1 Concrete Semantics of Scheme

Syntax Domains:

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

$x \in \mathsf{Var}$ A set of identifiers

$op \in \mathsf{Prim}$ A set of primitives

Atomic Evaluation:

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

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:

$$\sigma \sqcup [a \mapsto v] \triangleq \sigma[a \mapsto v]$$
$$\sigma_\kappa \sqcup [a_\kappa \mapsto \kappa] \triangleq \sigma_\kappa[a_\kappa \mapsto \kappa]$$

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:

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

2 Abstract Semantics of Scheme

Abstract Semantic Domains:

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

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

Tick/Alloc:

$$\widehat{tick} : \hat{\Sigma} \times \widehat{Kont} \to \widehat{Time}$$
$$\widehat{tick}(\hat{\varsigma}, \hat{\kappa}) \triangleq 0$$
$$\widehat{alloc} : \hat{\Sigma} \times \widehat{Kont} \to \widehat{Addr}$$
$$\widehat{alloc}(\hat{\varsigma}, \hat{\kappa}) \triangleq 0$$

Abstract Atomic Evaluation:

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

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

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

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

Proceed by matching on $\hat{\kappa}$

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

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

$$\mathbf{letk}(e_{\hat{\varsigma}}, \mathit{vars}, \widehat{\mathit{done}}, [\,], e_b, \hat{\rho}_{\hat{\kappa}}, \hat{a}'_{\hat{\kappa}})$$
$$\rightsquigarrow \{E\langle e_b, \hat{\rho}_{\hat{\kappa}}, \hat{\sigma}', \hat{a}'_{\hat{\kappa}}\rangle\}$$
where $\hat{a}_i \triangleq \widehat{\mathit{balloc}}(\mathit{vars}_i, e_{\hat{\varsigma}})$

$$\hat{\rho}'_{\hat{\kappa}} \triangleq \hat{\rho}_{\hat{\kappa}}[\mathit{vars}_0 \mapsto \hat{a}_0\ ...\ \mathit{vars}_{n-1} \mapsto \hat{a}_{n-1}]$$

$$\widehat{\mathit{done}}' \triangleq \widehat{\mathit{done}} +\!\!+ [\hat{v}]$$

$$\hat{\sigma}' \triangleq \hat{\sigma} \sqcup [\hat{a}_0 \mapsto \widehat{\mathit{done}}'_0 ...\ \hat{a}_{n-1} \mapsto \widehat{\mathit{done}}'_{n-1}]$$

$$\mathbf{letk}(e_{\hat{\varsigma}}, \mathit{vars}, \widehat{\mathit{done}}, e_h :: e_t, e_b, \hat{\rho}_{\hat{\kappa}}, \hat{a}'_{\hat{\kappa}})$$
$$\rightsquigarrow \{E\langle e_h, \hat{\rho}_{\hat{\kappa}}, \hat{\sigma}', \hat{a}''_{\hat{\kappa}}\rangle\}$$
where $\hat{a}''_{\hat{\kappa}} \triangleq \widehat{\mathit{kalloc}}(()e_{\hat{\varsigma}}, \hat{\rho}_{\hat{\kappa}})$

$$\hat{\kappa}' \triangleq \mathbf{letk}(e_{\hat{\varsigma}}, \mathit{vars}, \widehat{\mathit{done}} +\!\!+ [\hat{v}], e_t, e_b, \hat{\rho}_{\hat{\kappa}}, \hat{a}'_{\hat{\kappa}})$$

$$\hat{\sigma}' \triangleq \hat{\sigma} \sqcup [\hat{a}''_{\hat{\kappa}} \mapsto \hat{\kappa}']$$

$$\mathbf{setk}(\hat{a}, \hat{a}'_{\hat{\kappa}}) \rightsquigarrow \{A\langle \mathit{Void}, \hat{\sigma}', \hat{a}'_{\hat{\kappa}}\rangle\}$$
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 $\hat{v} \ni (\_, ((\lambda\ (x)\ e_b), \hat{\rho}_{\lambda}))$

$$\hat{a} \triangleq \widehat{\mathit{balloc}}(x, e_{\hat{\varsigma}})$$

$$\hat{\kappa}' \triangleq \hat{\sigma}(\hat{a}'_{\hat{\kappa}})$$

$$\hat{\rho}'_{\lambda} \triangleq \hat{\rho}_{\lambda}[x \mapsto \hat{a}]$$

$$\hat{\sigma}' \triangleq \hat{\sigma} \sqcup [\hat{a} \mapsto \hat{\kappa}']$$

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

$$\hat{\sigma}' \triangleq \hat{\sigma} \sqcup [\hat{a}_{\hat{\kappa}} \mapsto \hat{\kappa}']$$

