Syntax (new2)

$$
\begin{array}{llr}
e = x \mid \lambda x.e \mid e\, e & & \text{(Expressions)} \\
v = \lambda x.e & & \text{(Values)} \\
a = A[v] & & \text{(Answers)} \\
A = [\,] \mid A[\lambda x.A]\, e & & \text{(Answer Contexts)} \\
\hat{A} = [\,] \mid A[\hat{A}]\, e & & \text{(Partial Answer Contexts – outer)} \\
\check{A} = [\,] \mid A[\lambda x.\check{A}] & & \text{(Partial Answer Contexts – inner)} \\
E = [\,] \mid E\, e \mid A[E] \mid \hat{A}[A[\lambda x.\check{A}[E[x]]]\, E] & & \text{(Evaluation Contexts)} \\
\qquad\qquad \hat{A}[\check{A}] \in A & &
\end{array}
$$

Notions of Reduction (new2)

$$
\hat{A}[A_1[\lambda x.\check{A}[E[x]]]\, A_2[v]] \;\; \boldsymbol{\beta}_{\mathbf{need}} \;\; \hat{A}[A_1[A_2[\check{A}[E[x]]\{x := v\}]]]
$$
$$
\hat{A}[\check{A}] \in A
$$

$\rightarrow$: compatible closure of $\boldsymbol{\beta}_{\mathbf{need}}$

$\twoheadrightarrow$: reflexive, transitive closure of $\rightarrow$

$\Rightarrow$: parallel reduction of $\boldsymbol{\beta}_{\mathbf{need}}$ redexes

Definition 1 ($\Rightarrow$)**.** *Parallel reduction*

$$
\begin{array}{rcll}
e & \Rightarrow & e & (1) \\
\hat{A}[A_1[\lambda x.\check{A}[E[x]]]\, A_2[v]] & \Rightarrow & \hat{A}'[A_1'[A_2'[\check{A}'[E'[x]]\{x := v'\}]]] & (2) \\
 & \textit{if} & \hat{A}[\check{A}] \in A,\ \hat{A}'[\check{A}'] \in A, & \\
 & & \hat{A} \Rightarrow \hat{A}',\ A_1 \Rightarrow A_1',\ A_2 \Rightarrow A_2',\ \check{A} \Rightarrow \check{A}',\ E \Rightarrow E',\ v \Rightarrow v' & \\
e_1\, e_2 & \Rightarrow & e_1'\, e_2' & (3) \\
 & \textit{if} & e_1 \Rightarrow e_1',\ e_2 \Rightarrow e_2' & \\
\lambda x.e & \Rightarrow & \lambda x.e' & (4) \\
 & \textit{if} & e \Rightarrow e' &
\end{array}
$$

Definition 2 ($\Rightarrow$ for contexts)**.** *Parallel reduction of contexts.*

If all subterms in a context E parallel reduce, then $E \Rightarrow E'$, where each e in E is replaced with e' in E', and $e \Rightarrow e'$.

$$
\begin{array}{cccl}
[\,] & \Rightarrow & [\,] & \\
A_1[\lambda x.A_2]\, e & \Rightarrow & A_1'[\lambda x.A_2']\, e' & \textit{if } A_1 \Rightarrow A_1',\ A_2 \Rightarrow A_2',\ e \Rightarrow e' \\
A[\hat{A}]\, e & \Rightarrow & A'[\hat{A}']\, e' & \textit{if } A \Rightarrow A',\ \hat{A} \Rightarrow \hat{A}',\ e \Rightarrow e' \\
A[\lambda x.\check{A}] & \Rightarrow & A'[\lambda x.\check{A}'] & \textit{if } A \Rightarrow A',\ \check{A} \Rightarrow \check{A}' \\
E\, e & \Rightarrow & E'\, e' & \textit{if } E \Rightarrow E',\ e \Rightarrow e' \\
A[E] & \Rightarrow & A'[E'] & \textit{if } A \Rightarrow A',\ E \Rightarrow E' \\
\hat{A}[A[\lambda x.\check{A}[E_1[x]]]\, E_2] & \Rightarrow & \hat{A}'[A'[\lambda x.\check{A}'[E_1'[x]]]\, E_2'] & \textit{if } \hat{A}[\check{A}],\ \hat{A} \Rightarrow \hat{A}',\ A \Rightarrow A',\ \check{A} \Rightarrow \check{A}',\ E_1 \Rightarrow E_1',\ E_2 \Rightarrow E_2'
\end{array}
$$

