On the expressiveness of effect handlers and monadic reflection

Yannick Forster
supvervised by Ohad Kammar and Marcelo Fiore

UNIVERSITY OF CAMBRIDGE

A LITTLE SURVEY

- Who has ever tried to prove a functional program correct?
- Who has ever tried for a program involving reference cells or exceptions?
- Who has succeeded?
- Who thought it was fun?

UNIVERSITY OF CAMBRIDGE

HOW TO INCORPORATE EFFECTS?

Effects are …

- global store (i.e. references),
- exceptions,
- I/O,
- random,
- nondeterminism,
- or concurrency

UNIVERSITY OF CAMBRIDGE

An example

```
exception Error
val r = ref 0
fun error () = raise Error

fun test () = (r := 5; error() handle Error => !r)
```

`test()` evaluates to?

Why not to `0`?

WOULD BE COOL:

User definable effects on top of a functional language

There is more than one solution available!

GOAL

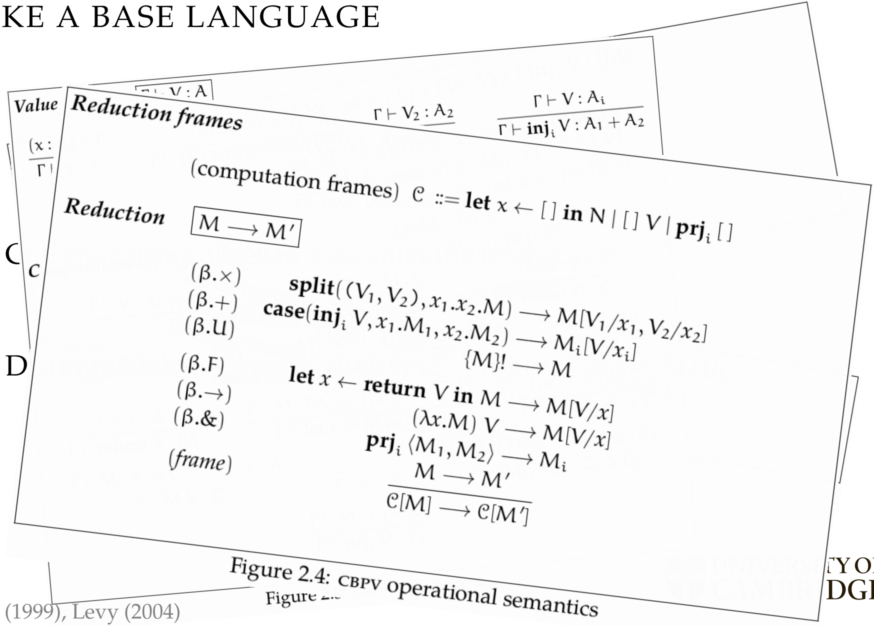
Compare two existing approaches in their expressiveness

A bit like “Compare expressiveness of recursion and for-loops”

APPROACH

- take a base language (functional, typed, no recursion)
- add each concept to the language
- define denotational semantics to each resulting calculus
- prove denotational semantics to be adequate
- use this to compare expressiveness

Take a base language



Reduction frames

(computation frames) $\mathcal{C} ::= \mathbf{let}\ x \leftarrow [\,]\ \mathbf{in}\ N \mid [\,]\ V \mid \mathbf{prj}_i\ [\,]$

Reduction $\boxed{M \longrightarrow M'}$

$$
\begin{array}{lr}
(\beta.\times) & \mathbf{split}((V_1, V_2), x_1.x_2.M) \longrightarrow M[V_1/x_1, V_2/x_2] \\
(\beta.+) & \mathbf{case}(\mathbf{inj}_i\ V, x_1.M_1, x_2.M_2) \longrightarrow M_i[V/x_i] \\
(\beta.U) & \{M\}! \longrightarrow M \\
(\beta.F) & \mathbf{let}\ x \leftarrow \mathbf{return}\ V\ \mathbf{in}\ M \longrightarrow M[V/x] \\
(\beta.\to) & (\lambda x.M)\ V \longrightarrow M[V/x] \\
(\beta.\&) & \mathbf{prj}_i\ \langle M_1, M_2 \rangle \longrightarrow M_i \\
(frame) & \dfrac{M \longrightarrow M'}{\mathcal{C}[M] \longrightarrow \mathcal{C}[M']}
\end{array}
$$

