# A Framework for Equivalence of Cast Calculi

by Lu Kuang-Chen

## 1 Definitions

**Definition 1.** *$s \approx^* s'$ if and only if there exists a state $t$ such that $s \longrightarrow^* t$ and $t \approx s'$. Intuitively, $s \approx^* s'$ means $s$ will be related to $s'$.*

**Notation 2.** *$[\![c]\!]$ means $\lambda V.\mathrm{applyCast}(V, c)$.*

**Lemma 3.** *If $U \approx U' : A$ then $[\![A \Rightarrow^l B]\!](U) \approx [\![\lceil A \Rightarrow^l B \rceil]\!](U') : B$.*

**Definition 4.** $\mathrm{extCont}(c, [\Box\langle d \rangle]k) = [\Box\langle \mathrm{seq}(c, d)\rangle]k$

**Lemma 5.** $\mathrm{extCont}(\mathrm{seq}(c, d), \kappa) = \mathrm{extCont}(c, \mathrm{extCont}(d, \kappa))$

**Lemma 6.** *If $\kappa \approx \kappa'$ then $[\Box\langle A \Rightarrow^l B \rangle]\kappa \approx \mathrm{extCont}(\lceil A \Rightarrow^l B \rceil, \kappa')$*

**Lemma 7.** *For all $U \approx U'$ and $V \approx V'$ and $k \approx k'$ then $\mathrm{doApp}(U, V, \kappa) \approx^\star \mathrm{doApp}'(U', V', \kappa')$.*

**Proof.** By induction on $U \approx U'$. There are two cases.

**Case** $\langle \lambda x : A.\, e : B, E\rangle \approx \langle \lambda x : \mathrm{id}(A).\, e : \mathrm{id}(B), E'\rangle$

In this case

$$\begin{array}{rcl} & & \mathrm{doApp}(U, V, k) \\ & = & \mathrm{doApp}(\langle \lambda x : A.\, e : B, E\rangle, V, \kappa) \\ & = & \langle e, E[x \leftarrow V], \kappa\rangle \end{array}$$

$$\begin{array}{rcl} & & \mathrm{doApp}'(U', V', \kappa') \\ & = & \mathrm{doApp}'(\langle \lambda x : \mathrm{id}(A).\, e : \mathrm{id}(B), E'\rangle, V', \kappa') \\ & = & [\![\mathrm{id}(A)]\!](V') \gg \lambda V'.\langle e, E'[x \leftarrow V'], \kappa'\rangle \\ (\star) & = & \underline{\mathrm{Just}\, V'} \gg \lambda V'.\langle e, E'[x \leftarrow V'], \kappa'\rangle \\ & = & \langle e, E'[x \leftarrow V'], \kappa'\rangle \end{array}$$

The step marked with $(\star)$ follows from the basic property of $\mathrm{id}(\cdot)$, that is

$$[\![\mathrm{id}(A)]\!](V) = \mathrm{Just}\, V$$

By definition, $\langle e, E[x \leftarrow V], \kappa\rangle \approx \langle e, E'[x \leftarrow V'], \kappa'\rangle$.

Thus $\langle e, E[x \leftarrow V], \kappa\rangle \approx^* \langle e, E'[x \leftarrow V'], \kappa'\rangle$.

**Case** $\langle U_1 \langle A \rightarrow B \Rightarrow^l C \rightarrow D\rangle, E\rangle \approx \langle \lambda(x : \mathrm{seq}(\lceil C \Rightarrow^l A \rceil, c)) : \mathrm{seq}(d, \lceil B \Rightarrow^l D \rceil).\, e, E'\rangle$

Let $U_1' = \langle \lambda x^{c,d}.e, E'\rangle$ , we also have $U_1 \approx U_1^{'}$.

In this case,

$$
\begin{array}{rl}
 & \mathrm{doApp}(U, V, \kappa) \\
= & \mathrm{doApp}(\langle U_1\langle A \to B \Rightarrow^l C \to D\rangle, E\rangle, V, \kappa) \\
\longrightarrow & [\![C \Rightarrow^l A]\!](V) \gg\!= \lambda V_1.\langle V_1, [(U_1\,\square)][B \Rightarrow^l D]\kappa\rangle
\end{array}
$$

$$
\begin{array}{rrl}
 & & \mathrm{doApp}'(U', V', k') \\
 & = & \mathrm{doApp}'(\langle \lambda(x : \mathrm{seq}(\lceil C \Rightarrow^l A\rceil, c)) : \mathrm{seq}(d, \lceil B \Rightarrow^l D\rceil).\, e, E'\rangle, V', \kappa') \\
 & = & [\![\mathrm{seq}(\lceil C \Rightarrow^l A\rceil, c)]\!](V') \gg\!= \lambda V_1'.\langle e, E'[x \leftarrow V_1'], \mathrm{extCont}(\mathrm{seq}(d, \lceil B \Rightarrow^l D\rceil), \kappa')\rangle \\
(\star) & = & \underline{\mathrm{Return}\, V' \gg\!= [\![c]\!] \gg\!= [\![d]\!]} \gg\!= \lambda V_1'.\langle e, E'[x \leftarrow V_1'], \mathrm{extCont}(\mathrm{seq}(d, \lceil B \Rightarrow^l D\rceil), \kappa')\rangle \\
 & = & [\![c]\!](V') \gg\!= [\![d]\!] \gg\!= \lambda V_1'.\langle e, E'[x \leftarrow V_1'], \mathrm{extCont}(\mathrm{seq}(d, \lceil B \Rightarrow^l D\rceil), \kappa')\rangle
\end{array}
$$

