Polar Type Inference with Intersection Types and ω

Sébastien Carlier

EPITA Research and Development Laboratory

ITRS'02, Copenhagen, July 26, 2002

1. Introduction
2. System $\mathbf{P}$
3. System $\mathbf{P}'$
4. System $\mathbf{I}_\omega$
5. Future Work

Introduction

Goal: design a type system for a language featuring

- Static typing
- Total type inference
- Greater expressivity than ML
- Incremental analysis

Example of code we would like to write:

```
map (λo. o!f o!v)
    [ { f = λx. x, v = 1 }
    , { f = λy. length y, v = [1,2,3] } ]
```

System P

Trevor Jim, A Polar Type System, ITRS 2000.

A restriction of the system with $\forall$ and $\wedge$.

$$
\begin{array}{rcll}
\tau \in \mathbb{T}_0 & ::= & \alpha \mid \tau \to \tau & \text{simple types} \\
\pi \in \mathbb{T}^+ & ::= & \tau \mid \nu \to \pi \mid \forall\alpha.\pi & \text{capabilities} \\
\nu \in \mathbb{T}^- & ::= & \tau \mid \pi \to \nu \mid \nu_1 \wedge \nu_2 & \text{requirements}
\end{array}
$$

System **P**:

- is decidable,
- types more terms than ML
- has principal typings (allows for compositional analysis)
- type inference is relatively simple

System P - Typing Inference

Key $\leq$-unification rules:

$$(4) \qquad \pi \leq (\nu_1 \wedge \nu_2) \quad \Rightarrow \quad \{\pi \leq \nu_1, \pi \leq \nu_2\}$$

(great if π is a $\forall$-type!)

$$(5) \qquad \forall\alpha.\pi \leq \nu \quad \Rightarrow \quad \exists\alpha.\{\pi \leq \nu\}$$

if ν is not an $\wedge$-type, and $\alpha \notin \mathtt{FV}(\nu)$

Problem:

In order to satisfy the polarity discipline,

$\forall$-bound variables are instantiated to simple types.

For instance, $(\lambda x.x)(\lambda y.yy)$ is **not typable**

System P'

An attempt to lift the "simple types" restriction.

We would like to replace:

$$(1) \qquad \tau_1 \leq \tau_2 \quad \Rightarrow \quad \{\tau_1 = \tau_2\}$$

with

$$(1^+) \qquad \alpha \leq \nu \quad \Rightarrow \quad \{\alpha \mapsto \nu\}$$

$$(1^-) \qquad \pi \leq \alpha \quad \Rightarrow \quad \{\alpha \mapsto \pi\}$$

For instance, if $M : \pi$, when typing $(\lambda x.x)M$:

$$\{\forall\alpha.\alpha \to \alpha \leq \pi \to \beta\} \quad \Rightarrow \quad \cdots \quad \Rightarrow \quad \{\beta \mapsto \pi\}$$

(typability is undecidable in the resulting system)

System P'

Problem: It only works when principal typings are linear.

$$\{\alpha \to \alpha \to \alpha \le \pi_1 \to \pi_2 \to \nu\} \quad \not\Rightarrow \quad \{\alpha \mapsto \pi_1, \alpha \mapsto \pi_2, \alpha \mapsto \nu\}$$

Only one rule can duplicate free type variables:

$$(4) \qquad \pi \le (\nu_1 \wedge \nu_2) \quad \Rightarrow \quad \{\pi \le \nu_1, \pi \le \nu_2\}$$

Solution: replace (4) with

$$(4') \qquad \pi \le (\nu_1 \wedge \nu_2) \quad \Rightarrow \quad \{\pi \le \nu_1, \pi \le \nu_2\} \qquad \underline{\text{if } \mathtt{FV}(\pi) = \emptyset}$$

System P'

Another problem: single type variables (e.g., in $\lambda x.\lambda y.x$).

Solution: add a type constant ω, with the typing rule:

$$\frac{A \vdash M : \pi}{A \vdash \lambda x.M : \omega \to \pi}\ \mathtt{Abs}_\omega$$

and the rewriting rule:

$$\pi \leq \omega \quad \Rightarrow \quad \{\} \qquad \text{if } \mathtt{FV}(\pi) = \emptyset$$

For instance:

$$\vdash \lambda x.\lambda y.x : \forall\alpha.\alpha \to \omega \to \alpha$$

Then, every type variable has **exactly** one positive and one negative occurrence in a typing inferred by the algorithm.

