1 Concrete Semantics of Scheme

Syntax Domains:

$$
\begin{aligned}
e \in \mathsf{Exp} ::= {} & æ \\
& \mid (\texttt{if}\ e\ e\ e) \\
& \mid (\texttt{let}\ ([x\ e]\ ...)\ e) \\
& \mid (\texttt{call/cc}\ e) \\
& \mid (\texttt{set!}\ x\ e) \\
& \mid (\texttt{prim}\ op\ e\ ...) \\
& \mid (\texttt{apply-prim}\ op\ e) \\
& \mid (\texttt{apply}\ e\ e) \\
& \mid (e\ e\ ...) \\
æ \in \mathsf{AExp} ::= {} & x \mid lam \mid \mathbb{Z} \mid \texttt{\#t} \mid \texttt{\#f} \\
& \mid (\texttt{quote}\ e) \\
lam \in \mathsf{Lam} ::= {} & (\lambda\ (x...)\ e) \mid (\lambda\ x\ e) \\
x \in \mathsf{Var} \quad & \text{A set of identifiers} \\
op \in \mathsf{Prim} \quad & \text{A set of primitives}
\end{aligned}
$$

Atomic Evaluation:

$$
\begin{aligned}
& \mathcal{A} : \Sigma_E \rightharpoonup Val \\
\mathcal{A}(\langle lam, \rho, ...\rangle) & \triangleq (lam, \rho) \\
\mathcal{A}(\langle (\texttt{quote}\ e), ...\rangle) & \triangleq \mathbf{quote}(e) \\
\mathcal{A}(\langle x, \rho, \sigma, ...\rangle) & \triangleq \sigma(\rho(x)) \\
\mathcal{A}(\langle æ, ...\rangle) & \triangleq æ
\end{aligned}
$$

Injection:

$$\mathcal{I} : \mathsf{Exp} \to \Sigma_E$$

$$\mathcal{I}(e) \triangleq E\langle e, \varnothing, \varnothing, \{(0,0) : \mathbf{mt}\}, (0,0)\rangle$$

Store Joining:

$$\sigma \sqcup [a \mapsto v] \triangleq \sigma[a \mapsto v]$$

Collecting Semantics:

$$eval(e) = \{\varsigma \mid \mathcal{I}(e) \mapsto^* \varsigma\}$$

Semantic Domains:

$$
\begin{aligned}
\varsigma \in \Sigma & \triangleq E\langle Eval\rangle + A\langle Apply\rangle \\
Eval & \triangleq \mathsf{Exp} \times Env \times Store \times Addr \\
Apply & \triangleq Val \times Store \times Addr \\
\rho \in Env & \triangleq \mathsf{Var} \rightharpoonup Addr \\
\sigma \in Store & \triangleq BAddr \rightharpoonup Val \\
& \quad \times KAddr \rightharpoonup Kont \\
v \in Val & \triangleq Clo + Kont + \mathbb{Z} \\
& \quad + \{\texttt{\#t}, \texttt{\#f}, Null, Void\} \\
& \quad + \{\mathbf{quote}(e), \mathbf{cons}(v, v)\} \\
clo \in Clo & \triangleq \mathsf{Lam} \times Env \\
a \in BAddr & \triangleq \mathbb{N} \times \mathbb{N} \\
a_\kappa \in KAddr & \triangleq \mathbb{N} \\
\kappa \in Kont & ::= \mathbf{mt} \\
& \mid \mathbf{ifk}(e, e, \rho, a_\kappa) \\
& \mid \mathbf{letk}(vars, done, todo, e, \rho, a_\kappa) \\
& \mid \mathbf{callcck}(a_\kappa) \\
& \mid \mathbf{setk}(x, \rho, a_\kappa) \\
& \mid \mathbf{primk}(op, done, todo, \rho, a_\kappa) \\
& \mid \mathbf{appprimk}(op, a_\kappa) \\
& \mid \mathbf{appk}(done, todo, \rho, a_\kappa) \\
& \mid \mathbf{appappk}(val?, e, \rho, a_\kappa) \\
done & \triangleq Val^* \\
todo & \triangleq \mathsf{Exp}^* \\
vars & \triangleq \mathsf{Var}^*
\end{aligned}
$$

Address Allocation:

$$
\begin{aligned}
& balloc : \Sigma \times \mathbb{N} \to BAddr \\
& kalloc : \Sigma \to KAddr \\
balloc(\langle \sigma, ...\rangle, n) & \triangleq (|\sigma|, n) \\
kalloc(\langle \sigma, ...\rangle) & \triangleq (|\sigma|)
\end{aligned}
$$

Eval Rules

Rules for when the control is an expression

$\varsigma = E\langle e_\varsigma, \rho, \sigma, a_\kappa \rangle$ Proceed by matching on e_ς

$$æ \rightsquigarrow A\langle v, \sigma, a_\kappa \rangle$$

$$(\texttt{if}\ e_c\ e_t\ e_f) \rightsquigarrow E\langle e_c, \rho, \sigma', a'_\kappa \rangle$$
where $a'_\kappa \triangleq \mathit{kalloc}(\varsigma)$
$\kappa \triangleq \mathbf{ifk}(e_t, e_f, \rho, a_\kappa)$
$\sigma' \triangleq \sigma \sqcup [a'_\kappa \mapsto \kappa]$

$$(\texttt{let}\ ()\ e_b) \rightsquigarrow E\langle e_b, \rho, \sigma, a_\kappa \rangle$$