0.1 Context Composition Lemmas

Lemma 1. *If $C_1 \Rightarrow C_1'$ and $C_2 \Rightarrow C_2'$, then $C_1[C_2] \Rightarrow C_1'[C_2']$.*

Proof. By structural induction on C_1. □

Lemma 2. *If $C \Rightarrow C'$ and $e \Rightarrow e'$, $C[e] \Rightarrow C'[e']$*

Proof. By structural induction on C. □

0.2 Diamond Lemma

Lemma 3 (Diamond Property of $\Rightarrow$). *If $e \Rightarrow e_1$ and $e \Rightarrow e_2$, there exists e' s.t. $e_1 \Rightarrow e'$ and $e_2 \Rightarrow e'$.*

Proof. By structural induction on proof of $e \Rightarrow e_1$.

Case $e \Rightarrow e_1$ by $\Rightarrow def(1)$, so $e = e_1$. (base)

Then $e' = e_2$ because $e_1 = e \Rightarrow e_2$ and $e_2 \Rightarrow e_2$.

Case $e \Rightarrow e_1$ by $\Rightarrow def(2)$.

Subcase $e \Rightarrow e_2$ by $\Rightarrow def(2)$.

$e = \hat{A}[A_1[\lambda x.\check{A}[E[x]]]\, A_2[v]]$,
$e_1 = \hat{A}'[A_1'[A_2'[\check{A}'[E'[x]]\{x := v'\}]]]$, $\quad \hat{A}{\Rightarrow}\hat{A}', A_1{\Rightarrow}A_1', A_2{\Rightarrow}A_2', \check{A}{\Rightarrow}\check{A}', E{\Rightarrow}E', v{\Rightarrow}v'$
$e_2 = \hat{A}''[A_1''[A_2''[\check{A}''[E''[x]]\{x := v''\}]]]$, $\quad \hat{A}{\Rightarrow}\hat{A}'', A_1{\Rightarrow}A_1'', A_2{\Rightarrow}A_2'', \check{A}{\Rightarrow}\check{A}'', E{\Rightarrow}E'', v{\Rightarrow}v''$
By IH, subterms satisfy diamond property. By lemma 4, subcontexts satisfy diamond property.
So $\exists e' = \hat{A}'''[A_1'''[A_2'''[\check{A}'''[E'''[x]]\{x := v'''\}]]]$ s.t.

- $e_1 \Rightarrow e'$ by composition lemmas and subst lemma
- $e_2 \Rightarrow e'$ by composition lemmas and subst lemma

Subcase $e \Rightarrow e_2$ by $\Rightarrow def(3)$.

$e = \hat{A}[A_1[\lambda x.\check{A}[E[x]]]\, A_2[v]]$,
$e_1 = \hat{A}'[A_1'[A_2'[\check{A}'[E'[x]]\{x := v'\}]]]$, $\quad \hat{A}{\Rightarrow}\hat{A}', A_1{\Rightarrow}A_1', A_2{\Rightarrow}A_2', \check{A}{\Rightarrow}\check{A}', E{\Rightarrow}E', v{\Rightarrow}v'$
$e_2 = \hat{A}''[A_1''[\lambda x.\check{A}''[E''[x]]]\, A_2''[v'']]$, $\quad \hat{A}{\Rightarrow}\hat{A}'', A_1{\Rightarrow}A_1'', A_2{\Rightarrow}A_2'', \check{A}{\Rightarrow}\check{A}'', E{\Rightarrow}E'', v{\Rightarrow}v''$, by lemma 5
By IH, subterms satisfy diamond property. By lemma 4, subcontexts satisfy diamond property.
So $\exists e' = \hat{A}'''[A_1'''[A_2'''[\check{A}'''[E'''[x]]\{x := v'''\}]]]$ s.t.

