COMPILING POLYMORPHISM USING INTENSIONAL TYPE ANALYSIS

R. HARPER & G. MORRISETT
POPL 1995

PRESENTED BY Oukseh Lee
ROPAS, KAIST

Nov. 4, 2000

Languages

- Source : Mini-ML

$$\tau ::= t \mid int \mid \tau \to \tau \mid \tau \times \tau$$
$$\sigma ::= \tau \mid \forall t.\sigma$$
$$e ::= x \mid n \mid (e, e) \mid \pi_1 e \mid \pi_2 e$$
$$\mid \lambda x.e \mid e\ e \mid \text{let } x = v \text{ in } e$$
$$v ::= x \mid n \mid (v_1, v_2) \mid \lambda x.e$$

- Target : λ_i^{ML}

$$\kappa ::= \Omega \mid \kappa \to \kappa$$
$$\mu ::= t \mid Int \mid \to(\mu, \mu) \mid \times(\mu, \mu)$$
$$\mid \lambda t :: \kappa.\mu \mid \mu[\mu] \mid \text{Typerec } \mu \text{ of } (\mu_i, \mu_\to, \mu_\times)$$
$$\sigma ::= T(\mu) \mid int \mid \sigma \to \sigma \mid \sigma \times \sigma$$
$$\mid \forall t :: \kappa.\sigma$$

$$e ::= x \mid n \mid \lambda x : \sigma.\ e \mid @^{\sigma}\ e\ e$$
$$\mid (e, e)^{\sigma,\sigma} \mid \pi_1^{\sigma,\sigma} e \mid \pi_2^{\sigma,\sigma} e$$
$$\mid \Lambda t :: \kappa.\ e \mid e[\mu]$$
$$\mid \text{typerec } \mu \text{ of } [t.\sigma]\ (e_i, e_\to, e_\times)$$

typerec / Typerec <u>examples</u>

- $$\begin{aligned}\text{Typerec}\ & \text{RecArray}[\text{Int}] = \text{intarray}\\ \mid\ & \text{RecArray}[\text{Real}] = \text{realarray}\\ \mid\ & \text{RecArray}[\tau_1 \times \tau_2] = \\ & \qquad \text{RecArray}[\tau_1] \times \text{RecArray}[\tau_2]\end{aligned}$$

  $\text{RecArray} \equiv$

  $$\lambda t :: \Omega.\ \text{Typerec}\ t\ \text{of}$$
  $$(\text{intarray},\ \text{realarray},$$
  $$\lambda t_1 \cdots t_4.\ \times(t_3, t_4))$$

- $$\begin{aligned}\text{typerec}\ & \text{sub}[\text{Int}] = \text{intsub}\\ \mid\ & \text{Sub}[\text{Real}] = \text{realsub}\\ \mid\ & \text{Sub}[\tau_1 \times \tau_2] = \\ & \quad \lambda((x, y), i).\ (\text{sub}[\tau_1](x, i),\\ & \qquad\qquad\qquad \text{Sub}[\tau_2](y, i))\end{aligned}$$

  $\text{sub} \equiv$

  $$\Lambda t :: \Omega.\ \text{typerec}\ t\ \text{of}\ \left[t.\ \begin{matrix}T(\text{RecArray}[t])\\ \times\ \text{int} \to T(t)\end{matrix}\right]$$
  $$\left(\begin{aligned}& \text{intsub},\ \text{realsub},\\ & \Lambda t_1, t_2 :: \Omega.\ \lambda \text{sub}_{\tau_1}.\ \lambda \text{sub}_{\tau_2}.\\ & \quad \lambda((x, y), i).\ (\text{sub}_{\tau_1}(x, i),\ \text{Sub}_{\tau_2}(y, i))\end{aligned}\right)$$

$$(int) \quad \Delta;\Gamma \rhd \bar{n} : \mathsf{int}$$

$$(var) \quad \frac{FTV([\tau_n/t_n](\cdots([\tau_1/t_1]\tau)\cdots)) \subseteq \Delta}{\Delta;\Gamma \uplus \{x : \forall t_1,\ldots,t_n.\tau\} \rhd x : [\tau_n/t_n](\cdots([\tau_1/t_1]\tau)\cdots)}$$

