpha,\beta]\{-\} \mid \{-\}M \mid V\{-\}
\end{aligned}
$$

It is easily shown that if we have the following claims:

$$
\begin{aligned}
&[\![E_w\{\mu\alpha.S\}]\!] =_v [\![\mu\beta.S[{}^{[\beta]E_w}/_{[\alpha]\{-\}}]]\!] \text{ and}\\
&[\![D_w\{\mu\alpha.S\}]\!] =_v [\![S[{}^{D_w\{-\}}/_{[\alpha]\{-\}}]]\!],
\end{aligned}
$$

then we can show this case. We can prove the claim by a case analysis of E_w and D_w. We consider the key cases:

- E_w is $x\{-\}$:

$$
\begin{aligned}
[\![x\mu\alpha.S]\!] &\equiv x\mu\alpha.[\![S]\!] =_{\eta_\supset} (\lambda z.xz)\mu\alpha.[\![S]\!]\\
&=_\zeta \mu\beta.[\![S]\!][{}^{[\beta](\lambda z.xz)\{-\}}/_{[\alpha]\{-\}}]\\
&=_{\eta_\supset} \mu\beta.[\![S]\!][{}^{[\beta]x\{-\}}/_{[\alpha]\{-\}}]\\
&\overset{(*)}{\equiv} \mu\beta.[\![S[{}^{[\beta]x\{-\}}/_{[\alpha]\{-\}}]]\!]
\end{aligned}
$$

$(*)$ can be shown by induction on terms and statements. The key case is proved as follows.

$$
\begin{aligned}
&[\![[\alpha]M]\!][{}^{[\beta]x\{-\}}/_{[\alpha]\{-\}}]\\
&\quad\equiv ([\alpha][\![M]\!])[{}^{[\beta]x\{-\}}/_{[\alpha]\{-\}}]\\
&\quad\equiv [\beta]x([\![M]\!][{}^{[\beta]x\{-\}}/_{[\alpha]\{-\}}])\\
&\quad\overset{I.H.}{\equiv} [\beta]x([\![M[{}^{[\beta]x\{-\}}/_{[\alpha]\{-\}}]]\!]\\
&\quad\equiv [\![([\alpha]M)[{}^{[\beta]x\{-\}}/_{[\alpha]\{-\}}]]\!]
\end{aligned}
$$

- D_w is $x\{-\}$: this can be shown in a way similar to the above case.
- D_w is $[\alpha,\beta]\{-\}$:

$$
\begin{aligned}
&[\![[\alpha,\beta]\mu\gamma.S]\!] \equiv \delta([\![\mu\gamma.S]\!],x.[\alpha]x,y.[\beta]y)\\
&\quad\equiv \delta(\mu\gamma.[\![S]\!],x.[\alpha]x,y.[\beta]y)\\
&\quad=_\zeta [\![S]\!][{}^{\delta(\{-\},x.[\alpha]x,y.[\beta]y)}/_{[\gamma]\{-\}}]\\
&\quad\overset{(*)}{\equiv} [\![S[{}^{[\alpha,\beta]\{-\}}/_{[\gamma]\{-\}}]]\!]
\end{aligned}
$$

$(*)$ is already shown in the proof of Proposition 2.

Case of $(\eta_\vee)$-rule :

$$
\begin{aligned}
&[\![\mu(\alpha,\beta).[\alpha,\beta]M]\!]\\
&\quad\equiv \mu\gamma.[\![[\alpha,\beta]M]\!]\\
&\qquad[{}^{[\gamma]\mathrm{inl}\{-\}}/_{[\alpha]\{-\}},{}^{[\gamma]\mathrm{inr}\{-\}}/_{[\beta]\{-\}}]\\
&\quad\equiv \mu\gamma.\delta([\![M]\!],x.[\alpha]x,y.[\beta]y)\\
&\qquad[{}^{[\gamma]\mathrm{inl}\{-\}}/_{[\alpha]\{-\}},{}^{[\gamma]\mathrm{inr}\{-\}}/_{[\beta]\{-\}}]\\
&\quad\equiv \mu\gamma.\delta([\![M]\!],x.[\gamma]\mathrm{inl}(x),y.[\gamma]\mathrm{inr}(y))\\
&\quad=_{(\pi)} \mu\gamma.[\gamma]\delta([\![M]\!],x.\mathrm{inl}(x),y.\mathrm{inr}(y))\\
&\quad=_{(\eta_\mu)} \delta([\![M]\!],x.\mathrm{inl}(x),y.\mathrm{inr}(y))\\
&\quad=_{(\eta_\vee)} [\![M]\!]
\end{aligned}
$$

Case of (*name*)-rule : Let D be a statement context of the $\lambda\mu_v^{wad-}$-calculus, then we can obtain (*name*)-rule of the $\lambda\mu_v^{wad-}$-calculus from $\lambda\mu_v^{wad-} \vdash (\lambda x.[\alpha]x)M =_v [\alpha]M$ with (η_μ), and (ζ)-rule in the following way.

$$
\begin{aligned}
&D\{M\} =_{(\eta_\mu)} D\{\mu\alpha.[\alpha]M\}\\
&\quad=_v D\{\mu\alpha.(\lambda x.[\alpha]x)M\} =_{(\zeta)} (\lambda x.D\{x\})M
\end{aligned}
$$

Therefore, it is sufficient to prove $(\lambda x.[\alpha]x)M]\!]$ $=_v [\![[\alpha]M]\!]$ in our call-by-value $\lambda\mu$-calculus.

$$[\![(\lambda x.[\alpha]x)M]\!] \equiv (\lambda x.[\alpha]x)[\![M]\!]$$
$$=_{(name)} [\![[\alpha]M]\!]$$

Case of $(comp)$-rule : Let D be a statement context of the $\lambda\mu_v^{wad-}$-calculus, then we can obtain $(comp)$-rule of the $\lambda\mu_v^{wad-}$-calculus from $[\alpha]((\lambda x.M)N) =_v (\lambda x.[\alpha]M)N$, (η_μ), and (ζ)-rule in the following way.

$$D\{(\lambda x.M)N\} =_{\eta_\mu} D\{\mu\alpha.[\alpha](\lambda x.M)N\}$$
$$=_v D\{\mu\alpha.(\lambda x.[\alpha]M)N\} =_\zeta \lambda x.D\{M\})N$$

Therefore, it is sufficient to prove $[\![[\alpha]((\lambda x.M)$ $N)]\!] =_v [\![\lambda x.[\alpha]N]\!]$.

$$[\![[\alpha]((\lambda x.M)N)]\!] \equiv [\alpha]((\lambda x.[\![M]\!])[\![N]\!])$$
$$=_{(comp)} (\lambda x.[\alpha][\![M]\!])[\![N]\!] \equiv [\![(\lambda x.[\alpha]M)N]\!]$$

(3) We can show this by induction on term M and statement S of the $\lambda\mu_v^\eta$-calculus. We consider $\mathrm{inl}(M)$, $\mathrm{inr}(M)$, $\delta(O,x.M,y.N)$ and $\delta(O,x.S,y.T)$.

Case of $\mathrm{inl}(M)$:

$$[\![\langle\!\langle \mathrm{inl}(M)\rangle\!\rangle]\!] \equiv [\![\mu(\alpha,\beta).[\alpha]\langle\!\langle M\rangle\!\rangle]\!]$$
$$\equiv \mu\gamma.\big([\alpha][\![\langle\!\langle M\rangle\!\rangle]\!]\big)$$
$$\big[{}^{[\gamma]\mathrm{inl}\{-\}}/_{[\alpha]\{-\}}, {}^{[\gamma]\mathrm{inr}\{-\}}/_{[\beta]\{-\}}\big]$$
$$\equiv \mu\gamma.[\gamma]\mathrm{inl}\big([\![\langle\!\langle M\rangle\!\rangle]\!]\big) \overset{I.H.}{=_v} \mu\gamma.[\gamma]\mathrm{inl}(M)$$
$$=_{\eta_\mu} \mathrm{inl}(M)$$

$\mathrm{inr}(M)$ can be shown similarly.

Case of $\delta(O,x.M,y.N)$:

$$[\![\langle\!\langle \delta(O,x.M,y.N)\rangle\!\rangle]\!]$$
$$\equiv [\![\mu\gamma.(\lambda y.[\gamma]\langle\!\langle N\rangle\!\rangle)(\mu\beta.(\lambda x.[\gamma]\langle\!\langle M\rangle\!\rangle)$$
$$(\mu\alpha.[\alpha,\beta]\langle\!\langle O\rangle\!\rangle)))]\!]$$
$$\equiv \mu\gamma.(\lambda y.[\gamma][\![\langle\!\langle N\rangle\!\rangle]\!])(\mu\beta.(\lambda x.[\gamma][\![\langle\!\langle M\rangle\!\rangle]\!])$$
$$(\mu\alpha.[\![\,[\alpha,\beta]\langle\!\langle O\rangle\!\rangle\,]\!]))$$
$$\overset{I.H.}{=_v} \mu\gamma.(\lambda y.[\gamma]N)(\mu\beta.(\lambda x.[\gamma]M)$$
$$(\mu\alpha.[\![\,[\alpha,\beta]\langle\!\langle O\rangle\!\rangle\,]\!]))$$
$$\equiv \mu\gamma.(\lambda y.[\gamma]N)(\mu\beta.(\lambda x.[\gamma]M)$$
$$(\mu\alpha.\delta([\![\langle\!\langle O\rangle\!\rangle]\!],x.[\alpha]x,y.[\beta]y)))$$
$$\overset{I.H.}{=_v} \mu\gamma.(\lambda y.[\gamma]N)(\mu\beta.(\lambda x.[\gamma]M)$$
$$(\mu\alpha.\delta(O,x.[\alpha]x,y.[\beta]y)))$$
$$=_\zeta \mu\gamma.\delta(O,x.(\lambda x.[\gamma]M)x,y.(\lambda y.[\gamma]N)y)$$
$$=_{\beta_\neg} \mu\gamma.\delta(O,x.[\gamma]M,y.[\gamma]N)$$
$$=_\pi \mu\gamma.[\gamma]\delta(O,x.M,y.N)$$
$$=_{\eta_\mu} \delta(O,x.M,y.N)$$

The other cases are shown similarly.

Case of $\delta(O,x.S,y.T)$: this case is similar to the above case.

(4) We can show this by induction on term M and statement S of the $\lambda\mu_v^{wad-}$-calculus. We consider only $\mu(\alpha,\beta).S$ and $[\alpha,\beta]M$.

Case of $\mu(\alpha,\beta).S$:

$$\langle\!\langle[\![\mu(\alpha,\beta).S]\!]\rangle\!\rangle$$
$$\equiv \langle\!\langle \mu\gamma.[\![S]\!]\big[{}^{[\gamma]\mathrm{inl}\{-\}}/_{[\alpha]\{-\}}, {}^{[\gamma]\mathrm{inr}\{-\}}/_{[\beta]\{-\}}\big]\rangle\!\rangle$$
$$\equiv \mu\gamma.\langle\!\langle[\![S]\!]\big[{}^{[\gamma]\mathrm{inl}\{-\}}/_{[\alpha]\{-\}}, {}^{[\gamma]\mathrm{inr}\{-\}}/_{[\beta]\{-\}}\big]\rangle\!\rangle$$
$$\equiv \mu\gamma.\langle\!\langle[\![S]\!]\rangle\!\rangle$$
$$\big[{}^{[\gamma][\![\mathrm{inl}\{-\}]\!]}/_{[\alpha]\{-\}}, {}^{[\gamma][\![\mathrm{inr}\{-\}]\!]}/_{[\beta]\{-\}}\big]$$
$$=_{\eta_\vee} \mu(\alpha_1,\beta_1).[\alpha_1,\beta_1]$$
$$\mu\gamma.\langle\!\langle[\![S]\!]\rangle\!\rangle$$
$$\big[{}^{[\gamma][\![\mathrm{inl}\{-\}]\!]}/_{[\alpha]\{-\}}, {}^{[\gamma][\![\mathrm{inr}\{-\}]\!]}/_{[\beta]\{-\}}\big]$$
$$=_{\zeta_\vee} \mu(\alpha_1,\beta_1).\langle\!\langle[\![S]\!]\rangle\!\rangle$$
$$\big[{}^{[\alpha_1,\beta_1][\![\mathrm{inl}\{-\}]\!]}/_{[\alpha]\{-\}}, {}^{[\alpha_1,\beta_1][\![\mathrm{inr}\{-\}]\!]}/_{[\beta]\{-\}}\big]$$
$$=_{\beta_\vee} \mu(\alpha,\beta).\langle\!\langle[\![S]\!]\rangle\!\rangle$$
$$\big[{}^{[\alpha_1]\{-\}}/_{[\alpha]\{-\}}, {}^{[\beta_1]\{-\}}/_{[\beta]\{-\}}\big]$$
$$\equiv \mu(\alpha,\beta).\langle\!\langle[\![S]\!]\rangle\!\rangle$$
$$\overset{I.H.}{=_v} \mu(\alpha,\beta).S$$

Case of $[\alpha,\beta]M$:

$$\langle\!\langle[\![[\alpha,\beta]M]\!]\rangle\!\rangle \equiv \langle\!\langle\delta([\![M]\!],x.[\alpha]x,y.[\beta]y)\rangle\!\rangle$$
$$\equiv \big((\lambda y.[\beta]y)(\mu\beta'.(\lambda x.[\alpha]x)$$
$$(\mu\alpha'.[\alpha',\beta']\langle\!\langle[\![M]\!]\rangle\!\rangle))\big)$$
$$=_{name} [\beta]\mu\beta'.[\alpha]\mu\alpha'.[\alpha',\beta']\langle\!\langle[\![M]\!]\rangle\!\rangle$$
$$=_{\beta_\mu} [\alpha,\beta]\langle\!\langle[\![M]\!]\rangle\!\rangle$$
$$\overset{I.H.}{=_v} [\alpha,\beta]M \qquad \Box$$

Proof of Lemma 10 (1) is proved by an induction on M and S. We give the sums, i.e., $\mathrm{inl}(O)$, $\mathrm{inr}(O)$, $\delta(O,x.M,y.N)$, and $\delta(O,x.S,y.T)$.