Figure 2.4: CBPV operational semantics

Levy (1999), Levy (2004)

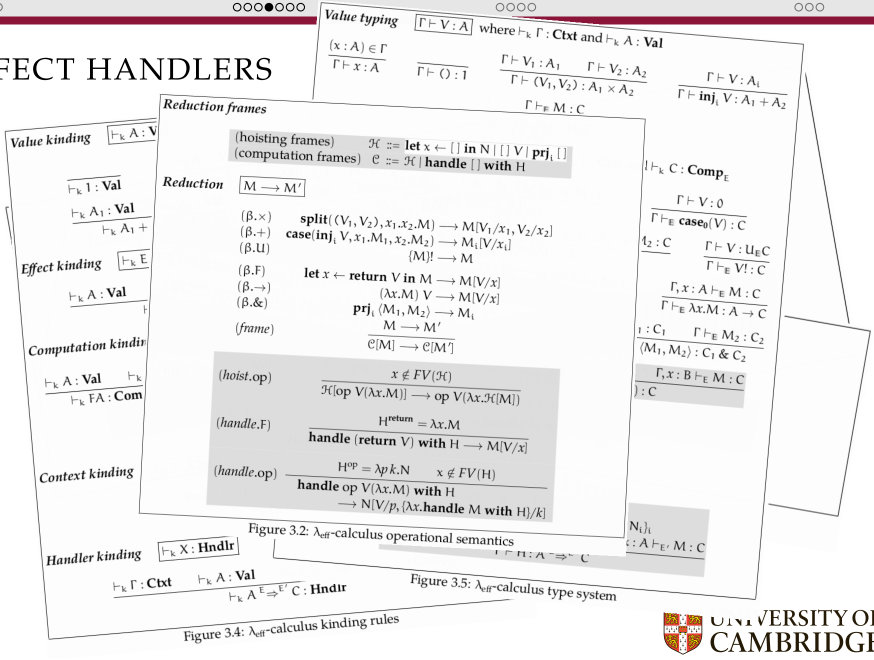
ADD EACH CONCEPT

- Effects and handler calculus λ_{eff}
- Monadic reflection calculus λ_{mon}

UNIVERSITY OF CAMBRIDGE

Kammar, Lindley, and Oury (2013), Filinski (2010)

EFFECT HANDLERS



Reduction frames

(hoisting frames) $\mathcal{H} ::= \textbf{let}\ x \leftarrow [\,]\ \textbf{in}\ N \mid [\,]\ V \mid \textbf{prj}_i\ [\,]$

(computation frames) $\mathcal{C} ::= \mathcal{H} \mid \textbf{handle}\ [\,]\ \textbf{with}\ H$

Reduction $\boxed{M \longrightarrow M'}$

$$
\begin{array}{ll}
(\beta.\times) & \textbf{split}(\langle V_1, V_2\rangle, x_1.x_2.M) \longrightarrow M[V_1/x_1, V_2/x_2] \\
(\beta.+) & \textbf{case}(\textbf{inj}_i\ V, x_1.M_1, x_2.M_2) \longrightarrow M_i[V/x_i] \\
(\beta.U) & \{M\}! \longrightarrow M \\
(\beta.F) & \textbf{let}\ x \leftarrow \textbf{return}\ V\ \textbf{in}\ M \longrightarrow M[V/x] \\
(\beta.\to) & (\lambda x.M)\ V \longrightarrow M[V/x] \\
(\beta.\&) & \textbf{prj}_i\ \langle M_1, M_2\rangle \longrightarrow M_i
\end{array}
$$

$$(\textit{frame})\quad \frac{M \longrightarrow M'}{\mathcal{C}[M] \longrightarrow \mathcal{C}[M']}$$

$$(\textit{hoist}.\mathrm{op})\quad \frac{x \notin FV(\mathcal{H})}{\mathcal{H}[\mathrm{op}\ V(\lambda x.M)] \longrightarrow \mathrm{op}\ V(\lambda x.\mathcal{H}[M])}$$

$$(\textit{handle}.F)\quad \frac{H^{\mathrm{return}} = \lambda x.M}{\textbf{handle}\ (\textbf{return}\ V)\ \textbf{with}\ H \longrightarrow M[V/x]}$$

$$(\textit{handle}.\mathrm{op})\quad \frac{H^{\mathrm{op}} = \lambda p\,k.N \qquad x \notin FV(H)}{\textbf{handle}\ \mathrm{op}\ V(\lambda x.M)\ \textbf{with}\ H \longrightarrow N[V/p, \{\lambda x.\textbf{handle}\ M\ \textbf{with}\ H\}/k]}$$

