Affine TES: Type and Effect System

Orpheas van Rooij

May 2023

1 Language

$$
\begin{aligned}
\mathrm{op} &::= + \mid - \mid * \mid / \mid \mathit{not} \mid \mathit{and} \mid \mathit{or} \\
v &::= () \mid \ell\ (\in \mathit{Loc}) \mid b\ (\in \mathit{Bool}) \mid i\ (\in \mathit{Int}) \mid \lambda x.\, e \mid \odot\ (\in \mathrm{op}) \mid (v, v) \mid \texttt{cont}\ \ell\ N \\
e &::= v \mid x \mid e\ e \mid (e, e) \mid \texttt{let}\ (x, x) = e\ \texttt{in}\ e \mid \texttt{if}\ e\ \texttt{then}\ e\ \texttt{else}\ e \\
&\quad \mid \texttt{do}\ e \mid \texttt{shallow-try}\ e\ \texttt{with}\ e \mid e \mid \texttt{deep-try}\ e\ \texttt{with}\ v \mid e \mid \texttt{eff}\ e\ K \\
N &::= \bullet \mid N\ e \mid v\ N \mid (N, e) \mid (v, N) \mid \texttt{let}\ (x, x) = N\ \texttt{in}\ e \mid \texttt{if}\ N\ \texttt{then}\ e\ \texttt{else}\ e \\
&\quad \mid \texttt{do}\ N \mid \texttt{deep-try}\ e\ \texttt{with}\ v \mid N \\
K &::= N \mid \texttt{shallow-try}\ K\ \texttt{with}\ e \mid e \mid \texttt{deep-try}\ K\ \texttt{with}\ v \mid e
\end{aligned}
$$