$$(pair) \quad \frac{\Delta;\Gamma \rhd e_1 : \tau_1 \qquad \Delta;\Gamma \rhd e_2 : \tau_2}{\Delta;\Gamma \rhd \langle e_1, e_2\rangle : \tau_1 \times \tau_2}$$

$$(\pi) \quad \frac{\Delta;\Gamma \rhd e : \tau_1 \times \tau_2}{\Delta;\Gamma \rhd \pi_i\, e : \tau_i} \quad (i = 1, 2)$$

$$(abs) \quad \frac{\Delta;\Gamma \uplus \{x : \tau_1\} \rhd e : \tau_2}{\Delta;\Gamma \rhd \lambda x.\, e : \tau_1 \to \tau_2}$$

$$(app) \quad \frac{\Delta;\Gamma \rhd e_1 : \tau' \to \tau \qquad \Delta;\Gamma \rhd e_2 : \tau'}{\Delta;\Gamma \rhd e_1\, e_2 : \tau}$$

$$(let) \quad \frac{\begin{array}{c}\Delta \uplus \{t_1,\ldots,t_n\};\Gamma \rhd v : \tau' \\ \Delta;\Gamma \uplus \{x : \forall t_1,\ldots,t_n.\tau'\} \rhd e : \tau\end{array}}{\Delta;\Gamma \rhd \mathsf{let}\, x = v \,\mathsf{in}\, e : \tau}$$

Figure 1: Mini-ML Typing Rules

$$(val) \quad v \hookrightarrow v$$

$$(pair) \quad \frac{e_1 \hookrightarrow v_1 \qquad e_2 \hookrightarrow v_2}{\langle e_1, e_2 \rangle^{\sigma_1, \sigma_2} \hookrightarrow \langle v_1, v_2 \rangle^{\sigma_1, \sigma_2}}$$

$$(proj) \quad \frac{e \hookrightarrow \langle v_1, v_2 \rangle^{\sigma'_1, \sigma'_2}}{\pi_i^{\sigma_1, \sigma_2}\, e \hookrightarrow v_i} \quad (i = 1, 2)$$

$$(app) \quad \frac{e_1 \hookrightarrow \lambda x{:}\sigma'.\, e \qquad e_2 \hookrightarrow v' \qquad [v'/x]e \hookrightarrow v}{@^{\sigma}\, e_1\, e_2 \hookrightarrow v}$$

$$(tapp) \quad \frac{e \hookrightarrow \Lambda t{::}\kappa.\, e' \qquad [\mu/t]e' \hookrightarrow v}{e[\mu] \hookrightarrow v}$$

$$(trec\text{-}int) \quad \frac{\mu \hookrightarrow \mathsf{Int} \qquad e_{\mathsf{i}} \hookrightarrow v}{\mathsf{typerec}\ \mu\ \mathsf{of}\ [t.\sigma](e_{\mathsf{i}} | e_{\rightarrow} | e_{\times}) \hookrightarrow v}$$

$$(trec\text{-}fn) \quad \frac{\begin{array}{c} \mu \hookrightarrow\ \rightarrow(\mu_1, \mu_2) \\ \mathsf{typerec}\ \mu_1\ \mathsf{of}\ [t.\sigma](e_{\mathsf{i}} | e_{\rightarrow} | e_{\times}) \hookrightarrow v_1 \\ \mathsf{typerec}\ \mu_2\ \mathsf{of}\ [t.\sigma](e_{\mathsf{i}} | e_{\rightarrow} | e_{\times}) \hookrightarrow v_2 \\ @^{[\mu_2/t]\sigma}(@^{[\mu_1/t]\sigma}(e_{\rightarrow}[\mu_1][\mu_2])\ v_1)\ v_2 \hookrightarrow v \end{array}}{\mathsf{typerec}\ \mu\ \mathsf{of}\ [t.\sigma](e_{\mathsf{i}} | e_{\rightarrow} | e_{\times}) \hookrightarrow v}$$

