A Proof Theoretical Account of Continuation Passing Style

Ichiro Ogata

Information Technology Research Institute
AIST, JAPAN
i.ogata@aist.go.jp

Motivation:

- To give a proof-theoretical account of Continuation Passing Style.
- To show Gentzen's sequent style is *good* syntax.

Main Results:

The CBV normalization for **CND** (Parigot 92) can be simulated by the cut-elimination procedure for **LKQ** (Danos-Joinet-Schellinx 93), namely the q-protocol.

Theorem 1. *If* $p \Rightarrow q$ *then* $\overline{p} \Rightarrow \overline{q}$

A Proof Theoretical Account of CPS

CPS translation is the one from Natural Deduction style logic to Gentzen's sequent style logic.

- The source language of CBV CPS is **CND** with CBV normalization procedure.
- The target language of CBV CPS is **LKQ**.
- The translation from **CND** into **LKQ** is the CBV CPS translation.

- Different from the traditional one, our translation does not raise translation on types.
- Plotkin-style and Hofmann-Streicher-style CPS translation can be recovered by considering *intuitionistic decoration* of **LKQ**.

What is **CND**? (Parigot 92)

- **CND**(Classical Natural Deduction) has classical provability.
- **CND** is a “classical” extension of natural deduction.
- Formation rules take the form of left and right introduction.

- $\lambda\mu_{\mathrm{n}}$ is the term calculus of **CND**.
- $\lambda\mu_{\mathrm{n}}$ is a classical extension to CBN λ-calculus.
- $\lambda\mu_{\mathrm{n}}$ has a SN and CR normalization procedure.

What is Call-By-Value **CND**? (Ong 96)

- **CND**'s term calculus is $\lambda\mu_{\mathrm{V}}$.
- $\lambda\mu_{\mathrm{V}}$ is a classical extension to CBV λ-calculus.
- Our $\lambda\mu_{\mathrm{V}}$ is a variant of Ong's.
- We also give a new, simple proof of SN and CR property of normalization.

Typical **CND** sequents with term assignment:

$$M: \quad A^x, B^y, (A \to B)^z \Rightarrow D, {A'}^{\alpha}, {B'}^{\beta}, (A' \to B')^{\gamma}$$

$$\frac{p: \quad A^x, \mathit{\Gamma} \Rightarrow B, \mathit{\Delta}}{\lambda x^A.p: \quad \mathit{\Gamma} \Rightarrow A \to B, \mathit{\Delta}} \to\mathcal{I}$$

$$\frac{M: \quad \mathit{\Gamma}_0 \Rightarrow A \to B, \mathit{\Delta}_0 \qquad N: \quad \mathit{\Gamma}_1 \Rightarrow A, \mathit{\Delta}_1}{M\,N: \quad \mathit{\Gamma}_0, \mathit{\Gamma}_1 \Rightarrow B, \mathit{\Delta}_0, \mathit{\Delta}_1} \to\mathcal{E}$$

What is **LKQ**? (Danos, Joinet and Schellinx 93)

– **LKQ** has classical provability.

– **LKQ** is a Gentzen's sequent-style classical logic.

– Formation rules take the form of left and right introduction.

– **LKQ** is a variant of Gentzen's **LK**.

Typical **LKQ** sequents with term assignment:

$$T: \quad A^x, B^y, (A \to B)^z \Rightarrow A'^{\alpha}, B'^{\beta}, (A' \to B')^{\gamma}\,;\; C$$

$$\frac{T: \quad A^x \Rightarrow B^{\beta}\,;\; \emptyset}{\overline{\lambda} x^A . \overline{\mu} \beta^B . T: \quad \Rightarrow\,;\; A \to B}\ \mathrm{R}{\to}$$

$$\frac{V: \quad \Rightarrow\,;\; A \qquad U: \quad B^y \Rightarrow\,;\; \emptyset}{\text{let } y = zV \text{ in } U: \quad (A \to B)^z \Rightarrow\,;\; \emptyset}\ \mathrm{L}{\to}$$

Curry-Howard isomorphism:

- Proofs of **CND** as programs.
- The normalization procedure as computation.

- Proofs of **LKQ** as programs.
- The cut-elimination procedure (i.e., the q-protocol) as computation.

Gentzen's sequent-style classical is good syntax.

– 2nd order extension is well-known.

– There also is a well-known algebraic semantics.

