Fully Reflexive Intensional Type Analysis in Type Erasure Semantics

Bratin Saha, Valery Trifonov, and Zhong Shao

ICFP 2000, TIC 2000

Sukyoung Ryu

November 22, 2000

Motivation

- Harper and Morrisett's intensional type analysis

  only allows the analysis of monotypes; it does not support analysis of polymorphic or recursive types

$$\texttt{typerec}\ \mu\ \texttt{of}\ [t.\sigma]\ (e_{\mathsf{int}};\ e_{\rightarrow})$$

- *fully reflexive* intensional type analysis

  type-analyzing operations are applicable to the type of any runtime value in the language

$$\texttt{typecase}[\tau]\ \tau'\ \texttt{of}\ (e_{\mathsf{int}};\ e_{\rightarrow};\ e_{\forall};\ e_{\forall^+};\ e_{\mu})$$

Harper and Morrisett's λ_i^{ML}

- a predicative variant of Girard's F_w

- `Typerec` operator

$$\texttt{Typerec int of } (\tau_{\mathsf{int}};\, \tau_{\to}) \leadsto \tau_{\mathsf{int}}$$

$$\texttt{Typerec } (\tau_1 \to \tau_2) \texttt{ of } (\tau_{\mathsf{int}};\, \tau_{\to}) \leadsto \\ \tau_{\to}\, \tau_1\, \tau_2\, (\texttt{Typerec } \tau_1 \texttt{ of } (\tau_{\mathsf{int}};\, \tau_{\to}))\, (\texttt{Typerec } \tau_2 \texttt{ of } (\tau_{\mathsf{int}};\, \tau_{\to}))$$

Analyzing Polymorphic Types

- arrow constructor

$$\begin{aligned} \twoheadrightarrow \quad &: \quad \Omega \to \Omega \to \Omega \\ \tau \to \tau' &\equiv ((\twoheadrightarrow)\, \tau)\, \tau' \end{aligned}$$

- forall constructor for predicative F_w

$$\begin{aligned} \forall \quad &: \quad (\Omega \to \Omega) \to \Omega \\ \forall \alpha\!:\!\Omega.\, \tau &\equiv \forall\!\!\!\forall\, (\lambda \alpha\!:\!\Omega.\, \tau) \end{aligned}$$

- forall constructor for impredicative F_w

$$\begin{aligned} \forall_\kappa \quad &: \quad (\kappa \to \Omega) \to \Omega \\ \forall \alpha\!:\!\kappa.\, \tau &\equiv \forall\!\!\!\forall_\kappa\, (\lambda \alpha\!:\!\kappa.\, \tau) \end{aligned}$$

Kind Polymorphism (I)

- $\mathbb{\forall} : \forall\chi.\,(\chi \to \Omega) \to \Omega$
  $\forall\alpha\!:\!\kappa.\,\tau \equiv (\mathbb{\forall}\,[\kappa])\,(\lambda\alpha\!:\!\kappa.\,\tau)$

- `Typerec` $(\mathbb{\forall}\,[\kappa]\,\tau)$ `of` $(\tau_{\mathsf{int}};\ \tau_{\to};\ \tau_{\forall};\ \tau_{\forall^+})$
  $\tau_{\forall}\ :\ \forall\chi.\,(\chi \to \Omega) \to (\chi \to \Omega) \to \Omega$
  $\qquad \Lambda\chi.\,\lambda\alpha\!:\!\chi \to \Omega.\,\lambda\beta\!:\!\chi \to \Omega.\,\tau'_{\forall}$

- `typecase[`τ`]` $(\mathbb{\forall}\,[\kappa]\,\tau')$ `of` $(e_{\mathsf{int}};\ e_{\to};\ e_{\forall};\ e_{\forall^+})$
  $e_{\forall} : \forall^{+}\chi.\,\forall\alpha\!:\!\chi \to \Omega.\,\tau\,(\mathbb{\forall}\,[\chi]\,\alpha)$
  $\qquad \Lambda^{+}\chi.\,\Lambda\alpha\!:\!\chi \to \Omega.\,e'_{\forall}$

Kind Polymorphism (II)

- $\mathbb{\forall}^{+} : (\forall\chi.\,\Omega) \to \Omega$
  $\forall^{+}\chi.\,\tau \equiv \mathbb{\forall}^{+}(\Lambda\chi.\,\tau)$