- $e_1 \Rightarrow e'$ by composition lemmas and subst lemma
- $e_2 \Rightarrow e'$ by $\Rightarrow def(2)$

Case $e \Rightarrow e_1$ by $\Rightarrow def(3)$.

$e \Rightarrow e_2$ subcases by $\Rightarrow def(2), (3)$, analogous to above case.

Case $e \Rightarrow e_1$ by $\Rightarrow def(4)$.

Claim holds by IH.

□

Lemma 4 (Diamond Property for Contexts). *If $C \Rightarrow C_1$ and $C \Rightarrow C_2$, then $\exists C'$ s.t. $C_1 \Rightarrow C'$ and $C_2 \Rightarrow C'$.*

Proof. By structural induction on C. □

Lemma 5. *If $e = \hat{A}[A_1[\lambda x.\check{A}[E[x]]]\, A_2[v]]$, $\hat{A}[\check{A}] \in A$, and $e \Rightarrow e'$ by $\Rightarrow def(3)$, then e' has shape $\hat{A}'[A_1'[\lambda x.\check{A}'[E'[x]]]\, A_2'[v']]$, and $\hat{A}{\Rightarrow}\hat{A}'$, $A_1{\Rightarrow}A_1'$, $A_2{\Rightarrow}A_2'$, $\check{A}{\Rightarrow}\check{A}'$, $E{\Rightarrow}E'$, $v{\Rightarrow}v'$*

Proof. By structural induction on $\hat{A}$.

Case $\hat{A} = [\,]$.

Since $\hat{A}[\check{A}] \in A$, $\check{A} = [\,]$, so $e = A_1[\lambda x.E[x]]\, A_2[v]$
Since $e \Rightarrow e'$ by $\Rightarrow def(3)$:

- $A_1[\lambda x.E[x]] \Rightarrow A_1'[\lambda x.E'[x]]$, with $A_1{\Rightarrow}A_1', E{\Rightarrow}E'$, by $A[v]$ and $E[x]$ shape lemmas (9 and 10).
- $A_2[v] \Rightarrow A_2'[v']$, with $A_2{\Rightarrow}A_2', v{\Rightarrow}v'$ by $A[v]$ lemma (9)

Thus, $e' = \hat{A}'[A_1'[\lambda x.\check{A}'[E'[x]]]\, A_2'[v']]$, where $\hat{A}' = \check{A}' = [\,]$

Case $\hat{A} = A[\hat{A}_1]e_1$**.**

$e = A[\hat{A}_1[A_1[\lambda x.\check{A}[E[x]]]\, A_2[v]]]\, e_1$
by $\Rightarrow def(3)$, $e \Rightarrow e' = A'[\hat{A}_1'[A_1'[\lambda x.\check{A}'[E'[x]]]\, A_2'[v']]]\, e_1'$, where:

- $A[\hat{A}_1[A_1[\lambda x.\check{A}[E[x]]]\, A_2[v]]] \Rightarrow A'[\hat{A}_1'[A_1'[\lambda x.\check{A}'[E'[x]]]\, A_2'[v']]]$
  bc by IH, $\hat{A}_1[A_1[\lambda x.\check{A}[E[x]]]\, A_2[v]] \Rightarrow \hat{A}_1'[A_1'[\lambda x.\check{A}'[E'[x]]]\, A_2'[v']]$,
  with $\hat{A}_1 \Rightarrow \hat{A}_1'$, $A_1 \Rightarrow A_1'$, $\check{A} \Rightarrow \check{A}'$, $E \Rightarrow E'$, $A_2 \Rightarrow A_2'$, $v \Rightarrow v'$
  Since, we lost an argument, $A[\hat{A}_1[A_1[\lambda x.\check{A}[E[x]]]\, A_2[v]]]$ is an answer, so it's also true that $A \Rightarrow A'$, by $A[v]$ lemma (9)
- $e_1 \Rightarrow e_1'$

□

0.3 Substitution Lemmas

Lemma 6 (Substitution)**.** *If* $e_1 \Rightarrow e_1'$ *and* $e_2 \Rightarrow e_2'$*, then* $e_1\{x := e_2\} \Rightarrow e_1'\{x := e_2'\}$*.*

Proof. By structural induction on $e_1 \Rightarrow e_1'$.

