Hinting
A nice presentation of algorithmic typing

Wojciech Kołowski

STLC with Hints

STLC with Hints is a flavour of STLC inspired by Bidirectional STLC. The main insight behind it is that in Bidirectional STLC, we have a hard time deciding whether a rule should be in checking mode or in inference mode, so why not both? This way, we would have some input type that guides us, but also produce an output type, which is in some sense "better". This is a bit silly if we already have the correct type as input, but we can make this idea work by introducing hints, which are types with holes, and insisting that the input is not a type, but merely a hint.

Hints

$$H ::= \mathbf{?} \mid H_1 \to H_2 \mid H_1 \times H_2 \mid H_1 + H_2 \mid \mathbf{1} \mid \mathbf{0}$$

Intuitively, hints are partial types. They are built like types, except that there's one additional constructor, **?**, which can be read as "hole" or "unknown".

We use the letter H for hints. When we use letters like A, B, C which usually stand in for types, it means that the hint **is** a type, i.e. it doesn't contain any **?**s.

Order on hints

$$\frac{}{\mathbf{?} \sqsubseteq H}$$

$$\frac{H_1 \sqsubseteq H_1' \quad H_2 \sqsubseteq H_2'}{H_1 \to H_2 \sqsubseteq H_1' \to H_2'}$$

$$\frac{H_1 \sqsubseteq H_1' \quad H_2 \sqsubseteq H_2'}{H_1 \times H_2 \sqsubseteq H_1' \times H_2'}$$

$$\frac{H_1 \sqsubseteq H_1' \quad H_2 \sqsubseteq H_2'}{H_1 + H_2 \sqsubseteq H_1' + H_2'}$$

$$\frac{}{\mathbf{1} \sqsubseteq \mathbf{1}} \qquad \frac{}{\mathbf{0} \sqsubseteq \mathbf{0}}$$

Order on hints – intuition

The order can be intuitively interpreted as information increase: $H_1 \sqsubseteq H_2$ means that hint H_2 is more informative than H_1, but in a compatible way. In other words, H_1 and H_2 have the same structure, but some **?**s from H_1 were possibly refined to something more informative in H_2.

Combining hints

$$
\begin{aligned}
\mathbf{?} \sqcup H &= H \\
H \sqcup \mathbf{?} &= H \\
(H_1 \to H_2) \sqcup (H_1' \to H_2') &= (H_1 \sqcup H_1') \to (H_2 \sqcup H_2') \\
(H_1 \times H_2) \sqcup (H_1' \times H_2') &= (H_1 \sqcup H_1') \times (H_2 \sqcup H_2') \\
(H_1 + H_2) \sqcup (H_1' + H_2') &= (H_1 \sqcup H_1') + (H_2 \sqcup H_2') \\
\mathbf{1} \sqcup \mathbf{1} &= \mathbf{1} \\
\mathbf{0} \sqcup \mathbf{0} &= \mathbf{0}
\end{aligned}
$$

The order on hints induces a partial operation $\sqcup$, which computes the least upper bound of two hints when it exists. Intuitively, $\sqcup$ combines two hints which share the same structure, filling the $\mathbf{?}$s in the leaves with something more informative coming from the other argument. For hints with incompatible structure the result is undefined.

Combining hints – properties

If all relevant results are defined, then:

- $(H_1 \sqcup H_2) \sqcup H_3 = H_1 \sqcup (H_2 \sqcup H_3)$
- $H_1 \sqcup H_2 = H_2 \sqcup H_1$
- $\mathbf{?} \sqcup H = H = H \sqcup \mathbf{?}$
- $H \sqcup H = H$

If $\sqcup$ were not partial, $(H, \sqcup, \mathbf{?})$ would be a commutative idempotent monoid. But since it is partial, meh...

