Type reconstruction example

Andrew Appel
Princeton University

October 20, 2000

Claim. The following program can be well typed.

```
fun g(f) is fun h(x) is f (f x) end end (fun j(x) is x end)
```

Definitions.

$$
\begin{array}{rcl}
H & \equiv & \texttt{fun}\, h\, (x\!:\!\alpha_1)\!:\!\alpha_2\, \texttt{is}\, f(f(x))\, \texttt{end} \\
G & \equiv & \texttt{fun}\, g\, (f\!:\!\alpha_3)\!:\!\alpha_4\, \texttt{is}\, H\, \texttt{end} \\
J & \equiv & \texttt{fun}\, j\, (x\!:\!\alpha_5)\!:\!\alpha_6\, \texttt{is}\, x\, \texttt{end}
\end{array}
$$

Theorem. $\vdash (GJ)^- \leadsto GJ : \tau$, for some τ.

Derivation using rules 11.19–11.26.

$$
\dfrac{\dfrac{\dfrac{\dfrac{}{g, f:\alpha_3, h, x \vdash f \leadsto f : \alpha_3[\top]} \quad \dfrac{\dfrac{}{g, f:\alpha_3, h, x \vdash f \leadsto f : \alpha_3[\top]} \quad \dfrac{}{g, f, h, x : \alpha_1 \vdash x \leadsto x : \alpha_1[\top]}}{g, f:\alpha_3, h, x:\alpha_1 \vdash fx \leadsto fx : \alpha_8[\phi_3]}}{g:\alpha_3 \to \alpha_4, f:\alpha_3, h:\alpha_1 \to \alpha_2, x:\alpha_1 \vdash f(fx) \leadsto f(fx) : \alpha_9[\phi_4]}}{g:\alpha_3 \to \alpha_4, f:\alpha_3 \vdash H^- \leadsto H : \alpha_1 \to \alpha_2[\phi_5]}}{\vdash G^- \leadsto G : \alpha_3 \to \alpha_4[\phi_6]}
$$

$$
\dfrac{\dfrac{\vdots}{\vdash G^- \leadsto G : \alpha_3 \to \alpha_4[\phi_6]} \quad \dfrac{\dfrac{}{j:\alpha_5 \to \alpha_6, x:\alpha_5 \vdash x \leadsto x : \alpha_5[\top]}}{\vdash J^- \leadsto J : \alpha_5 \to \alpha_6[\phi_2]}}{\vdash (GJ)^- \leadsto GJ : \alpha_7[\phi_7]}
$$

where

$$
\begin{array}{rcl}
\phi_2 & = & \alpha_5 = \alpha_6 \\
\phi_3 & = & \alpha_3 = \alpha_1 \to \alpha_8 \\
\phi_4 & = & \phi_3 \wedge \alpha_3 = \alpha_8 \to \alpha_9 \\
\phi_5 & = & \phi_4 \wedge \alpha_9 = \alpha_2 \\
\phi_6 & = & \phi_5 \wedge \alpha_1 \to \alpha_2 = \alpha_4 \\
\phi_7 & = & \phi_6 \wedge \phi_2 \wedge \alpha_3 \to \alpha_4 = (\alpha_5 \to \alpha_6) \to \alpha_7
\end{array}
$$