$$(trec\text{-}pair) \quad \frac{\begin{array}{c} \mu \hookrightarrow\ \times(\mu_1, \mu_2) \\ \mathsf{typerec}\ \mu_1\ \mathsf{of}\ [t.\sigma](e_{\mathsf{i}} | e_{\rightarrow} | e_{\times}) \hookrightarrow v_1 \\ \mathsf{typerec}\ \mu_2\ \mathsf{of}\ [t.\sigma](e_{\mathsf{i}} | e_{\rightarrow} | e_{\times}) \hookrightarrow v_2 \\ @^{[\mu_2/t]\sigma}(@^{[\mu_1/t]\sigma}(e_{\times}[\mu_1][\mu_2])\ v_1)\ v_2 \hookrightarrow v \end{array}}{\mathsf{typerec}\ \mu\ \mathsf{of}\ [t.\sigma](e_{\mathsf{i}} | e_{\rightarrow} | e_{\times}) \hookrightarrow v}$$

Figure 3: Operational Semantics for λ_i^{ML}

$$\frac{\Delta \uplus \{t :: \kappa'\} \triangleright \mu_1 :: \kappa \qquad \Delta \triangleright \mu_2 :: \kappa'}{\Delta \triangleright (\lambda t :: \kappa'.\mu_1)[\mu_2] \equiv [\mu_2/t]\mu_1 :: \kappa} \qquad \frac{\begin{array}{c}\Delta \triangleright \mu_{\mathsf{i}} :: \kappa \\ \Delta \triangleright \mu_{\to} :: \Omega \to \Omega \to \kappa \to \kappa \to \kappa \\ \Delta \triangleright \mu_{\times} :: \Omega \to \Omega \to \kappa \to \kappa \to \kappa\end{array}}{\Delta \triangleright \mathsf{Typerec\ Int\ of}\ (\mu_{\mathsf{i}}|\mu_{\to}|\mu_{\times}) \equiv \mu_{\mathsf{i}} :: \kappa}$$

$$\frac{\begin{array}{c}\Delta \triangleright \mu_1 :: \Omega \qquad \Delta \triangleright \mu_2 :: \Omega \qquad \Delta \triangleright \mu_{\mathsf{i}} :: \kappa \\ \Delta \triangleright \mu_{\to} :: \Omega \to \Omega \to \kappa \to \kappa \to \kappa \qquad \Delta \triangleright \mu_{\times} :: \Omega \to \Omega \to \kappa \to \kappa \to \kappa\end{array}}{\left\{\begin{array}{c}\Delta \triangleright \mathsf{Typerec}\ (\to(\mu_1,\mu_2))\ \mathsf{of}\ (\mu_{\mathsf{i}}|\mu_{\to}|\mu_{\times}) \equiv \mu_{\to}\ \mu_1\ \mu_2\ (\mathsf{Typerec}\ \mu_1\ \mathsf{of}\ (\mu_{\mathsf{i}}|\mu_{\to}|\mu_{\times}))\ (\mathsf{Typerec}\ \mu_2\ \mathsf{of}\ (\mu_{\mathsf{i}}|\mu_{\to}|\mu_{\times})) :: \kappa \\ \Delta \triangleright \mathsf{Typerec}\ (\times(\mu_1,\mu_2))\ \mathsf{of}\ (\mu_{\mathsf{i}}|\mu_{\to}|\mu_{\times}) \equiv \mu_{\times}\ \mu_1\ \mu_2\ (\mathsf{Typerec}\ \mu_1\ \mathsf{of}\ (\mu_{\mathsf{i}}|\mu_{\to}|\mu_{\times}))\ (\mathsf{Typerec}\ \mu_2\ \mathsf{of}\ (\mu_{\mathsf{i}}|\mu_{\to}|\mu_{\times})) :: \kappa\end{array}\right\}}$$

Figure 2: Constructor Equivalence

$$(var) \quad \frac{\Delta \triangleright \sigma}{\Delta; \Gamma \uplus \{x : \sigma\} \triangleright x : \sigma} \qquad (int) \quad \Delta; \Gamma \triangleright \bar{n} : \mathsf{int}$$

$$(pair) \quad \frac{\Delta; \Gamma \triangleright e_1 : \sigma_1 \qquad \Delta; \Gamma \triangleright e_2 : \sigma_2}{\Delta; \Gamma \triangleright \langle e_1, e_2 \rangle^{\sigma_1,\sigma_2} : \sigma_1 \times \sigma_2} \qquad (\pi) \quad \frac{\Delta; \Gamma \triangleright e : \sigma_1 \times \sigma_2}{\Delta; \Gamma \triangleright \pi_i^{\sigma_1,\sigma_2}\ e : \sigma_i} \quad (i = 1, 2)$$