Figure 1: Syntax of effect values, values, expressions, and evaluation contexts

2 Head Reduction

$$
\begin{array}{rcl}
(\lambda x.\, e)\ v\ /\ \sigma & \rightsquigarrow & e[v/x]\ /\ \sigma \\
\odot\ v\ /\ \sigma & \rightsquigarrow & v'\ /\ \sigma \\
 & & [\![\odot]\!]\ v = v' \\
\odot\ v_1\ v_2\ /\ \sigma & \rightsquigarrow & v\ /\ \sigma \\
 & & v_1\ [\![\odot]\!]\ v_2 = v \\
\texttt{let}\ (x_1, x_2) = (v_1, v_2)\ \texttt{in}\ e_3\ /\ \sigma & \rightsquigarrow & e_3\,[v_1/x_1]\,[v_2/x_2]\ /\ \sigma \\
\texttt{if}\ \texttt{true}\ \texttt{then}\ e_1\ \texttt{else}\ e_2\ /\ \sigma & \rightsquigarrow & e_1\ /\ \sigma \\
\texttt{if}\ \texttt{false}\ \texttt{then}\ e_1\ \texttt{else}\ e_2\ /\ \sigma & \rightsquigarrow & e_2\ /\ \sigma \\
\texttt{do}\ v\ /\ \sigma & \rightsquigarrow & \texttt{eff}\ v\ \bullet\ /\ \sigma \\
\texttt{shallow-try}\ v\ \texttt{with}\ h \mid r\ /\ \sigma & \rightsquigarrow & r\ v\ /\ \sigma \\
\texttt{deep-try}\ v\ \texttt{with}\ h \mid r\ /\ \sigma & \rightsquigarrow & r\ v\ /\ \sigma \\
\texttt{shallow-try}\ (\texttt{eff}\ v\ N)\ \texttt{with}\ h \mid r\ /\ \sigma & \rightsquigarrow & h\ v\ (\texttt{cont}\ \ell\ N)\ /\ \sigma[\ell \mapsto \texttt{false}] \\
 & & \ell \notin \mathrm{dom}\ \sigma \\
\texttt{deep-try}\ (\texttt{eff}\ v\ N)\ \texttt{with}\ h \mid r\ /\ \sigma & \rightsquigarrow & h\ v\ (\texttt{deep-try}\ (\texttt{cont}\ \ell\ N)\ \texttt{with}\ h \mid r)\ /\ \sigma[\ell \mapsto \texttt{false}] \\
 & & \ell \notin \mathrm{dom}\ \sigma \\
(\texttt{cont}\ \ell\ N)\ v\ /\ \sigma[\ell \mapsto \texttt{false}] & \rightsquigarrow & N[v]\ /\ \sigma[\ell \mapsto \texttt{true}] \\
 & & \\
(\texttt{eff}\ v_1\ N)\ e_2\ /\ \sigma & \rightsquigarrow & \texttt{eff}\ v_1\ (N\ e_2)\ /\ \sigma \\
v_1\ (\texttt{eff}\ v_2\ N)\ /\ \sigma & \rightsquigarrow & \texttt{eff}\ v_2\ (v_1\ N)\ /\ \sigma \\
(\texttt{eff}\ v_1\ N, e_2)\ /\ \sigma & \rightsquigarrow & \texttt{eff}\ v_1\ (N, e_2)\ /\ \sigma \\
(v_1, \texttt{eff}\ v_2\ N)\ /\ \sigma & \rightsquigarrow & \texttt{eff}\ v_2\ (v_1, N)\ /\ \sigma \\
\texttt{let}\ (x_1, x_2) = (\texttt{eff}\ v_1\ N)\ \texttt{in}\ e_2\ /\ \sigma & \rightsquigarrow & \texttt{eff}\ v_1\ (\texttt{let}\ (x_1, x_2) = N\ \texttt{in}\ e_2)\ /\ \sigma \\
\texttt{if}\ (\texttt{eff}\ v\ N)\ \texttt{then}\ e\ \texttt{else}\ e\ /\ \sigma & \rightsquigarrow & \texttt{eff}\ v\ (\texttt{if}\ N\ \texttt{then}\ e\ \texttt{else}\ e)\ /\ \sigma \\
\texttt{do}\ (\texttt{eff}\ v\ N)\ /\ \sigma & \rightsquigarrow & \texttt{eff}\ v\ (\texttt{do}\ N)\ /\ \sigma \\
\texttt{deep-try}\ e_1\ \texttt{with}\ v_2 \mid (\texttt{eff}\ v_3\ N)\ /\ \sigma & \rightsquigarrow & \texttt{eff}\ v_3\ (\texttt{deep-try}\ e_1\ \texttt{with}\ v_2 \mid N)\ /\ \sigma
\end{array}
$$

Figure 2: The head reduction relation

3 Types

$$\begin{aligned} \tau,\, \kappa,\, \iota &::= \texttt{unit} \mid \texttt{bool} \mid \texttt{int} \mid \tau \overset{\rho}{\multimap} \tau \mid \tau * \tau \\ \rho &::= \langle\rangle \mid \tau \Rightarrow \tau \end{aligned}$$

Figure 3: Syntax of types, and row signatures

4 Typing Rules

$$\textsc{Unit}\quad \Gamma \vDash () : \rho : \texttt{unit}$$

$$\textsc{Bool}\quad \Gamma \vDash b : \rho : \texttt{bool}$$

$$\textsc{Int}\quad \Gamma \vDash i : \rho : \texttt{int}$$

$$\textsc{Op}\quad \frac{\odot \vDash \tau : \kappa :}{\Gamma \vDash \odot : \rho : \tau \overset{\rho}{\multimap} \kappa}$$

$$\textsc{Sub}\quad \frac{\Gamma \vDash e : \langle\rangle : \tau}{\Gamma \vDash e : \rho : \tau}$$

$$\textsc{Fun}\quad \frac{\Gamma, x : \tau \vDash e : \rho : \kappa}{\Gamma \vDash \lambda x.\, e : \langle\rangle : \tau \overset{\rho}{\multimap} \kappa}$$

$$\textsc{App}\quad \frac{\begin{array}{c}\Gamma_1 \vDash e : \rho : \tau \overset{\rho}{\multimap} \kappa \\ \Gamma_2 \vDash e' : \rho : \tau\end{array}}{\Gamma_1 +\!\!+ \Gamma_2 \vDash e\, e' : \rho : \kappa}$$