$$(\texttt{let}\ \mathit{binds}\ e_b) \rightsquigarrow E\langle e_0, \rho, \sigma', a'_\kappa \rangle$$
where $\mathit{binds} = ([x_0\ e_0]\ [x_s\ e_s]\ ...)$
$a'_\kappa \triangleq \mathit{kalloc}(\varsigma)$
$\kappa \triangleq \mathbf{letk}(x_0 :: x_s, [\,], e_s, e_b, \rho, a_\kappa)$
$\sigma' \triangleq \sigma \sqcup [a'_\kappa \mapsto \kappa]$

$$(\texttt{call/cc}\ e) \rightsquigarrow E\langle e, \rho, \sigma', a'_\kappa \rangle$$
where $a'_\kappa \triangleq \mathit{kalloc}(\varsigma)$
$\kappa \triangleq \mathbf{callcck}(a_\kappa)$
$\sigma' \triangleq \sigma \sqcup [a_\kappa \mapsto \kappa]$

$$(\texttt{set!}\ x\ e) \rightsquigarrow E\langle e, \rho, \sigma', a'_\kappa \rangle$$
where $a'_\kappa \triangleq \mathit{kalloc}(\varsigma)$
$a \triangleq \rho(x)$
$\kappa \triangleq \mathbf{setk}(a, a_\kappa)$
$\sigma' \triangleq \sigma \sqcup [a'_\kappa \mapsto \kappa]$

$$(\texttt{prim}\ \mathit{op}) \rightsquigarrow A\langle v, \sigma, a_\kappa \rangle$$
where $v = \mathit{op}$ applied to 0 arguments

$$(\texttt{prim}\ \mathit{op}\ e_0\ e_s\ ...) \rightsquigarrow E\langle e_0, \rho, \sigma', a'_\kappa \rangle$$
where $a'_\kappa \triangleq \mathit{kalloc}(\varsigma)$
$\kappa \triangleq \mathbf{primk}(\mathit{op}, [\,], e_s, \rho, a_\kappa)$
$\sigma' \triangleq \sigma \sqcup [a'_\kappa \mapsto \kappa]$

$$(\texttt{apply-prim}\ \mathit{op}\ e) \rightsquigarrow E\langle e, \rho, \sigma', a'_\kappa \rangle$$
where $a'_\kappa \triangleq \mathit{kalloc}(\varsigma)$
$\kappa \triangleq \mathbf{appprimk}(\mathit{op}, a_\kappa)$
$\sigma' \triangleq \sigma \sqcup [a'_\kappa \mapsto \kappa]$

$$(\texttt{apply}\ e_f\ e_x) \rightsquigarrow E\langle e_f, \rho, \sigma', a'_\kappa \rangle$$
where $a'_\kappa \triangleq \mathit{kalloc}(\varsigma)$
$\kappa \triangleq \mathbf{appappk}(\varnothing, e_x, \rho, a_\kappa)$
$\sigma' \triangleq \sigma \sqcup [a'_\kappa \mapsto \kappa]$

$$(e_f\ e_s\ ...) \rightsquigarrow E\langle e_f, \rho, \sigma', a'_\kappa \rangle$$
where $a'_\kappa \triangleq \mathit{kalloc}(\varsigma)$
$\kappa \triangleq \mathbf{appk}([\,], e_s, \rho, a_\kappa)$
$\sigma' \triangleq \sigma \sqcup [a'_\kappa \mapsto \kappa]$

Apply Rules

Rules for when the control is a value

$\varsigma = A\langle v, \sigma, a_\kappa \rangle$

$\kappa \triangleq \sigma(a_\kappa)$

Proceed by matching on κ

$\mathbf{mt} \rightsquigarrow \varsigma$

$\mathbf{ifk}(\_, e_f, \rho_\kappa, a'_\kappa) \rightsquigarrow E\langle e_f, \rho_\kappa, \sigma, a'_\kappa \rangle$
where $v = \texttt{\#f}$

$\mathbf{ifk}(e_t, \_, \rho_\kappa, a'_\kappa) \rightsquigarrow E\langle e_t, \rho_\kappa, \sigma, a'_\kappa \rangle$
where $v \neq \texttt{\#f}$

$\mathbf{letk}(vars, done, [\,], e_b, \rho_\kappa, a'_\kappa) \rightsquigarrow E\langle e_b, \rho'_\kappa, \sigma', a'_\kappa \rangle$
where

$$a_i \triangleq balloc(\varsigma, i)$$
$$done' \triangleq done +\!\!+ [v]$$
$$\rho'_\kappa \triangleq \rho_\kappa[vars_0 \mapsto a_0 \ldots vars_{n-1} \mapsto a_{n-1}]$$
$$\sigma' \triangleq \sigma \sqcup [a_0 \mapsto done'_0 \ldots a_{n-1} \mapsto done'_{n-1}]$$

$\mathbf{letk}(vars, done, e_h :: e_t, e_b, \rho_\kappa, a'_\kappa) \rightsquigarrow E\langle e_h, \rho_\kappa, \sigma', a''_\kappa \rangle$
where

$$a''_\kappa \triangleq kalloc(\varsigma)$$
$$\kappa' \triangleq \mathbf{letk}(vars, done +\!\!+ [v], e_t, e_b, \rho_\kappa, a'_\kappa)$$
$$\sigma' \triangleq \sigma \sqcup [a''_\kappa \mapsto \kappa']$$