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

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

$$\mathbf{primk}(\mathit{op}, \widehat{\mathit{done}}, e_h :: e_t, \hat{\rho}_{\hat{\kappa}}, \hat{a}'_{\hat{\kappa}})$$
$$\rightsquigarrow \{E\langle e_h, \hat{\rho}\hat{\kappa}, \hat{\sigma}', \hat{a}''_{\hat{\kappa}}\rangle\}$$
where $\hat{a}''_{\hat{\kappa}} \triangleq \widehat{\mathit{kalloc}}(e_h, \hat{\rho}_{\hat{\kappa}})$

$$\hat{\kappa}' \triangleq \mathbf{primk}(\mathit{op}, \widehat{\mathit{done}} +\!\!+ [\hat{v}]\ e_t, \hat{\rho}_{\hat{\kappa}}, \hat{a}'_{\hat{\kappa}})$$

$$\hat{\sigma}' \triangleq \hat{\sigma} \sqcup [\hat{a}''_{\hat{\kappa}} \mapsto \hat{\kappa}']$$

More Apply Rules

Rules for when the control is a value

$$\begin{array}{l}\textbf{appappk}(e_{\hat\varsigma}, \hat v_f, \_, \_, \hat a'_{\hat\kappa}) \\ \leadsto \{E\langle e_b, \hat\rho'_\lambda, \hat\sigma', \hat a'_{\hat\kappa}\rangle\}\end{array}$$

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

$$\begin{array}{l}\textbf{appappk}(e_{\hat\varsigma}, \hat v_f, \_, \_, \hat a'_{\hat\kappa}) \\ \leadsto \{E\langle e_b, \hat\rho'_\lambda, \hat\sigma', \hat a'_{\hat\kappa}\rangle\}\end{array}$$

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

$$\begin{array}{l}\textbf{appappk}(e_{\hat\varsigma}, \varnothing, e_x, \hat\rho_{\hat\kappa}, \hat a'_{\hat\kappa}) \\ \leadsto \{E\langle e_x, \hat\rho_{\hat\kappa}, \hat\sigma', \hat a''_{\hat\kappa}\rangle\}\end{array}$$

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

$$\begin{array}{l}\textbf{appk}(e_{\hat\varsigma}, \hat v_h :: \hat v_t, [\,], \_, \hat a'_{\hat\kappa}) \\ \leadsto \{E\langle e_b, \hat\rho'_\lambda, \hat\sigma', \hat a'_{\hat\kappa}\rangle\}\end{array}$$

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

$$\begin{array}{l}\textbf{appk}(e_{\hat\varsigma}, \hat v_h :: \hat v_t, [\,], \_, \hat a'_{\hat\kappa}) \\ \leadsto \{E\langle e_b, \hat\rho'_\lambda, \hat\sigma', \hat a'_{\hat\kappa}\rangle\}\end{array}$$

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

$$\textbf{appk}(e_{\hat\varsigma}, [\hat v_f], [\,].\hat\rho_{\hat\kappa}, \_) \leadsto \{A\langle \hat v, \hat\sigma', \hat a'_{\hat\kappa}\rangle\}$$

$$\begin{aligned} \text{where } \hat v_f &= (\hat\kappa_\lambda, \_) \\ \hat a'_{\hat\kappa} &\triangleq \widehat{kalloc}(e_{\hat\varsigma}, \hat\rho_{\hat\kappa}) \\ \hat\sigma' &\triangleq \hat\sigma \sqcup [\hat a'_{\hat\kappa} \mapsto \hat\kappa_\lambda] \end{aligned}$$

$$\begin{array}{l}\textbf{appk}(e_{\hat\varsigma}, \widehat{done}, e_h :: e_t, \hat\rho_{\hat\kappa}, \hat a'_{\hat\kappa}) \\ \leadsto \{E\langle e_h, \hat\rho_{\hat\kappa}, \hat\sigma', \hat a''_{\hat\kappa}\rangle\}\end{array}$$

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