- $\texttt{Typerec}\ (\mathbb{\forall}^{+}\tau)\ \texttt{of}\ (\tau_{\mathsf{int}};\ \tau_{\to};\ \tau_{\forall};\ \tau_{\forall^{+}})$
  $\tau_{\mathbb{\forall}^{+}} :\ (\forall\chi.\,\Omega) \to (\forall\chi.\,\Omega) \to \Omega$

- $\texttt{typecase[}\tau\texttt{]}\ (\mathbb{\forall}^{+}\tau')\ \texttt{of}\ (e_{\mathsf{int}};\ e_{\to};\ e_{\forall};\ e_{\forall^{+}})$
  $e_{\forall^{+}} : \forall\alpha{:}(\forall\chi.\,\Omega).\,\tau\,(\mathbb{\forall}^{+}\alpha)$

λ_i^{ML} + Analyzing Polymorphic Types

$$
\begin{array}{llll}
(kinds) & \kappa & ::= & \Omega \mid \kappa \to \kappa' \mid \chi \mid \forall\chi.\,\kappa \\[1em]
(types) & \tau & ::= & \texttt{int} \mid \twoheadrightarrow \mid \forall\!\!\!\forall \mid \forall\!\!\!\forall^{+} \mid \alpha \mid \Lambda\chi.\,\tau \mid \lambda\alpha\!:\!\kappa.\,\tau \mid \tau\,[\kappa] \mid \tau\,\tau' \\
 & & \mid & \texttt{Typerec}\ \tau\ \texttt{of}\ (\tau_{\mathsf{int}};\ \tau_{\to};\ \tau_{\forall};\ \tau_{\forall^{+}}) \\[1em]
(values) & v & ::= & i \mid \Lambda^{+}\chi.\,v \mid \Lambda\alpha\!:\!\kappa.\,v \mid \lambda x\!:\!\tau.\,e \\[1em]
(terms) & e & ::= & v \mid x \mid e\,[\kappa]^{+} \mid e\,[\tau] \mid e\,e' \\
 & & \mid & \texttt{typecase[}\tau\texttt{]}\ \tau'\ \texttt{of}\ (e_{\mathsf{int}};\ e_{\to};\ e_{\forall};\ e_{\forall^{+}})
\end{array}
$$

Formation rules

$$\frac{\mathcal{E} \vdash \Delta}{\begin{array}{ll}\mathcal{E};\Delta \vdash \texttt{int} & : \Omega \\ \mathcal{E};\Delta \vdash (\twoheadrightarrow) & : \Omega \to \Omega \to \Omega \\ \mathcal{E};\Delta \vdash \mathbb{\forall} & : \forall\chi.\,(\chi \to \Omega) \to \Omega \\ \mathcal{E};\Delta \vdash \mathbb{\forall}^{+} & : (\forall\chi.\,\Omega) \to \Omega\end{array}}$$

$$\frac{\begin{array}{ll}\mathcal{E};\Delta \vdash \tau & : \Omega \\ \mathcal{E};\Delta \vdash \tau_{\mathsf{int}} & : \Omega \\ \mathcal{E};\Delta \vdash \tau_{\to} & : \Omega \to \Omega \to \Omega \to \Omega \to \Omega \\ \mathcal{E};\Delta \vdash \tau_{\forall} & : \forall\chi.\,(\chi \to \Omega) \to (\chi \to \Omega) \to \Omega \\ \mathcal{E};\Delta \vdash \tau_{\forall^{+}} & : (\forall\chi.\,\Omega) \to (\forall\chi.\,\Omega) \to \Omega\end{array}}{\mathcal{E};\Delta \vdash \texttt{Typerec}\ \tau\ \texttt{of}\ (\tau_{\mathsf{int}};\ \tau_{\to};\ \tau_{\forall};\ \tau_{\forall^{+}})\ :\ \Omega}$$