$\mathbf{setk}(a, a'_\kappa) \rightsquigarrow A\langle Void, \sigma', a'_\kappa \rangle$
where $\sigma' \triangleq \sigma \sqcup [a \mapsto v]$

$\mathbf{callcck}(a'_\kappa) \rightsquigarrow E\langle e, \rho'_\lambda, \sigma', a'_\kappa \rangle$
where $v = ((\lambda\ (x)\ e), \rho_\lambda)$

$$a \triangleq balloc(\varsigma, 0)$$
$$\kappa' \triangleq \sigma(a'_\kappa)$$
$$\rho'_\lambda \triangleq \rho_\lambda[x \mapsto a]$$
$$\sigma' \triangleq \sigma \sqcup [a \mapsto \kappa']$$

$\mathbf{callcck}(\_) \rightsquigarrow A\langle \kappa, \sigma', a'_\kappa \rangle$
where $v = \kappa'$

$$a'_\kappa \triangleq kalloc(\varsigma)$$
$$\sigma' \triangleq \sigma \sqcup [a'_\kappa \mapsto \kappa']$$

$\mathbf{appprimk}(op, a'_\kappa) \rightsquigarrow A\langle v', \sigma, a'_\kappa \rangle$
where $v' \triangleq op$ applied to v

$\mathbf{primk}(op, done, [\,], \_, a'_\kappa) \rightsquigarrow A\langle v', \sigma, a'_\kappa \rangle$
where $v' \triangleq op$ applied to $(done +\!\!+ [v])$

$\mathbf{primk}(op, done, e_h :: e_t, \rho_\kappa, a'_\kappa) \rightsquigarrow E\langle e_h, \rho_\kappa, \sigma', a''_\kappa \rangle$
where

$$a''_\kappa \triangleq kalloc(\varsigma)$$
$$\kappa' \triangleq \mathbf{primk}(op, done +\!\!+ [v], e_t, \rho_\kappa, a'_\kappa)$$
$$\sigma' \triangleq \sigma \sqcup [a''_\kappa \mapsto \kappa']$$

More Apply Rules

Rules for when the control is a value

$$\textbf{appappk}(v_f, \_, \_, a'_\kappa) \rightsquigarrow E\langle e_b, \rho'_\lambda, \sigma', a'_\kappa\rangle$$
$$\text{where } v_f = ((\lambda\ (x_s...)\ e_b), \rho_\lambda)$$
$$\begin{aligned} a_i &\triangleq balloc(\varsigma, i) \\ \rho'_\lambda &\triangleq \rho_\lambda[x_0 \mapsto a_0\ ... \\ &\qquad\quad x_{n-1} \mapsto a_{n-1}] \\ \sigma' &\triangleq \sigma \sqcup [a_0 \mapsto v_0\ ... \\ &\qquad\quad a_{n-1} \mapsto v_{n-1}] \end{aligned}$$

$$\textbf{appappk}(v_f, \_, \_, a'_\kappa) \rightsquigarrow E\langle e_b, \rho'_\lambda, \sigma', a'_\kappa\rangle$$
$$\text{where } v_f = ((\lambda\ x\ e_b), \rho_\lambda)$$
$$\begin{aligned} a &\triangleq balloc(\varsigma, 0) \\ \rho'_\lambda &\triangleq \rho_\lambda[x \mapsto a] \\ \sigma' &\triangleq \sigma \sqcup [a \mapsto v] \end{aligned}$$

$$\textbf{appappk}(\varnothing, e, \rho_\kappa, a'_\kappa) \rightsquigarrow E\langle e, \rho_\kappa, \sigma', a''_\kappa\rangle$$
$$\text{where } a''_\kappa \triangleq kalloc(\varsigma)$$
$$\begin{aligned} \kappa' &\triangleq \textbf{appappk}(v, e, \rho_\kappa, a'_\kappa) \\ \sigma' &\triangleq \sigma \sqcup [a''_\kappa \mapsto \kappa'] \end{aligned}$$

$$\textbf{appk}(done, [\,], \_, a'_\kappa) \rightsquigarrow E\langle e_b, \rho'_\lambda, \sigma', a'_\kappa\rangle$$
$$\text{where } done = ((\lambda\ (x_s...)\ e_b), \rho_\lambda) :: v_s$$
$$\begin{aligned} a_i &\triangleq balloc(\varsigma, i) \\ v'_s &\triangleq v_s +\!\!+ [v] \\ \rho'_\lambda &\triangleq \rho_\lambda[x_0 \mapsto a_0\ ... \\ &\qquad\quad x_{n-1} \mapsto a_{n-1}] \\ \sigma' &\triangleq \sigma \sqcup [a_0 \mapsto v'_0\ ... \\ &\qquad\quad a_{n-1} \mapsto v'_{n-1}] \end{aligned}$$

$$\textbf{appk}(done, [\,], \_, a'_\kappa) \rightsquigarrow E\langle e_b, \rho'_\lambda, \sigma', a'_\kappa\rangle$$
$$\text{where } done = ((\lambda\ x\ e_b), \rho_\lambda) :: v_s$$
$$\begin{aligned} a &\triangleq balloc(\varsigma, 0) \\ v'_s &\triangleq v_s +\!\!+ [v] \\ \rho'_\lambda &\triangleq \rho_\lambda[x \mapsto a] \\ \sigma' &\triangleq \sigma \sqcup [a \mapsto v'_s] \end{aligned}$$