Case of $\mathrm{inl}(O)$:

$$\langle\!\langle\mathrm{inl}(O)\rangle\!\rangle^* \equiv (\mu(\alpha,\beta).[\alpha]\langle\!\langle O\rangle\!\rangle)^*$$
$$\equiv \big(\langle(\langle(\langle\!\langle O\rangle\!\rangle^* \bullet \alpha).\beta\rangle\mathrm{inr}\bullet\gamma).\alpha\rangle\mathrm{inr}\bullet\gamma\big).\gamma$$
$$\overset{I.H.}{=^n} \big(\langle(\langle(O^\circledast \bullet \alpha).\beta\rangle\mathrm{inr}\bullet\gamma).\alpha\rangle\mathrm{inr}\bullet\gamma\big).\gamma$$
$$=^n_{\eta_\vee} \big(\langle(\langle(O^\circledast \bullet \alpha).\beta\rangle\mathrm{inr}\bullet\gamma).\alpha\rangle\mathrm{inr}\bullet\widehat{\gamma}\big).\gamma$$
$$=^n_{\beta_\vee} \big(\langle(\langle(O^\circledast \bullet \alpha).\beta\rangle\mathrm{inr}\bullet\gamma).\alpha \bullet y.(\langle y\rangle\mathrm{inr}$$
$$\bullet\gamma)\big).\gamma$$
$$=^n_{\eta_\vee} \big(\langle(\langle(O^\circledast \bullet \alpha).\beta\rangle\mathrm{inr}\bullet\widehat{\gamma}).\alpha \bullet y.(\langle y\rangle\mathrm{inr}$$
$$\bullet\gamma)\big).\gamma$$
$$=^n_{\beta_\vee} \big((O^\circledast \bullet \alpha).\beta \bullet x.(\langle x\rangle\mathrm{inl}\bullet\gamma)).\alpha$$
$$\bullet y.(\langle y\rangle\mathrm{inr}\bullet\gamma)\big).\gamma$$
$$=^n_{(\beta_R)} \big(O^\circledast \bullet y.(\langle y\rangle\mathrm{inr}\bullet\gamma)\big).\gamma$$
$$=^n_{\beta_L} (\langle O^\circledast\rangle\mathrm{inr}\bullet\gamma).\gamma$$
$$=^n_{\eta_L} \langle O^\circledast\rangle\mathrm{inr}$$

where $\widehat{\gamma}$ is an abbreviation of $[x.(\langle x\rangle\mathrm{inl}\ \bullet$

$\gamma), y.(\langle y\rangle\mathrm{inr}\bullet\gamma)]$. Note that these equations are the $DC^{\eta=}$ equation, that is, Wadler's system (2005).

Case of $\mathrm{inr}(O)$: this case is proved in a way similar to the above case.

Case of $\delta(O, x.M, y.N)$:

$$\begin{aligned}
&\langle\!\langle \delta(O,x.M,y.N)\rangle\!\rangle^*\\
&\quad\equiv \big(\mu\gamma.(\lambda y.[\gamma]\langle\!\langle N\rangle\!\rangle)(\mu\beta.(\lambda x.[\gamma]\langle\!\langle M\rangle\!\rangle)\\
&\qquad \mu\alpha.[\alpha,\beta]\langle\!\langle O\rangle\!\rangle)\big)^*\\
&\quad\equiv \big([\widehat{N}]\mathrm{not}\bullet\mathrm{not}\langle([\widehat{M}]\mathrm{not}\bullet\mathrm{not}\langle(\langle\!\langle O\rangle\!\rangle^*\\
&\qquad \bullet[\alpha,\beta]).\alpha\rangle).\beta\rangle\big).\gamma\\
&\quad=^n_{\beta_\neg} \big(((\langle\!\langle O\rangle\!\rangle^*\bullet[\alpha,\beta]).\alpha\bullet\widehat{M}).\beta\bullet\widehat{N}\big).\gamma\\
&\quad\overset{I.H.}{=^n} \big(((O^\circledast\bullet[\alpha,\beta]).\alpha\bullet x.(M^\circledast\bullet\gamma)).\beta\\
&\qquad \bullet y.(N^\circledast\bullet\gamma)\big).\gamma\\
&\quad=^n_{(name)} \big((O^\circledast\bullet[x.(M^\circledast\bullet\gamma),\beta]).\beta\\
&\qquad \bullet y.(N^\circledast\bullet\gamma)\big).\gamma\\
&\quad=^n_{(name)} \big(O^\circledast\bullet[x.(M^\circledast\bullet\gamma), y.(N^\circledast\bullet\gamma)]\big).\gamma\\
&\quad\equiv (\delta(O,x.M,y.N))^\circledast
\end{aligned}$$

where $\widehat{M}$ and $\widehat{N}$ are abbreviations of $x.(\langle\!\langle M\rangle\!\rangle^*\bullet\gamma)$ and $y.(\langle\!\langle N\rangle\!\rangle^*\bullet\gamma)$ respectively.

Case of $\delta(O, x.S, y.T)$: this case is proved in a way similar to the above case.

(2) is also proved by an induction on M and S. The key cases are terms and statements for sums, and these cases are shown in a way similar to (1). □

Proof of Lemma 12 We prove (1), (2), and (3) by a simultaneous induction on M, S. If (1) is shown for some term M, (2) of M is easily shown by: $M^\circledast \longrightarrow^n_{\eta_R} (M^\circledast\bullet\alpha).\alpha \overset{(1)}{\longrightarrow^{n*}} (M :_n \alpha).\alpha \equiv M^\sharp$. Therefore we prove (1) and (3).

Case of x : this case is immediate.

Case of $\lambda x.M$, $\langle M, N\rangle$, $\mathrm{inl}(O)$, and $\mathrm{inr}(O)$: these cases are easily shown by the induction hypothesis of (2).

Case of MN (MN is a term, M is not a λ-abstraction) : this case is also easily shown by the induction hypotheses of (1) and (2).

Case of $\lambda x.S$: this case is easily shown by the induction hypothesis of (3).

Case of MN (MN is a statement, M is not a λ-abstraction) : this case is also easily shown by the induction hypotheses of (1) and (2).

Case of $\delta(O, x.S, y.T)$: this case is also easily shown by the induction hypotheses of (2) and (3).

Case of $[\alpha]M$: this case is also easily shown by the induction hypothesis of (1).

Case of $(\lambda x.M)N$:

$$\begin{aligned}
&((\lambda x.M)N)^\circledast\bullet P\\
&\quad\equiv ((\lambda x.M^\circledast)\bullet(N^\circledast @\alpha)).\alpha\bullet P\\
&\quad\longrightarrow^n_{(\beta_L)} (\lambda x.M^\circledast)\bullet(N^\circledast @P)\\
&\quad\longrightarrow^n_{(\beta_\supset)} N^\circledast\bullet x.(M^\circledast\bullet P)\\
&\quad\overset{I.H.(1)}{\longrightarrow^{n*}} N^\sharp\bullet x.(M^\circledast\bullet P)\\
&\quad\overset{I.H.(2)}{\longrightarrow^{n*}} N^\sharp\bullet x.(M :_n P)\equiv((\lambda x.M)N) :_n P
\end{aligned}$$

Case of $\mathrm{fst}(O)$:

$$\begin{aligned}
&\mathrm{fst}(O)^\circledast\bullet P\equiv(O^\circledast\bullet\mathrm{fst}[\alpha]).\alpha\bullet P\\
&\qquad\longrightarrow^n_{(\beta_L)} O^\circledast\bullet\mathrm{fst}[P]\overset{I.H.(1)}{\longrightarrow^{n*}} O :_n \mathrm{fst}[P]\\
&\qquad\equiv\mathrm{fst}(O) :_n P
\end{aligned}$$

Case of $\mathrm{snd}(O)$: this case is shown in a way similar to the above case.

Case of $\delta(O, x.M, y.N)$:

$$\begin{aligned}
&\delta(O,x.M,y.N)^\circledast\bullet P\\
&\quad\equiv(O^\circledast\bullet[x.(M^\circledast\bullet\alpha), y.(N^\circledast\bullet\alpha)]).\alpha\bullet P\\
&\quad\longrightarrow^n_{(\beta_L)} O^\circledast\bullet[x.(M^\circledast\bullet P), y.(N^\circledast\bullet P)]\\
&\quad\overset{I.H.(1)}{\longrightarrow^{n*}} O^\circledast\bullet[x.(M :_n P), y.(N :_n P)]\\
&\quad\overset{I.H.(2)}{\longrightarrow^{n*}} O^\sharp\bullet[x.(M :_n P), y.(N :_n P)]\\
&\quad\equiv\delta(O,x.M,y.N) :_n P
\end{aligned}$$

Case of $\mu\alpha.S$:

$$\begin{aligned}
&(\mu\alpha.S)^\circledast\bullet P\equiv S^\circledast.\alpha\bullet P\overset{I.H.(3)}{\longrightarrow^{n*}} S^\sharp.\alpha\bullet P\\
&\qquad\longrightarrow^n_{(\beta_L)} S^\sharp[{}^P/_\alpha]\equiv\mu\alpha.S :_n P
\end{aligned}$$

Case of $(\lambda x.S)N$:

$$\begin{aligned}
&((\lambda x.S)N)^\circledast\\
&\quad\equiv[x.S^\circledast]\mathrm{not}\bullet\mathrm{not}\langle N^\circledast\rangle\longrightarrow^n_{(\beta_\neg)} N^\circledast\bullet x.S^\circledast\\
&\quad\overset{I.H.(2),(3)}{\longrightarrow^{n*}} N^\sharp\bullet x.S^\sharp\equiv((\lambda x.S)N)^\sharp
\end{aligned}$$
□

Proof of Lemma 14 (1) we can prove this claim by a straightforward induction on M and S. The key case is :

$$\begin{aligned}
&(x :_n P)[{}^{N^\sharp}/_x]\\
&\quad\equiv(x\bullet P)[{}^{N^\sharp}/_x]\equiv N^\sharp\bullet P[{}^{N^\sharp}/_x]\\
&\quad\equiv(N :_n \alpha).\alpha\bullet P[{}^{N^\sharp}/_x]\longrightarrow^n N :_n P[{}^{N^\sharp}/_x]
\end{aligned}$$

(2)

$$\begin{aligned}
&M\bullet[x.(x\bullet P), y.(y\bullet Q)]\\
&\quad\longrightarrow^n_{(name)} (M\bullet[x.(x\bullet P),\beta]).\beta\bullet y.(y\bullet Q)\\
&\quad\longrightarrow^n_{(\beta_L)} (M\bullet[x.(x\bullet P),\beta]).\beta\bullet Q\\
&\quad\longrightarrow^n_{(\beta_R)} M\bullet[x.(x\bullet P), Q]\\
&\quad\longrightarrow^n_{(name)} (M\bullet[\alpha, Q]).\alpha\bullet x.(x\bullet P)
\end{aligned}$$

$$\longrightarrow^n_{(\beta_L)} (M \bullet [\alpha, Q]).\alpha \bullet P$$
$$\longrightarrow^n_{(\beta_R)} M \bullet [P, Q]$$

(3) this claim can be shown by an induction on term O.

Cases of x, $\lambda x.M$, $\lambda x.S$, $\mu\alpha.S$, $\langle M, N\rangle$, $\mathrm{inl}(M)$, and $\mathrm{inr}(M)$: these are easily shown by the definition.

Case of MN (MN is a term, M is not a λ-abstraction) :

$$MN :_n K$$
$$\equiv M :_n N^\sharp @K \stackrel{I.H.}{\longrightarrow}{}^{n*} M^\sharp \bullet (N^\sharp @K)$$
$$\longrightarrow^n_{(name)} (M^\sharp \bullet (N^\sharp @\alpha)).\alpha \bullet K$$
$$\longrightarrow^n (M :_n N^\sharp @\alpha).\alpha \bullet K$$
$$\equiv (MN :_n \alpha).\alpha \bullet K$$
$$\equiv (MN)^\sharp \bullet K$$

Case of $(\lambda x.M)N$:

$$(\lambda x.M)N :_n K \equiv (N^\sharp \bullet x.(M :_n \alpha)).\alpha \bullet K$$
$$\equiv ((\lambda x.M)N :_n \alpha).\alpha \bullet K$$
$$\equiv ((\lambda x.M)N)^\sharp \bullet K$$

Case of $\mathrm{fst}(O)$:

$$\mathrm{fst}(O) :_n K \equiv O :_n \mathrm{fst}[K] \stackrel{I.H.}{\longrightarrow}{}^{n*} O^\sharp \bullet \mathrm{fst}[K]$$
$$\longrightarrow^n_{name} (O^\sharp \bullet \mathrm{fst}[\alpha]).\alpha \bullet K$$
$$\longrightarrow^n (M :_n \mathrm{fst}[\alpha]).\alpha \bullet K$$
$$\equiv (\mathrm{fst}(O) :_n \alpha).\alpha \bullet K \equiv \mathrm{fst}(O)^\sharp \bullet K$$

Case of $\mathrm{snd}(O)$: this case is shown in a way similar to the above case.

Case of $\delta(O, x.M, y.N)$:

$$\delta(O, x.M, y.N) :_n K$$
$$\equiv (O^\sharp, [x.(M :_n \alpha), y.(N :_n \alpha)]).\alpha \bullet K$$
$$\equiv (\delta(O, x.M, y.N) :_n \alpha).\alpha \bullet K$$
$$\equiv \delta(O, x.M, y.N)^\sharp \bullet K$$

(4) If M is not a λ-abstraction, then the claim is immediately shown. We consider the remaining cases.

$$(\lambda x.M) : (N^\sharp @P) \equiv (\lambda x.M^\sharp) \bullet (N^\sharp @P)$$
$$\longrightarrow^n_{\beta_\supset} N^\sharp \bullet x.(M^\sharp \bullet P)$$
$$\longrightarrow^n_{\beta_L} N^\sharp \bullet x.(M :_n P) \equiv (\lambda x.M)N :_n P$$
$$(\lambda x.S) : \mathrm{not}\langle N^\sharp\rangle \equiv [x.S^\sharp]\mathrm{not} \bullet \mathrm{not}\langle N^\sharp\rangle$$
$$\longrightarrow^n_{\beta_\neg} N^\sharp \bullet x.S^\sharp \equiv ((\lambda x.S)N)^\sharp \qquad \square$$

Proof of Lemma 15 (1) is proved by an induction on E_n and D_n. For E_nN, we can prove the claim by the induction hypothesis, since $E_n\{M\}$ is not a λ-abstraction by the assumption of M. We now consider $\delta(E_n, x.E'_n\{x\}, y.E''_n\{y\})$ and $\delta(E_n, x.D_n\{x\}, y.D'_n\{y\})$.