$$\frac{\begin{array}{ll}\mathcal{E};\Delta \vdash \tau & : \Omega \to \Omega \\ \mathcal{E};\Delta \vdash \tau' & : \Omega \\ \mathcal{E};\Delta;\Gamma \vdash e_{\mathsf{int}} & : \tau\ \texttt{int} \\ \mathcal{E};\Delta;\Gamma \vdash e_{\to} & : \forall\alpha{:}\Omega.\,\forall\alpha'{:}\Omega.\,\tau\,(\alpha \to \alpha') \\ \mathcal{E};\Delta;\Gamma \vdash e_{\forall} & : \forall^{+}\chi.\,\forall\alpha{:}\chi \to \Omega.\,\tau\,(\mathbb{\forall}\,[\chi]\,\alpha) \\ \mathcal{E};\Delta;\Gamma \vdash e_{\forall^{+}} & : \forall\alpha{:}(\forall\chi.\,\Omega).\,\tau\,(\mathbb{\forall}^{+}\alpha)\end{array}}{\mathcal{E};\Delta;\Gamma \vdash \texttt{typecase[}\tau\texttt{]}\ \tau'\ \texttt{of}\ (e_{\mathsf{int}};\ e_{\to};\ e_{\forall};\ e_{\forall^{+}})\ :\ \tau\,\tau'}$$

Reduction Rules

- Selected λ_i^{P+} type reduction rules

$$\frac{\mathcal{E};\Delta,\alpha:\kappa' \vdash \texttt{Typerec}\ (\tau\ \alpha)\ \texttt{of}\ (\tau_{\mathsf{int}};\ \tau_{\rightarrow};\ \tau_{\forall};\ \tau_{\forall^+}) \leadsto \tau'\ :\ \Omega}{\begin{array}{l}\mathcal{E};\Delta \vdash \texttt{Typerec}\ (\forall\,[\kappa']\,\tau)\ \texttt{of}\ (\tau_{\mathsf{int}};\ \tau_{\rightarrow};\ \tau_{\forall};\ \tau_{\forall^+})\\ \qquad \leadsto \tau_{\forall}\,[\kappa']\,\tau\,(\lambda\alpha:\kappa'.\,\tau')\ :\ \Omega\end{array}}$$

$$\frac{\mathcal{E},\chi;\Delta \vdash \texttt{Typerec}\ (\tau\,[\chi])\ \texttt{of}\ (\tau_{\mathsf{int}};\ \tau_{\rightarrow};\ \tau_{\forall};\ \tau_{\forall^+}) \leadsto \tau'\ :\ \Omega}{\mathcal{E};\Delta \vdash \texttt{Typerec}\ (\forall^+\tau)\ \texttt{of}\ (\tau_{\mathsf{int}};\ \tau_{\rightarrow};\ \tau_{\forall};\ \tau_{\forall^+}) \leadsto \tau_{\forall^+}\,\tau\,(\Lambda\chi.\,\tau')\ :\ \Omega}$$

- Selected term reduction rules of λ_i^{P+}

$$\texttt{typecase}[\tau]\ (\forall\,[\kappa]\,\tau')\ \texttt{of}\ (e_{\mathsf{int}};\ e_{\rightarrow};\ e_{\forall};\ e_{\forall^+};\ e_{\mu}) \leadsto e_{\forall}\,[\kappa]^{+}[\tau']$$

$$\texttt{typecase}[\tau]\ (\forall^+\tau')\ \texttt{of}\ (e_{\mathsf{int}};\ e_{\rightarrow};\ e_{\forall};\ e_{\forall^+};\ e_{\mu}) \quad \leadsto e_{\forall^+}\,[\tau']$$

Example: Marshalling (I)

$$\texttt{newid} : \forall\alpha_1{:}\Omega.\,\forall\alpha_2{:}\Omega.\,(\texttt{Tran}\,\alpha_1{\to}\texttt{Tran}\,\alpha_2){\to}\texttt{Tran}\,(\alpha_1{\to}\alpha_2)$$
$$\texttt{rapp} : \forall\alpha_1{:}\Omega.\,\forall\alpha_2{:}\Omega.\,\texttt{Tran}\,(\alpha_1{\to}\alpha_2){\to}\texttt{Tran}\,\alpha_1{\to}\texttt{Tran}\,\alpha_2$$

$$\begin{aligned}
\texttt{M}\,[\texttt{int}] &= \lambda x{:}\texttt{int}.\,x \\
\texttt{M}\,[\alpha_1 \to \alpha_2] &= \lambda x{:}\alpha_1 \to \alpha_2. \\
&\qquad \texttt{newid}\,[\alpha_1]\,[\alpha_2]\,(\lambda x'{:}\texttt{Tran}\,\alpha_1.\,\texttt{M}\,[\alpha_2]\,(x\,(\texttt{U}\,[\alpha_1]\,x')))
\end{aligned}$$