Case $e_1 = e_1'$**.** (base)

Case $e_1 \Rightarrow e_1'$ by $\Rightarrow def(2)$**.**

$e_1 = \hat{A}[A_1[\lambda y.\check{A}[E[y]]]\, A_2[v]]$,
$e_1' = \hat{A}'[A_1'[A_2'[\check{A}'[E'[y]]\{y := v'\}]]]$
by IH,
$\hat{A}\{x := e_2\} \Rightarrow \hat{A}'\{x := e_2'\}$
$A_1\{x := e_2\} \Rightarrow A_1'\{x := e_2'\}$
$\check{A}[E[y]]\{x := e_2\} \Rightarrow \check{A}'[E'[y]]\{x := e_2'\}$
$A_2\{x := e_2\} \Rightarrow A_2'\{x := e_2'\}$
$v\{x := e_2\} \Rightarrow v'\{x := e_2'\}$

$$\begin{aligned}
& e_1\{x := e_2\} \\
= & \hat{A}\{x := e_2\}[A_1\{x := e_2\}[\lambda y.\check{A}[E[y]]\{x := e_2\}]\, A_2[v]\{x := e_2\}] && \text{(distribute subst)} \\
\Rightarrow & \hat{A}'\{x := e_2'\}[A_1'\{x := e_2'\}[A_2'\{x := e_2'\}[\check{A}'[E'[y]]\{x := e_2'\}\{y := v'\{x := e_2'\}\}]]] && (\Rightarrow def(2)\text{+IH}) \\
= & \hat{A}'\{x := e_2'\}[A_1'\{x := e_2'\}[A_2'\{x := e_2'\}[\check{A}'[E'[y]]\{y := v'\}\{x := e_2'\}]]] && \text{(property of subst)} \\
= & \hat{A}'[A_1'[A_2'[\check{A}'[E'[y]]\{y := v'\}]]]\{x := e_2'\} && \text{(undist subst)} \\
= & e_1'\{x := e_2'\}
\end{aligned}$$

Case $e_1 \Rightarrow e_1'$ by $\Rightarrow def(3)$**.**

by IH and distributing and undistributing subst

Case $e_1 \Rightarrow e_1'$ by $\Rightarrow def(4)$**.**

by IH

□

Corollary 1 (Substitution)**.** *If* $\hat{A} \Rightarrow \hat{A}'$*,* $e_1 \Rightarrow e_1'$*, and* $e_2 \Rightarrow e_2'$*, then* $\hat{A}[e_1\{x := e_2\}] \Rightarrow \hat{A}'[e_1'\{x := e_2'\}]$*, etc.*

Proof. By lemmas 2 and 6. □

0.4 Not $\boldsymbol{\beta}_{\mathbf{need}}$ Redex Lemmas

Note: $\boldsymbol{\beta}_{\mathbf{need}}$ redex $= \hat{A}[A_1[\lambda x.\check{A}[E[x]]]\, A_2[v]]$, $\hat{A}[\check{A}] \in A$

Lemma 7. *$A[v]$ is not a $\boldsymbol{\beta}_{\mathbf{need}}$ redex.*

Lemma 8. *$E[x]$, where E does not bind x, is not a $\boldsymbol{\beta}_{\mathbf{need}}$ redex.*

0.5 Shape Preserving Lemmas

Lemma 9. *If $e = A[v]$ and $e \Rightarrow e'$, then $e' = A'[v']$ where $A \Rightarrow A'$ and $v \Rightarrow v'$.*

Proof. By structural induction on A. Proceed by cases on proof of $e \Rightarrow e'$.

Case $e \Rightarrow e'$ by $\Rightarrow def(1)$**.**

$A' = A$, $v = v'$

Case $e \Rightarrow e'$ by $\Rightarrow def(2)$**.**

Impossible, because e is not a $\boldsymbol{\beta}_{\mathbf{need}}$ redex, by lemma 7.