$$\textbf{appk}([\kappa_\lambda], [\,], \rho_\kappa, a'_\kappa) \rightsquigarrow A\langle v, \sigma', a''_\kappa\rangle$$
$$\text{where } a''_\kappa \triangleq kalloc(\varsigma)$$
$$\sigma' \triangleq \sigma \sqcup [a''_\kappa \mapsto \kappa_\lambda]$$

$$\textbf{appk}(done, e_h :: e_t, \rho_\kappa, a'_\kappa)$$
$$\rightsquigarrow E\langle e_h, \rho_\kappa, \sigma', a''_\kappa\rangle$$
$$\text{where } a''_\kappa \triangleq kalloc(\varsigma)$$
$$\begin{aligned} \kappa' &\triangleq \textbf{appk}(done +\!\!+ [v], e_t, \rho_\kappa, a'_\kappa) \\ \sigma' &\triangleq \sigma \sqcup [a''_\kappa \mapsto \kappa'] \end{aligned}$$

2 Abstract Semantics of Scheme

Abstract Semantic Domains:

$$
\begin{aligned}
\hat{\varsigma} \in \hat{\Sigma} &\triangleq E\langle \mathit{Eval} \rangle + E\langle \mathit{Apply} \rangle \\
\mathit{Eval} &\triangleq \mathsf{Exp} \times \widehat{\mathit{Env}} \times \widehat{\mathit{Store}} \times \widehat{\mathit{KAddr}} \\
\mathit{Apply} &\triangleq \widehat{\mathit{Val}} \times \widehat{\mathit{Store}} \times \widehat{\mathit{KAddr}} \\
\hat{\rho} \in \widehat{\mathit{Env}} &\triangleq \mathsf{Var} \rightharpoonup \widehat{\mathit{Addr}} \\
\hat{\sigma} \in \widehat{\mathit{Store}} &\triangleq \widehat{\mathit{BAddr}} \rightharpoonup \widehat{\mathit{Val}} \\
&\quad \times \widehat{\mathit{KAddr}} \rightharpoonup \mathcal{P}(\widehat{\mathit{Kont}}) \\
\hat{v} \in \widehat{\mathit{Val}} &\triangleq (\widehat{\mathit{CVal}} + \top + \bot) \times \mathcal{P}(\widehat{\mathit{Clo}}) \\
\hat{cv} \in \widehat{\mathit{CVal}} &\triangleq \widehat{\mathit{Kont}} + \mathbb{Z} \\
&\quad + \{\texttt{\#t}, \texttt{\#f}, \mathit{Null}, \mathit{Void}\} \\
&\quad + \{\mathbf{quote}(e), \mathbf{cons}(\hat{v}, \hat{v})\} \\
\widehat{\mathit{clo}} \in \widehat{\mathit{Clo}} &\triangleq \mathsf{Lam} \times \widehat{\mathit{Env}} \\
\hat{a} \in \widehat{\mathit{BAddr}} &\triangleq \mathsf{Var} \times \mathsf{Expr} \\
\hat{a}_{\hat{\kappa}} \in \widehat{\mathit{KAddr}} &\triangleq \mathsf{Expr} \times \widehat{\mathit{Env}} \\
\widehat{\mathit{done}} &\triangleq \widehat{\mathit{Val}}^{*}
\end{aligned}
$$

$$
\begin{aligned}
\hat{\kappa} \in \widehat{\mathit{Kont}} ::=\ & \mathbf{mt} \\
&\mid \mathbf{ifk}(e, e, \hat{\rho}, \hat{a}_{\hat{\kappa}}) \\
&\mid \mathbf{letk}(e, vars, \widehat{\mathit{done}}, \mathit{todo}, e, \hat{\rho}, \hat{a}_{\hat{\kappa}}) \\
&\mid \mathbf{callcck}(e, \hat{a}_{\hat{\kappa}}) \\
&\mid \mathbf{setk}(\hat{a}, \hat{a}_{\hat{\kappa}}) \\
&\mid \mathbf{primk}(op, \widehat{\mathit{done}}, \mathit{todo}, \hat{\rho}, \hat{a}_{\hat{\kappa}}) \\
&\mid \mathbf{appprimk}(op, \hat{a}_{\hat{\kappa}}) \\
&\mid \mathbf{appk}(e, \widehat{\mathit{done}}, \mathit{todo}, \hat{\rho}, \hat{a}_{\hat{\kappa}}) \\
&\mid \mathbf{appappk}(e, \hat{v}?, e, \hat{\rho}, \hat{a}_{\hat{\kappa}})
\end{aligned}
$$

Abstract Atomic Evaluation:

$$\mathcal{A} : \mathit{Eval} \to \widehat{\mathit{Val}}$$
$$\mathcal{A}(E\langle lam, \hat{\rho}, ... \rangle) \triangleq (\bot, \{(lam, \hat{\rho})\})$$
$$\mathcal{A}(æ) \triangleq (æ, \emptyset)$$

Allocation:

$$\widehat{\mathit{balloc}} : \mathsf{Var} \times \mathsf{Expr} \to \widehat{\mathit{BAddr}}$$
$$\widehat{\mathit{balloc}}(x, e) \triangleq (x, e)$$
$$\widehat{\mathit{kalloc}} : \mathsf{Expr} \times \widehat{\mathit{Env}} \to \widehat{\mathit{KAddr}}$$
$$\widehat{\mathit{kalloc}}(e, \hat{\rho}) \triangleq (e, \hat{\rho})$$