Hints for term constructors

$$\begin{aligned}
\mathtt{hint}(\lambda x.\, e) &= \mathbf{?} \to \mathbf{?} \\
\mathtt{hint}((e_1, e_2)) &= \mathbf{?} \times \mathbf{?} \\
\mathtt{hint}(\mathtt{inl}\ e) &= \mathbf{?} + \mathbf{?} \\
\mathtt{hint}(\mathtt{inr}\ e) &= \mathbf{?} + \mathbf{?} \\
\mathtt{hint}(\mathtt{unit}) &= \mathbf{1}
\end{aligned}$$

Terms

Terms:

$e ::=$

$x \mid (e : H) \mid$

$\lambda x.\, e \mid e_1\ e_2 \mid$

$(e_1, e_2) \mid \mathtt{outl}\ e \mid \mathtt{outr}\ e \mid$

$\mathtt{inl}\ e \mid \mathtt{inr}\ e \mid \mathtt{case}\ e\ \mathtt{of}\ (e_1, e_2) \mid$

$\mathtt{unit} \mid \mathtt{exfalso}\ e$

Note: red color marks differences from Bidirectional STLC.

Judgements:

$\Gamma \vdash e \Leftarrow H \Rightarrow A$ – in context Γ, term e checks with hint H and infers type A

Declarative typing – differences

$$\frac{\Gamma \vdash e : A \quad H \sqsubseteq A}{\Gamma \vdash (e : H) : A}\textsc{Annot}$$

Hinting – basic rules

$$\frac{(x : A) \in \Gamma \quad H \sqsubseteq A}{\Gamma \vdash x \Leftarrow H \Rightarrow A}\textsc{Var}$$

$$\frac{\Gamma \vdash e \Leftarrow H_1 \sqcup H_2 \Rightarrow A}{\Gamma \vdash (e : H_1) \Leftarrow H_2 \Rightarrow A}\textsc{Annot}$$

$$\frac{\Gamma \vdash e \Leftarrow \mathtt{hint}(e) \Rightarrow A \quad e \ \mathtt{constructor}}{\Gamma \vdash e \Leftarrow \mathbf{?} \Rightarrow A}\textsc{Hole}$$

Note that the rule Hole can only be applied once, because $\mathtt{hint}(e)$ can never be $\mathbf{?}$. After applying Hole, the only applicable rules are the type-directed ones.

Hinting – type-directed rules

$$\frac{\Gamma, x : A \vdash e \Leftarrow H \Rightarrow B}{\Gamma \vdash \lambda x.\, e \Leftarrow A \to H \Rightarrow A \to B}$$

$$\frac{\Gamma \vdash f \Leftarrow \mathbf{?} \to H \Rightarrow A \to B \quad \Gamma \vdash a \Leftarrow A \Rightarrow A}{\Gamma \vdash f\ a \Leftarrow H \Rightarrow B}$$

$$\frac{\Gamma \vdash a \Leftarrow H_A \Rightarrow A \quad \Gamma \vdash b \Leftarrow H_B \Rightarrow B}{\Gamma \vdash (a, b) \Leftarrow H_A \times H_B \Rightarrow A \times B}$$

$$\frac{\Gamma \vdash e \Leftarrow H \times \mathbf{?} \Rightarrow A \times B}{\Gamma \vdash \mathtt{outl}\ e \Leftarrow H \Rightarrow A} \qquad \frac{\Gamma \vdash e \Leftarrow \mathbf{?} \times H \Rightarrow A \times B}{\Gamma \vdash \mathtt{outr}\ e \Leftarrow H \Rightarrow B}$$

Hinting – type-directed rules

$$\frac{\Gamma \vdash e \Leftarrow H \Rightarrow A}{\Gamma \vdash \texttt{inl}\ e \Leftarrow H + B \Rightarrow A + B}$$

$$\frac{\Gamma \vdash e \Leftarrow H \Rightarrow B}{\Gamma \vdash \texttt{inr}\ e \Leftarrow A + H \Rightarrow A + B}$$

$$\frac{\Gamma \vdash e \Leftarrow \mathbf{?} + \mathbf{?} \Rightarrow A + B \qquad \begin{array}{c}\Gamma \vdash f \Leftarrow A \to H \Rightarrow A \to C \\ \Gamma \vdash g \Leftarrow B \to C \Rightarrow B \to C\end{array}}{\Gamma \vdash \texttt{case}\ e\ \texttt{of}\ (f, g) \Leftarrow H \Rightarrow C}$$

$$\frac{}{\Gamma \vdash \texttt{unit} \Leftarrow \mathbf{1} \Rightarrow \mathbf{1}} \qquad \frac{\Gamma \vdash e \Leftarrow \mathbf{0} \Rightarrow \mathbf{0}}{\Gamma \vdash \texttt{exfalso}\ e \Leftarrow A \Rightarrow A}$$