$$\textsc{Pair}\quad \frac{\Gamma_1 \vDash e_1 : \rho : \tau \qquad \Gamma_2 \vDash e_2 : \rho : \kappa}{\Gamma_1 +\!\!+ \Gamma_2 \vDash (e_1, e_2) : \rho : \tau * \kappa}$$

$$\textsc{Pair-Elimination}\quad \frac{\Gamma_1 \vDash e_1 : \rho : \tau * \kappa \qquad \Gamma_2, x_1 : \tau, x_2 : \kappa \vDash e_2 : \rho : \iota}{\Gamma_1 +\!\!+ \Gamma_2 \vDash \texttt{let}\ (x_1, x_2) = e_1\ \texttt{in}\ e_2 : \rho : \iota}$$

$$\textsc{If-Then-Else}\quad \frac{\Gamma_1 \vDash e_1 : \rho : \texttt{bool} \qquad \Gamma_2 \vDash e_2 : \rho : \tau \qquad \Gamma_2 \vDash e_3 : \rho : \tau}{\Gamma_1 +\!\!+ \Gamma_2 \vDash \texttt{if}\ e_1\ \texttt{then}\ e_2\ \texttt{else}\ e_3 : \rho : \tau}$$

$$\textsc{Do}\quad \frac{\begin{array}{c}\rho = (\iota \Rightarrow \kappa) \\ \Gamma \vDash e : \rho : \iota\end{array}}{\Gamma \vDash \texttt{do}\ e : \rho : \kappa}$$

$$\textsc{Shallow-Handler}\quad \frac{\begin{array}{c}\Gamma_1 \vDash e : \rho : \tau \qquad \rho = (\iota \Rightarrow \kappa) \\ \Gamma_2 \vDash h : \langle\rangle : \iota \multimap (\kappa \overset{\rho}{\multimap} \tau) \overset{\rho}{\multimap} \tau' \qquad \Gamma_2 \vDash r : \langle\rangle : \tau \overset{\rho}{\multimap} \tau'\end{array}}{\Gamma_1 +\!\!+ \Gamma_2 \vDash \texttt{shallow-try}\ e\ \texttt{with}\ h \mid r : \rho : \tau'}$$

$$\textsc{Deep-Handler}\quad \frac{\begin{array}{c}\Gamma_1 \vDash e : \rho : \tau \qquad \rho = (\iota \Rightarrow \kappa) \\ \varnothing \vDash h : \langle\rangle : \iota \multimap (\kappa \overset{\rho'}{\multimap} \tau') \overset{\rho'}{\multimap} \tau' \qquad \Gamma_2 \vDash r : \langle\rangle : \tau \overset{\rho'}{\multimap} \tau'\end{array}}{\Gamma_1 +\!\!+ \Gamma_2 \vDash \texttt{deep-try}\ e\ \texttt{with}\ h \mid r : \rho' : \tau'}$$

Figure 4: Semantic typing rules

5 Protocol

Protocol

$$\begin{aligned} \mathit{Protocol} &\triangleq \mathit{Val} \to (\mathit{Val} \to \mathit{iProp}) \to \mathit{iProp} \\ !\,\vec{x}\,(v)\,\{P\}.\,?\,\vec{y}\,(w)\,\{Q\} &\triangleq \lambda u\,\Psi.\,\exists \vec{x}.\,\ulcorner u = v \urcorner \ast P \ast (\forall \vec{y}.\, Q \mathrel{-\!\!\ast} \Psi(w)) \\ \bot &\triangleq !\,x\,(x)\,\{\mathsf{False}\}.\,?\,y\,(y)\,\{\mathsf{True}\} \end{aligned}$$

Figure 5: Definition of a protocol

6 Extended Weakest Precondition

The extended Weakest Precondition that we will use for the semantic typing is an enhancement of the usual weakest precondition that captures safety to incorporate reasoning with effects and effect handlers.

The *ewp* e $\langle\Psi\rangle\{\Phi\}$ specifies that expression e can either call an effect according to protocol Ψ or it evaluates safely such that if it evaluates to a value that value satisfies Φ.