Store Joining:

$$\sigma \sqcup [\hat{a}_{\hat{\kappa}} \mapsto \hat{\kappa}] \triangleq \sigma[\hat{a}_{\hat{\kappa}} \mapsto \sigma(\hat{a}_{\hat{\kappa}}) \cup \{\hat{\kappa}\}]$$
$$\sigma \sqcup [a \mapsto (\hat{cv}_i, \widehat{clo}_i)] \triangleq$$
match on $\sigma[a]$

$$
\sigma[a \mapsto \begin{cases}
\texttt{empty} & (\hat{cv}_i, \widehat{clo}_i) \\
(\bot, \widehat{clo}_e) & (\hat{cv}_i, \widehat{clo}_i \cup \widehat{clo}_e) \\
(\hat{cv}_e, \widehat{clo}_e) & \\
\quad \wedge\ \hat{cv}_i = \bot & (\hat{cv}_e, \widehat{clo}_i \cup \widehat{clo}_e)) \\
(\hat{cv}_e, \widehat{clo}_e) & \\
\quad \wedge\ \hat{cv}_i = \hat{cv}_e & (\hat{cv}_i, \widehat{clo}_i \cup \widehat{clo}_e) \\
(\_, \widehat{clo}_e) & (\top, \widehat{clo}_i \cup \widehat{clo}_e)
\end{cases}]
$$

Abstract Eval Rules

Rules for when the control is an expression

$$\hat\varsigma = E\langle e_{\hat\varsigma}, \hat\rho, \hat\sigma, \hat a_{\hat\kappa}\rangle$$

Proceed by matching on $e_{\hat\varsigma}$

$$\text{æ} \rightsquigarrow A\langle \hat v, \hat\sigma, \hat a_{\hat\kappa}\rangle$$
$$\text{where } \hat v \triangleq \hat{\mathcal{A}}(\hat\varsigma)$$

$$(\texttt{if}\ e_c\ e_t\ e_f) \rightsquigarrow E\langle e_c, \hat\rho, \hat\sigma', \hat a_{\hat\kappa}\rangle$$
$$\begin{aligned} \text{where } \hat a'_{\hat\kappa} &\triangleq \widehat{kalloc}(e_{\hat\varsigma}, \hat\rho) \\ \hat\kappa &\triangleq \mathbf{ifk}(e_t, e_f, \hat\rho, \hat a_{\hat\kappa}) \\ \hat\sigma' &\triangleq \hat\sigma \sqcup [\hat a'_{\hat\kappa} \mapsto \hat\kappa] \end{aligned}$$

$$(\texttt{let}\ ()\ e_b) \rightsquigarrow E\langle e_b, \hat\rho, \hat\sigma, \hat a_{\hat\kappa}\rangle$$

$$(\texttt{let}\ bnds\ e_b) \rightsquigarrow E\langle e_0, \hat\rho, \hat\sigma', \hat a'_{\hat\kappa}\rangle$$
$$\begin{aligned} \text{where } bnds &= ([x_0\ e_0]\ [x_s\ e_s]\ ...) \\ \hat a'_{\hat\kappa} &\triangleq \widehat{kalloc}(e_{\hat\varsigma}, \hat\rho) \\ \hat\kappa &\triangleq \mathbf{letk}(e_{\hat\varsigma}, x_0 :: x_s, [\,], \\ &\qquad e_s, e_b, \hat\rho, \hat a_{\hat\kappa}) \\ \hat\sigma' &\triangleq \hat\sigma \sqcup [\hat a'_{\hat\kappa} \mapsto \hat\kappa] \end{aligned}$$

$$(\texttt{call/cc}\ e) \rightsquigarrow E\langle e, \hat\rho, \hat\sigma', \hat a'_{\hat\kappa}\rangle$$
$$\begin{aligned} \text{where } \hat a'_{\hat\kappa} &\triangleq \widehat{kalloc}(e_{\hat\varsigma}, \hat\rho) \\ \hat\kappa &\triangleq \mathbf{callcck}(e_{\hat\varsigma}, \hat\rho, \hat a_{\hat\kappa}) \\ \hat\sigma' &\triangleq \hat\sigma \sqcup [\hat a'_{\hat\kappa} \mapsto \hat\kappa] \end{aligned}$$

$$(\texttt{set!}\ x\ e) \rightsquigarrow E\langle e, \hat\rho, \hat\sigma', \hat a'_{\hat\kappa}\rangle$$
$$\begin{aligned} \text{where } \hat a'_{\hat\kappa} &\triangleq \widehat{kalloc}(e_{\hat\varsigma}, \hat\rho) \\ \hat a &\triangleq \hat\rho(x) \\ \hat\kappa &\triangleq \mathbf{setk}(\hat a, \hat a_{\hat\kappa}) \\ \hat\sigma' &\triangleq \hat\sigma \sqcup [\hat a'_{\hat\kappa} \mapsto \hat\kappa] \end{aligned}$$

$$(\texttt{prim}\ op) \rightsquigarrow A\langle \hat v, \hat\sigma, \hat a_{\hat\kappa}\rangle$$
$$\text{where } \hat v = op \text{ applied to 0 arguments}$$

$$(\texttt{prim}\ op\ e_0\ e_s\ ...) \rightsquigarrow E\langle e_0, \hat\rho, \hat\sigma', \hat a'_{\hat\kappa}\rangle$$
$$\begin{aligned} \text{where } \hat a'_{\hat\kappa} &\triangleq \widehat{kalloc}(e_{\hat\varsigma}, \hat\rho) \\ \hat\kappa &\triangleq \mathbf{primk}(op, [\,], e_s, \hat\rho, \hat a_{\hat\kappa}) \\ \hat\sigma' &\triangleq \hat\sigma \sqcup [\hat a'_{\hat\kappa} \mapsto \hat\kappa] \end{aligned}$$