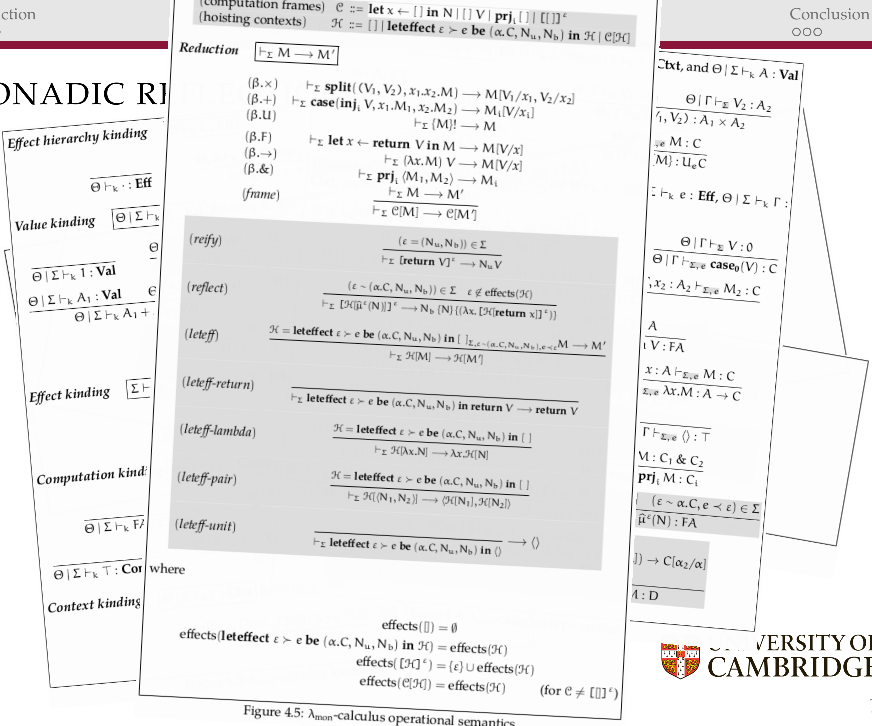
Figure 3.2: λ_{eff}-calculus operational semantics

Figure 3.5: λ_{eff}-calculus type system

Figure 3.4: λ_{eff}-calculus kinding rules

UNIVERSITY OF CAMBRIDGE

MONADIC RE



(computation frames) $\mathcal{C} ::= \mathbf{let}\ x \leftarrow [\,]\ \mathbf{in}\ N \mid [\,]\ V \mid \mathbf{prj}_i\,[\,] \mid [[\,]]^{\varepsilon}$

(hoisting contexts) $\mathcal{H} ::= [\,] \mid \mathbf{leteffect}\ \varepsilon \succ e\ \mathbf{be}\ (\alpha.C, N_u, N_b)\ \mathbf{in}\ \mathcal{H} \mid \mathcal{C}[\mathcal{H}]$

Reduction $\boxed{\vdash_\Sigma M \longrightarrow M'}$

$(\beta.\times)\quad \vdash_\Sigma \mathbf{split}((V_1,V_2), x_1.x_2.M) \longrightarrow M[V_1/x_1, V_2/x_2]$

$(\beta.+)\quad \vdash_\Sigma \mathbf{case}(\mathbf{inj}_i\,V, x_1.M_1, x_2.M_2) \longrightarrow M_i[V/x_i]$

$(\beta.U)\quad \vdash_\Sigma \{M\}! \longrightarrow M$

$(\beta.F)\quad \vdash_\Sigma \mathbf{let}\ x \leftarrow \mathbf{return}\ V\ \mathbf{in}\ M \longrightarrow M[V/x]$