CBV Evaluation Context:

We define **CBV evaluation contexts**, ranged over by E, as follows:

$$E := [\text{-}] \mid EN \mid vE \mid EB$$

Traditionally, evaluation contexts are devised so that every non-normal closed term M can be written uniquely as $E[R]$, where R is a redex.

Instead, we use the notion of evaluation context to uniquely define a ζ-redex in every μ-rename.

Reduction Rules for $\lambda\mu_{\mathrm{V}}$:

$$\begin{array}{lll}
(\beta_{\mathrm{V}}) & (\lambda x.L)v & \Rightarrow\; L\,[x := v] \\
(\zeta_{\mathrm{V}}) & \mu\alpha.[\beta]\,E[\mu\gamma.[\delta]\,M] & \Rightarrow\; \mu\alpha.([\delta]\,M)\,[\gamma := [\beta]\,E[\text{-}]] \\
(\text{polymorphic}) & (\mathit{\Lambda} X.L)B & \Rightarrow\; L\,[X := B]
\end{array}$$

Where $M\,[\gamma := [\beta]\,E[\text{-}]]$ means “in M, replace all subterms of the form $[\gamma]\,L$ by the term $[\beta]\,E[L]$”.

Difference from Ong-Stewart’s CBV $\lambda\mu_{\mathrm{V}}$(Ong96):

We pack $n+1$ length of “reduction sequence” into single reduction.

Simulating CBV language by $\lambda\mu_{\mathrm{V}}$

The CBV language we consider is typed CBV λ-calculus with continuations(catch/throw) called λ_{V}. Its reduction rules are:

$$\begin{aligned}
(\lambda x.L)v &\Rightarrow \text{catch } \beta\ L\,[x := v] \qquad\qquad \beta \notin \mathrm{FN}(L\,[x := v])\\
\text{catch } \beta\ E[\text{catch } \alpha\ M] &\Rightarrow \text{catch } \beta\ E[M\ [\alpha := [\beta]\,E[\text{-}]]]\\
\text{throw } \beta\ E[\text{catch } \alpha\ M] &\Rightarrow \text{throw } \beta\ E[M\ [\alpha := [\beta]\,E[\text{-}]]]\\
\text{catch } \alpha\ E_e[\text{throw } \alpha\ N] &\Rightarrow \text{catch } \alpha\ N\\
\text{throw } \beta\ E_e[\text{throw } \alpha\ N] &\Rightarrow \text{throw } \alpha\ N
\end{aligned}$$

The embedding: $\ulcorner\_\urcorner : \lambda_{\mathrm{V}}$-terms $\mapsto$ $\lambda\mu_{\mathrm{V}}$-terms is defined as follows:

$$\begin{gathered}
\ulcorner x \urcorner = x \\
\ulcorner \lambda x^A\!.L \urcorner = \lambda x^A\!.\mu\beta^B\!.[\beta]\ulcorner L \urcorner \\
\ulcorner MN \urcorner = \ulcorner M \urcorner \ulcorner N \urcorner \\
\ulcorner \text{catch } \alpha^A N \; \urcorner = \mu\alpha^A\!.[\alpha]\ulcorner N \urcorner \\
\ulcorner \text{throw } \alpha \; N \urcorner = \mu\delta^B\!.[\alpha]\ulcorner N \urcorner
\end{gathered}$$

Where $\beta \notin \mathrm{FN}(\ulcorner L \urcorner)$ and $\delta \notin \mathrm{FN}(\ulcorner N \urcorner)$.

Theorem 2. *If* $M \Rightarrow N$ *then* $\ulcorner M \urcorner \Rightarrow \ulcorner N \urcorner$

Reduction Rules for $\overline{\lambda\mu}_{\mathrm{q}}$:

$$\begin{array}{lll}
(\mathrm{S1}) & \mathrm{let}\ x = \overline{\mu}\alpha.S\ \mathrm{in}\ T & \Rightarrow S\,[\alpha := (\mathrm{let}\ x = \_\ \mathrm{in}\ T)] \\
(\mathrm{S2}) & \mathrm{let}\ x = V\ \mathrm{in}\ S & \Rightarrow S\,[x := V] \\
(\mathrm{L}_{\rightarrow}) & (\overline{\lambda}x^{A}.\overline{\mu}\beta^{B}.T)V & \Rightarrow \overline{\mu}\beta^{B}.(\mathrm{let}\ x = V\ \mathrm{in}\ T) \\
(\mathrm{L}_{\forall}) & (\overline{\mathit{\Lambda}}X.\overline{\mu}\alpha^{A}.T)B & \Rightarrow \overline{\mu}\alpha^{A}.T\,[X := B]
\end{array}$$