Solution of the constraints using rules 11.27–11.28

$$
\begin{array}{l}
(\{\},\quad \alpha_3 = \alpha_1 \to \alpha_8 \wedge \alpha_3 = \alpha_8 \to \alpha_9 \wedge \alpha_9 = \alpha_2 \wedge \alpha_1 \to \alpha_2 = \alpha_4 \wedge \alpha_5 = \alpha_6 \\
\qquad\qquad \wedge \alpha_3 \to \alpha_4 = (\alpha_5 \to \alpha_6) \to \alpha_7) \\
(\{\alpha_3 = \alpha_1 \to \alpha_8\},\quad \alpha_1 \to \alpha_8 = \alpha_8 \to \alpha_9 \wedge \alpha_9 = \alpha_2 \wedge \alpha_1 \to \alpha_2 = \alpha_4 \wedge \alpha_5 = \alpha_6 \\
\qquad\qquad \wedge(\alpha_1 \to \alpha_8) \to \alpha_4 = (\alpha_5 \to \alpha_6) \to \alpha_7) \\
(\{\alpha_3 = \alpha_1 \to \alpha_8\},\quad \alpha_1 = \alpha_8 \wedge \alpha_8 = \alpha_9 \wedge \alpha_9 = \alpha_2 \wedge \alpha_1 \to \alpha_2 = \alpha_4 \wedge \alpha_5 = \alpha_6 \\
\qquad\qquad \wedge(\alpha_1 \to \alpha_8) \to \alpha_4 = (\alpha_5 \to \alpha_6) \to \alpha_7) \\
(\{\alpha_1 = \alpha_8, \alpha_3 = \alpha_8 \to \alpha_8\},\quad \alpha_8 = \alpha_9 \wedge \alpha_9 = \alpha_2 \wedge \alpha_8 \to \alpha_2 = \alpha_4 \wedge \alpha_5 = \alpha_6 \\
\qquad\qquad \wedge(\alpha_8 \to \alpha_8) \to \alpha_4 = (\alpha_5 \to \alpha_6) \to \alpha_7) \\
(\{\alpha_8 = \alpha_9, \alpha_1 = \alpha_9, \alpha_3 = \alpha_9 \to \alpha_9\},\quad \alpha_9 = \alpha_2 \wedge \alpha_9 \to \alpha_2 = \alpha_4 \wedge \alpha_5 = \alpha_6 \\
\qquad\qquad \wedge(\alpha_9 \to \alpha_9) \to \alpha_4 = (\alpha_5 \to \alpha_6) \to \alpha_7) \\
(\{\alpha_9 = \alpha_2, \alpha_8 = \alpha_2, \alpha_1 = \alpha_2, \alpha_3 = \alpha_2 \to \alpha_2\},\quad \alpha_2 \to \alpha_2 = \alpha_4 \wedge \alpha_5 = \alpha_6 \\
\qquad\qquad \wedge(\alpha_2 \to \alpha_2) \to \alpha_4 = (\alpha_5 \to \alpha_6) \to \alpha_7) \\
(\{\alpha_4 = \alpha_2 \to \alpha_2, \alpha_9 = \alpha_2, \alpha_8 = \alpha_2, \alpha_1 = \alpha_2, \alpha_3 = \alpha_2 \to \alpha_2\},\quad \alpha_5 = \alpha_6 \\
\qquad\qquad \wedge(\alpha_2 \to \alpha_2) \to (\alpha_2 \to \alpha_2) = (\alpha_5 \to \alpha_6) \to \alpha_7) \\
(\{\alpha_5 = \alpha_6, \alpha_4 = \alpha_2 \to \alpha_2, \alpha_9 = \alpha_2, \alpha_8 = \alpha_2, \alpha_1 = \alpha_2, \alpha_3 = \alpha_2 \to \alpha_2\}, \\
\qquad\qquad (\alpha_2 \to \alpha_2) \to (\alpha_2 \to \alpha_2) = (\alpha_6 \to \alpha_6) \to \alpha_7) \\
(\{\alpha_5 = \alpha_6, \alpha_4 = \alpha_2 \to \alpha_2, \alpha_9 = \alpha_2, \alpha_8 = \alpha_2, \alpha_1 = \alpha_2, \alpha_3 = \alpha_2 \to \alpha_2\}, \\
\qquad\qquad (\alpha_2 \to \alpha_2) = (\alpha_6 \to \alpha_6) \wedge (\alpha_2 \to \alpha_2) = \alpha_7) \\
(\{\alpha_5 = \alpha_6, \alpha_4 = \alpha_2 \to \alpha_2, \alpha_9 = \alpha_2, \alpha_8 = \alpha_2, \alpha_1 = \alpha_2, \alpha_3 = \alpha_2 \to \alpha_2\}, \\
\qquad\qquad \alpha_2 = \alpha_6 \wedge \alpha_2 = \alpha_6 \wedge (\alpha_2 \to \alpha_2) = \alpha_7) \\
(\{\alpha_2 = \alpha_6, \alpha_5 = \alpha_6, \alpha_4 = \alpha_6 \to \alpha_6, \alpha_9 = \alpha_6, \alpha_8 = \alpha_6, \alpha_1 = \alpha_6, \alpha_3 = \alpha_6 \to \alpha_6\}, \\
\qquad\qquad \alpha_6 = \alpha_6 \wedge (\alpha_6 \to \alpha_6) = \alpha_7) \\
(\{\alpha_2 = \alpha_6, \alpha_5 = \alpha_6, \alpha_4 = \alpha_6 \to \alpha_6, \alpha_9 = \alpha_6, \alpha_8 = \alpha_6, \alpha_1 = \alpha_6, \alpha_3 = \alpha_6 \to \alpha_6\},\quad (\alpha_6 \to \alpha_6) = \alpha_7) \\
(\{\alpha_7 = \alpha_6 \to \alpha_6, \alpha_2 = \alpha_6, \alpha_5 = \alpha_6, \alpha_4 = \alpha_6 \to \alpha_6, \alpha_9 = \alpha_6, \alpha_8 = \alpha_6, \alpha_1 = \alpha_6, \alpha_3 = \alpha_6 \to \alpha_6\},\quad \top)
\end{array}
$$