$(\beta.\to)\quad \vdash_\Sigma (\lambda x.M)\ V \longrightarrow M[V/x]$

$(\beta.\&)\quad \vdash_\Sigma \mathbf{prj}_i\,\langle M_1, M_2\rangle \longrightarrow M_i$

$(\mathit{frame})\quad \dfrac{\vdash_\Sigma M \longrightarrow M'}{\vdash_\Sigma \mathcal{C}[M] \longrightarrow \mathcal{C}[M']}$

$(\mathit{reify})\quad \dfrac{(\varepsilon = (N_u, N_b)) \in \Sigma}{\vdash_\Sigma [\mathbf{return}\ V]^{\varepsilon} \longrightarrow N_u\,V}$

$(\mathit{reflect})\quad \dfrac{(\varepsilon \sim (\alpha.C, N_u, N_b)) \in \Sigma \quad \varepsilon \notin \mathsf{effects}(\mathcal{H})}{\vdash_\Sigma [\mathcal{H}[\widehat{\mu}^{\varepsilon}(N)]]^{\varepsilon} \longrightarrow N_b\,\{N\}\,\{(\lambda x.\,[\mathcal{H}[\mathbf{return}\ x]]^{\varepsilon})\}}$

$(\mathit{leteff})\quad \dfrac{\mathcal{H} = \mathbf{leteffect}\ \varepsilon \succ e\ \mathbf{be}\ (\alpha.C, N_u, N_b)\ \mathbf{in}\ [\,] \quad \vdash_{\Sigma, \varepsilon \sim (\alpha.C, N_u, N_b), e \prec \varepsilon} M \longrightarrow M'}{\vdash_\Sigma \mathcal{H}[M] \longrightarrow \mathcal{H}[M']}$

$(\mathit{leteff\text{-}return})\quad \dfrac{}{\vdash_\Sigma \mathbf{leteffect}\ \varepsilon \succ e\ \mathbf{be}\ (\alpha.C, N_u, N_b)\ \mathbf{in}\ \mathbf{return}\ V \longrightarrow \mathbf{return}\ V}$

$(\mathit{leteff\text{-}lambda})\quad \dfrac{\mathcal{H} = \mathbf{leteffect}\ \varepsilon \succ e\ \mathbf{be}\ (\alpha.C, N_u, N_b)\ \mathbf{in}\ [\,]}{\vdash_\Sigma \mathcal{H}[\lambda x.N] \longrightarrow \lambda x.\mathcal{H}[N]}$

$(\mathit{leteff\text{-}pair})\quad \dfrac{\mathcal{H} = \mathbf{leteffect}\ \varepsilon \succ e\ \mathbf{be}\ (\alpha.C, N_u, N_b)\ \mathbf{in}\ [\,]}{\vdash_\Sigma \mathcal{H}[\langle N_1, N_2\rangle] \longrightarrow \langle \mathcal{H}[N_1], \mathcal{H}[N_2]\rangle}$

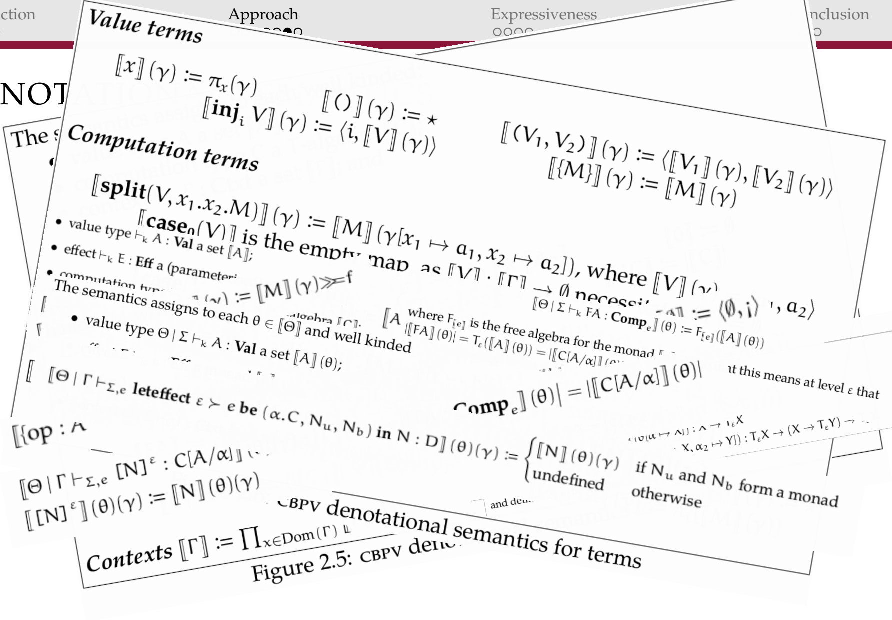
$(\mathit{leteff\text{-}unit})\quad \dfrac{}{\vdash_\Sigma \mathbf{leteffect}\ \varepsilon \succ e\ \mathbf{be}\ (\alpha.C, N_u, N_b)\ \mathbf{in}\ \langle\rangle \longrightarrow \langle\rangle}$