$$(abs) \quad \frac{\Delta \triangleright \sigma_1 \qquad \Delta; \Gamma \uplus \{x : \sigma_1\} \triangleright e : \sigma_2}{\Delta; \Gamma \triangleright \lambda x{:}\sigma_1.\, e : \sigma_1 \to \sigma_2}$$

$$(app) \quad \frac{\Delta; \Gamma \triangleright e_1 : \sigma' \to \sigma \qquad \Delta; \Gamma \triangleright e_2 : \sigma'}{\Delta; \Gamma \triangleright @^{\sigma'}\ e_1\ e_2 : \sigma}$$

$$(tabs) \quad \frac{\Delta \uplus \{t{::}\kappa\}; \Gamma \triangleright e : \sigma}{\Delta; \Gamma \triangleright \Lambda t{::}\kappa.\, e : \forall t{::}\kappa.\sigma} \qquad (tapp) \quad \frac{\begin{array}{c}\Delta \triangleright \mu{::}\kappa \\ \Delta; \Gamma \triangleright e : \forall t{::}\kappa.\sigma\end{array}}{\Delta; \Gamma \triangleright e[\mu] : [\mu/t]\sigma}$$

$$(trec) \quad \frac{\begin{array}{c}\Delta \triangleright \mu :: \Omega \qquad \Delta \uplus \{t{::}\Omega\} \triangleright \sigma \qquad \Delta; \Gamma \triangleright e_{\mathsf{i}} : [\mathsf{Int}/t]\sigma \\ \Delta; \Gamma \triangleright e_{\to} : \forall t_1, t_2{::}\Omega.[t_1/t]\sigma \to [t_2/t]\sigma \to [\to(t_1,t_2)/t]\sigma \\ \Delta; \Gamma \triangleright e_{\times} : \forall t_1, t_2{::}\Omega.[t_1/t]\sigma \to [t_2/t]\sigma \to [\times(t_1,t_2)/t]\sigma\end{array}}{\Delta; \Gamma \triangleright \mathsf{typerec}[t.\sigma](\mu; e_{\mathsf{i}}; e_{\times}; e_{\to}) : [\mu/t]\sigma}$$

Figure 5: Term Formation

$$
\begin{array}{rcl}
|t| & = & t \\
|\mathsf{int}| & = & \mathsf{Int} \\
|\tau_1 \to \tau_2| & = & \to(|\tau_1|, |\tau_2|) \\
|\tau_1 \times \tau_2| & = & \times(|\tau_1|, |\tau_2|) \\
\\
|\tau|_s & = & T(|\tau|) \\
|\forall t.\sigma|_s & = & \forall t{::}\Omega.|\sigma|_s
\end{array}
$$

$$(int) \quad \Delta; \Gamma \rhd \bar{n} : \mathsf{int} \Rightarrow \bar{n}$$

$$(var) \quad \frac{FTV([\tau_n/t_n](\cdots([\tau_1/t_1]\tau)\cdots)) \subseteq \Delta}{\Delta; \Gamma \uplus \{x : \forall t_1, \ldots, t_n.\tau\} \rhd x : [\tau_n/t_n](\cdots([\tau_1/t_1]\tau)\cdots) \Rightarrow x[|\tau_1|]\cdots[|\tau_n|]}$$

$$(pair) \quad \frac{\Delta; \Gamma \rhd e_1 : \tau_1 \Rightarrow e_1' \qquad \Delta; \Gamma \rhd e_2 : \tau_2 \Rightarrow e_2'}{\Delta; \Gamma \rhd \langle e_1, e_2 \rangle : \tau_1 \times \tau_2 \Rightarrow \langle e_1', e_2' \rangle^{|\tau_1|_s, |\tau_2|_s}}$$

$$(abs) \quad \frac{\Delta; \Gamma \uplus \{x : \tau_1\} \rhd e : \tau_2 \Rightarrow e'}{\Delta; \Gamma \rhd \lambda x.\, e : \tau_1 \to \tau_2 \Rightarrow \lambda x : |\tau_1|_s.\, e'}$$