Simulation of $\lambda\mu_{\mathrm{V}}$ by $\overline{\lambda\mu}_{\mathrm{q}}$:

1. The translation $\overline{(\_)}$, $\lambda\mu_{\mathrm{V}}$-μ-renames $\mapsto$ $\overline{\lambda\mu}_{\mathrm{q}}$-$\overline{\mu}$-abstractions is defined as follows:

$$\overline{\mu\beta^B.[\alpha]\,N} = \overline{\mu}\beta^B.(N : \mathrm{let}\ x = \_\ \mathrm{in}\ [\alpha]\,x)$$

2. The infix operator :, $\lambda\mu_{\mathrm{V}}$-terms$\times\overline{\lambda\mu}_{\mathrm{q}}$-contexts $\mapsto$ $\overline{\lambda\mu}_{\mathrm{q}}$-contexts is defined as follows:

$$\begin{array}{rcc}
v : \mathrm{let}\ \ y = \_\ \ \mathrm{in}\ \ S & = & S\,[y := \Psi(v)] \\
\mu\alpha.[\beta]\,M : \mathrm{let}\ \ y = \_\ \ \mathrm{in}\ \ S & = & \mathrm{let}\ \ y = \overline{\mu\alpha.[\beta]\,M}\ \ \mathrm{in}\ \ S \\
MN : \mathrm{let}\ \ y = \_\ \ \mathrm{in}\ \ S & = & M : \mathrm{let}\ \ z = \_\ \ \mathrm{in}\ \ (N : \mathrm{let}\ \ x = \_\ \ \mathrm{in}\ \ (\mathrm{let}\ \ y = zx\ \ \mathrm{in}\ \ S)) \\
MB : \mathrm{let}\ \ y = \_\ \ \mathrm{in}\ \ S & = & M : \mathrm{let}\ \ z = \_\ \ \mathrm{in}\ \ (\mathrm{let}\ \ y = zB\ \ \mathrm{in}\ \ S)
\end{array}$$

3. An auxiliary function Ψ, $\lambda\mu_{\mathrm{V}}$-values $\mapsto$ $\overline{\lambda\mu}_{\mathrm{q}}$-values, is defined as follows:

$$\Psi(x) = x; \quad \Psi(\lambda x.p) = \overline{\lambda}x.\overline{p}; \quad \Psi(\Lambda X.p) = \overline{\Lambda}X.\overline{p}$$

Proof theoretically, this translation is based on Prawitz's observation to simulate natural deduction-style by Gentzen's sequent style logic (Prawitz65).

For example, the application pq can be written in **LKQ** as follows:

$$\dfrac{U:\ \Rightarrow (A\to B)^{\gamma}\,;\ \emptyset \qquad \dfrac{S:\ \Rightarrow A^{\alpha}\,;\ \emptyset \qquad \dfrac{x:\quad A^{x} \Rightarrow\ ;\ A \qquad [\beta]\,y:\quad B^{y} \Rightarrow B^{\beta}\,;\ \emptyset}{\text{let }\ y = zx\ \text{ in }\ [\beta]\,y:\quad A^{x}, (A \to B)^{z} \Rightarrow B^{\beta}\,;\ \emptyset}\,\text{L}\!\to}{\text{let }\ x = \overline{\mu}\alpha^{A}.S\ \text{ in }\ (\text{let }\ y = zx\ \text{ in }\ [\beta]\,y):\quad (A \to B)^{z} \Rightarrow B^{\beta}\,;\ \emptyset}\,\text{mid}}{\text{let }\ z = \overline{\mu}\gamma^{A\to B}.U\ \text{ in }\ (\text{let }\ x = \overline{\mu}\alpha^{A}.S\ \text{ in }\ (\text{let }\ y = zx\ \text{ in }\ [\beta]\,y)):\quad \Rightarrow B^{\beta}\,;\ \emptyset}\,\text{mid}$$

where $\overline{p} = \overline{\mu}\gamma.U$ and $\overline{q} = \overline{\mu}\alpha.S$.