$$(\texttt{apply-prim}\ op\ e) \rightsquigarrow E\langle e, \hat\rho, \hat\sigma', \hat a'_{\hat\kappa}\rangle$$
$$\begin{aligned} \text{where } \hat a'_{\hat\kappa} &\triangleq \widehat{kalloc}(e_{\hat\varsigma}, \hat\rho) \\ \hat\kappa &\triangleq \mathbf{appprimk}(op, \hat a_{\hat\kappa}) \\ \hat\sigma' &\triangleq \hat\sigma \sqcup [\hat a'_{\hat\kappa} \mapsto \hat\kappa] \end{aligned}$$

$$(\texttt{apply}\ e_f\ e_x) \rightsquigarrow E\langle e_f, \hat\rho, \hat\sigma', \hat a'_{\hat\kappa}\rangle$$
$$\begin{aligned} \text{where } \hat a'_{\hat\kappa} &\triangleq \widehat{kalloc}(e_{\hat\varsigma}, \hat\rho) \\ \hat\kappa &\triangleq \mathbf{appappk}(e_{\hat\varsigma}, \varnothing, e_x, \hat\rho, \hat a_{\hat\kappa}) \\ \hat\sigma' &\triangleq \hat\sigma \sqcup [\hat a'_{\hat\kappa} \mapsto \hat\kappa] \end{aligned}$$

$$(e_f\ e_s\ ...) \rightsquigarrow E\langle e_f, \hat\rho, \hat\sigma', \hat a'_{\hat\kappa}\rangle$$
$$\begin{aligned} \text{where } \hat a'_{\hat\kappa} &\triangleq \widehat{kalloc}(e_{\hat\varsigma}, \hat\rho) \\ \hat\kappa &\triangleq \mathbf{appk}(e_{\hat\varsigma}, [\,], e_s, \hat\rho, \hat a_{\hat\kappa}) \\ \hat\sigma' &\triangleq \hat\sigma \sqcup [\hat a'_{\hat\kappa} \mapsto \hat\kappa] \end{aligned}$$

Apply Rules

Rules for when the control is a value

$\hat{\varsigma} = A\langle \hat{v}, \hat{\sigma}, \hat{a}_{\kappa\varsigma}\rangle$

$\kappa_\varsigma \in \hat{\sigma}(\hat{a}_{\kappa\varsigma})$

Proceed by matching on κ_ς

$$\mathbf{mt} \rightsquigarrow \emptyset$$

$$\mathbf{ifk}(\_, e_f, \hat{\rho}_{\hat{\kappa}}, \hat{a}_{\hat{\kappa}}) \rightsquigarrow \{E\langle e_f, \hat{\rho}_{\hat{\kappa}}, \hat{\sigma}, \hat{a}_{\hat{\kappa}}\rangle\}$$
where $\hat{v} = (\texttt{\#f}, \emptyset)$

$$\mathbf{ifk}(e_t, \_, \hat{\rho}_{\hat{\kappa}}, \hat{a}_{\hat{\kappa}}) \rightsquigarrow \{E\langle e_t, \hat{\rho}_{\hat{\kappa}}, \hat{\sigma}, \hat{a}_{\hat{\kappa}}\rangle\}$$
where $\hat{v} = (\mathit{not}\ \texttt{\#f}\ \mathit{nor}\ \top, \_)$

$$\mathbf{ifk}(e_t, e_f, \hat{\rho}_{\hat{\kappa}}, \hat{a}_{\hat{\kappa}}) \rightsquigarrow \{E\langle e_t, \hat{\rho}_{\hat{\kappa}}, \hat{\sigma}, \hat{a}_{\hat{\kappa}}\rangle, E\langle e_f, \hat{\rho}_{\hat{\kappa}}, \hat{\sigma}, \hat{a}_{\hat{\kappa}}\rangle\}$$
where $\hat{v} \in \{(\texttt{\#f}, \mathit{not}\ \emptyset), (\top, \_)\}$

$$\mathbf{letk}(e_{\hat{\varsigma}}, vars, \widehat{done}, [\,], e_b, \hat{\rho}_{\hat{\kappa}}, \hat{a}_{\hat{\kappa}}) \rightsquigarrow \{E\langle e_b, \hat{\rho}'_{\hat{\kappa}}, \hat{\sigma}', \hat{a}_{\hat{\kappa}}\rangle\}$$
where
$$\begin{aligned}
\hat{a}_i &\triangleq \widehat{balloc}(vars_i, e_{\hat{\varsigma}})\\
\hat{\rho}'_{\hat{\kappa}} &\triangleq \hat{\rho}_{\hat{\kappa}}[vars_0 \mapsto \hat{a}_0 \ldots\\
&\qquad vars_{n-1} \mapsto \hat{a}_{n-1}]\\
\widehat{done}' &\triangleq \widehat{done} \mathbin{+\!\!+} [\hat{v}]\\
\hat{\sigma}' &\triangleq \hat{\sigma} \sqcup [\hat{a}_0 \mapsto \widehat{done}'_0 \ldots\\
&\qquad \hat{a}_{n-1} \mapsto \widehat{done}'_{n-1}]
\end{aligned}$$