Case of $\delta(E_n, x.E'_n\{x\}, y.E''_n\{y\})$:

$$\delta(E_n\{M\}, x.E'_n\{x\}, y.E''_n\{y\}) :_n P$$
$$\equiv E_n\{M\}^\sharp \bullet [x.(E'_n\{x\} :_n P),$$
$$y.(E''_n\{y\} :_n P)]$$
$$\stackrel{I.H.}{\longrightarrow}{}^{n*} E_n\{M\}^\sharp \bullet [x.(x \bullet \Phi(E'_n, P)),$$
$$y.(y \bullet \Phi(E''_n, P))]$$
$$\stackrel{Lem\ 14(2)}{\longrightarrow}{}^{n*} E_n\{M\}^\sharp \bullet [\Phi(E'_n, P), \Phi(E''_n, P)]$$
$$\equiv (E_n\{M\} :_n \alpha).\alpha \bullet [\Phi(E'_n, P), \Phi(E''_n, P)]$$
$$\longrightarrow^n (E_n\{M\} :_n [\Phi(E'_n, P), \Phi(E''_n, P)]$$
$$\stackrel{I.H.}{\longrightarrow}{}^{n*} M :_n \Phi(E_n, [\Phi(E'_n, P), \Phi(E''_n, P)])$$
$$\equiv M :_n \Phi(\delta(E_n, x.E'_n\{x\}, y.E''_n\{y\}), P)$$

Case of $\delta(E_n, x.D_n\{x\}, y.D'_n\{y\})$: this case is proved in a way similar to the above case.

The other cases are easily shown using the induction hypothesis.

(2) is proved by an induction on E_n and D_n. If E_n is E_nN, then

$$M :_n \Phi(E_nN, P) \equiv M :_n \Phi(E_n, N^\sharp @P)$$
$$\stackrel{I.H.}{\longrightarrow}{}^{n*} E_n\{M\} :_n (N^\sharp @P)$$
$$\stackrel{Lem\ 14(4)}{\longrightarrow}{}^{n*} E_n\{M\}N :_n P$$

If D_n is E_nN, this case is also proved by the induction hypothesis and Lemma 14(4). If E_n is $\delta(E_n, x.E'_n\{x\}, y.E''_n\{y\})$, then

$$M :_n \Phi(\delta(E_n, x.E'_n\{x\}, y.E''_n\{y\}), P)$$
$$\equiv M :_n \Phi(E_n, [\Phi(E'_n, P), \Phi(E''_n, P)])$$
$$\stackrel{I.H.}{\longrightarrow}{}^{n*} E_n\{M\} :_n [\Phi(E'_n, P), \Phi(E''_n, P)]$$
$$\longrightarrow^{n*}_{\eta_L} E_n\{M\} :_n [x.(x \bullet \Phi(E'_n, P)),$$
$$y.(y \bullet \Phi(E''_n, P))]$$
$$\stackrel{I.H.}{\longrightarrow}{}^{n*} E_n\{M\} :_n [x.(E'_n\{x\} :_n P),$$
$$y.(E''_n\{y\} :_n P)]$$
$$\stackrel{Lem\ 14(3)}{\longrightarrow}{}^{n*} (E_n\{M\})^\sharp \bullet [x.(E'_n\{x\} :_n P),$$
$$y.(E''_n\{y\} :_n P)]$$
$$\equiv \delta(E_n\{M\}, x.E'_n\{x\}, y.E''_n\{y\}) :_n P$$

If D_n is $\delta(E_n, x.D_n\{x\}, y.D'_n\{y\})$, this case can be shown in a way similar to the above case. The remaining cases are proved easily using the the induction hypothesis.

(3) is proved by an induction on M and S. The key case is $[\alpha]M$.

$$([\alpha]M)^\sharp[{}^{\Phi(D_n)}/_\alpha] \equiv (M :_n \alpha)[{}^{\Phi(D_n)}/_\alpha]$$
$$\stackrel{I.H.}{\longrightarrow}{}^{n*} (M[{}^{D_n\{-\}}/_{[\alpha]\{-\}}] :_n \Phi(D_n))$$

$$\overset{(2)}{\longrightarrow}^{n*} \left(D_n\{M[^{D_n\{-\}}/_{[\alpha]\{-\}}]\}\right)^\sharp$$
$$\equiv \left(([\alpha]M)[^{D_n\{-\}}/_{[\alpha]\{-\}}]\right)^\sharp$$

The other cases are proved easily using the the induction hypothesis. □

Proof of Theorem 16 The claims are proved by simultaneous induction on $\longrightarrow_n$.

- Base cases are shown as follows.

$(\beta_\supset)$: $((\lambda x.M)N :_n P) \equiv N^\sharp \bullet x.(M :_n P)$
$\longrightarrow^n_{(\beta_L)} (M :_n P)[^{N^\sharp}/_x]$
$\overset{Lem\ 14(1)}{\longrightarrow}^{n*} (M[^N/_x] :_n P)$

$(\beta_\&)$: $(\mathrm{fst}\langle M, N\rangle :_n P) \equiv \langle M, N\rangle :_n \mathrm{fst}[P]$
$\equiv \langle M^\sharp, N^\sharp\rangle \bullet \mathrm{fst}[P] \longrightarrow^n_{(\beta_\vee)} M^\sharp \bullet P$
$\longrightarrow^n_{(\beta_R)} (M :_n P)$

The other rule of $(\beta_\&)$ can be proved similarly.

$(\beta_\vee)$: $(\delta(\mathrm{inl}(O), x.E_n\{x\}, y.E'_n\{y\}) :_n P)$
$\overset{Lem\ 15(1)}{\longrightarrow}^{n*} (\mathrm{inl}(O) :_n \Phi(\delta(\{-\}, x.E_n\{x\}, y.E'_n\{y\}), P))$
$\equiv (\mathrm{inl}(O) :_n \Phi(\{-\}, [\Phi(E_n, P), \Phi(E'_n, P)]))$
$\equiv (\mathrm{inl}(O) :_n [\Phi(E_n, P), \Phi(E'_n, P)])$
$\equiv (\langle O^\sharp\rangle\mathrm{inl} \bullet [\Phi(E_n, P), \Phi(E'_n, P)])$
$\longrightarrow^n_{(\beta_\vee)} O^\sharp \bullet \Phi(E_n, P)$
$\longrightarrow^n_{(\beta_R)} O :_n \Phi(E_n, P)$
$\overset{Lem\ 15(2)}{\longrightarrow}^{n*} (E_n\{O\} :_n P)$

The other rules of $(\beta_\vee)$ can be proved similarly.

$(\beta_\neg)$: $((\lambda x.S)N)^\sharp \equiv N^\sharp \bullet x.S^\sharp \longrightarrow^n_{(\beta_L)} S^\sharp[^{N^\sharp}/_x]$
$\overset{Lem\ 14(1)}{\longrightarrow}^{n*} (S[^N/_x])^\sharp$

(ζ) : $(E_n\{\mu\alpha.S\} :_n P)$
$\overset{Lem\ 15(1)}{\longrightarrow}^{n*} (\mu\alpha.S :_n \Phi(E_n, P))$
$\equiv S^\sharp[^{\Phi(E_n,P)}/_\alpha] \equiv S^\sharp[^{\Phi(E_n,\beta)}/_\alpha][^P/_\beta]$
$\equiv S^\sharp[^{\Phi([\beta]E_n)}/_\alpha][^P/_\beta]$
$\overset{Lem\ 15(3)}{\longrightarrow}^{n*} (S[^{[\beta]E_n\{-\}}/_{[\alpha]\{-\}}])^\sharp[^P/_\beta]$
$\equiv (\mu\beta.S[^{[\beta]E_n\{-\}}/_{[\alpha]\{-\}}] :_n P)$

The other rule of (ζ) can be proved similarly.

(η_μ) : $(\mu\alpha.[\alpha]M :_n P) \equiv ([\alpha]M)^\sharp[^P/_\alpha]$
$\equiv (M :_n \alpha)[^P/_\alpha] \equiv M :_n P$

(π) : $(E_n\{\delta(O, x.M, y.N)\} :_n P)$
$\overset{Lem\ 15(1)}{\longrightarrow}^{n*} (\delta(O, x.M, y.N) :_n \Phi(E_n, P))$
$\equiv (O^\sharp \bullet [x.(M :_n \Phi(E_n, P)), y.(N :_n \Phi(E_n, P))])$
$\overset{Lem\ 15(2)}{\longrightarrow}^{n*} (O^\sharp \bullet [x.(E_n\{M\} :_n P), y.(E_n\{N\} :_n P)])$
$\equiv (\delta(O, x.E_n\{M\}, y.E_n\{N\}) :_n P)$

The other rules of (π) can be proved similarly.

(ν) : Let T not be a simple form. Then

$(\delta(O, x.S, y.T))^\sharp \equiv O^\sharp \bullet [x.S^\sharp, y.T^\sharp]$
$\longrightarrow^n_{(name)} (O^\sharp \bullet [x.S^\sharp, \beta]).\beta \bullet y.T^\sharp$
$\longrightarrow^n_{(\eta_L)} (O^\sharp \bullet [x.S^\sharp, y'.(y' \bullet \beta)]).\beta \bullet y.T^\sharp$
$\equiv \delta(O, x.S, y'.[\beta]y')^\sharp.\beta \bullet y.T^\sharp$
$\equiv ((\lambda y.T)\mu\beta.\delta(O, x.S, y'.[\beta]y'))^\sharp$.

The other rule of (ν) can be proved similarly.

- Induction cases of (1) and (2).

We can easily show these cases by the induction hypothesis. We consider the less than obvious case: $ON \longrightarrow_n (\lambda x.M)N$ is obtained from $O \longrightarrow_n \lambda x.M$ and O is not a λ-abstraction.

$(ON :_n P) \equiv (O :_n (N^\sharp @ P))$
$\overset{I.H.}{\longrightarrow}^{n*} (\lambda x.M :_n (N^\sharp @ P))$
$\equiv \lambda x.M^\sharp \bullet (N^\sharp @ P)$
$\longrightarrow^n_{(\beta_\supset)} N^\sharp \bullet x.(M^\sharp \bullet P)$
$\longrightarrow^n_{(\beta_R)} N^\sharp \bullet x.(M :_n P)$
$\equiv ((\lambda x.M)N :_n P)$ □

Proof of Lemma 20 (1) Since M is not a λ-abstraction, we can immediately show $(D_v\{M\} :_v K) \equiv (M :_v \Psi(D_v))$ by the definition. If E_v is an elimination context or $(\lambda x.M)\{-\}$, then we have $(E_v\{M\} :_v K) \equiv (M :_v \Psi(E_v, K))$. For the introduction contexts, the claim is proved by a case analysis of E_v.

Case of $\langle\{-\}, N\rangle$:

$\langle M, N\rangle :_v K \equiv \langle M^\dagger, N^\dagger\rangle \bullet K$
$\longrightarrow^v_{(name)} M^\dagger \bullet x.(\langle x, N^\dagger\rangle \bullet K)$
$\longrightarrow^v_{(\beta_R)} (M :_v x.(\langle x, N^\dagger\rangle \bullet K))$
$\equiv (M :_v \Psi(\langle\{-\}, N\rangle, K))$

Case of $\langle V, \{-\}\rangle$:

$\langle V, M\rangle :_v K \equiv \langle V^\dagger, M^\dagger\rangle \bullet K$
$\longrightarrow^v_{(name)} V^\dagger \bullet x.(\langle x, M^\dagger\rangle \bullet K)$
$\longrightarrow^v_{(\beta_R)} V :_v x.(\langle x, M^\dagger\rangle \bullet K)$
$\overset{Lem\ 19(2)}{\longrightarrow}^{v} V^v \bullet x.(\langle x, M^\dagger\rangle \bullet K)$
$\longrightarrow^v_{(\beta_L)} \langle V^v, M^\dagger\rangle \bullet K$
$\longrightarrow^v_{(name)} M^\dagger \bullet y.(\langle V^v, y\rangle \bullet K)$
$\longrightarrow^v_{(\beta_R)} (M :_v y.(\langle V^v, y\rangle \bullet K))$
$\equiv (M :_v \Psi(\langle V, \{-\}\rangle, K))$

Case of $\mathrm{inl}(-)$:

$$\begin{array}{l}\mathrm{inl}(M) :_v K \equiv \langle M^\dagger\rangle\mathrm{inl}\bullet K\\ \quad\longrightarrow^v_{(name)} M^\dagger \bullet x.(\langle x\rangle\mathrm{inl}\bullet K)\\ \quad\longrightarrow^v_{(\beta_R)} M :_v x.(\langle x\rangle\mathrm{inl}\bullet K)\\ \quad\equiv M :_v \Psi(\mathrm{inl}(-), K)\end{array}$$

Case of $\mathrm{inr}(-)$: this case is proved in a way similar to the above case.

(2) can be shown by a case analysis of E and D_v. We give the key case of D_v as follows:

$$\begin{array}{l}\lambda x.S :_v \Psi(\{-\}N) \equiv \lambda x.S :_v \mathrm{not}\langle N^\dagger\rangle\\ \quad\equiv [x.S^\dagger]\mathrm{not}\bullet\mathrm{not}\langle N^\dagger\rangle \longrightarrow^v_{(\beta_\neg)} N^\dagger \bullet x.S^\dagger\\ \quad\longrightarrow^v_{(\beta_R)} N :_v x.S^\dagger \equiv ((\lambda x.S)N)^\dagger\end{array}$$

For E, the key case is: M is $\lambda x.M$ and E is $\{-\}N$. This case is shown in a way similar to the key case of D_v.