$$(\pi) \quad \frac{\Delta; \Gamma \rhd e : \tau_1 \times \tau_2 \Rightarrow e'}{\Delta; \Gamma \rhd \pi_i\, e : \tau_i \Rightarrow \pi_i^{|\tau_1|_s, |\tau_2|_s}\, e'} \quad (i = 1, 2)$$

$$(app) \quad \frac{\Delta; \Gamma \rhd e_1 : \tau' \to \tau \Rightarrow e_1' \qquad \Delta; \Gamma \rhd e_2 : \tau' \Rightarrow e_2'}{\Delta; \Gamma \rhd e_1\, e_2 : \tau \Rightarrow @^{|\tau'|_s}\, e_1'\, e_2'}$$

$$(let) \quad \frac{\begin{array}{c}\Delta \uplus \{t_1, \ldots, t_n\}; \Gamma \rhd v : \tau' \Rightarrow e_1' \\ \Delta; \Gamma \uplus \{x : \forall t_1, \ldots, t_n.\tau'\} \rhd e : \tau \Rightarrow e_2'\end{array}}{\begin{array}{c}\Delta; \Gamma \rhd \mathsf{let}\, x = v\, \mathsf{in}\, e : \tau \Rightarrow \\ @^{|\forall t_1, \ldots, t_n.\tau'|_s}\, (\lambda x : \forall t_1{::}\Omega, \ldots, t_n{::}\Omega.|\tau'|_s.\, e_2')\, (\Lambda t_1{::}\Omega, \ldots, t_n{::}\Omega.\, e_1')\end{array}}$$

Figure 4: Translation from Mini-ML to λ_i^{ML}

THEOREMS

- Target Language

  2.1 $\Delta; \Gamma \triangleright e : \sigma$ decidable

  2.2 (Type Preservation)

  $\emptyset; \emptyset \triangleright e : \sigma$ & $e \hookrightarrow v$

  $\Rightarrow$ $\emptyset; \emptyset \triangleright v : \sigma$

  2.3 (Termination)

  $\emptyset; \emptyset \triangleright e : \sigma \Rightarrow {}^{\exists v.}\, e \hookrightarrow v$

- Translation

  2.5 $\Delta; \Gamma \triangleright e : \tau \Rightarrow e'$

  $$\Downarrow$$

  $|\Delta|; |\Gamma| \triangleright e' : |\tau|$.

Some Application of typerec/Typerec

Equality

$$Eq :: \Omega \to \Omega$$

$$\begin{aligned}
&Eq[Int] = Int \\
&Eq[Bool] = Bool \\
&Eq[\times(\mu_1,\mu_2)] = \times(Eq[\mu_1], Eq[\mu_2]) \\
&Eq[\to(\mu_1,\mu_2)] = Void \\
&Eq[Void] = Void
\end{aligned}$$

$$\begin{aligned}
&eq[Int] = eqint \\
&eq[Bool] = eqbool \\
&eq[\times(\mu_1,\mu_2)] = \lambda x.\lambda y.\ eq[Eq[\mu_1]]\ (\pi_1 x)(\pi_1 y) \\
&\qquad\qquad \&\&\ eq[Eq[\mu_2]]\ (\pi_2 x)(\pi_2 y) \\
&eq[\to(\mu_1,\mu_2)] = \lambda x:void.\ \lambda y:void.\ false \\
&\qquad eq[Void] = \qquad ''
\end{aligned}$$

Flattening

$$|\tau_1 \times \tau_2| = Prod[|\tau_1|][|\tau_2|]$$

where

$$\begin{aligned}
&Prod[Int][\mu] = \times(Int, \mu) \\
&Prod[\to(\mu_1,\mu_2)][\mu] = \times(\to(\mu_1,\mu_2), \mu) \\
&Prod[\times(\mu_1,\mu_2)][\mu] = \times(\mu_1, Prod[\mu_2][\mu])
\end{aligned}$$

QUESTION

- No recursive function

- Efficiency
  - TIL reported that
    3.3 times faster than
    below 75% memory consumption of SML/NJ
  - Morrisett's thesis reported that
    slow-down in separate compilation
  - Leroy criticized that
    TIL's report was not conclusive