The step marked with $(\star)$ follows from the basic property of $\mathrm{seq}(\cdot,\cdot)$, that is

$$
[\![\mathrm{seq}(c, d)]\!](V) = \mathrm{Return}\, V \gg\!= [\![c]\!] \gg\!= [\![d]\!]
$$

By Lemma 3, we can case-split $[\![C \Rightarrow^l A]\!](V)$ and $[\![\lceil C \Rightarrow^l A\rceil]\!](V')$ at the same time.

**Case** $[\![C \Rightarrow^l A]\!](V) = \mathrm{Raise}\, l$ and $[\![\lceil C \Rightarrow^l A\rceil]\!](V') = \mathrm{Raise}\, l$

In this case

$$
\begin{array}{rl}
 & \mathrm{doApp}(U, V, \kappa) \\
\longrightarrow & \mathrm{Raise}\, l \gg\!= \lambda V_1.\langle V_1, [(U_1\,\square)][B \Rightarrow^l D]\kappa\rangle \\
\longrightarrow & \mathrm{Raise}\, l
\end{array}
$$

$$
\begin{array}{rl}
 & \mathrm{doApp}'(U', V', \kappa') \\
= & \mathrm{Raise}\, l \gg\!= [\![c]\!] \gg\!= \lambda V_1'.\langle e, E'[x \leftarrow V_1'], \mathrm{extCont}(\mathrm{seq}(d, \lceil B \Rightarrow^l D\rceil), \kappa')\rangle \\
= & \mathrm{Raise}\, l
\end{array}
$$

By definition, $\mathrm{Err}\, l \approx \mathrm{Err}\, l$.

Thus $\mathrm{doApp}(U, V, \kappa) \approx^* \mathrm{doApp}'(U', V', \kappa')$.

**Case** $[\![C \Rightarrow^l A]\!](V) = \mathrm{Return}\, V_1$ and $[\![\lceil C \Rightarrow^l A\rceil]\!](V') = \mathrm{Return}\, V_1'$ and $V_1 \approx V_1'$

In this case

$$
\begin{array}{rl}
 & \mathrm{doApp}(U, V, \kappa) \\
\longrightarrow & \mathrm{Return}\, V_1 \gg\!= \lambda V_1.\langle V_1, [(U_1\,\square)][B \Rightarrow^l D]\kappa\rangle \\
\longrightarrow & \langle V_1, [(U_1\,\square)][B \Rightarrow^l D]\kappa\rangle \\
\longrightarrow & \mathrm{doApp}(U_1, V_1, [\square\langle B \Rightarrow^l D\rangle]\kappa)
\end{array}
$$

$$
\begin{array}{rrl}
 & & \mathrm{doApp}'(U', V', \kappa') \\
 & = & \mathrm{Return}\, V_1' \gg\!= [\![c]\!] \gg\!= \lambda V_2'.\langle e, E'[x \leftarrow V_2'], \mathrm{extCont}(\mathrm{seq}(d, \lceil B \Rightarrow^l D\rceil), \kappa')\rangle \\
 & = & [\![c]\!](V_1') \gg\!= \lambda V_2'.\langle e, E'[x \leftarrow V_2'], \mathrm{extCont}(\mathrm{seq}(d, \lceil B \Rightarrow^l D\rceil), \kappa')\rangle \\
(\star) & = & [\![c]\!](V_1') \gg\!= \lambda V_2'.\langle e, E'[x \leftarrow V_2'], \underline{\mathrm{extCont}(d, \mathrm{extCont}(\lceil B \Rightarrow^l D\rceil, \kappa'))}\rangle \\
 & = & \mathrm{doApp}'(\langle \lambda(x : c) : d.\, e, E'\rangle, V_1', \mathrm{extCont}(\lceil B \Rightarrow^l D\rceil, \kappa')) \\
 & = & \mathrm{doApp}'(U_1', V_1', \mathrm{extCont}(\lceil B \Rightarrow^l D\rceil, \kappa'))
\end{array}
$$

The step marked with $(\star)$ follows from Lemma 5.

Now we can use the induction hypothesis two show that

$$\mathrm{doApp}(U_1, V_1, [\Box\langle B \Rightarrow^l D\rangle]\kappa) \approx^* \mathrm{doApp}'(U_1', V_1', \mathrm{extCont}(\lceil B \Rightarrow^l D\rceil, \kappa'))$$

because $U_1 \approx U_1'$ and $V_1 \approx V_1'$ and, by Lemma 6, $[\Box\langle B \Rightarrow^l D\rangle]\kappa \approx \mathrm{extCont}(\lceil B \Rightarrow^l D\rceil, \kappa')$. Thus $\mathrm{doApp}(U, V, \kappa) \approx^* \mathrm{doApp}'(U', V', \kappa')$.

$\Box$