where

$$\begin{aligned}
\mathsf{effects}([\,]) &= \emptyset \\
\mathsf{effects}(\mathbf{leteffect}\ \varepsilon \succ e\ \mathbf{be}\ (\alpha.C, N_u, N_b)\ \mathbf{in}\ \mathcal{H}) &= \mathsf{effects}(\mathcal{H}) \\
\mathsf{effects}([\mathcal{H}]^{\varepsilon}) &= \{\varepsilon\} \cup \mathsf{effects}(\mathcal{H}) \\
\mathsf{effects}(\mathcal{C}[\mathcal{H}]) &= \mathsf{effects}(\mathcal{H}) \qquad (\text{for } \mathcal{C} \neq [[\,]]^{\varepsilon})
\end{aligned}$$

Figure 4.5: λ_{mon}-calculus operational semantics

DENOT


Value terms
$[\![x]\!](\gamma) := \pi_x(\gamma)$
$[\![()]\!](\gamma) := \star$
$[\![\mathbf{inj}_i\, V]\!](\gamma) := \langle i, [\![V]\!](\gamma)\rangle$
$[\![(V_1, V_2)]\!](\gamma) := \langle [\![V_1]\!](\gamma), [\![V_2]\!](\gamma)\rangle$
$[\![\{M\}]\!](\gamma) := [\![M]\!](\gamma)$
Computation terms
$[\![\mathbf{split}(V, x_1.x_2.M)]\!](\gamma) := [\![M]\!](\gamma[x_1 \mapsto a_1, x_2 \mapsto a_2])$, where $[\![V]\!](\gamma)$
$[\![\mathbf{case}_0(V)]\!]$ is the empty map
The semantics assigns to each $\theta \in [\![\Theta]\!]$ and well kinded
value type $\Theta \mid \Sigma \vdash_k A : \mathbf{Val}$ a set $[\![A]\!](\theta)$;
$[\![\Theta \mid \Sigma \vdash_k FA : \mathbf{Comp}_e]\!](\theta) := F_{[\![e]\!]}([\![A]\!](\theta))$
where $F_{[\![e]\!]}$ is the free algebra for the monad
$[\![\Theta \mid \Gamma \vdash_{\Sigma,e} \mathbf{leteffect}\ \varepsilon \succ e\ \mathbf{be}\ (\alpha.C, N_u, N_b)\ \mathbf{in}\ N : D]\!](\theta)(\gamma) := \begin{cases} [\![N]\!](\theta)(\gamma) & \text{if } N_u \text{ and } N_b \text{ form a monad} \\ \text{undefined} & \text{otherwise} \end{cases}$
$[\![\Theta \mid \Gamma \vdash_{\Sigma,e} [N]^\varepsilon : C[A/\alpha]]\!]$
$[\![[N]^\varepsilon]\!](\theta)(\gamma) := [\![N]\!](\theta)(\gamma)$
Contexts $[\![\Gamma]\!] := \prod_{x \in \mathrm{Dom}(\Gamma)}$
Figure 2.5: CBPV denotational semantics for terms

ADEQUACY AND SOUNDNESS

Theorem 2.18 (adequacy). *Denotational equivalence implies contextual equivalence. Explicitly: Given a monad satisfying the mono requirement, then for all* $\Gamma \vdash \mathsf{P}, \mathsf{Q} : \mathsf{X}$*, if* $[\![\mathsf{P}]\!] = [\![\mathsf{Q}]\!]$ *then* $\mathsf{P} \simeq \mathsf{Q}$.