(3) can be shown by a case analysis of E_i.

Case of $\langle\{-\}, N\rangle$:

$$\begin{array}{l}(M :_v \Psi(\langle\{-\}, N\rangle, K))\\ \quad\equiv M :_v x.(\langle x, N^\dagger\rangle \bullet K)\\ \quad\longrightarrow^v_{(\eta_R)} M :_v x.(\langle x^\dagger, N^\dagger\rangle \bullet K)\\ \quad\equiv M :_v x.(\langle x, N\rangle :_v K)\\ \quad\equiv (\lambda x.\langle x, N\rangle)M :_v K\end{array}$$

Case of $\langle V, \{-\}\rangle$:

$$\begin{array}{l}M :_v \Psi(\langle V, \{-\}\rangle, K)\\ \quad\equiv (M :_v y.(\langle V^v, y\rangle \bullet K))\\ \quad\overset{Lem\ 19(1)}{\longrightarrow^v} (M :_v y.(\langle V^\dagger, y\rangle \bullet K))\\ \quad\longrightarrow^v_{(\eta_R)} (M :_v y.(\langle V^\dagger, y^\dagger\rangle \bullet K))\\ \quad\equiv (M :_v y.(\langle V, y\rangle :_v K))\\ \quad\equiv (\lambda y.\langle V, y\rangle)M :_v K\end{array}$$

Case of $\mathrm{inl}(-)$:

$$\begin{array}{l}M :_v \Psi(\mathrm{inl}(-), K) \equiv M :_v x.(\langle x\rangle\mathrm{inl}\bullet K)\\ \quad\equiv M :_v x.(\mathrm{inl}(x) :_v K)\\ \quad\equiv (\lambda x.\mathrm{inl}(x))M :_v K\end{array}$$

Case of $\mathrm{inr}(-)$: this case is proved in a way similar to the above case.

(4) can be shown by an induction on M and S. The key case is:

$$\begin{array}{l}([\alpha]M)[{}^{\Psi(E,\beta)}/_\alpha] \equiv (M :_v \alpha)[{}^{\Psi(E,\beta)}/_\alpha]\\ \quad\overset{I.H.}{\longrightarrow^{v*}} M[{}^{[\beta]E\{-\}}/_{[\alpha]\{-\}}] :_v \Psi(E,\beta)\\ \quad\overset{(2)}{\longrightarrow^{v*}} E\{M[{}^{[\beta]E\{-\}}/_{[\alpha]\{-\}}]\} :_v \beta\\ \quad\equiv ([\beta]E\{M[{}^{[\beta]E\{-\}}/_{[\alpha]\{-\}}]\})^\dagger\\ \quad\equiv (([\alpha]M)[{}^{[\beta]E\{-\}}/_{[\alpha]\{-\}}])^\dagger\end{array}$$

(5) can be shown by an induction on M and S. The key case is:

$$\begin{array}{l}([\alpha]M)[{}^{\Psi(D_v)}/_\alpha] \equiv (M :_v \alpha)[{}^{\Psi(D_v)}/_\alpha]\\ \quad\overset{I.H.}{\longrightarrow^{v*}} M[{}^{D_v\{-\}}/_{[\alpha]\{-\}}] :_v \Psi(D_v)\\ \quad\overset{(2)}{\longrightarrow^{v*}} (D_v\{M[{}^{D_v\{-\}}/_{[\alpha]\{-\}}]\})^\dagger\\ \quad\equiv (([\alpha]M)[{}^{D_v\{-\}}/_{[\alpha]\{-\}}])^\dagger\end{array}$$

□

Proof of Theorem 21 (1) and (2) are proved by simultaneous induction on $\longrightarrow_v$. Base cases are shown as follows.

$(\beta_\supset)$: $(\lambda x.M)V :_v K \equiv V :_v x.(M :_v K)$

$$\begin{array}{l}\quad\overset{Lem\ 19(2)}{\longrightarrow^{v*}} V^v \bullet x.(M :_v K)\\ \quad\longrightarrow^v_{(\beta_L)} (M :_v K)[{}^{V^v}/_x]\\ \quad\overset{Lem\ 19(3)}{\longrightarrow^{v*}} M[{}^V/_x] :_v K\end{array}$$

$(\beta_\&)$: $\mathrm{fst}\langle V, W\rangle :_v K \equiv \langle V, W\rangle :_v \mathrm{fst}[K]$

$$\begin{array}{l}\quad\overset{Lem\ 19(2)}{\longrightarrow^{v*}} \langle V, W\rangle^v\bullet\mathrm{fst}[K] \equiv \langle V^v, W^v\rangle\bullet\mathrm{fst}[K]\\ \quad\longrightarrow^v_{(\beta_\&)} V^v \bullet K \overset{Lem\ 19(1)}{\longrightarrow^{v*}} V :_v K\end{array}$$

The other rule of $(\beta_\&)$ is proved similarly.

$(\beta_\vee)$: $\delta(\mathrm{inl}(V), x.M, y.N) :_v K$

$$\begin{array}{l}\quad\equiv \mathrm{inl}(V) :_v [x.(M :_v K), y.(N :_v K)]\\ \quad\overset{Lem\ 19(2)}{\longrightarrow^{v*}} \langle V^v\rangle\mathrm{inl}\bullet[x.(M :_v K), y.(N :_v K)]\\ \quad\longrightarrow^v_{(\beta_\vee)} V^v \bullet x.(M :_v K)\\ \quad\longrightarrow^v_{(\beta_L)} (M :_v K)[{}^{V^v}/_x]\\ \quad\overset{Lem\ 19(3)}{\longrightarrow^{v*}} M[{}^V/_x] :_v K\end{array}$$

The other rules of $(\beta_\vee)$ are proved similarly.

$(\beta_\neg)$: $((\lambda x.S)V)^\dagger \equiv V :_v x.S^\dagger \overset{Lem\ 19(2)}{\longrightarrow^{v*}} V^v \bullet_v x.S^\dagger$

$$\quad\longrightarrow^v_{(\beta_L)} S^\dagger[{}^{V^v}/_x] \overset{Lem\ 19(3)}{\longrightarrow^{v*}} (S[{}^V/_x])^\dagger$$

(ζ): Let $E_{e\lambda}$ be an elimination context of $(\lambda x.M)\{-\}$, then

$$\begin{array}{l}E_{e\lambda}\{\mu\alpha.S\} :_v K \overset{Lem\ 20(1)}{\longrightarrow^{v*}} \mu\alpha.S :_v \Psi(E_{e\lambda}, K)\\ \equiv S^\dagger[{}^{\Psi(E_{e\lambda},K)}/_\alpha] \equiv S^\dagger[{}^{\Psi(E_{e\lambda},\beta)}/_\alpha][{}^K/_\beta]\\ \overset{Lem\ 20(4)}{\longrightarrow^{v*}} (S[{}^{[\beta]E_{e\lambda}\{-\}}/_{[\alpha]\{-\}}])^\dagger[{}^K/_\beta]\\ \equiv \mu\beta.S[{}^{[\beta]E_{e\lambda}\{-\}}/_{[\alpha]\{-\}}] :_V K\end{array}$$

The other rule of (ζ) is proved similarly.

$(comp)$: $E_{e\lambda}\{(\lambda x.M)N\} :_v K$

$$\begin{array}{l}\quad\overset{Lem\ 20(1)}{\longrightarrow^{v*}} (\lambda x.M)N :_v \Psi(E_{e\lambda}, K)\\ \quad\equiv N :_v x.(M :_v \Psi(E_{e\lambda}, K))\\ \quad\overset{Lem\ 20(2)}{\longrightarrow^{v*}} N :_v x.(E_{e\lambda}\{M\} :_v K)\\ \quad\equiv (\lambda x.E_{e\lambda}\{M\})N :_v K\end{array}$$

The other rule of (ζ) is proved similarly.

(π): $E_{e\lambda}\{\delta(O, x.M, y.N)\} :_v K$

$$\quad\overset{Lem\ 20(1)}{\longrightarrow^{v*}} \delta(O, x.M, y.N) :_v \Psi(E_{e\lambda}, K)$$

$\equiv O :_v [x.(M :_v \Psi(E_{e\lambda}, K)),$
$y.(N :_v \Psi(E_{e\lambda}, K))]$
$\overset{Lem\ 20(2)}{\longrightarrow^{v*}} O :_v [x.(E_{e\lambda}\{M\} :_v K),$
$y.(E_{e\lambda}\{N\} :_v K)]$
$\equiv \delta(O, x.E_{e\lambda}\{M\}, y.E_{e\lambda}\{N\}) :_v K$

The other rule of (ζ) is proved similarly.

(η_μ): $\mu\alpha.[\alpha]M :_v K \equiv ([\alpha]M)^\dagger[^K/_\alpha]$
$\equiv (M :_v \alpha)[^K/_\alpha] \equiv M :_v K$

$(name)$: Let O not be a value. Then $E_i\{O\} :_v K \overset{Lem\ 20(1)}{\longrightarrow^{v*}} O :_v \Psi(E_i, K)$
$\overset{Lem\ 20(3)}{\longrightarrow^{v*}} (\lambda x.E_i\{x\})O :_v K$; and
$E_e\{O\} :_v K \overset{Lem\ 20(1)}{\longrightarrow^{v*}} O :_v \Psi(E_e, K)$
$\longrightarrow^{v}_{\eta_L} O :_v x.(x \bullet \Psi(E_e, K))$
$\overset{Lem\ 20(2)}{\longrightarrow^{v*}} O :_v x.(E_e\{x\} :_v K)$
$\equiv (\lambda x.E_e\{x\})O :_v K$.

Induction cases (1) and (2) : These cases are similar to the proof for induction cases of call-by-name. □

Proof of Lemma 22 (1) is proved by an induction on M, K, and S. We give the cases of sums, i.e., $\langle M\rangle$inl, $\langle N\rangle$inl, and $[K, L]$.

Case of $\langle M\rangle$inl :

$[\![\langle M\rangle \mathrm{inl}_*]\!] \equiv [\![\mu(\alpha,\beta).[\alpha]M_*]\!]$
$\equiv \mu\gamma.([\alpha][\![M_*]\!])$
$[^{[\gamma]\mathrm{inl}(-)}/_{[\alpha]\{-\}}, ^{[\gamma]\mathrm{inr}(-)}/_{[\beta]\{-\}}]$
$\equiv \mu\gamma.[\gamma]\mathrm{inl}([\![M_*]\!]) \overset{I.H.}{=_n} \mu\gamma.[\gamma]\mathrm{inl}(M_\circledast)$
$=_n \mathrm{inl}(M_\circledast) \equiv (\langle M\rangle \mathrm{inl})_\circledast$

Case of $\langle N\rangle$inr : this case is proved in a way similar to the above case.

Case of $[K, L]$:

$[\![[K, L]_*\{O\}]\!] \equiv [\![L_*\{\mu\beta.K_*\{\mu\alpha.[\alpha,\beta]O\}\}]\!]$
$\overset{I.H.}{=_n} L_\circledast\{[\![\mu\beta.K_*\{\mu\alpha.[\alpha,\beta]O\}]\!]\}$
$\equiv L_\circledast\{\mu\beta.[\![K_*\{\mu\alpha.[\alpha,\beta]O\}]\!]\}$
$\overset{I.H.}{=_n} L_\circledast\{\mu\beta.K_\circledast\{[\![\mu\alpha.[\alpha,\beta]O]\!]\}\}$
$\equiv L_\circledast\{\mu\beta.K_\circledast\{\mu\alpha.\delta([\![O]\!], x.[\alpha]x, y.[\beta]y)\}\}$
$\overset{(*)}{=_n} \delta([\![O]\!], x.K_\circledast\{x\}, y.L_\circledast\{y\})$
$\equiv [K, L]_\circledast\{[\![O]\!]\}$

$(*)$ comes from the claim: $K_\circledast\{\mu\alpha.S\} =_n S[^{K_\circledast\{-\}}/_{[\alpha]\{-\}}]$. This claim is proved by a straightforward induction on K.

(2) is proved by an induction on M and S. The key cases are terms and statements for sums, and these cases are shown in a way similar to (1). □

Proof of Theorem 28 The claims are proved by simultaneous induction on the reduction relation $\longrightarrow^n$. Base cases are shown as follows.

$(\beta_\supset)$: $((\lambda x.M) \bullet (N@P))_\sharp \equiv (N@P)_\sharp\{\lambda x.M_\sharp\}$
$\equiv P_\sharp\{(\lambda x.M_\sharp)N_\sharp\} \longrightarrow_{(\beta_\supset)} P_\sharp\{M_\sharp[^{N_\sharp}/_x]\}$
$\equiv (P_\sharp\{M_\sharp\})[^{N_\sharp}/_x] \equiv (M \bullet P)_\sharp[^{N_\sharp}/_x]$
$\equiv (x.(M \bullet P))_\sharp\{N_\sharp\} \equiv (N \bullet x.(M \bullet P))_\sharp$

$(\beta_\&)$: $(\langle M, N\rangle \bullet \mathrm{fst}[P])_\sharp \equiv \mathrm{fst}[P]_\sharp\{\langle M_\sharp, N_\sharp\rangle\}$
$\equiv P_\sharp\{\mathrm{fst}\langle M_\sharp, N_\sharp\rangle\}$
$\longrightarrow_{(\beta_\&)} P_\sharp\{M_\sharp\} \equiv (M \bullet P)_\sharp$

The other $(\beta_\&)$ case can be shown similarly.

$(\beta_\vee)$: $(\langle M\rangle \mathrm{inl} \bullet [P, Q])_\sharp \equiv [P, Q]_\sharp\{\mathrm{inl}(M_\sharp)\}a$
$\equiv \delta(\mathrm{inl}(M_\sharp), x.P_\sharp\{x\}, y.Q_\sharp\{y\})$
$\longrightarrow_{(\beta_\vee)} P_\sharp\{M_\sharp\} \equiv (M \bullet P)_\sharp$

The other $(\beta_\&)$ case can be shown similarly.