$$\begin{aligned}
\texttt{U}\,[\texttt{int}] &= \lambda x{:}\texttt{Tran}\,(\texttt{int}).\,x \\
\texttt{U}\,[\alpha_1 \to \alpha_2] &= \lambda x{:}\texttt{Tran}\,(\alpha_1 \to \alpha_2).\,\lambda x'{:}\alpha_1. \\
&\qquad \texttt{U}\,[\alpha_2]\,(\texttt{rapp}\,[\alpha_1]\,[\alpha_2]\,x\,(\texttt{M}\,[\alpha_1]\,x'))
\end{aligned}$$

Example: Marshalling (II)

$$\begin{aligned}\texttt{newpid} &: \forall^{+}\chi.\, \forall\alpha{:}\chi \to \Omega.\, (\forall\alpha'{:}\chi.\, \texttt{Tran}\,(\alpha\,\alpha')) \to \texttt{Tran}\,(\mathbb{\forall}\,[\chi]\,\alpha)\\ \texttt{rtapp} &: \forall^{+}\chi.\, \forall\alpha{:}\chi \to \Omega.\, \texttt{Tran}\,(\mathbb{\forall}\,[\chi]\,\alpha) \to \forall\alpha'{:}\chi.\, \texttt{Tran}\,(\alpha\,\alpha')\end{aligned}$$

$$\texttt{M}\,[\mathbb{\forall}\,[\chi]\,\alpha] = \lambda x{:}\mathbb{\forall}\,[\chi]\,\alpha.\, \texttt{newpid}\,[\chi]^{+}\,[\alpha]\,(\Lambda\alpha'{:}\chi.\, \texttt{M}\,[\alpha\,\alpha']\,(x\,[\alpha']))$$

$$\texttt{U}\,[\mathbb{\forall}\,[\chi]\,\alpha] = \lambda x{:}\texttt{Tran}\,(\mathbb{\forall}\,[\chi]\,\alpha).\, \Lambda\alpha'{:}\chi.\, \texttt{U}\,[\alpha\,\alpha']\,(\texttt{rtapp}\,[\chi]^{+}\,[\alpha]\,x\,[\alpha'])$$

Example: Marshalling (III)

$$\mathtt{newkid} : \forall\alpha : (\forall\chi.\,\Omega).\,(\overset{+}{\forall}\chi.\,\mathtt{Tran}\,(\alpha\,[\chi])) \to \mathtt{Tran}\,(\overset{+}{\mathbb{\forall}}\alpha)$$

$$\mathtt{rkapp} : \forall\alpha : (\forall\chi.\,\Omega).\,\mathtt{Tran}\,(\overset{+}{\mathbb{\forall}}\alpha) \to \overset{+}{\forall}\chi.\,\mathtt{Tran}\,(\alpha\,[\chi])$$

$$\mathtt{M}\,[\overset{+}{\mathbb{\forall}}\alpha] = \lambda x : \overset{+}{\mathbb{\forall}}\alpha.\,\mathtt{newkid}\,[\alpha]\,(\overset{+}{\Lambda}\chi.\,\mathtt{M}\,[\alpha\,[\chi]]\,(x\,[\chi]^{+}))$$

$$\mathtt{U}\,[\overset{+}{\mathbb{\forall}}\alpha] = \lambda x : \mathtt{Tran}\,(\overset{+}{\mathbb{\forall}}\alpha).\,\overset{+}{\Lambda}\chi.\,\mathtt{U}\,[\alpha\,[\chi]]\,(\mathtt{rkapp}\,[\alpha]\,x\,[\chi]^{+})$$

The source language λ_i^{P+}

- λ_i^{ML} + analyzing polymorphic types + analyzing recursive types

$$
\begin{array}{llcl}
(\textit{kinds}) & \kappa & ::= & \ldots \\
(\textit{types}) & \tau & ::= & \ldots \mid \mu \mid \texttt{Place} \\
(\textit{values}) & v & ::= & \ldots \mid \texttt{fix}\, x\!:\!\tau.\, v \mid \texttt{fold[}\tau\texttt{]}\ v \\
(\textit{terms}) & e & ::= & \ldots \mid \texttt{fold[}\tau\texttt{]}\ e \mid \texttt{unfold[}\tau\texttt{]}\ e \\
 & & \mid & \texttt{typecase[}\tau\texttt{]}\ \tau'\ \texttt{of}\ (e_{\mathsf{int}};\ e_{\to};\ e_{\forall};\ e_{\forall^+};\ e_{\mu})
\end{array}
$$