Corollary 2.19 (soundness). *All well-typed closed ground returners reduce to a normal form. Explicitly: for all* $\vdash \mathsf{M} : \mathsf{FG}$ *there exists some* $\vdash \mathsf{V} : \mathsf{G}$ *such that:*

$$\mathsf{M} \longrightarrow^{\star} \textbf{return}\ \mathsf{V}$$

TYPED MACRO EXPRESSABILITY

One concept can express another if there is a *local* translation function $\underline{\ \ }$ that:

- is homomorphic on the base calculus
- replaces new syntactic constructs without rearranging the whole program
- translates terms $\emptyset \vdash M : X$ to terms $\emptyset \vdash \underline{M} : \underline{X}$

Felleisen (1990)

FOCUS IN THIS THESIS

Produce negative results: Prove that *no* translation exists with the help of denotational semantics

λ_{mon} CAN NOT TYPED MACRO EXPRESS λ_{eff}

- There are only finitely many terms for every type in λ_{mon}
- Some types in λ_{eff} have countably many observationally distinguishable terms
- Given a translation $\lambda_{\text{eff}} \to \lambda_{\text{mon}}$, take the type $\underline{F1}$
- $\underline{F1}$ has k terms
- $F1$ has more than k observationally distinguishable terms
- Derive a contradiction

UNIVERSITY OF CAMBRIDGE

THE BIG PICTURE

$$
\begin{tikzcd}
 & \text{CBPV} \arrow[dl, "\text{typed}"'] \arrow[dr, "\text{typed}"] & \\
\lambda_{\text{eff}} \arrow[rr, dashed, shift right] & & \lambda_{\text{mon}} \arrow[ll, "\text{untyped}"', shift right] \arrow[ll, dotted, shift left=3, "\text{typed}"]
\end{tikzcd}
$$

Contribution

- Adequacy proof for the set theoretic model for calculus of effect handlers λ_{eff}
- Adequate denotational semantics for calculus of monadic reflection λ_{mon}
- Definition of (typed) macro expressability
- Proof that λ_{mon} is macro expressible in λ_{eff}
- Proof that λ_{eff} is not macro typed expressible in λ_{mon}

UNIVERSITY OF CAMBRIDGE

FUTURE WORK

- Show that λ_{mon} is not typed macro expressible in λ_{eff};
- extend the type system of λ_{eff} to typed macro express λ_{mon};
- do similar comparison for calculus of delimited control.

Related work / Bibliography

- Paul Blain Levy. Call-By-Push-Value: A Functional/Imperative Synthesis, volume 2 of Semantics Structures in Computation. Springer, 2004.
- Ohad Kammar, Sam Lindley, and Nicolas Oury. Handlers in action. SIGPLAN Not. 48(9):145–158, September 2013.
- Andrzej Filinski. Monads in action. SIGPLAN Not., 45(1):483–494, January 2010.
- Matthias Felleisen. On the expressive power of programming languages. In Science of Computer Programming, pages 134–151. Springer-Verlag, 1990.

UNIVERSITY OF CAMBRIDGE

Value relations $\boxed{R_A \subseteq [\![A]\!] \times \lambda_{\mathbf{eff}\,A}}$

$$R_1 := \{\langle \star, V\rangle | V \simeq ()\}$$

$$R_{A_1 \times A_2} := \left\{\langle\langle a_1, a_2\rangle, V\rangle \middle| \exists V_1, V_2 \forall i. V_i : A_i. \langle a_i, V_i\rangle \in R_{A_i}, V \simeq (V_1, V_2)\right\} \qquad R_0 := \emptyset$$

$$R_{A_1 + A_2} = \bigcup_{i \in \{1,2\}} \left\{\langle \iota_i a, V\rangle \middle| \exists V' : A_i. \langle a, V'\rangle \in R_{A_i}, V \simeq \mathbf{inj}_i\, V'\right\}$$

$$R_{U_E C} := \{\langle a, V\rangle | \exists M : C. \langle a, M\rangle \in R_{E,C}, V \simeq \{M\}\}$$