$$\mathbf{letk}(e_{\hat{\varsigma}}, vars, \widehat{done}, e_h :: e_t, e_b, \hat{\rho}_{\hat{\kappa}}, \hat{a}_{\hat{\kappa}}) \rightsquigarrow \{E\langle e_h, \hat{\rho}_{\hat{\kappa}}, \hat{\sigma}', \hat{a}'_{\hat{\kappa}}\rangle\}$$
where
$$\begin{aligned}
\hat{a}'_{\hat{\kappa}} &\triangleq \widehat{kalloc}(e_{\hat{\varsigma}}, \hat{\rho}_{\hat{\kappa}})\\
\hat{\kappa} &\triangleq \mathbf{letk}(e_{\hat{\varsigma}}, vars, \widehat{done} \mathbin{+\!\!+} [\hat{v}],\\
&\qquad e_t, e_b, \hat{\rho}_{\hat{\kappa}}, \hat{a}_{\hat{\kappa}})\\
\hat{\sigma}' &\triangleq \hat{\sigma} \sqcup [\hat{a}'_{\hat{\kappa}} \mapsto \hat{\kappa}]
\end{aligned}$$

$$\mathbf{setk}(\hat{a}, \hat{a}_{\hat{\kappa}}) \rightsquigarrow \{A\langle (\mathit{Void}, \emptyset), \hat{\sigma}', \hat{a}_{\hat{\kappa}}\rangle\}$$
where $\hat{\sigma}' \triangleq \hat{\sigma} \sqcup [\hat{a} \mapsto \hat{v}]$

$$\mathbf{callcck}(e_{\hat{\varsigma}}, \hat{a}_{\hat{\kappa}}) \rightsquigarrow \{E\langle e_b, \hat{\rho}'_\lambda, \hat{\sigma}', \hat{a}_{\hat{\kappa}}\rangle\}$$
where $\hat{v} \ni (\_, ((\lambda\ (x)\ e_b), \hat{\rho}_\lambda))$
$$\begin{aligned}
\hat{a} &\triangleq \widehat{balloc}(x, e_{\hat{\varsigma}})\\
\hat{\kappa} &\in \hat{\sigma}(\hat{a}_{\hat{\kappa}})\\
\hat{\rho}'_\lambda &\triangleq \hat{\rho}_\lambda[x \mapsto \hat{a}]\\
\hat{\sigma}' &\triangleq \hat{\sigma} \sqcup [\hat{a} \mapsto \hat{\kappa}]
\end{aligned}$$

$$\mathbf{callcck}(\_, \_) \rightsquigarrow \{A\langle \kappa_\varsigma, \hat{\sigma}', \hat{a}_{\kappa\varsigma}\rangle\}$$
where $\hat{v} = (\hat{\kappa}, \_)$
$$\hat{\sigma}' \triangleq \hat{\sigma} \sqcup [\hat{a}_{\kappa\varsigma} \mapsto \hat{\kappa}]$$

$$\mathbf{appprimk}(op, \hat{a}_{\hat{\kappa}}) \rightsquigarrow \{A\langle \hat{v}', \hat{\sigma}, \hat{a}_{\hat{\kappa}}\rangle\}$$
where $\hat{v}' \triangleq op$ applied to $\hat{v}$

$$\mathbf{primk}(op, \widehat{done}, [\,], \_, \hat{a}_{\hat{\kappa}}) \rightsquigarrow \{A\langle \hat{v}', \hat{\sigma}, \hat{a}_{\hat{\kappa}}\rangle\}$$
where $\hat{v}' \triangleq op$ applied to $(\widehat{done} \mathbin{+\!\!+} [\hat{v}])$

$$\mathbf{primk}(op, \widehat{done}, e_h :: e_t, \hat{\rho}_{\hat{\kappa}}, \hat{a}_{\hat{\kappa}}) \rightsquigarrow \{E\langle e_h, \hat{\rho}_{\hat{\kappa}}, \hat{\sigma}', \hat{a}'_{\hat{\kappa}}\rangle\}$$
where
$$\begin{aligned}
\hat{a}'_{\hat{\kappa}} &\triangleq \widehat{kalloc}(e_h, \hat{\rho}_{\hat{\kappa}})\\
\hat{\kappa} &\triangleq \mathbf{primk}(op, \widehat{done} \mathbin{+\!\!+} [\hat{v}]\\
&\qquad e_t, \hat{\rho}_{\hat{\kappa}}, \hat{a}_{\hat{\kappa}})\\
\hat{\sigma}' &\triangleq \hat{\sigma} \sqcup [\hat{a}'_{\hat{\kappa}} \mapsto \hat{\kappa}]
\end{aligned}$$

More Apply Rules

Rules for when the control is a value

$$\begin{aligned}
&\textbf{appappk}(e_{\hat\varsigma}, \hat v_f, _, _, \hat a_{\hat\kappa}) \\
&\leadsto \{E\langle e_b, \hat\rho'_\lambda, \hat\sigma', \hat a_{\hat\kappa}\rangle\} \\
\text{where } &\hat v_f \ni (_, ((\lambda\ (x_s ...)\ e_b), \hat\rho_\lambda)) \\
&\hat a_i \triangleq \widehat{balloc}(x_i, e_{\hat\varsigma}) \\
&\hat\rho'_\lambda \triangleq \hat\rho_\lambda[x_0 \mapsto \hat a_0\ ... \\
&\qquad x_{n-1} \mapsto \hat a_{n-1}] \\
&\hat\sigma' \triangleq \hat\sigma \sqcup [\hat a_0 \mapsto \hat v_0\ ... \\
&\qquad \hat a_{n-1} \mapsto \hat v_{n-1}]
\end{aligned}$$