If we want to, we can apply this substition to the type-inference proof, yielding a proof using rules 4.1–4.12:

$$
\dfrac{
  \dfrac{
    \dfrac{}{g, f : (\alpha_6 \to \alpha_6), h, x \vdash f : (\alpha_6 \to \alpha_6)}
    \qquad
    \dfrac{
      \dfrac{}{g, f : (\alpha_6 \to \alpha_6), h, x \vdash f : (\alpha_6 \to \alpha_6)}
      \quad
      \dfrac{}{g, f, h, x : \alpha_6 \vdash x : \alpha_6}
    }{g, f : (\alpha_6 \to \alpha_6), h, x : \alpha_6 \vdash fx : \alpha_6}
  }{
    \dfrac{g : (\alpha_6 \to \alpha_6) \to (\alpha_6 \to \alpha_6), f : (\alpha_6 \to \alpha_6), h : \alpha_6 \to \alpha_6, x : \alpha_6 \vdash f(fx) : \alpha_9}{g : (\alpha_6 \to \alpha_6) \to (\alpha_6 \to \alpha_6), f : (\alpha_6 \to \alpha_6) \vdash H : \alpha_6 \to \alpha_6}
  }
}{\vdash G : (\alpha_6 \to \alpha_6) \to (\alpha_6 \to \alpha_6)}
$$

$$
\dfrac{
  \begin{array}{c}\vdots \\ \vdash G : (\alpha_6 \to \alpha_6) \to (\alpha_6 \to \alpha_6)\end{array}
  \qquad
  \dfrac{\dfrac{}{j : \alpha_6 \to \alpha_6, x : \alpha_6 \vdash x : \alpha_6}}{\vdash J : \alpha_6 \to \alpha_6}
}{\vdash GJ : \alpha_6 \to \alpha_6}
$$

But this isn't a proof, because α_6 isn't a type. What we have discovered is that this expression is polymorphic; it can have many types. Since MinML is not a polymorphic language, we can make this into a real proof of static semantics by choosing some arbitrary type, for example `int`:

$$
\begin{array}{c}\vdots \\ \vdash GJ : \texttt{int} \to \texttt{int}\end{array}
$$