$(\beta_\neg)$: $([K]\mathrm{not} \bullet \mathrm{not}\langle M\rangle)_\sharp$
$\equiv (\mathrm{not}\langle M\rangle)_\sharp\{\lambda x.K_\sharp\{x\}\} \equiv (\lambda x.K_\sharp\{x\})M_\sharp$
$\longrightarrow_{(\beta_\neg)} K_\sharp\{M_\sharp\} \equiv (M \bullet K)_\sharp$

(β_R): $(S.\alpha \bullet P)_\sharp \equiv P_\sharp\{\ \mu\alpha.S_\sharp\ \}$
$\longrightarrow_{(\zeta)} S_\sharp[^{P_\sharp\{-\}}/_{[\alpha](-)}] \overset{Lem\ 26(4)}{\equiv} (S[^P/_\alpha])_\sharp$

(β_L): $(M \bullet x.S)_\sharp \equiv (x.S)_\sharp\{M_\sharp\}$
$\equiv S_\sharp[^{M_\sharp}/_x] \overset{Lem\ 26(2)}{\equiv} (S[^M/_x])_\sharp$

(η_R): $M_\sharp \longrightarrow_{(\eta_\mu)} \mu\alpha.[\alpha]M_\sharp \equiv (\ (M \bullet \alpha).\alpha\ )_\sharp$

(η_L): $K_\sharp\{O\} \equiv (K_\sharp\{x\})[^O/_x]$
$\equiv (x \bullet K)_\sharp[^O/_x] \equiv (x.(x \bullet K))_\sharp\{O\}$

$(name)$: Note that K is not a covalue.
$(\ M \bullet F\{K\}\ )_\sharp \equiv F\{K\}_\sharp\{M_\sharp\}$
$\overset{Lem\ 27(1)}{\longrightarrow^*_n} K_\sharp\{F_\sharp\{M_\sharp\}\}$
$\longrightarrow_{(\eta_\mu)} K_\sharp\{\mu\alpha.[\alpha]F_\sharp\{M_\sharp\}\}$
$\equiv K_\sharp\{\mu\alpha.\alpha_\sharp\{\ F_\sharp\{M_\sharp\}\ \}\}$
$\overset{Lem\ 27(2)}{\longrightarrow^*_n} K_\sharp\{\mu\alpha.F\{\alpha\}_\sharp\{M_\sharp\}\}$
$\equiv ((M \bullet F\{\alpha\}).\alpha \bullet K)_\sharp$

Induction cases can be shown easily. □

Proof of Lemma 29 (1) is immediately shown. We show (2) when O is not a value by a case analysis of K.

Case of α: $\alpha_\dagger\{O\} \equiv [\alpha]O \longrightarrow_{(name)} (\lambda z.[\alpha]z)O \equiv \alpha_\dagger[O]$

Case of $[K, L]$:

$[K, L]_\dagger\{O\} \equiv \delta(O, x.K_\dagger\{x\}, y.L_\dagger\{y\})$
$\longrightarrow_{(name)} (\lambda z.\delta(z, x.K_\dagger\{x\}, y.L_\dagger\{y\}))O$
$\equiv [K, L]_\dagger[O]$

Case of fst$[K]$:

$\mathrm{fst}[K]_\dagger\{O\} \equiv K_\dagger[\mathrm{fst}(O)] \equiv (\lambda z.K_\dagger\{z\})\mathrm{fst}(O)$
$\longrightarrow_{(name)} (\lambda z.K_\dagger\{z\})((\lambda x.\mathrm{fst}(x))O)$
$\longrightarrow_{(comp)} (\lambda x.(\lambda z.K_\dagger\{z\})\mathrm{fst}(x))O$

$\equiv (\lambda x.(K_\dagger[\mathrm{fst}(x)])O$
$\equiv (\lambda x.(\mathrm{fst}[K]_\dagger\{x\})O \equiv \mathrm{fst}[K]_\dagger[O]$

Case of $\mathrm{snd}[K]$: This case can be shown in a way similar to the above case.
Case of $\mathrm{not}\langle M\rangle$:

$$\mathrm{not}\langle M\rangle_\dagger\{O\} \equiv OM_\dagger \longrightarrow_{(name)} (\lambda z.zM_\dagger)O \equiv (\lambda z.\mathrm{not}\langle M\rangle_\dagger\{z\})O \equiv \mathrm{not}\langle M\rangle_\dagger[O]$$

Case of $M@K$:

$$(M@K)_\dagger\{O\} \equiv K_\dagger[OM_\dagger] \equiv (\lambda z.K_\dagger\{z\})OM_\dagger \longrightarrow_{(name)} (\lambda z.K_\dagger\{z\})((\lambda x.xM_\dagger)O) \longrightarrow_{(comp)} (\lambda x.(\lambda z.K_\dagger\{z\})(xM_\dagger))O \equiv (\lambda x.(K_\dagger[xM_\dagger])O \equiv (\lambda x.(M@K)_\dagger\{x\})O \equiv (M@K)_\dagger[O]$$

Case of $x.S$:

$$(x.S)_\dagger\{O\} \equiv (\lambda x.S_\dagger)O \equiv (\lambda z.S_\dagger[^z/_x])O \equiv (\lambda z.(x.S)_\dagger\{z\})O \equiv (x.S)_\dagger[O] \qquad \square$$

Proof of Lemma 30 The claims are shown by a simultaneous induction on M, K, and S. We consider the key cases.
Case of $\mathrm{fst}[K]$:

$$\mathrm{fst}[K]_\circledast\{O\} \equiv K_\circledast\{\mathrm{fst}(O)\} \overset{I.H.}{\longrightarrow_v^*} K_\dagger\{\mathrm{fst}(O)\} \overset{Lem\ 29}{\longrightarrow_v^*} K_\dagger[\mathrm{fst}(O)] \equiv \mathrm{fst}[K]_\dagger[O]$$

Case of $\mathrm{snd}[K]$: This case is proved similarly.
Case of $M@K$:

$$(M@K)_\circledast\{O\} \equiv K_\circledast\{OM_\circledast\} \overset{I.H.}{\longrightarrow_v^*} K_\dagger\{OM_\dagger\} \overset{Lem\ 29}{\longrightarrow_v^*} K_\dagger[OM_\dagger] \equiv (M@K)_\dagger\{O\}$$

Case of $x.S$:

$$(x.S)_\circledast\{O\} \equiv (\lambda x.S_\circledast)O \overset{I.H.}{\longrightarrow_v^*} (\lambda x.S_\dagger)O \longrightarrow_v^* (\overline{\lambda}_v x.S_\dagger)O \equiv (x.S)_\dagger\{O\}$$

Case of $M \bullet K$:

$$(M \bullet K)_\circledast \equiv K_\circledast\{M_\circledast\} \overset{I.H.}{\longrightarrow_v^*} K_\dagger\{M_\dagger\} \overset{Lem\ 29}{\longrightarrow_v^*} K_\dagger[M_\dagger] \equiv (M \bullet K)_\dagger \qquad \square$$

Proof of Lemma 32 (1) The claim is proved by induction on K.
(2) We claim that if we have $(K_\dagger\{O\})[^{V_\dagger}/_x] \equiv (K[^V/_x])_\dagger\{O[^{V_\dagger}/_x]\}$, then we obtain $(K_\dagger[O])[^{V_\dagger}/_x] \equiv (K[^V/_x])_\dagger[O[^{V_\dagger}/_x]]$. We show this claim. Assume O is a value, then $O[^V/_x]$ is also a value. Hence we have

$$(K_\dagger[O])[^{V_\dagger}/_x] \equiv (K_\dagger\{O\})[^{V_\dagger}/_x] \equiv (K[^V/_x])_\dagger\{O[^{V_\dagger}/_x]\} \equiv (K[^V/_x])_\dagger[O[^{V_\dagger}/_x]].$$

If O is not a value, then $O[^V/_x]$ is also not a value, so we have

$$(K_\dagger[O])[^{V_\dagger}/_x] \equiv ((\lambda z.K_\dagger\{z\})O)[^{V_\dagger}/_x] \equiv (\lambda z.(K_\dagger\{z\})[^{V_\dagger}/_x])(O[^{V_\dagger}/_x])) \equiv (\lambda z.(K[^V/_x])_\dagger\{z\})(O[^{V_\dagger}/_x])) \equiv (K[^V/_x])_\dagger[O[^{V_\dagger}/_x]].$$

Therefore we show the other claims by induction on M, K, and S. We give the key case: $x_\dagger[^{V_\dagger}/_x] \equiv x[^{V_\dagger}/_x] \equiv V_\dagger$. $\square$

Proof of Lemma 33 We can prove this lemma by a simultaneous induction on M, K, and S. We give two cases.
Case of α:

$$(\alpha_\dagger\{O\})[^{(\lambda y.L_\dagger\{y\})\{-\}}/_{[\alpha]\{-\}}] \equiv ([\alpha]O)[^{(\lambda y.L_\dagger\{y\})\{-\}}/_{[\alpha]\{-\}}] \equiv (\lambda y.L_\dagger\{y\})(O[^{(\lambda y.L_\dagger\{y\})\{-\}}/_{[\alpha]\{-\}}]) \longrightarrow_v^* (\overline{\lambda}_v y.L_\dagger\{y\})(O[^{(\lambda y.L_\dagger\{y\})\{-\}}/_{[\alpha]\{-\}}]) \equiv L_\dagger[O[^{(\lambda y.L_\dagger\{y\})\{-\}}/_{[\alpha]\{-\}}]]$$

Case of $\mathrm{fst}[K]$:

$$((\mathrm{fst}[K])_\dagger\{O\})[^{(\lambda y.L_\dagger\{y\})\{-\}}/_{[\alpha]\{-\}}] \equiv (K_\dagger[\mathrm{fst}(O)])[^{(\lambda y.L_\dagger\{y\})\{-\}}/_{[\alpha]\{-\}}] \overset{I.H.}{\longrightarrow_v^*} (K[^L/_\alpha])_\dagger \big[\mathrm{fst}(O)[^{(\lambda y.L_\dagger\{y\})\{-\}}/_{[\alpha]\{-\}}]\big] \equiv (K[^L/_\alpha])_\dagger \big[\mathrm{fst}(O[^{(\lambda y.L_\dagger\{y\})\{-\}}/_{[\alpha]\{-\}}])\big] \equiv (\mathrm{fst}(K[^L/_\alpha]))_\dagger \big\{\mathrm{fst}(O[^{(\lambda y.L_\dagger\{y\})\{-\}}/_{[\alpha]\{-\}}])\big\} \overset{Lem\ 29(2)}{\longrightarrow_v^*} (\mathrm{fst}(K[^L/_\alpha]))_\dagger \big[\mathrm{fst}(O[^{(\lambda y.L_\dagger\{y\})\{-\}}/_{[\alpha]\{-\}}])\big] \qquad \square$$

Proof of Theorem 34 (1) – (3) are proved by simultaneous induction on the reduction relation $\longrightarrow^v$. We claim that if $K_\dagger\{O\} \longrightarrow_v^* L_\dagger\{O\}$, then $K_\dagger[O] \longrightarrow_v^* L_\dagger[O]$. We first show this claim. If O is a value, the claim is immediately shown. Otherwise, $K_\dagger[O] \equiv (\lambda x.K_\dagger\{x\})O \longrightarrow_v^* (\lambda x.L_\dagger\{x\})O \equiv L_\dagger[O]$.

We often use the following shortcuts:
(a) $(V \bullet K)_\dagger \equiv K_\dagger[V_\dagger] \equiv K_\dagger\{V_\dagger\}$
(b) $(\lambda x.K_\dagger\{x\})M_\dagger \longrightarrow_v^* (\overline{\lambda}_v x.K_\dagger\{x\})M_\dagger \equiv K_\dagger[M_\dagger] \equiv (M \bullet K)_\dagger$
(c) $(\lambda x.S_\dagger)M_\dagger \longrightarrow_v^* (\overline{\lambda}_v x.S_\dagger)M_\dagger \equiv (x.S)_\dagger\{M_\dagger\} \overset{Lem\ 29}{\longrightarrow_v^*} (x.S)_\dagger[M_\dagger] \equiv (M \bullet x.S)_\dagger$

Base cases are shown as follows.
$(\beta_\supset)$: $((\lambda x.M) \bullet (N@K))_\dagger$

$\overset{(a)}{\equiv} (N@K)_\dagger\{(\lambda x.M)_\dagger\}$
$\equiv K_\dagger[(\lambda x.M_\dagger)N_\dagger]$
$\equiv (\lambda z.K_\dagger\{z\})((\lambda x.M_\dagger)N_\dagger)$
$\longrightarrow_{(comp)} (\lambda x.(\lambda z.K_\dagger\{z\})M_\dagger)N_\dagger$
$\overset{(b)}{\longrightarrow_v^*} (\lambda x.(M\bullet K)_\dagger)N_\dagger \overset{(c)}{\longrightarrow_v^*} (N\bullet x.(M\bullet K))_\dagger$

$(\beta_\&)$: $(\langle V,W\rangle \bullet \mathrm{fst}[K])_\dagger \overset{(a)}{\equiv} \mathrm{fst}[K]_\dagger\{(\langle V,W\rangle)_\dagger\}$
$\equiv K_\dagger[\,\mathrm{fst}\langle V_\dagger, W_\dagger\rangle\,] \longrightarrow_{(\beta_\&)} K_\dagger[V_\dagger]$
$\equiv (V\bullet K)_\dagger$
The other case of $(\beta_\&)$ can be shown similarly.

$(\beta_\vee)$: $(\langle V\rangle\mathrm{inl}\bullet[K,L])_\dagger \overset{(a)}{\equiv} [K,L]_\dagger\{\,\langle V\rangle\mathrm{inl}_\dagger\,\}$
$\equiv \delta(\mathrm{inl}(V_\dagger), x.K_\dagger\{x\}, y.L_\dagger\{y\})$
$\longrightarrow_{(\beta_\vee)} K_\dagger\{V_\dagger\}$
$\overset{(a)}{\equiv} (V\bullet K)_\dagger$
The other case of $(\beta_\&)$ can be shown similarly.