$$
\begin{prooftree}
\AxiomC{}
\RightLabel{Ax}
\UnaryInfC{$A \Rightarrow\ ;\ A$}
\AxiomC{}
\RightLabel{Ax}
\UnaryInfC{$B \Rightarrow\ ;\ B$}
\RightLabel{D}
\UnaryInfC{$B \Rightarrow B\ ;$}
\RightLabel{L$\to$}
\BinaryInfC{$A, A \to B \Rightarrow B\ ;$}
\AxiomC{$\vdots$}
\noLine
\UnaryInfC{$\Rightarrow A\ ;$}
\RightLabel{mid}
\BinaryInfC{$A \to B \Rightarrow B\ ;$}
\AxiomC{$\vdots$}
\noLine
\UnaryInfC{$\Rightarrow A \to B\ ;$}
\RightLabel{mid}
\BinaryInfC{$\Rightarrow B\ ;$}
\end{prooftree}
$$

$$
\begin{prooftree}
\AxiomC{}
\RightLabel{Ax}
\UnaryInfC{$A \Rightarrow\ ;\ A$}
\AxiomC{}
\RightLabel{Ax}
\UnaryInfC{$B \Rightarrow\ ;\ B$}
\RightLabel{D}
\UnaryInfC{$B \Rightarrow B\ ;$}
\RightLabel{L$\to$}
\BinaryInfC{$A, A \to B \Rightarrow B\ ;$}
\AxiomC{$\vdots$}
\noLine
\UnaryInfC{$\Rightarrow A \to B\ ;$}
\RightLabel{mid}
\BinaryInfC{$A \Rightarrow B\ ;$}
\AxiomC{$\vdots$}
\noLine
\UnaryInfC{$\Rightarrow A\ ;$}
\RightLabel{mid}
\BinaryInfC{$\Rightarrow B\ ;$}
\end{prooftree}
$$

There are two ways to map **CND** into **LKQ**:

We choose the ζ_{V}-redex in the application from left-to-right(LR) order. Of course, the opposite order should be studied in its own right.

This phenomena is known in the previous study of CPS; the CBV right-to-left(RL) evaluation method. This kind of CPS-translation was shown, for example, by Murthy(Murthy92).

RL reduction system:

RL Evaluation Context:

$$E := [\text{-}] \mid ME \mid Ev \mid EB$$

RL Translation:

$$MN : \text{let } y = \_ \text{ in } S = N : \text{let } x = \_ \text{ in } (M : \text{let } z = \_ \text{ in } (\text{let } y = zx \text{ in } S))$$

Danos-Joinet-Schellinx's theory give a proof theoretical explanation to this phenomenon – they say there are two ways to map **LK** derivations to **LKQ** derivations.

Intuitionistic Decoration:

One can embed **LKQ** into (multiplicative) **LJ** by means of **intuitionistic decoration** method. Formulas of **LJ** are that of second order propositional logic constructed from $\cap$ (multiplicative conjunction) and $\supset$(multiplicative implication). The introduced **negations** are implications; it is defined as $\overset{\phi}{\neg} A \equiv A \supset \phi$, where ϕ is an arbitrary fixed formula.

Definition 1 (intuitionistic decoration on formulas). *Intuitionistic decoration on formulas are defined as follows:*

$$\begin{array}{cc}
A^q := A \ \ (\textit{for } A \textit{ atomic}) & (A \to B)^q := \overset{\phi}{\neg}(A^q \cap \overset{\phi}{\neg} B^q) \\
(A \vee B)^q := \overset{\phi}{\neg}(\overset{\phi}{\neg} A^q \cap \overset{\phi}{\neg} B^q) & (A \wedge B)^q := A^q \cap B^q \\
(\forall X.A)^q := \forall X. \overset{\phi}{\neg}\overset{\phi}{\neg} A^q &
\end{array}$$

Definition 2 (intuitionistic decoration on sequents). *Intuitionistic decoration on sequents are defined as follows:*

$$\mathbf{LKQ} \vdash \mathit{\Gamma} \Rightarrow \mathit{\Delta}\,;\ \mathit{\Pi} \qquad \textit{iff} \qquad \mathbf{LJ} \vdash \mathit{\Gamma}^q, \overset{\phi}{\neg} \mathit{\Delta}^q \Rightarrow \mathit{\Pi}^q$$

The direct translation from $\lambda\mu_{\mathrm{V}}$ to λ_{V} allows us to recover Hofmann-Streicher-style CPS-translation as follows:

Definition 3 (Hofmann-Streicher-style CPS-translation).