Hinting – alternative rules

$$\frac{\Gamma \vdash a \Leftarrow \mathbf{?} \Rightarrow H_A \quad \Gamma \vdash f \Leftarrow H_A \to H_B \Rightarrow A \to B}{\Gamma \vdash f\ a \Leftarrow H_B \Rightarrow B}\ \textsc{AltApp}$$

$$\frac{\begin{array}{c}\Gamma \vdash f \Leftarrow \mathbf{?} \to H_C \Rightarrow H_A \to H'_C \\ \Gamma \vdash g \Leftarrow \mathbf{?} \to H'_C \Rightarrow H_B \to C\end{array} \quad \Gamma \vdash e \Leftarrow H_A + H_B \Rightarrow A + B}{\Gamma \vdash \mathtt{case}\ e\ \mathtt{of}\ (f, g) \Leftarrow H_C \Rightarrow C}\ \textsc{AltCase}$$

We could have made some different choices. For application, we could try to infer the argument type first and then feed it to the function as a hint. For case, we could try to infer domains of the branches first, then feed these as hints when checking the discriminee.

Notations and derived terms

We can introduce some handy notations:

- $\Gamma \vdash e \Leftarrow A :\equiv \Gamma \vdash e \Leftarrow A \Rightarrow A$
- $\Gamma \vdash e \Rightarrow A :\equiv \Gamma \vdash e \Leftarrow \mathbf{?} \Rightarrow A$

We can embed Intrinsic STLC terms:

- $\lambda x : A.\, e :\equiv (\lambda x.\, e : A \to \mathbf{?})$
- $\mathtt{inl}_B\ e :\equiv (\mathtt{inl}\ e : \mathbf{?} + B)$
- $\mathtt{inr}_A\ e :\equiv (\mathtt{inr}\ e : A + \mathbf{?})$
- $\mathtt{exfalso}_A\ e :\equiv (\mathtt{exfalso}\ e : A)$

We can also embed Dual Intrinsic STLC terms:

- $\mathtt{app}_A\ f\ \ a :\equiv (f : A \to \mathbf{?})\ a$
- $\mathtt{outl}_B\ e :\equiv \mathtt{outl}\ (e : \mathbf{?} \times B)$
- $\mathtt{outr}_A\ e :\equiv \mathtt{outr}\ (e : A \times \mathbf{?})$
- $\mathtt{case}_{A,B}\ e\ \mathtt{of}\ (f, g) :\equiv \mathtt{case}\ (e : A + B)\ \mathtt{of}\ (f, g)$

Rules for derived terms

$$\frac{\Gamma, x : A \vdash e \Rightarrow B}{\Gamma \vdash \lambda x : A.\, e \Rightarrow A \to B} \qquad \frac{\Gamma \vdash f \Leftarrow A \to B \quad \Gamma \vdash a \Leftarrow A}{\Gamma \vdash \mathtt{app}_A\ f\ a \Leftarrow B}$$

$$\frac{\Gamma \vdash e \Leftarrow A \times B}{\Gamma \vdash \mathtt{outl}_B\ e \Leftarrow A} \qquad \frac{\Gamma \vdash e \Leftarrow A \times B}{\Gamma \vdash \mathtt{outr}_A\ e \Leftarrow B}$$

$$\frac{\Gamma \vdash e \Rightarrow A}{\Gamma \vdash \mathtt{inl}_B\ e \Rightarrow A + B} \qquad \frac{\Gamma \vdash e \Rightarrow B}{\Gamma \vdash \mathtt{inr}_A\ e \Rightarrow A + B}$$

$$\frac{\Gamma \vdash e \Leftarrow A + B \quad \Gamma \vdash f \Leftarrow A \to C \quad \Gamma \vdash g \Leftarrow B \to C}{\Gamma \vdash \mathtt{case}_{A,B}\ e\ \mathtt{of}\ (f, g) \Leftarrow C}$$

$$\frac{\Gamma \vdash e \Rightarrow \mathbf{0}}{\Gamma \vdash \mathtt{exfalso}_A\ e \Rightarrow A}$$

Information order on terms

The order on hints induces an order on terms: it is the smallest relation which preserves term constructors and subsumes hint ordering on annotated terms.

$$\frac{e_1 \sqsubseteq e_2 \quad H_1 \sqsubseteq H_2}{(e_1 : H_1) \sqsubseteq (e_2 : H_2)}$$