$(\beta_\neg)$: $([K]\mathrm{not}\bullet \mathrm{not}\langle M\rangle)_\dagger$
$\overset{(a)}{\equiv} (\mathrm{not}\langle M\rangle)_\dagger\{\,[K]\mathrm{not}_\dagger\,\}$
$\equiv (\lambda x.K_\dagger\{x\})M_\dagger \overset{(b)}{\longrightarrow_v^*} (M\bullet K)_\dagger$

(β_R): $(S.\alpha\bullet K)_\dagger \equiv K_\dagger[\,(S.\alpha)_\dagger\,]$
$\equiv (\lambda x.K_\dagger\{x\})\mu\alpha.S_\dagger$
$\longrightarrow_{(\zeta)} S_\dagger[{}^{(\lambda x.K_\dagger\{x\})\{-\}}/_{[\alpha](-)}]$
$\overset{Lem\ 33}{\equiv} (S[{}^{K}/_{\alpha}])_\dagger$

(β_L): $(V\bullet x.S)_\dagger \overset{(a)}{\equiv} (x.S)_\dagger\{V_\dagger\} \equiv S_\dagger[{}^{V_\dagger}/_x]$
$\overset{Lem\ 32(2)}{\equiv} (S[{}^{V}/_x])_\dagger$

(η_R): $M_\dagger \longrightarrow_{(\eta_\mu)} \mu\alpha.[\alpha]M_\dagger \equiv \mu\alpha.\alpha_\dagger\{M\}$
$\overset{Lem\ 29(2)}{\longrightarrow_v^*} \mu\alpha.\alpha_\dagger[M] \equiv \mu\alpha.(M\bullet\alpha)_\dagger$
$\equiv (\,(M\bullet\alpha).\alpha\,)_\dagger$

(η_L): If O is a value V, then
$K_\dagger\{V\} \equiv (K_\dagger\{x\})[{}^{V}/_x] \overset{(a)}{\equiv} (x\bullet K)_\dagger[{}^{V}/_x]$
$\equiv (x.(x\bullet K))_\dagger\{V\}$.
If O is not a value, then
$K_\dagger\{O\} \overset{Lem\ 29(2)}{\longrightarrow_v^*} K_\dagger[O] \equiv (\lambda x.K_\dagger\{x\})O$
$\overset{(a)}{\equiv} (\lambda x.(x\bullet K)_\dagger)O \equiv (x.(x\bullet K))_\dagger\{O\}$.

$(name)$: Note that $M_\dagger$ is not a value because M is not a value.

$(\,\langle M,N\rangle\bullet K\,)_\dagger \equiv K_\dagger[\,\langle M,N\rangle_\dagger\,]$
$\equiv (\lambda z.K_\dagger\{z\})\langle M_\dagger, N_\dagger\rangle$
$\longrightarrow_{(name)} (\lambda z.K_\dagger\{z\})((\lambda x.\langle x, N_\dagger\rangle)M_\dagger)$
$\longrightarrow_{(comp)} (\lambda x.(\lambda z.K_\dagger\{z\})\langle x, N_\dagger\rangle)M_\dagger$
$\overset{(b)}{\longrightarrow_v^*} (\lambda x.(\langle x,N\rangle\bullet K)_\dagger)M_\dagger$
$\overset{(c)}{\longrightarrow_v^*} (M\bullet x.(\langle x,N\rangle\bullet K))_\dagger$

$(\,\langle V,M\rangle\bullet K\,)_\dagger \equiv K_\dagger[\,\langle V,M\rangle_\dagger\,]$
$\equiv (\lambda z.K_\dagger\{z\})\langle V_\dagger, M_\dagger\rangle$
$\longrightarrow_{(name)} (\lambda z.K_\dagger\{z\})((\lambda x.\langle V_\dagger, x\rangle)M_\dagger)$
$\longrightarrow_{(comp)} (\lambda x.(\lambda z.K_\dagger\{z\})\langle V_\dagger, x\rangle)M_\dagger$
$\overset{(b)}{\longrightarrow_v^*} (\lambda x.(\langle V,x\rangle\bullet K)_\dagger)M_\dagger$
$\overset{(c)}{\longrightarrow_v^*} (M\bullet x.(\langle V,x\rangle\bullet K))_\dagger$

$(\,\langle M\rangle\mathrm{inl}\bullet K\,)_\dagger \equiv K_\dagger[\,\langle M\rangle\mathrm{inl}_\dagger\,]$
$\equiv (\lambda z.K_\dagger\{z\})\mathrm{inl}(M_\dagger)$
$\longrightarrow_{(name)} (\lambda z.K_\dagger\{z\})((\lambda x.\mathrm{inl}(x))M_\dagger)$
$\longrightarrow_{(comp)} (\lambda x.(\lambda z.K_\dagger\{z\})\mathrm{inl}(x))M_\dagger$
$\overset{(b)}{\longrightarrow_v^*} (\lambda x.(\langle x\rangle\mathrm{inl}\bullet K)_\dagger)M_\dagger$
$\overset{(c)}{\longrightarrow_v^*} (M\bullet x.(\langle x\rangle\mathrm{inl}\bullet K))_\dagger$

The last case, $\langle M\rangle\mathrm{inr}\bullet K \longrightarrow^v M\bullet x.(\langle x\rangle\mathrm{inr}\bullet K)$, is also shown similarly.

Induction cases can be easily shown. □

Proof of Proposition 35 (1) If we have $P_\sharp\{O\} \longrightarrow_n^* (O :_n P)_\sharp$ for any covalue P, then we can obtain $O \longrightarrow_{(\eta_\mu)} \mu\alpha.[\alpha]O \equiv \mu\alpha.\alpha_\sharp\{O\} \longrightarrow_n^* \mu\alpha.(O :_n \alpha)_\sharp \equiv ((O :_n \alpha).\alpha)_\sharp \equiv (O^\sharp)_\sharp$. We prove the rest of (1) by a simultaneous induction on O and S.

Case of x: $P_\sharp\{x\} \equiv (x\bullet P)_\sharp \equiv (x :_n P)_\sharp$

Case of $\langle M,N\rangle$:

$$P_\sharp\{\langle M,N\rangle\} \overset{I.H.}{\longrightarrow_n^*} P_\sharp\{\langle (M^\sharp)_\sharp, (N^\sharp)_\sharp\rangle\}$$
$$\equiv P_\sharp\{\langle M^\sharp, N^\sharp\rangle_\sharp\}$$
$$\equiv (\langle M,N\rangle^\sharp\bullet P)_\sharp \longrightarrow_n^* (\langle M,N\rangle :_n P)_\sharp$$

Case of $\mathrm{fst}(O)$:

$$P_\sharp\{\mathrm{fst}(O)\}$$
$$\equiv \mathrm{fst}[P]_\sharp\{O\} \overset{I.H.}{\longrightarrow_n^*} \mathrm{fst}[P]_\sharp\{(O^\sharp)_\sharp\}$$
$$\equiv (O^\sharp\bullet\mathrm{fst}[P])_\sharp \longrightarrow_n^* (O :_n \mathrm{fst}[P])_\sharp$$
$$\equiv (\mathrm{fst}(O) :_n P)_\sharp$$

The case of $\mathrm{snd}(O)$ is shown similarly.

Case of $\mathrm{inl}(O)$:

$$P_\sharp\{\mathrm{inl}(O)\} \overset{I.H.}{\longrightarrow_n^*} P_\sharp\{\mathrm{inl}((O^\sharp)_\sharp)\}$$
$$\equiv P_\sharp\{\langle O^\sharp\rangle\mathrm{inl}_\sharp\} \equiv (\langle O^\sharp\rangle\mathrm{inl}\bullet P)_\sharp$$
$$\equiv (\mathrm{inl}(O) :_n P)_\sharp$$

The case of $\mathrm{inr}(O)$ is shown similarly.

Case of $\delta(O, x.M, y.N)$:

$$P_\sharp\{\delta(O,x.M,y.N)\}$$
$$\longrightarrow_{(\pi)} \delta(O, x.P_\sharp\{M\}, y.P_\sharp\{N\})$$
$$\overset{I.H.}{\longrightarrow_n^*} \delta\big(O, x.(M :_n P), y.(N :_n P)\big)$$
$$\equiv \delta\big(O, x'.(x.(M :_n P))_\sharp\{x'\},$$

$$y'.(y.(N :_n P))_\sharp\{y'\}\big)$$
$$\equiv [x.(M :_n P), y.(N :_n P)]_\sharp\{O\}$$
$$\xrightarrow{I.H.}{}^{*}_{n} [x.(M :_n P), y.(N :_n P)]_\sharp\{(O^\sharp)_\sharp\}$$
$$\equiv \big(O^\sharp \bullet [x.(M :_n P), y.(N :_n P)]\big)_\sharp$$
$$\equiv (\delta(O, x.M, y.N) :_n P)_\sharp$$

Case of $\lambda x.M$:

$$P_\sharp\{\lambda x.M\} \xrightarrow{I.H.}{}^{*}_{n} P_\sharp\{\lambda x.(M^\sharp)_\sharp\}$$
$$\equiv ((\lambda x.M^\sharp) \bullet P)_\sharp \equiv (\lambda x.M :_n P)_\sharp$$

Case of MN(MN is a term, and M is not a λ-abstraction):

$$P_\sharp\{\, MN\,\} \xrightarrow{I.H.}{}^{*}_{n} P_\sharp\{\, M(N^\sharp)_\sharp\,\}$$
$$\equiv (N^\sharp @ P)_\sharp\{\, M\,\}$$
$$\xrightarrow{I.H.}{}^{*}_{n} (M^\sharp :_n (N^\sharp @ P))_\sharp \equiv (MN :_n P)_\sharp$$

Case of $(\lambda x.M)N$:

$$P_\sharp\{\,(\lambda x.M)N\,\} \longrightarrow_{(\beta_\supset)} P_\sharp\{\, M[{}^N/_x]\,\}$$
$$\equiv (x.(P_\sharp\{M\}))_\sharp\{N\}$$
$$\xrightarrow{I.H.}{}^{*}_{n} (x.(M :_n P))_\sharp\{N\}$$
$$\xrightarrow{I.H.}{}^{*}_{n} (x.(M :_n P))_\sharp\{\,(N^\sharp)_\sharp\,\}$$
$$\equiv (N^\sharp \bullet x.(M :_n P))_\sharp$$
$$\equiv (\,(\lambda x.M)N :_n P\,)_\sharp$$

Case of $\lambda x.S$:

$$P_\sharp\{\lambda x.S\} \xrightarrow{I.H.}{}^{*}_{n} P_\sharp\{\lambda x.(S^\sharp)_\sharp\}$$
$$\equiv P_\sharp\big\{\lambda y.(x.S^\sharp)_\sharp\{y\}\big\} \equiv \big([x.S^\sharp]\mathrm{not} \bullet P\big)_\sharp$$
$$\equiv (\,\lambda x.S :_n P\,)_\sharp$$

Case of $\mu\alpha.S$:

$$P_\sharp\{\mu\alpha.S\} \xrightarrow{I.H.}{}^{*}_{n} P_\sharp\{\mu\alpha.(S^\sharp)_\sharp\}$$
$$\longrightarrow_{(\zeta)} (S^\sharp)_\sharp[{}^{P_\sharp\{-\}}/_{[\alpha]\{-\}}]$$
$$\longrightarrow^{*}_{n} (S^\sharp[{}^P/_\alpha])_\sharp \equiv (\,\mu\alpha.S :_n P\,)_\sharp$$

Case of $[\alpha]O$:

$$[\alpha]O \equiv \alpha_\sharp\{O\} \xrightarrow{I.H.}{}^{*}_{n} (O :_n \alpha)_\sharp$$
$$\equiv (([\alpha]O)^\sharp)_\sharp$$

Case of $\delta(O, x.S, y.T)$:

$$\delta(O, x.S, y.T) \xrightarrow{I.H.}{}^{*}_{n} \delta\big(O, x.(S^\sharp)_\sharp, y.(T^\sharp)_\sharp\big)$$
$$\equiv \delta\big(O, x'.((x.S^\sharp)_\sharp\{x'\}),$$
$$y'.((y.T^\sharp)_\sharp\{y'\})\big)$$
$$\equiv [x.S^\sharp, y.T^\sharp]_\sharp\{O\}$$
$$\xrightarrow{I.H.}{}^{*}_{n} [x.S^\sharp, y.T^\sharp]_\sharp\{(O^\sharp)_\sharp\}$$
$$\equiv \big(O^\sharp \bullet [x.S^\sharp, y.T^\sharp]\big)_\sharp$$
$$\equiv (\delta(O, x.S, y.T)^\sharp)_\sharp$$

Case of MN(MN is a statement, and M is not a λ-abstraction):

$$MN \xrightarrow{I.H.}{}^{*}_{n} M(N^\sharp)_\sharp \equiv \mathrm{not}\langle N^\sharp\rangle_\sharp\{M\}$$
$$\xrightarrow{I.H.}{}^{*}_{n} (M :_n \mathrm{not}\langle N^\sharp\rangle)_\sharp \equiv (\,(MN)^\sharp\,)_\sharp$$

Case of $(\lambda x.S)N$:

$$(\lambda x.S)N \xrightarrow{I.H.}{}^{*}_{n} (\lambda x.(S^\sharp)_\sharp)N$$
$$\longrightarrow_{(\beta_\neg)} (S^\sharp)_\sharp[{}^N/_x]$$
$$\equiv (x.S^\sharp)_\sharp\{N\} \xrightarrow{I.H.}{}^{*}_{n} (x.S^\sharp)_\sharp\{(N^\sharp)_\sharp\}$$
$$\equiv \big(N^\sharp \bullet x.S^\sharp\big)_\sharp \equiv \big(((\lambda x.S)N)^\sharp\big)_\sharp$$

(2) We can easily show $O \longrightarrow^{*}_{v} (O^\dagger)_\dagger$ from $K_\dagger[O] \longrightarrow^{*}_{v} (O :_v K)_\dagger$ in a way similar to the proof in (1). In the following, we prove the rest of (2) by a simultaneous induction on O and S.