$$\begin{aligned}
&\textbf{appappk}(e_{\hat\varsigma}, \hat v_f, _, _, \hat a_{\hat\kappa}) \\
&\leadsto \{E\langle e_b, \hat\rho'_\lambda, \hat\sigma', \hat a_{\hat\kappa}\rangle\} \\
\text{where } &\hat v_f \ni (_, ((\lambda\ x\ e_b), \hat\rho_\lambda)) \\
&\hat a \triangleq \widehat{balloc}(x, e_{\hat\varsigma}) \\
&\hat\rho'_\lambda \triangleq \hat\rho_\lambda[x \mapsto \hat a] \\
&\hat\sigma' \triangleq \hat\sigma \sqcup [\hat a \mapsto \hat v]
\end{aligned}$$

$$\begin{aligned}
&\textbf{appappk}(e_{\hat\varsigma}, \varnothing, e_x, \hat\rho_{\hat\kappa}, \hat a_{\hat\kappa}) \\
&\leadsto \{E\langle e_x, \hat\rho_{\hat\kappa}, \hat\sigma', \hat a'_{\hat\kappa}\rangle\} \\
\text{where } &\hat a'_{\hat\kappa} \triangleq \widehat{kalloc}(e_x, \hat\rho_{\hat\kappa}) \\
&\hat\kappa \triangleq \textbf{appappk}(e_{\hat\varsigma}, \hat v, e_x, \hat\rho_{\hat\kappa}, \hat a_{\hat\kappa}) \\
&\hat\sigma' \triangleq \hat\sigma \sqcup [\hat a'_{\hat\kappa} \mapsto \hat\kappa]
\end{aligned}$$

$$\begin{aligned}
&\textbf{appk}(e_{\hat\varsigma}, \hat v_h :: \hat v_t, [\,], _, \hat a_{\hat\kappa}) \\
&\leadsto \{E\langle e_b, \hat\rho'_\lambda, \hat\sigma', \hat a_{\hat\kappa}\rangle\} \\
\text{where } &\hat v_h \ni (_, ((\lambda\ (x_s ...)\ e_b), \hat\rho_\lambda)) \\
&\hat a_i \triangleq \widehat{balloc}(x_i, e_{\hat\varsigma}) \\
&\hat\rho'_\lambda \triangleq \hat\rho_\lambda[x_0 \mapsto \hat a_0 ... \\
&\qquad x_{n-1} \mapsto \hat a_{n-1}] \\
&\hat v'_t \triangleq \hat v_t +\!\!+ [\hat v] \\
&\hat\sigma' \triangleq \hat\sigma \sqcup [\hat a_0 \mapsto \hat v'_0 ... \\
&\qquad \hat a_{n-1} \mapsto \hat v'_{n-1}]
\end{aligned}$$

$$\begin{aligned}
&\textbf{appk}(e_{\hat\varsigma}, \hat v_h :: \hat v_t, [\,], _, \hat a_{\hat\kappa}) \\
&\leadsto \{E\langle e_b, \hat\rho'_\lambda, \hat\sigma', \hat a_{\hat\kappa}\rangle\} \\
\text{where } &\hat v_h \ni (_, ((\lambda\ x\ e_b), \hat\rho_\lambda)) \\
&\hat a \triangleq \widehat{balloc}(x, e_{\hat\varsigma}) \\
&\hat\rho'_\lambda \triangleq \hat\rho_\lambda[x \mapsto \hat a] \\
&\hat v'_t \triangleq \hat v_t +\!\!+ [\hat v] \\
&\hat\sigma' \triangleq \hat\sigma \sqcup [\hat a \mapsto \hat v'_t]
\end{aligned}$$

$$\begin{aligned}
&\textbf{appk}(e_{\hat\varsigma}, [\hat v_f], [\,].\hat\rho_{\hat\kappa}, _) \leadsto \{A\langle \hat v, \hat\sigma', \hat a_{\hat\kappa}\rangle\} \\
\text{where } &\hat v_f = (\hat\kappa_\lambda, _) \\
&\hat a_{\hat\kappa} \triangleq \widehat{kalloc}(e_{\hat\varsigma}, \hat\rho_{\hat\kappa}) \\
&\hat\sigma' \triangleq \hat\sigma \sqcup [\hat a_{\hat\kappa} \mapsto \hat\kappa_\lambda]
\end{aligned}$$

$$\begin{aligned}
&\textbf{appk}(e_{\hat\varsigma}, \widehat{done}, e_h :: e_t, \hat\rho_{\hat\kappa}, \hat a_{\hat\kappa}) \\
&\leadsto \{E\langle e_h, \hat\rho_{\hat\kappa}, \hat\sigma', \hat a'_{\hat\kappa}\rangle\} \\
\text{where } &\hat a'_{\hat\kappa} \triangleq \widehat{kalloc}(e_h, \hat\rho_{\hat\kappa}) \\
&\hat\kappa \triangleq \textbf{appk}(e_{\hat\varsigma}, \widehat{done} +\!\!+ [\hat v], e_t, \hat\rho_{\hat\kappa}, \hat a_{\hat\kappa}) \\
&\hat\sigma' \triangleq \hat\sigma \sqcup [\hat a'_{\hat\kappa} \mapsto \hat\kappa]
\end{aligned}$$