Case $e \Rightarrow e'$ by $\Rightarrow def(3)$**.**

If $A = [\,]$, trivial.
If $A = A_1[\lambda x.A_2]\, e_1$, lemma holds by IH.

Case $e \Rightarrow e'$ by $\Rightarrow def(4)$**.**

$A' = A = [\,]$.

□

Lemma 10. *If $e = E[x]$, where E does not bind x, and $e \Rightarrow e'$, then $e' = E'[x]$, E' does not bind x, and $E \Rightarrow E'$.*

Proof. By induction on E. Proceed by cases on proof of $e \Rightarrow e'$.

Case $e \Rightarrow e'$ by $\Rightarrow def(1)$**.**

$E' = E$

Case $e \Rightarrow e'$ by $\Rightarrow def(2)$**.**

Impossible because e is not a $\boldsymbol{\beta}_{\mathbf{need}}$ redex, by lemma 8

Case $e \Rightarrow e'$ by $\Rightarrow def(3)$**.**

Subcase $E = [\,]$**.**

$E' = [\,]$

Subcase $E = E_1\, e_1$**.**

Claim holds by IH.

Subcase $E = A[E_1]$**.**

If $A = [\,]$, trivial.
If $A = A_1[\lambda x.A_2]\, e_1$, then lemma holds for $A_2[E_1[x]]$ by IH, and $A_1 \Rightarrow A_1'$ by lemma 9, so $E' = A_1'[\lambda x.A_2'[E_1']]\, e_1'$.

Subcase $E = \hat{A}[A[\lambda y.\check{A}[E_1[y]]]\, E_2]$, $\hat{A}[\check{A}] \in A$**.**

Claim holds for $E_2[x]$ by IH. $\check{A} \Rightarrow \check{A}'$ by lemma 11. $A \Rightarrow A'$ by lemma 9. $\hat{A} \Rightarrow \hat{A}'$ by lemma 12 so $E' = \hat{A}'[A'[\lambda y.\check{A}'[E_1'[y]]]\, E_2']$

Case $e \Rightarrow e'$ by $\Rightarrow def(4)$**.**

Impossible because no E has shape $\lambda x. \ldots$.

□

Lemma 11. *If $e = \check{A}[E[x]]$, where $\check{A}[E]$ does not bind x, and $e \Rightarrow e'$, then $e' = \check{A}'[E'[x]]$, $\check{A}'[E']$ does not bind x, and $\check{A} \Rightarrow \check{A}'$ and $E \Rightarrow E'$.*

Proof. By induction on $\check{A}$. $E[x] \Rightarrow E'[x]$ by lemma 10.

Case $\check{A} = [\,]$**.**

$\check{A}' = [\,]$

Case $\check{A} = A[\lambda y.\check{A}_1]$**.**

Claim holds by IH and lemma 9.

□

Lemma 12. *If $e = \hat{A}[e_1]$, $e \Rightarrow e'$ by $\Rightarrow def(3)$, and $e_1 \Rightarrow e_1'$, then $e' = \hat{A}'[e_1']$, where $\hat{A} \Rightarrow \hat{A}'$.*

Proof. By induction on $\hat{A}$.

Case $\hat{A} = [\,]$**.**

$\hat{A}' = [\,]$

Case $\hat{A} = A[\hat{A}_1]\, e_2$**.**

Subcase $A = [\,]$**.**

Claim holds by IH.

Subcase $A = A_1[\lambda x.A_2]\, e_3$**.**

Claim holds by IH and lemmas 9 and 13.

□

Lemma 13. *If $e = A[e_1]$, $e \Rightarrow e'$ by $\Rightarrow def(3)$, and $e_1 \Rightarrow e_1'$, then $e' = A'[e_1']$ where $A \Rightarrow A'$.*

Proof. By induction on A.

Case $A = [\,]$**.**

$A' = [\,]$

Case $A = A_1[\lambda x.A_2]\, e_2$**.**

$A_2 \Rightarrow A_2'$ by IH and $A_1 \Rightarrow A_1'$ by lemma 9, so claim holds with $A' = A_1'[\lambda x.A_2']\, e_2'$

□