Case of x: $K_\dagger[x] \equiv (x \bullet K)_\dagger \equiv (x :_v K)_\dagger$

Case of $\langle M, N\rangle$:

$$K_\dagger[\langle M, N\rangle] \xrightarrow{I.H.}{}^{*}_{v} K_\dagger[\langle (M^\dagger)_\dagger, (N^\dagger)_\dagger\rangle]$$
$$\equiv K_\dagger[\,\langle M^\dagger, N^\dagger\rangle_\dagger\,] \equiv (\langle M, N\rangle^\dagger \bullet K)_\dagger$$
$$\longrightarrow^{*}_{v} (\langle M, N\rangle :_v K)_\dagger$$

Case of $\mathrm{fst}(O)$:

$$K_\dagger[\mathrm{fst}(O)] \equiv \mathrm{fst}[K]_\dagger\{O\}$$
$$\xrightarrow{I.H.}{}^{*}_{v} \mathrm{fst}[K]_\dagger\{(O^\dagger)_\dagger\}$$
$$\xrightarrow{Lem\ 29}{}^{*}_{v} \mathrm{fst}[K]_\dagger[(O^\dagger)_\dagger] \equiv (O^\dagger \bullet \mathrm{fst}[K])_\dagger$$
$$\longrightarrow^{*}_{v} (O :_v \mathrm{fst}[K])_\dagger \equiv (\mathrm{fst}(O) :_v K)_\dagger$$

The case of $\mathrm{snd}(O)$ is shown similarly.

Case of $\mathrm{inl}(O)$:

$$K_\dagger[\mathrm{inl}(O)] \xrightarrow{I.H.}{}^{*}_{v} K_\dagger[\mathrm{inl}((O^\dagger)_\dagger)]$$
$$\equiv K_\dagger[\langle O^\dagger\rangle \mathrm{inl}_\dagger]$$
$$\equiv (\langle O^\dagger\rangle \mathrm{inl} \bullet K)_\dagger \equiv (\mathrm{inl}(O) :_v K)_\dagger$$

The case of $\mathrm{inr}(O)$ is shown similarly.

Case of $\delta(O, x.M, y.N)$:

$$K_\dagger[\delta(O, x.M, y.N)]$$
$$\equiv (\lambda z.K_\dagger\{z\})\delta(O, x.M, y.N)$$
$$\longrightarrow_{(\pi)} \delta\big(O, x.(\lambda z.K_\dagger\{z\})M,$$
$$y.(\lambda z.K_\dagger\{z\})N\big)$$
$$\longrightarrow^{*}_{v} \delta\big(O, x.K_\dagger[M], y.K_\dagger[N]\big)$$
$$\xrightarrow{I.H.}{}^{*}_{v} \delta\big(O, x.(M :_v K), y.(N :_v K)\big)$$
$$\equiv \delta\big(O, x'.(x.(M :_v K))_\dagger\{x'\},$$
$$y'.(y.(N :_v K))_\dagger\{y'\}\big)$$

$$\equiv [x.(M :_v K), y.(N :_v K)]_\dagger\{O\}$$
$$\xrightarrow{Lem\ 29}{}_v^* [x.(M :_v K), y.(N :_v K)]_\dagger[O]$$
$$\xrightarrow{I.H.}{}_v^* \big(O :_v [x.(M :_v K), y.(N :_v K)]\big)_\dagger$$
$$\equiv (\,\delta(O, x.M, y.N) :_v K\,)_\dagger$$

Case of $\lambda x.M$:

$$K_\dagger[\lambda x.M] \xrightarrow{I.H.}{}_v^* K_\dagger[\lambda x.(M^\dagger)_\dagger]$$
$$\equiv ((\lambda x.M^\dagger) \bullet K)_\dagger \equiv (\lambda x.M :_v K)_\dagger$$

Case of MN(MN is a term, and M is not a λ-abstraction):

$$K_\dagger[\,MN\,] \xrightarrow{I.H.}{}_v^* K_\dagger[\,M(N^\dagger)_\dagger\,]$$
$$\equiv (N^\dagger @K)_\dagger\{\,M\,\} \xrightarrow{Lem\ 29}{}_v^* (N^\dagger @K)_\dagger[\,M\,]$$
$$\xrightarrow{I.H.}{}_v^* (M :_v (N^\dagger @K))_\dagger \equiv (MN :_v K)_\dagger$$

Case of $(\lambda x.M)N$:

$$K_\dagger[\,(\lambda x.M)N\,] \equiv (\lambda z.K_\dagger\{z\})((\lambda x.M)N)$$
$$\longrightarrow_{(comp)} (\lambda x.(\lambda z.K_\dagger\{z\})M)N$$
$$\longrightarrow_v^* (\lambda x.K_\dagger[M])N$$
$$\xrightarrow{I.H.}{}_v^* (\lambda x.(M :_v K)_\dagger)N$$
$$\longrightarrow_v^* (x.(M :_v K))_\dagger\{N\}$$
$$\xrightarrow{I.H.}{}_v^* (N :_v x.(M :_v K))_\dagger$$
$$\equiv (\,(\lambda x.M)N :_v K\,)_\dagger$$

Case of $\lambda x.S$:

$$K_\dagger[\lambda x.S] \xrightarrow{I.H.}{}_v^* K_\dagger[\lambda x.(S^\dagger)_\dagger]$$
$$\equiv K_\dagger\big[\lambda y.(x.S^\dagger)_\dagger\{y\}\big]$$
$$\equiv \big([x.S^\dagger]\mathrm{not} \bullet K\big)_\dagger \equiv (\,\lambda x.S :_v K\,)_\dagger$$

Case of $\mu\alpha.S$:

$$K_\dagger[\mu\alpha.S] \xrightarrow{I.H.}{}_v^* K_\dagger[\mu\alpha.(S^\dagger)_\dagger]$$
$$\equiv (\lambda z.K_\dagger\{z\})\mu\alpha.(S^\dagger)_\dagger$$
$$\longrightarrow_{(\zeta)} (S^\dagger)_\dagger[{}^{(\lambda z.K_\dagger\{z\})\{-\}}/{}_{[\alpha]\{-\}}]$$
$$\xrightarrow{Lem\ 33}{}_v^* (S^\dagger[{}^K/{}_\alpha])_\dagger \equiv (\,\mu\alpha.S :_v K\,)_\dagger$$

Case of $[\alpha]O$:

$$[\alpha]O \equiv \alpha_\dagger\{O\} \xrightarrow{Lem\ 29}{}_v^* \alpha_\dagger[O]$$
$$\xrightarrow{I.H.}{}_v^* (O :_v \alpha)_\dagger \equiv (([\alpha]O)^\dagger)_\dagger$$

Case of $\delta(O, x.S, y.T)$:

$$\delta(O, x.S, y.T) \xrightarrow{I.H.}{}_v^* \delta\big(O, x.(S^\dagger)_\dagger, y.(T^\dagger)_\dagger\big)$$
$$\equiv \delta\big(O, x'.((x.S^\dagger)_\dagger\{x'\}),$$
$$y'.((y.T^\dagger)_\dagger\{y'\})\big)$$
$$\equiv [x.S^\dagger, y.T^\dagger]_\dagger\{O\}$$
$$\xrightarrow{Lem\ 29}{}_v^* [x.S^\dagger, y.T^\dagger]_\dagger[O]$$
$$\xrightarrow{I.H.}{}_v^* \big(O :_v [x.S^\dagger, y.T^\dagger]\big)_\dagger$$
$$\equiv (\delta(O, x.S, y.T)^\dagger)_\dagger$$

Case of MN(MN is a statement, and M is not a λ-abstraction):

$$MN \xrightarrow{I.H.}{}_v^* M(N^\dagger)_\dagger \equiv \mathrm{not}\langle N^\dagger\rangle_\dagger\{M\}$$
$$\xrightarrow{I.H.}{}_v^* (M :_v \mathrm{not}\langle N^\dagger\rangle)_\dagger \equiv (\,(MN)^\dagger\,)_\dagger$$

Case of $(\lambda x.S)N$:

$$(\lambda x.S)N \xrightarrow{I.H.}{}_v^* (\lambda x.(S^\dagger)_\dagger)N$$
$$\longrightarrow_v^* (x.S^\dagger)_\dagger[N]$$
$$\xrightarrow{I.H.}{}_v^* (N :_v x.S^\dagger)_\dagger \equiv \big(((\lambda x.S)N)^\dagger\big)_\dagger \quad \square$$

Proof of Proposition 36 (1) If we establish the following claims: (a) $M \bullet Q \longrightarrow^{n*} (M_\sharp :_n Q)$; (b) $O^\sharp \bullet K \longrightarrow^{n*} K_\sharp\{O\}$ if K is not a covalue; (c) $(O :_n P) \longrightarrow^{n*} (P_\sharp\{O\})^\sharp$; and (d) $S \longrightarrow^{n*} (S_\sharp)^\sharp$, we can easily obtain (1). Therefore, we show these claims by a simultaneous induction on M, K, P, and S.

Case of x: $x \bullet Q \equiv x :_n P$

Case of $\langle M\rangle\mathrm{inl}$:

$$\langle M\rangle\mathrm{inl} \bullet Q \xrightarrow{I.H.(a)}{}^{n*} \langle (M_\sharp)^\sharp\rangle\mathrm{inl} \bullet Q$$
$$\equiv \mathrm{inl}(M_\sharp) :_n Q \equiv \langle M\rangle\mathrm{inl}_\sharp :_n Q$$

The case of $\langle M\rangle\mathrm{inr}$ is shown similarly.

Case of $\langle M, N\rangle$:

$$\langle M, N\rangle \bullet Q \xrightarrow{I.H.(a)}{}^{n*} \langle (M_\sharp)^\sharp, (N_\sharp)^\sharp\rangle \bullet Q$$
$$\equiv (\langle M_\sharp, N_\sharp\rangle :_n Q) \equiv (\langle M, N\rangle_\sharp :_n Q)$$

Case of $[K]\mathrm{not}$:

$$[K]\mathrm{not} \bullet Q \longrightarrow^n_{(\eta_L)} [x.(x \bullet K)]\mathrm{not} \bullet Q$$
$$\longrightarrow^n_{(\eta_R)} [x.(x^\sharp \bullet K)]\mathrm{not} \bullet Q$$
$$\xrightarrow{I.H.(b)}{}^{n*} [x.(K_\sharp\{x\})^\sharp]\mathrm{not} \bullet Q$$
$$\equiv (\lambda x.(K_\sharp\{x\}) :_n Q) \equiv ([K]\mathrm{not}_\sharp :_n Q)$$

Case of $\lambda x.M$:

$$(\lambda x.M) \bullet P \xrightarrow{I.H.(a)}{}^{n*} (\lambda x.(M_\sharp)^\sharp) \bullet P$$
$$\equiv (\,\lambda x.M_\sharp :_n P\,) \equiv (\,(\lambda x.M)_\sharp :_n P\,)$$

Case of $S.\alpha$:

$$S.\alpha \bullet Q \xrightarrow{I.H.(d)}{}^{n*} (S_\sharp)^\sharp.\alpha \bullet Q$$
$$\longrightarrow^n_{(\beta_R)} (S_\sharp)^\sharp[{}^Q/{}_\alpha]$$
$$\equiv (\mu\alpha.S_\sharp :_n Q) \equiv ((S.\alpha)_\sharp :_n Q)$$

Case of α:

$$(O :_n \alpha) \equiv ([\alpha]O)^\sharp \equiv (\alpha_\sharp\{O\})^\sharp$$

Case of $[P, Q]$:

$$\begin{aligned}&(O :_n [P, Q]) \\ &\longrightarrow^{n*}_{(\eta_L)} (O :_n [x.(x \bullet P),\, y.(y \bullet Q)]) \\ &\overset{I.H.(c)}{\longrightarrow^{n*}} (O :_n [x.(P_\sharp\{x\})^\sharp,\, y.(Q_\sharp\{y\})^\sharp]) \\ &\longrightarrow^{n*} O^\sharp \bullet [x.(P_\sharp\{x\})^\sharp,\, y.(Q_\sharp\{y\})^\sharp] \\ &\equiv \delta(O, x.P_\sharp\{x\}, y.Q_\sharp\{y\})^\sharp \\ &\equiv ([P, Q]_\sharp\{O\})^\sharp\end{aligned}$$

Case of $\mathrm{fst}[P]$:

$$\begin{aligned}&(O :_n \mathrm{fst}[P]) \equiv (\mathrm{fst}(O) :_n P) \\ &\overset{I.H.(c)}{\longrightarrow^{n*}} (P_\sharp\{\mathrm{fst}(O)\})^\sharp \equiv (\mathrm{fst}[P]_\sharp\{O\})^\sharp\end{aligned}$$

The case of $\mathrm{snd}[P]$ can be shown similarly.

Case of $\mathrm{not}\langle M\rangle$:

$$\begin{aligned}&(O :_n \mathrm{not}\langle M\rangle) \overset{I.H.(a)}{\longrightarrow^{n*}} (O :_n \mathrm{not}\langle (M_\sharp)^\sharp\rangle) \\ &\longrightarrow^{n*} (OM_\sharp)^\sharp \equiv (\mathrm{not}\langle M\rangle_\sharp\{O\})^\sharp\end{aligned}$$

Case of $M@P$:

$$\begin{aligned}&(O :_n (M@P)) \overset{I.H.(a)}{\longrightarrow^{n*}} \big(O :_n ((M_\sharp)^\sharp @P)\big) \\ &\longrightarrow^{n*} (OM_\sharp :_n P) \overset{I.H.(c)}{\longrightarrow^{n*}} (P_\sharp\{OM_\sharp\})^\sharp \\ &\equiv (\,(M@P)_\sharp\{O\}\,)^\sharp\end{aligned}$$

Case of $[K, Q]$ (where $[K, Q]$ is not a covalue):

$$\begin{aligned}&(O^\sharp \bullet [K, L]) \\ &\longrightarrow^{n*}_{(\eta_L)} O^\sharp \bullet [x.(x \bullet K),\, y.(y \bullet L)] \\ &\overset{I.H.(b)}{\longrightarrow^{n*}} O^\sharp \bullet [x.(K_\sharp\{x\})^\sharp,\, y.(L_\sharp\{y\})^\sharp] \\ &\equiv \delta(O, x.K_\sharp\{x\}, y.L_\sharp\{y\}) \\ &\equiv ([K, L]_\sharp\{O\})^\sharp\end{aligned}$$

Case of $\mathrm{fst}[K]$ (where K is not a covalue):

$$\begin{aligned}&(O^\sharp \bullet \mathrm{fst}[K]) \\ &\longrightarrow^{n}_{(name)} (O^\sharp \bullet \mathrm{fst}[\alpha]).\alpha \bullet K \\ &\longrightarrow^{n*} (O :_n \mathrm{fst}[\alpha]).\alpha \bullet K \\ &\equiv (\mathrm{fst}(O) :_n \alpha).\alpha \bullet K \equiv (\mathrm{fst}(O))^\sharp \bullet K \\ &\overset{I.H.(b)}{\longrightarrow^{n*}} \big(K_\sharp\{\mathrm{fst}(O)\}\big)^\sharp \equiv \big(\mathrm{fst}[K]_\sharp\{O\}\big)^\sharp\end{aligned}$$

The case of $\mathrm{snd}[K]$ can be shown similarly.