Extended weakest precondition

$$\begin{aligned} \mathit{ewp}\ v\ \langle\Psi\rangle\{\Phi\} &\triangleq \mathrel{\models\!\!\!\Rrightarrow} \Phi(v) \\ \mathit{ewp}\ (\texttt{eff}\ v\ N)\ \langle\Psi\rangle\{\Phi\} &\triangleq (\uparrow\Psi)\ v\ (\lambda w.\, \triangleright\, \mathit{ewp}\ N[w]\ \langle\Psi\rangle\{\Phi\}) \\ \mathit{ewp}\ e\ \langle\Psi\rangle\{\Phi\} &\triangleq \forall \sigma.\, S(\sigma) \Rrightarrow\!\!\ast \\ &\quad \begin{cases} \exists e', \sigma'.\, e \,/\, \sigma \longrightarrow e' \,/\, \sigma' \ast \\ \forall e', \sigma'.\, e \,/\, \sigma \longrightarrow e' \,/\, \sigma' \Rrightarrow\!\!\ast \triangleright \mathrel{\models\!\!\!\Rrightarrow} \\ \qquad\qquad S(\sigma') \ast \mathit{ewp}\ e'\ \langle\Psi\rangle\{\Phi\} \end{cases} \end{aligned}$$

Upward closure

$$(\uparrow\Psi)\ v\ \Phi \triangleq \exists \Phi'.\, \Psi\, v\, \Phi' \ast (\forall w.\, \Phi'(w) \mathrel{-\!\!\ast} \Phi(w))$$

Figure 6: Definition of the weakest precondition

7 Semantic Interpretation

Interpretation of types

$$
\begin{aligned}
\mathcal{V}[\![\texttt{unit}]\!](v) &\triangleq \ulcorner v = () \urcorner \\
\mathcal{V}[\![\texttt{bool}]\!](v) &\triangleq \exists\, b.\, \ulcorner v = \#b \urcorner \\
\mathcal{V}[\![\texttt{int}]\!](v) &\triangleq \exists\, i.\, \ulcorner v = \#i \urcorner \\
\mathcal{V}[\![\tau \xrightarrow{\rho}\!\circ\, \kappa]\!](v) &\triangleq \forall\, w.\, \mathcal{V}[\![\tau]\!](w) \mathrel{-\!\!*} \mathit{ewp}\ (v\ w)\ \langle \mathcal{R}[\![\rho]\!] \rangle \{ \mathcal{V}[\![\kappa]\!] \} \\
\mathcal{V}[\![\tau * \kappa]\!](v) &\triangleq \exists\, v_1\, v_2.\, \ulcorner v = (v_1, v_2) \urcorner * \mathcal{V}[\![\tau]\!](v_1) * \mathcal{V}[\![\kappa]\!](v_2)
\end{aligned}
$$

Interpretation of a row

$$
\begin{aligned}
\mathcal{R}[\![\langle\rangle]\!] &\triangleq \bot \\
\mathcal{R}[\![\tau \Rightarrow \iota]\!] &\triangleq\ !\, x\, (x)\, \{ \mathcal{V}[\![\tau]\!](x) \}.\, ?\, y\, (y)\, \{ \mathcal{V}[\![\kappa]\!](y) \}
\end{aligned}
$$

Interpretation of typing judgments

$$
\begin{aligned}
\Gamma \vDash e : \rho : \tau &\triangleq \forall\, vs.\, \mathcal{G}[\![\Gamma]\!](vs) \mathrel{-\!\!*} \mathit{ewp}\ e[vs]\ \langle \mathcal{R}[\![\rho]\!] \rangle \{ \mathcal{V}[\![\tau]\!] \} \\
\mathcal{G}[\![\Gamma]\!](vs) &\triangleq \forall\, \{ x \mapsto \tau \} \subseteq \Gamma.\, \mathcal{V}[\![\tau]\!](vs(x))
\end{aligned}
$$

Figure 7: Interpretation of types, rows, and typing judgments