1. *The translation* $\overline{(\_)}, \lambda\mu_{\mathcal{V}}$*-*$\mu$*-renames* $\mapsto \lambda_{\mathcal{V}}$*-values is defined as follows:*

$$\overline{\mu\alpha.[\beta]\,M} = \lambda\alpha.(M : \lambda y.\beta y)$$

2. *The infix operator* $:, \lambda\mu_{\mathcal{V}}$*-terms*$\times\lambda_{\mathcal{V}}$*-abstractions* $\mapsto \lambda_{\mathcal{V}}$*-applications is defined as follows:*

$$\begin{array}{rcl} v : \lambda y.S & = & S\,[y := \Psi(v)] \\ \mu\alpha.[\beta]\,M : \lambda y.S & = & (\overline{\mu\alpha.[\beta]\,M})\lambda y.S \\ MN : \lambda y.S & = & M : \lambda z.(N : \lambda x.z\langle x, \lambda y.S\rangle) \\ MB : \lambda y.S & = & M : \lambda z.z\langle B, \lambda y.S\rangle \end{array}$$

3. *An auxiliary function* $\Psi, \lambda\mu_{\mathcal{V}}$*-values* $\mapsto \lambda_{\mathcal{V}}$*-values, is defined as follows:*

$$\Psi(x) = x; \quad \Psi(\lambda x.p) = \lambda x.\overline{p}; \quad \Psi(\Lambda X.p) = \Lambda X.\overline{p}$$

Plotkin's' modified CPS-translation:

Above translation can be thought of a variant of Plotkin's modified CPS-translation(Plotkin 75). What is new here is that we use two colons on the translation.

RL system in Hofmann-Streicher-style:

Of course, RL translation is also possible:

$$MN : \lambda y.S = N : \lambda x.(M : \lambda z.z\langle x, \lambda y.S\rangle)$$

Semantics:

Let $(A, \cap, 1)$ with partial order $(\leq)$ be a meet-semilattice. A Heyting semilattice $(A, \cap, 1, \supset)$ is a meet-semilattice with an additional binary operation $\supset$ satisfying

$$a \cap b \leq c \Leftrightarrow a \leq b \supset c$$

Moreover, we have distinguished element $\phi \in A$. We define pseudonegation as follows:

$$\overset{\phi}{\neg} a \overset{\text{def}}{=} a \supset \phi$$

Now, we define following operations:

$$\begin{aligned}
a^* &= a \\
(a \rightarrow b)^* &= \overset{\phi}{\neg}(a^* \cap \overset{\phi}{\neg} b^*) \\
(a \wedge b)^* &= a^* \cap b^* \\
(a \vee b)^* &= \overset{\phi}{\neg}(\overset{\phi}{\neg} a^* \cap \overset{\phi}{\neg} b^*) \\
(\neg a)^* &= \overset{\phi}{\neg}(a^*) \\
1^* &= 1 \\
0^* &= \phi \\
(a \ll b)^* &= \overset{\phi}{\neg} b^* \leq \overset{\phi}{\neg} a^*
\end{aligned}$$

$(A^*, \wedge, \vee, 1, 0, \neg)$ is a boolean algebra with partial order $\ll$.

- Every proof made in A^* (as boolean algebra) can be translated into the proof in A(as Heyting-semilattice).

- As before, there are two ways to map A^* proof into A proof. We cannot distinguish two proofs in algebraic method. Algebra don't care a bit about the order of application of transitive law.

- However, the order of arrow composition matters in category theory — we may have different arrows according to the order of composition.

- Hence A^* raise a co-control category (Selinger01). Specifically, $\rightarrow$ and $\wedge$ are **binoidal functors** in co-control category. That is, they are not **bifunctors**.

$$\frac{}{x:\quad A^{x} \Rightarrow A}\,\text{Ax}$$

$$\frac{N:\quad \Rightarrow A, \mathit{\Delta}}{\mu\beta^{B}.[\alpha]\,N:\quad \Rightarrow B, ((A^{\alpha}, \mathit{\Delta}) \setminus B^{\beta})}\,\text{rename}$$

$$\frac{p:\quad A^{x} \Rightarrow B}{\lambda x^{A}.p:\quad \Rightarrow A \to B}\,{\to}\mathcal{I}$$

$$\frac{M:\quad \Rightarrow A \to B \qquad N:\quad \Rightarrow A}{M\,N:\quad \Rightarrow B}\,{\to}\mathcal{E}$$

$$\frac{p[X := Y]:\quad \Rightarrow A[X := Y]}{\mathit{\Lambda}X.p:\quad \Rightarrow \forall X.A}\,\forall^{2}\mathcal{I}^{*}$$

$$\frac{M:\quad \Rightarrow \forall X.A}{M\,B:\quad \Rightarrow A[X := B]}\,\forall^{2}\mathcal{E}$$