System P'

Yet another problem:

How can the polarity discipline still be respected?

$$\forall \alpha.\alpha \to \alpha \quad \leq^{+} \quad ? \to \pi$$

We introduce a negation:

$$\forall \alpha.\alpha \to \alpha \quad \leq^{+} \quad \pi^{\perp} \to \pi$$

$\tau^{\perp}$ intuitively means "τ is consumed"

The inference algorithm never yields typings mentioning $(\cdot)^{\perp}$.

Shortcomings of System P'

Inference fails on open terms that are not used exactly once.

$$\{f : \alpha_1 \to \alpha_2\} \vdash (\lambda x.fx) : \alpha_1 \to \alpha_2, \qquad \mathtt{FV}(\alpha_1 \to \alpha_2) \neq \emptyset$$

How can this typing be refined?

$$\{f : \alpha_1^0 \vee \alpha_1^1 \to \alpha_2^0 \wedge \alpha_2^1\} \vdash (\lambda x.fx) : (\alpha_1^0 \to \alpha_2^0) \wedge (\alpha_1^1 \to \alpha_2^1)$$
$$\{f : (\alpha_1^0 \to \alpha_2^0) \wedge (\alpha_1^1 \to \alpha_2^1)\} \vdash (\lambda x.fx) : (\alpha_1^0 \to \alpha_2^0) \wedge (\alpha_1^1 \to \alpha_2^1)$$

We need some marker to note that intersections should appear at $\alpha_1 \to \alpha_2$ and $\alpha_1 \to \alpha_2$ simultaneously.

Solution:

Kfoury & Wells' recent technology of **expansion variables**.

Expansion variables

What are expansion variables ?

$$\frac{\{x_1 : \nu_i, \cdots, x_i : \nu_i\} \vdash M : \pi}{\{x_1 : F\nu_i, \cdots, x_i : F\nu_i\} \vdash M : F\pi}\ \mathtt{F}$$

Expansions:

$$e \in \mathbb{E} ::= \omega \mid \Box \mid e_0 \wedge e_1 \mid F\, e$$

They can be seen as a **substitution based** tool to manipulate derivations.

$\wedge$-expansion

$$\dfrac{\dfrac{\vdots}{\{x_1 : \nu_i, \cdots, x_i : \nu_i\} \vdash M : \pi}\, r}{\{x_1 : F\nu_i, \cdots, x_i : F\nu_i\} \vdash M : F\pi}\, \mathtt{F}$$

$$\{F \mapsto \Box \wedge \Box\}$$

$$\Downarrow$$

$$\dfrac{\dfrac{\vdots}{\{x_1 : \nu_i^0, \cdots, x_i : \nu_i^0\} \vdash M : \pi^0}\, r \qquad \dfrac{\vdots}{\{x_1 : \nu_i^1, \cdots, x_i : \nu_i^1\} \vdash M : \pi^1}\, r}{\{x_1 : \nu_i^0 \wedge \nu_0^1, \cdots, x_i : \nu_i^0 \wedge \nu_i^1\} \vdash M : \pi^0 \wedge \pi^1}\, \wedge$$

ω-expansion

$$\dfrac{\dfrac{\{x : {}^{-}\alpha_1\} \vdash x : {}^{+}\alpha_1}{\{x : {}^{-}\alpha_1\} \vdash (\lambda y.x) : \omega \to {}^{+}\alpha_1}\ \texttt{Abs} \qquad \dfrac{\{z : {}^{-}\alpha_2\} \vdash z : {}^{+}\alpha_2}{\{z : F_1{}^{-}\alpha_2\} \vdash z : F_1{}^{+}\alpha_2}\ \texttt{F}}{? \vdash ((\lambda y.x)z) : ?}\ \texttt{App}$$

$$\Downarrow$$

$$\dfrac{\dfrac{\{x : {}^{-}\alpha_1\} \vdash x : {}^{+}\alpha_1}{\{x : {}^{-}\alpha_1\} \vdash (\lambda y.x) : \omega \to {}^{+}\alpha_1}\ \texttt{Abs} \qquad \dfrac{\{{}^{-}\alpha_2\} \vdash z : {}^{+}\alpha_2}{\{\} \vdash z : \omega}\ \omega}{\{x : {}^{-}\alpha_1\} \vdash ((\lambda y.x)z) : {}^{+}\alpha_1}\ \texttt{App}$$