Information order on terms

$$\frac{}{x \sqsubseteq x} \qquad \frac{e_1 \sqsubseteq e_2}{\lambda x.\, e_1 \sqsubseteq \lambda x.\, e_2} \qquad \frac{f_1 \sqsubseteq f_2 \quad a_1 \sqsubseteq a_2}{f_1 \; a_1 \sqsubseteq f_2 \; a_2}$$

$$\frac{a_1 \sqsubseteq a_2 \quad b_1 \sqsubseteq b_2}{(a_1, b_1) \sqsubseteq (a_2, b_2)} \qquad \frac{e_1 \sqsubseteq e_2}{\mathtt{outl}\; e_1 \sqsubseteq \mathtt{outl}\; e_2} \qquad \frac{e_1 \sqsubseteq e_2}{\mathtt{outl}\; e_1 \sqsubseteq \mathtt{outr}\; e_2}$$

$$\frac{e_1 \sqsubseteq e_2}{\mathtt{inl}\; e_1 \sqsubseteq \mathtt{inl}\; e_2} \qquad \frac{e_1 \sqsubseteq e_2}{\mathtt{inr}\; e_1 \sqsubseteq \mathtt{inr}\; e_2}$$

$$\frac{e_1 \sqsubseteq e_2 \quad f_1 \sqsubseteq f_2 \quad g_1 \sqsubseteq g_2}{\mathsf{case}\; e_1\; \mathsf{of}\; (f_1, g_1) \sqsubseteq \mathsf{case}\; e_2\; \mathsf{of}\; (f_2, g_2)}$$

$$\frac{}{\mathtt{unit} \sqsubseteq \mathtt{unit}} \qquad \frac{e_1 \sqsubseteq e_2}{\mathtt{exfalso}\; e_1 \sqsubseteq \mathtt{exfalso}\; e_2}$$

Metatheory 1

If $\Gamma \vdash e \Leftarrow H \Rightarrow A$ then:

- (Soundness) $\Gamma \vdash e : A$ (proof: induction)
- (Compatibility) $H \sqsubseteq A$ (proof: induction)
- (Squeeze) If $H \sqsubseteq H'$ and $H' \sqsubseteq A$, then $\Gamma \vdash e \Leftarrow H' \Rightarrow A$ (proof: induction)
- (Upper bound) If $\Gamma \vdash e \Leftarrow H' \Rightarrow A$, then $\Gamma \vdash e \Leftarrow H \sqcup H' \Rightarrow A$ (proof: follows from the above)
- (Decidability) For Γ, e, H it is decidable whether there exists A such that $\Gamma \vdash e \Leftarrow H \Rightarrow A$ (proof: the rules are literally the algorithm)
- (Minimality) There exists $H' \sqsubseteq H$ such that $\Gamma \vdash e \Leftarrow H' \Rightarrow A$ and for all $H'' \sqsubset H'$ it is not the case that $\Gamma \vdash e \Leftarrow H'' \Rightarrow A$.

Metatheory 2

If $\Gamma \vdash e : A$, then:

- (Annotability) There exists e' such that $e \sqsubseteq e'$ and $\Gamma \vdash e' \Leftarrow A \Rightarrow A$
- (Minimal annotability) There exists e' such that $e \sqsubseteq e'$ and $\Gamma \vdash e' \Leftarrow A \Rightarrow A$ and for all e'' such that $e \sqsubseteq e''$ and $e'' \sqsubseteq e'$ it is not the case that $\Gamma \vdash e'' \Leftarrow A \Rightarrow A$
- There exist $e \sqsubseteq e_1$ and $H_1 \sqsubseteq A$ such that $\Gamma \vdash e_1 \Leftarrow H_1 \Rightarrow A$ and for all $e \sqsubseteq e_2 \sqsubseteq e_1$ and $H \sqsubseteq H' \sqsubseteq A$ it is not the case that $\Gamma \vdash e_2 \Leftarrow H_2 \Rightarrow A$

(Non)uniqueness of typing

Similarly to Extrinsic STLC, STLC with Hints does not enjoy uniqueness of typing. This is because we still can have terms like $\lambda x.\, x$ with hint **?**, which can be typed with any type of the form $A \to A$. However, if the hint is informative enough, then the type is unique. Moreover, every typable term can be given a hint which makes its type unique.