Computation relations $\boxed{R_{E,C} \subseteq |[\![C]\!]| \times \lambda_{\mathbf{eff}\,E,C}}$

$$\begin{aligned}
R_{E,FA} &:= R'_{E,FA} \cup \{\langle \mathbf{return}\ a, M\rangle | \exists V : A. \langle a, V\rangle \in R_A, M \simeq \mathbf{return}\ V\} \\
R_{E,A\to C} &:= R'_{E,A\to C} \cup \{\langle f, M\rangle | \forall \langle a, V\rangle \in R_{E,A}, \langle f(a), M\ V\rangle \in R_{E,C}\} \\
R_{E,\top} &:= R'_{E,\top} \cup \{\langle \star, M\rangle | M \simeq \langle\rangle\} \\
R_{E,C_1 \& C_2} &:= R'_{E,C_1\&C_2} \cup \left\{\langle\langle c_1, c_2\rangle, M\rangle \middle| \exists M_1 : C_1, M_2 : C_2. M \simeq \langle M_1, M_2\rangle, \forall i. \langle c_i, M_i\rangle \in R_{E,C_i}\right\}
\end{aligned}$$

where

$$\begin{aligned}
R'_{E,C} := \{ \quad & \\
& \langle \mathrm{op}_a(\lambda b.c), N\rangle \mid \exists VM. \\
& \langle a, V\rangle \in R_{E,A} \quad (\mathrm{op} : A \to B) \in E \\
& \langle \lambda b.c, \lambda x.M\rangle \in R_{E,B\to C} \\
& N \simeq \mathrm{op}\ V\ (\lambda x.M) \\
\} \quad &
\end{aligned}$$

Handler relations $\boxed{R_{A^{E} \Rightarrow^{E'} C} \subseteq [\![A \ {}^{E}\!\Rightarrow^{E'}\ C]\!] \times \mathbf{handlers}(A \ {}^{E}\!\Rightarrow^{E'}\ C)}$

$$\begin{aligned}
R_{A^{E} \Rightarrow^{E'} C} := \{ \quad & \\
& \langle\langle D_\gamma, f_\gamma\rangle, H\rangle \mid \\
& (\mathrm{op}_i : A \to B) \in E, \quad D_\gamma = \langle |[\![C]\!]|, [\![-]\!]\,(\gamma)\rangle, \\
& \forall i. \exists M N_i. \langle f_\gamma, \lambda x.M\rangle \in R_{E',A\to C} \wedge, \\
& \forall \langle a, V\rangle \in R_{E,A}, \langle k, \lambda x.M'\rangle \in RB \to C. \\
& \left\langle [\![\mathrm{op}_i]\!]\,(\gamma)\,(\lambda x.kx \gg\!= f)\ a, N_i[V/p, (\lambda x.\mathbf{handle}\ M'\ \mathbf{with}\ H)/k]\right\rangle \\
\} \quad &
\end{aligned}$$

Figure 3.8: λ_{eff}-calculus logical relations

Value relations $\boxed{R_{\Theta,\Sigma,A}(\rho) \subseteq [\![A]\!](\theta) \times \lambda^{\Sigma}_{\mathbf{mon}}(A)}$

$$R_{\Theta,\Sigma,1}(\rho) := \{\langle \star, V\rangle | V \simeq ()\}$$

$$R_{\Theta,\Sigma,A_1 \times A_2}(\rho) := \left\{\langle\langle a_1, a_2\rangle, V\rangle \middle| \exists V_1, V_2 \forall i. V_i : A_i. \langle a_i, V_i\rangle \in R_{\Sigma,A_i}(\rho), V \simeq (V_1, V_2)\right\}$$

$$R_{\Theta,\Sigma,0}(\rho) := \emptyset$$

$$R_{\Theta,\Sigma,A_1+A_2}(\rho) = \bigcup_{i=1,2} \left\{\langle \iota_i a, V\rangle \middle| \exists V' : A_i. \langle a, V'\rangle \in R_{\Theta,\Sigma,A_i}(\rho), V \simeq \mathbf{inj}_i V'\right\}$$

$$R_{\Theta,\Sigma,U_e C}(\rho) := \{\langle a, V\rangle | \exists M : C. \langle a, M\rangle \in R_{\Theta,\Sigma,e,C}(\rho), V \simeq \{M\}\} \qquad R_{\Theta,\Sigma,\alpha}(\rho) := \rho(\alpha)$$