To be compared with: $\{x : {}^{-}\alpha_1, y : {}^{-}\alpha_2\} \vdash ((\lambda y.x)z) : {}^{+}\alpha_1$

Types

Positive types: capabilities/outputs

$$\pi \in \mathbb{P} ::= \omega \mid \bar{\pi} \mid (\pi_0 \wedge \pi_1) \mid (F\ \pi)$$
$$\bar{\pi} \in \bar{\mathbb{P}} ::= {}^{+}\alpha \mid (\nu \to \bar{\pi})$$

Negative types: requirements/inputs

$$\nu \in \mathbb{N} ::= \omega \mid \bar{\nu} \mid (\nu_0 \wedge \nu_1) \mid (F\ \nu)$$
$$\bar{\nu} \in \bar{\mathbb{N}} ::= {}^{-}\alpha \mid (\pi \to \bar{\nu})$$

Types:

$$\tau \in \mathbb{T} ::= \nu \mid \pi$$

Typing rules

$$\frac{\nu \perp \pi}{\{x : \nu\} \vdash x : \pi}\ \texttt{VAR}$$

$$\frac{A \vdash M : \pi}{F\,A \vdash M : F\pi}\ F$$

$$\frac{A \vdash M : \pi}{A \setminus x \vdash \lambda x.M : A(x) \to \pi}\ \texttt{ABS}$$

$$\frac{A \vdash M : \pi_0 \qquad B \vdash M : \pi_1}{A \wedge B \vdash M : \pi_0 \wedge \pi_1}\ \wedge$$

$$\frac{A \vdash M : \pi_1 \qquad B \vdash N : \pi_2 \qquad (\pi_2 \to {\pi_3}^{\perp}) \perp \pi_1}{A \wedge B \vdash M\,N : \pi_3}\ \texttt{APP}$$

$$\frac{A \vdash M : \pi}{\{\} \vdash M : \omega}\ \omega$$

Typing inference algorithm

$$
\begin{aligned}
&\mathtt{PP}(x) = \\
&\quad \text{let } \alpha \text{ be fresh} \\
&\quad \text{in } \langle \{x : {}^{-}\alpha\}, {}^{+}\alpha \rangle \\
&\mathtt{PP}(\lambda x.M) = \\
&\quad \text{let } \langle A, \pi \rangle = \mathtt{PP}(M) \\
&\quad \text{in } \langle A \setminus x, A(x) \to \pi \rangle \\
&\mathtt{PP}(MN) = \\
&\quad \text{let } \langle A, \pi_1 \rangle = \mathtt{PP}(M) \\
&\quad\quad \langle B, \pi_2 \rangle = \mathtt{PP}(N) \\
&\quad\quad \alpha \text{ and } F \text{ be fresh} \\
&\quad\quad \mathbf{S} = \mathtt{MGS}((F\pi_2 \to {}^{-}\alpha) \perp \pi_1) \\
&\quad \text{in } \langle \bar{\mathbf{S}}(\Box, A \wedge FB), \bar{\mathbf{S}}(\Box, {}^{+}\alpha) \rangle
\end{aligned}
$$

Lifted substitution

$$
\begin{aligned}
\bar{\mathbf{S}}(\Box, \omega) &= \omega \\
\bar{\mathbf{S}}(\Box, {}^{-}\alpha) &= \mathbf{S}({}^{-}\alpha) \\
\bar{\mathbf{S}}(\Box, {}^{+}\alpha) &= \mathbf{S}({}^{+}\alpha) \\
\bar{\mathbf{S}}(\Box, F\tau) &= \bar{\mathbf{S}}(\mathbf{S}(F), \tau) \\
\bar{\mathbf{S}}(\Box, \tau \to \tau') &= \bar{\mathbf{S}}(\Box, \tau) \to \bar{\mathbf{S}}(\Box, \tau') \\
\bar{\mathbf{S}}(\Box, \tau_0 \wedge \tau_1) &= \bar{\mathbf{S}}(\Box, \tau_0) \wedge \bar{\mathbf{S}}(\Box, \tau_1)
\end{aligned}
$$