Case of $M@K$ (where K is not a covalue):

$$\begin{aligned}&(O^\sharp \bullet (M@K)) \\ &\longrightarrow^{n}_{(name)} (O^\sharp \bullet (M@\alpha)).\alpha \bullet K \\ &\longrightarrow^{n*} (\,O :_n (M@\alpha)\,).\alpha \bullet K \\ &\overset{I.H.(a)}{\longrightarrow^{n*}} \big(\,O :_n ((M_\sharp)^\sharp @\alpha)\,\big).\alpha \bullet K \\ &\longrightarrow^{n*} (\,OM_\sharp :_n \alpha\,).\alpha \bullet K \\ &\equiv (OM_\sharp)^\sharp \bullet K \overset{I.H.(b)}{\longrightarrow^{n*}} (\,K_\sharp\{OM_\sharp\}\,)^\sharp \\ &\equiv (\,(M@K)_\sharp\{O\}\,)^\sharp\end{aligned}$$

Case of $x.S$:

$$\begin{aligned}&(O^\sharp \bullet x.S) \overset{I.H.(d)}{\longrightarrow^{n*}} (O^\sharp \bullet x.(S_\sharp)^\sharp) \\ &\longrightarrow^{n}_{(\beta_L)} (S_\sharp)^\sharp[^{O^\sharp}/_x] \\ &\overset{Lem\ 14(1)}{\longrightarrow^{n*}} (S_\sharp[^{O}/_x])^\sharp \equiv ((x.S)_\sharp\{O\})^\sharp\end{aligned}$$

Case of $M \bullet K$:

$$\begin{aligned}&(M \bullet K) \overset{I.H.(a)}{\longrightarrow^{n*}} (M_\sharp)^\sharp \bullet K \\ &\overset{I.H.}{\longrightarrow^{n*}} (\,K_\sharp\{M_\sharp\}\,)^\sharp \equiv ((M \bullet K)_\sharp)^\sharp\end{aligned}$$

(2) We show these claims by a simultaneous induction on M, K, and S. If we establish $(O :_v K) \longrightarrow^{v*} (K_\dagger\{O\})^\dagger$, then we can easily obtain $(O^\dagger \bullet K) \longrightarrow^{v*} (K_\dagger\{O\})^\dagger$. Therefore, we show this claim instead of the second clause of (2).

Case of x: $x \longrightarrow^{v}_{(\eta_R)} (x \bullet \alpha).\alpha \equiv (x :_v \alpha).\alpha \equiv x^\dagger \equiv (x_\dagger)^\dagger$

Case of $\langle M\rangle\mathrm{inl}$:

$$\begin{aligned}&\langle M\rangle\mathrm{inl} \overset{I.H.}{\longrightarrow^{v*}} \langle (M_\dagger)^\dagger\rangle\mathrm{inl} \\ &\longrightarrow^{v}_{(\eta_R)} (\langle (M_\dagger)^\dagger\rangle\mathrm{inl} \bullet \alpha).\alpha \\ &\equiv (\mathrm{inl}(M_\dagger) :_v \alpha).\alpha \equiv (\mathrm{inl}(M_\dagger))^\dagger \\ &\equiv (\langle M\rangle\mathrm{inl}_\dagger)^\dagger\end{aligned}$$

$\langle M\rangle\mathrm{inr}$ is shown similarly.

Case of $\langle M, N\rangle$:

$$\begin{aligned}&\langle M, N\rangle \overset{I.H.}{\longrightarrow^{v*}} \langle (M_\dagger)^\dagger,\, (N_\dagger)^\dagger\rangle \\ &\longrightarrow^{v}_{(\eta_R)} (\langle (M_\dagger)^\dagger,\, (N_\dagger)^\dagger\rangle \bullet \alpha).\alpha \\ &\equiv (\langle M_\dagger,\, N_\dagger\rangle :_v \alpha).\alpha \equiv \langle M_\dagger,\, N_\dagger\rangle^\dagger \\ &\equiv (\langle M,\, N\rangle_\dagger)^\dagger\end{aligned}$$

Case of $[K]\mathrm{not}$:

$$\begin{aligned}&[K]\mathrm{not} \longrightarrow^{v}_{(\eta_L)} [x.(x \bullet K)]\mathrm{not} \\ &\overset{I.H.}{\longrightarrow^{v*}} [x.(K_\dagger\{x\})^\dagger]\mathrm{not} \\ &\longrightarrow^{v}_{(\eta_R)} \big([x.(K_\dagger\{x\})^\dagger]\mathrm{not} \bullet \alpha\big).\alpha \\ &\equiv (\lambda x.K_\dagger\{x\} :_v \alpha).\alpha \\ &\equiv (\lambda x.K_\dagger\{x\})^\dagger \equiv ([K]\mathrm{not}_\dagger)^\dagger\end{aligned}$$

Case of $\lambda x.M$:

$$\begin{aligned}&\lambda x.M \overset{I.H.}{\longrightarrow^{v*}} \lambda x.(M_\dagger)^\dagger \\ &\longrightarrow^{v}_{(\eta_R)} (\,\lambda x.(M_\dagger)^\dagger \bullet \alpha\,).\alpha \\ &\equiv (\,\lambda x.M_\dagger :_v \alpha\,).\alpha \equiv (\,\lambda x.M_\dagger\,)^\dagger \\ &\equiv (\,(\lambda x.M)_\dagger\,)^\dagger\end{aligned}$$

Case of $S.\alpha$:

$$\begin{aligned} S.\alpha &\stackrel{I.H.}{\longrightarrow}{}^{v*} (S_\dagger)^\dagger.\alpha \equiv (S_\dagger)^\dagger[{}^\beta/_\alpha].\beta \\ &\equiv (\mu\alpha.S_\dagger :_v \beta).\beta \equiv (\mu\alpha.S_\dagger)^\dagger \\ &\equiv ((S.\alpha)_\dagger)^\dagger \end{aligned}$$

Case of α: $(O :_v \alpha) \equiv ([\alpha]O)^\dagger \equiv (\alpha_\dagger\{O\})^\dagger$

Case of $[K, L]$:

$$\begin{aligned} &(O :_v [K, L]) \\ &\longrightarrow^{v*}_{(\eta_L)} (O :_v [x.(x \bullet K),\, y.(y \bullet L)]) \\ &\stackrel{I.H.}{\longrightarrow}{}^{v*} (O :_v [x.(K_\dagger\{x\})^\dagger,\, y.(L_\dagger\{y\})^\dagger]) \\ &\equiv \delta(O, x.K_\dagger\{x\}, y.L_\dagger\{y\})^\dagger \\ &\equiv ([K, L]_\dagger\{O\})^\dagger \end{aligned}$$

Case of $\mathrm{fst}[K]$:

$$\begin{aligned} &(O :_v \mathrm{fst}[K]) \equiv (\mathrm{fst}(O) :_v K) \\ &\stackrel{I.H.}{\longrightarrow}{}^{v*} (K_\dagger\{\mathrm{fst}(O)\})^\dagger \\ &\stackrel{Lem\ 29}{\longrightarrow}{}^{v*} (K_\dagger[\mathrm{fst}(O)])^\dagger \equiv (\mathrm{fst}[K]_\dagger\{O\})^\dagger \end{aligned}$$

$\mathrm{snd}[K]$ can be shown similarly.

Case of $\mathrm{not}\langle M\rangle$:

$$\begin{aligned} &(O :_v \mathrm{not}\langle M\rangle) \stackrel{I.H.}{\longrightarrow}{}^{v*} (O :_v \mathrm{not}\langle (M_\dagger)^\dagger\rangle) \\ &\longrightarrow^{v*} (OM_\dagger)^\dagger \equiv (\mathrm{not}\langle M\rangle_\dagger\{O\})^\dagger \end{aligned}$$

Case of $M@K$:

$$\begin{aligned} &(O :_v (M@K)) \stackrel{I.H.}{\longrightarrow}{}^{v*} \big(O :_v ((M_\dagger)^\dagger@K)\big) \\ &\longrightarrow^{v*} (OM_\dagger :_v K) \stackrel{I.H.}{\longrightarrow}{}^{v*} (K_\dagger\{OM_\dagger\})^\dagger \\ &\stackrel{Lem\ 29}{\longrightarrow}{}^{v*} (K_\dagger[OM_\dagger])^\dagger \equiv ((M@K)_\dagger\{O\})^\dagger \end{aligned}$$

Case of $x.S$:

$$\begin{aligned} &(O :_v \bullet x.S) \stackrel{I.H.}{\longrightarrow}{}^{v*} (O :_v x.(S_\dagger)^\dagger) \\ &\equiv ((\lambda x.S_\dagger)O)^\dagger \longrightarrow^{v*} ((x.S)_\dagger\{O\})^\dagger \end{aligned}$$

Case of $M \bullet K$:

$$\begin{aligned} &(M \bullet K) \stackrel{I.H.}{\longrightarrow}{}^{v*} (M_\dagger)^\dagger \bullet K \\ &\longrightarrow^{v}_{(\beta_L)} (M_\dagger :_v K) \stackrel{I.H.}{\longrightarrow}{}^{v*} (K_\dagger\{M_\dagger\})^\dagger \\ &\stackrel{Lem\ 29}{\longrightarrow}{}^{v*} (K_\dagger[M_\dagger])^\dagger \equiv ((M \bullet K)_\dagger)^\dagger \qquad \square \end{aligned}$$

Proof of Proposition 37 (1) We show the first line of (1). Suppose $\lambda\mu \vdash O \longrightarrow^*_n M$ and $\lambda\mu \vdash O \longrightarrow^*_n M'$, then $DC \vdash (O)^\sharp \longrightarrow^{n*} (M)^\sharp$ and $DC \vdash (O)^\sharp \longrightarrow^{n*} (M')^\sharp$ by Theorem 16. By the Church-Rosser property of the dual calculus, there is a term N of the dual calculus such that $DC \vdash (M)^\sharp \longrightarrow^{n*} N$ and $DC \vdash (M')^\sharp \longrightarrow^{n*} N$. Hence we have $\lambda\mu \vdash ((M)^\sharp)_\sharp \longrightarrow^*_n (N)_\sharp$ and $\lambda\mu \vdash ((M')^\sharp)_\sharp \longrightarrow^*_n (N)_\sharp$ by Theorem 28. Therefore we obtain $\lambda\mu \vdash M \longrightarrow^*_n (N)_\sharp$ and $\lambda\mu \vdash M' \longrightarrow^*_n (N)_\sharp$ by Proposition 35. The second line of (1) is shown similarly. (2) is also shown in a way similar to (1). $\square$

Proof of Theorem 40 (1) is shown by using Theorem 16, Theorem 34, and Proposition 38.
(2) is shown by using Theorem 21, Theorem 28, and Proposition 38.
(3) follows from Proposition 35, 36, and 38. We show the third line first.

$$\begin{aligned} S &\stackrel{Prop\ 35(1)}{\longrightarrow^*_n} (S^\sharp)_\sharp \equiv (S^{\sharp\circ\circ})_\sharp \\ &\stackrel{Prop\ 36(2)}{\longrightarrow^*_n} (((S^{\sharp\circ})_\dagger)^{\dagger\circ})_\sharp \equiv (S_\circ)_\bullet \end{aligned}$$

The second line is shown as follows.

$$\begin{aligned} &O_\bullet\{M\} \equiv (O^{\dagger\circ})_\sharp\{M\} \\ &\stackrel{Prop\ 35(1)}{\longrightarrow^*_n} (O^{\dagger\circ})_\sharp\{(M^\sharp)_\sharp\} \\ &\equiv (M^\sharp \bullet O^{\dagger\circ})_\sharp \equiv (M^{\sharp\circ\circ} \bullet O^{\dagger\circ})_\sharp \\ &\equiv ((O^\dagger \bullet M^{\sharp\circ})^\circ)_\sharp \\ &\stackrel{Prop\ 36(2)}{\longrightarrow^*_n} (((M^{\sharp\circ})_\dagger\{O\})^{\dagger\circ})_\sharp \equiv (M_\circ\{O\})_\bullet \end{aligned}$$

The first line follows from the second line.

$$\begin{aligned} &M \longrightarrow_{(\eta_\mu)} \mu\alpha.[\alpha]M \\ &\stackrel{(*)}{\equiv} \mu\alpha.\alpha_\bullet\{M\} \longrightarrow^*_n \mu\alpha.(M_\circ\{\alpha\})_\bullet \end{aligned}$$

$(*)$ is shown by

$$\begin{aligned} &\alpha_\bullet\{M\} \equiv (\alpha^{\dagger\circ})_\sharp\{M\} \equiv ((\alpha \bullet \beta).\beta)^\circ)_\sharp\{M\} \\ &\quad \equiv (\beta.(\beta \bullet \alpha))_\sharp\{M\} \equiv ([\alpha]\beta)[{}^M/_\beta] \equiv [\alpha]M. \end{aligned}$$

(4) can be shown in a way similar to (3). $\square$

(Received August 21, 2006)
(Accepted January 9, 2007)
(Online version of this article can be found in the IPSJ Digital Courier, Vol.3, pp.207–243.)

Daisuke Kimura was born in 1977. He received his M.S. degree from Kyoto University in 2003. He has been a Ph.D. student at the Graduate University for Advanced Studies since 2003. He has done research area on theoretical computer science and mathematical logic. His current research interest is type theory for classical logic. He is a member of IPSJ and JSSST.