Computation relations $\boxed{R^{v}_{\Theta,\Sigma,e,C}(\rho) \subseteq |[\![C]\!](\theta)| \times \lambda^{\Sigma,e}_{\mathbf{mon}}(C)}$

$$R^{v}_{\Theta,\Sigma,\perp,FA}(\rho) := \{\langle a, \mathbf{return}\ V\rangle | \exists \langle a', V\rangle \in R_{\Sigma,A}(\rho), a = \mathbf{return}\ a'\}$$

$$R^{v}_{\Theta,\Sigma,\varepsilon,FA}(\rho) := \{\langle a, \mathbf{return}\ V\rangle | \exists \langle a', V\rangle \in R_{\Sigma,A}(\rho), a = \mathbf{return}\ a'\} \cup \left\{\langle a, \widehat{\mu}^{\varepsilon}(N)\rangle \middle| (\varepsilon \sim \alpha.C) \in \Sigma, \langle a, N\rangle \in R_{\Theta,\Sigma,e,C[A/\alpha]}(\rho)\right\}$$

$$R^{v}_{\Theta,\Sigma,e,A\to C}(\rho) := \{\langle f, \lambda x.M\rangle | \forall \langle a, V\rangle \in R_{\Theta,\Sigma,e,A}(\rho), \langle f(a), (\lambda x.M)\ V\rangle \in R_{\Theta,\Sigma,e,C}(\rho)\}$$

$$R^{v}_{\Theta,\Sigma,e,\top}(\rho) := \{\langle \star, \langle\rangle\rangle\} \qquad R^{v}_{\Theta,\Sigma,e,C_1\& C_2}(\rho) := \left\{\langle\langle c_1, c_2\rangle, (M_1, M_2)\rangle \middle| \langle c_i, M_i\rangle \in R_{\Theta,\Sigma,e,C_i}(\rho)\right\}$$

$\top$- ***and*** $\top\top$-***liftings*** $\boxed{(R^{v}_{\Theta,\Sigma,e,C}(\rho))^{\top} \subseteq (|[\![C]\!](\theta)| \to |[\![FG]\!](\theta)|) \times \mathcal{X}^{\Sigma,e}_{mon}(FG[C])}$ ***and*** $\boxed{(R^{v}_{\Theta,\Sigma,e,C}(\rho))^{\top\top} \subseteq |[\![C]\!](\theta)| \times \lambda^{\Theta,\Sigma,e}_{\mathbf{mon}}(C)}$

$$(R^{v}_{\Theta,\Sigma,e,C}(\rho))^{\top} := \left\{\langle f, \mathcal{X}\rangle \middle| \emptyset[\Theta] \mid \emptyset[\emptyset] \vdash_{\Sigma[\Sigma],\perp[e]} \mathcal{X}[\ ] : FG[C]. \forall \langle a, M\rangle \in R^{v}_{\Theta,\Sigma,e,C}(\rho).\ \exists V. \mathcal{X}[M] \simeq \mathbf{return}\ V \wedge \langle fa, \mathbf{return}\ V\rangle \in R^{v}_{\Sigma,\perp,FG}(\rho)\right\}$$

$$(R^{v}_{\Theta,\Sigma,e,C}(\rho))^{\top\top} := \left\{\langle c, M\rangle \middle| \forall \langle f, \mathcal{X}\rangle \in (R^{v}_{\Theta,\Sigma,e,C}(\rho))^{\top}. \exists V. \mathcal{X}[M] \simeq \mathbf{return}\ V \wedge \langle fc, \mathbf{return}\ V\rangle \in R^{v}_{\Sigma,\perp,FG}(\rho)\right\}$$

$$R_{\Theta,\Sigma,e,C}(\rho) := (R^{v}_{\Theta,\Sigma,e,C}(\rho))^{\top\top}$$

Figure 4.8: λ_{mon}-calculus logical relations