$$
\begin{aligned}
\bar{\mathbf{S}}(\Box, \Box) &= \Box \\
\bar{\mathbf{S}}(\Box, Fe) &= \bar{\mathbf{S}}(\mathbf{S}(F), e) \\
\bar{\mathbf{S}}(\Box, e_0 \wedge e_1) &= \bar{\mathbf{S}}(\Box, e_0) \wedge \bar{\mathbf{S}}(\Box, e_1)
\end{aligned}
$$

$$
\begin{aligned}
\bar{\mathbf{S}}(\omega, \tau) &= \omega \\
\bar{\mathbf{S}}(e_0 \wedge e_1, \tau) &= \bar{\mathbf{S}}(e_0, \langle\tau\rangle^0) \wedge \bar{\mathbf{S}}(e_1, \langle\tau\rangle^1) \\
\bar{\mathbf{S}}(F\, e, \tau) &= F\, \bar{\mathbf{S}}(e, \tau)
\end{aligned}
$$

$$
\begin{aligned}
\bar{\mathbf{S}}(\omega, e) &= \omega \\
\bar{\mathbf{S}}(e_0 \wedge e_1, e') &= \bar{\mathbf{S}}(e_0, \langle e'\rangle^0) \wedge \bar{\mathbf{S}}(e_1, \langle e'\rangle^1) \\
\bar{\mathbf{S}}(F\, e, e') &= F\, \bar{\mathbf{S}}(e, e')
\end{aligned}
$$

Computing the most general solution

$$
\begin{aligned}
\mathtt{MGS}(\bar{\nu} \perp {}^{+}\alpha) &= \{\alpha \mapsto \langle \bar{\nu}, \bar{\nu}^{\perp} \rangle\} \\
\mathtt{MGS}({}^{-}\alpha \perp \bar{\pi}) &= \{\alpha \mapsto \langle \bar{\pi}^{\perp}, \bar{\pi} \rangle\} \\
\mathtt{MGS}(\omega \perp F\pi) &= \{F \mapsto \omega\} \\
\mathtt{MGS}(\bar{\nu} \perp F\pi) &= \{F \mapsto \Box\} \; ; \; (\bar{\nu} \perp \pi) \\
\mathtt{MGS}((\nu_0 \wedge \nu_1) \perp F\pi) &= \{F \mapsto F^0\Box \wedge F^1\Box\} \; ; \\
& \qquad (\nu_0 \perp \langle F\pi \rangle^0) \; ; \; (\nu_1 \perp \langle F\pi \rangle^1) \\
\mathtt{MGS}(F\nu \perp F\pi) &= \mathtt{MGS}(\nu \perp \pi) \\
\mathtt{MGS}(G\nu \perp F\pi) &= \{F \mapsto GH\Box\} \; ; \; (\nu \perp H\pi) \qquad \text{with } H \text{ fresh} \\
\mathtt{MGS}((\pi_2 \to \nu_2) \perp (\nu_1 \to \pi_1)) &= \mathtt{MGS}(\nu_1 \perp \pi_2) \; ; \; (\nu_2 \perp \pi_1)
\end{aligned}
$$

$$
\mathbf{S} \; ; \; (\nu \perp \pi) = \mathtt{MGS}(\bar{\mathbf{S}}(\Box, \nu) \perp \bar{\mathbf{S}}(\Box, \pi)) \diamond \mathbf{S}
$$

System $\mathbf{I}_\omega$

Two ways to look at System $\mathbf{I}_\omega$:

- System $\mathbf{I} + \omega +$ polarity discipline
- System $\mathbf{P}' +$ expansion variables $-$ quantifiers

- The power of $\forall$ is subsumed by $\wedge$ with expansion variables. (typings may be refined an arbitrary finite number of times)

- The polarity discipline provided good sanity checks in our implementations, thanks to the stronger invariants.

- It may help getting an intuition by considering inference in System $\mathbf{I}$ as propagation of capabilities and requirements.

Future Work

- Prove general results on System $\mathbf{I}_\omega$
- Allow for cruder analysis ($\forall, \exists$ help)
- Find practical decidable restrictions (beside rank)
- Fit union types into the framework of expansion variables
- Further exploit the "proof manipulation" idea of expansion variables