Analyzing Recursive Types

- $\mu : (\Omega \to \Omega) \to \Omega$
  $\texttt{Place} : \Omega \to \Omega$

- Selected formation rules of λ_i^{P+}

$$\frac{\mathcal{E}; \Delta \vdash \tau \;:\; \Omega \to \Omega \quad \mathcal{E}; \Delta; \Gamma \vdash e \;:\; \tau\,(\mu\tau)}{\mathcal{E}; \Delta; \Gamma \vdash \texttt{fold[}\tau\texttt{]}\ e \;:\; \mu\tau}$$

$$\frac{\mathcal{E}; \Delta \vdash \tau \;:\; \Omega \to \Omega \quad \mathcal{E}; \Delta; \Gamma \vdash e \;:\; \mu\tau}{\mathcal{E}; \Delta; \Gamma \vdash \texttt{unfold[}\tau\texttt{]}\ e \;:\; \tau\,(\mu\tau)}$$

$$\frac{\mathcal{E}; \Delta; \Gamma \vdash e_\mu \;:\; \forall \alpha : \Omega \to \Omega.\, \tau\,(\mu\,\alpha)}{\mathcal{E}; \Delta; \Gamma \vdash \texttt{typecase[}\tau\texttt{]}\ \tau'\ \texttt{of}\ (e_{\mathsf{int}};\ e_{\to};\ e_{\forall};\ e_{\forall^+};\ e_\mu) \;:\; \tau\,\tau'}$$

Reduction Rules

- Selected λ_i^{P+} type reduction rules

$$\frac{\mathcal{E};\Delta,\alpha\!:\!\Omega \vdash \texttt{Typerec}\ (\tau\ (\texttt{Place}\,\alpha))\ \texttt{of}\ (\tau_{\mathsf{int}};\ \tau_{\rightarrow};\ \tau_{\forall};\ \tau_{\forall^+}) \leadsto \tau'\ :\ \Omega}{\mathcal{E};\Delta \vdash \texttt{Typerec}\ (\mu\,\tau)\ \texttt{of}\ (\tau_{\mathsf{int}};\ \tau_{\rightarrow};\ \tau_{\forall};\ \tau_{\forall^+}) \leadsto \mu\,(\lambda\alpha\!:\!\Omega.\,\tau')\ :\ \Omega}$$

$$\frac{\mathcal{E};\Delta \vdash \texttt{Typerec}\ (\texttt{Place}\,\tau)\ \texttt{of}\ (\tau_{\mathsf{int}};\ \tau_{\rightarrow};\ \tau_{\forall};\ \tau_{\forall^+})\ :\ \Omega}{\mathcal{E};\Delta \vdash \texttt{Typerec}\ (\texttt{Place}\,\tau)\ \texttt{of}\ (\tau_{\mathsf{int}};\ \tau_{\rightarrow};\ \tau_{\forall};\ \tau_{\forall^+}) \leadsto \tau\ :\ \Omega}$$

- Selected λ_i^{P+} term reduction rules

$$\texttt{typecase}[\tau]\ (\mu\,\tau')\ \texttt{of}\ (e_{\mathsf{int}};\ e_{\rightarrow};\ e_{\forall};\ e_{\forall^+};\ e_{\mu}) \leadsto e_{\mu}\,[\tau']$$

$$\begin{aligned}&\texttt{typecase}[\tau]\ (\texttt{Place}\,\tau')\ \texttt{of}\ (e_{\mathsf{int}};\ e_{\rightarrow};\ e_{\forall};\ e_{\forall^+};\ e_{\mu}) \leadsto \\ &\quad \texttt{typecase}[\tau]\ (\texttt{Place}\,\tau')\ \texttt{of}\ (e_{\mathsf{int}};\ e_{\rightarrow};\ e_{\forall};\ e_{\forall^+};\ e_{\mu})\end{aligned}$$