Table 1. $\lambda\mu_{\mathrm{V}}$-term Assignment for **CND**

$$\frac{}{x : \quad A^x \Rightarrow \,;\, A}\ \mathrm{Ax}$$

$$\frac{V : \quad \Rightarrow \,;\, A}{[\alpha]V : \quad \Rightarrow A^\alpha \,;\, \emptyset}\ \mathrm{D}$$

$$\frac{V : \quad \Rightarrow \,;\, A \qquad\qquad U : \quad B^y \Rightarrow \,;\, \emptyset}{\text{let } y = zV \text{ in } U : \quad (A \to B)^z \Rightarrow \,;\, \emptyset}\ \mathrm{L}{\to}$$

$$\frac{T : \quad A^x \Rightarrow B^\beta \,;\, \emptyset}{\overline{\lambda} x^A . \overline{\mu} \beta^B . T : \quad \Rightarrow \,;\, A \to B}\ \mathrm{R}{\to}$$

Table 2. $\overline{\lambda\mu}_{\mathrm{q}}$-term Assignment for **LKQ**

$$\frac{V : \ \Rightarrow\ ;\ A \qquad T : \ A^x \Rightarrow\ ;\ \emptyset}{\text{let } x = V \text{ in } T : \ \Rightarrow\ ;\ \emptyset}\,\text{tail}$$

$$\frac{S : \ \Rightarrow A^\alpha ;\ \emptyset \qquad T : \ A^x \Rightarrow\ ;\ \emptyset}{\text{let } x = \overline{\mu}\alpha.S \text{ in } T : \ \Rightarrow\ ;\ \emptyset}\,\text{mid}$$

$$\frac{U : \ (A[X := B])^x \Rightarrow\ ;\ \emptyset}{\text{let } x = zB \text{ in } U : \ (\forall X.A)^z \Rightarrow\ ;\ \emptyset}\,\mathrm{L}\forall^2$$

$$\frac{T[X := Y] : \ \Rightarrow (A[X := Y])^\alpha ;\ \emptyset}{\overline{\Lambda}X.\overline{\mu}\alpha^A.T : \ \Rightarrow\ ;\ \forall X.A}\,\mathrm{R}\forall^{2*}$$

Table 3. $\overline{\lambda\mu}_{\mathrm{q}}$-term Assignment for **LKQ** — continued

$$\frac{S:\quad \Rightarrow A \qquad\qquad U:\quad B^{y} \Rightarrow C}{(\lambda y^{B}.U)(zx):\quad z^{(A\supset B)} \Rightarrow C}\,\mathrm{L}\supset$$

$$\frac{T:\quad A^{x} \Rightarrow B}{\lambda x^{A}.T:\quad \Rightarrow A \supset B}\,\mathrm{R}\supset$$

$$\frac{T:\quad A^{x}, B^{y} \Rightarrow C}{T:\quad (A \cap B)^{\langle x,y\rangle} \Rightarrow C}\,\mathrm{L}\cap$$

$$\frac{S:\quad \Rightarrow A \qquad\qquad T:\quad \Rightarrow B}{\langle S,T\rangle:\quad \Rightarrow A \cap B}\,\mathrm{R}\cap$$

Table 4. λ-term Assignment for **LJ**

$$\frac{}{x: \quad A^x \Rightarrow A}\,\text{Ax}$$

$$\frac{S: \quad \Rightarrow A \qquad\quad T: \quad A^x \Rightarrow C}{(\lambda x^A.T)S: \quad \Rightarrow C}\,\text{cut}$$

$$\frac{U: \quad (A[X:=B])^x \Rightarrow C}{(\lambda x^{A[X:=B]}.U)(zB): \quad (\forall X.A)^z \Rightarrow C}\,\text{L}\forall^2$$

$$\frac{T[X:=Y]: \quad \Rightarrow A[X:=Y]}{\mathit{\Lambda} X.T: \quad \Rightarrow \forall X.A}\,\text{R}\forall^{2*}$$

Table 5. λ-term Assignment for **LJ** — continued