Higher-Ranked Exception Types

Ruud Koot

April 4, 2014

1 The $\lambda^{\cup}$-calculus

Types

$$
\begin{array}{lrll}
\tau \in \mathbf{Ty} & ::= & \mathcal{P} & \text{(base type)} \\
 & | & \tau_1 \to \tau_2 & \text{(function type)}
\end{array}
$$

Terms

$$
\begin{array}{lrll}
t \in \mathbf{Tm} & ::= & x & \text{(variable)} \\
 & | & \lambda x : \tau . t & \text{(abstraction)} \\
 & | & t_1\ t_2 & \text{(application)} \\
 & | & \emptyset & \text{(empty)} \\
 & | & \{c\} & \text{(singleton)} \\
 & | & t_1 \cup t_2 & \text{(union)}
\end{array}
$$

Environments

$$
\Gamma \in \mathbf{Env} ::= \cdot \quad | \quad \Gamma, x : \tau
$$

1.1 Typing relation

$$
\frac{}{\Gamma, x : \tau \vdash x : \tau}\ [\text{T-Var}] \qquad
\frac{\Gamma, x : \tau_1 \vdash t : \tau_2}{\Gamma \vdash \lambda x : \tau_1 . t : \tau_1 \to \tau_2}\ [\text{T-Abs}] \qquad
\frac{\Gamma \vdash t_1 : \tau_1 \to \tau_2 \quad \Gamma \vdash t_2 : \tau_1}{\Gamma \vdash t_1\ t_2 : \tau_2}\ [\text{T-App}]
$$

$$
\frac{}{\Gamma \vdash \emptyset : \mathcal{P}}\ [\text{T-Empty}] \qquad
\frac{}{\Gamma \vdash \{c\} : \mathcal{P}}\ [\text{T-Con}] \qquad
\frac{\Gamma \vdash t_1 : \tau \quad \Gamma \vdash t_2 : \tau}{\Gamma \vdash t_1 \cup t_2 : \tau}\ [\text{T-Union}]
$$

1.2 Semantics

1.3 Reduction relation

Definition 1. *Let $\prec$ be a strict total order on* $\mathbf{Con} + \mathbf{Var}$*, with $c \prec x$ for all $c \in$* $\mathbf{Con}$ *and $x \in$* $\mathbf{Var}$.

$$
\begin{aligned}
(\lambda x : \tau.t_1)\ \ t_2 &\longrightarrow [t_2/x]\,t_1 && (\beta\text{-reduction})\\
(t_1 \cup t_2)\ \ t_3 &\longrightarrow t_1\ t_3 \cup t_2\ t_3 && \\
(\lambda x : \tau.t_1) \cup (\lambda x : \tau.t_2) &\longrightarrow \lambda x : \tau.\,(t_1 \cup t_2) && \text{(congruences)}\\
x\ t_1 \cdots t_n \cup x'\ t_1' \cdots t_n' &\longrightarrow x\ \ (t_1 \cup t_1') \cdots (t_n \cup t_n') && \\
(t_1 \cup t_2) \cup t_3 &\longrightarrow t_1 \cup (t_2 \cup t_3) && \text{(associativity)}\\
\varnothing \cup t &\longrightarrow t && \\
t \cup \varnothing &\longrightarrow t && \text{(unit)}\\
x \cup x &\longrightarrow x && \\
x \cup (x \cup t) &\longrightarrow x \cup t && \\
\{c\} \cup \{c\} &\longrightarrow \{c\} && \text{(idempotence)}\\
\{c\} \cup (\{c\} \cup t) &\longrightarrow \{c\} \cup t && \\
&\longrightarrow && \text{(commutativity)}
\end{aligned}
$$

Conjecture 1. *The reduction relation $\longrightarrow$ preserves meaning.*

Conjecture 2. *The reduction relation $\longrightarrow$ is strongly normalizing.*

Conjecture 3. *The reduction relation $\longrightarrow$ is locally confluent.*

Corollary 1. *The reduction relation $\longrightarrow$ is confluent.*

Proof. Follows from SN, LC and Newman's Lemma. □

Corollary 2. *The $\lambda^{\cup}$-calculus has unique normal forms.*

Corollary 3. *Equality of $\lambda^{\cup}$-terms can be decided by normalization.*

2 Completion

$$
\begin{aligned}
\kappa \in \mathbf{Kind} \quad &::= \ \textsc{exn} && \text{(exception)}\\
&\ \ \mid\ \ \kappa_1 \Rightarrow \kappa_2 && \text{(exception operator)}\\[1em]
\chi \in \mathbf{Exn} \quad &::= \ e && \text{(exception variables)}\\
&\ \ \mid\ \ \lambda e : \kappa.\chi && \text{(exception abstraction)}\\[1em]
\widehat{\tau} \in \mathbf{ExnTy} \quad &::= \ \forall e :: \kappa.\widehat{\tau} && \text{(exception quantification)}\\
&\ \ \mid\ \ \widehat{\text{bool}} && \text{(boolean type)}\\
&\ \ \mid\ \ [\widehat{\tau}\ \mathbf{throws}\ \chi] && \text{(list type)}\\
&\ \ \mid\ \ \widehat{\tau}_1\ \mathbf{throws}\ \chi_1 \to \widehat{\tau}_2\ \mathbf{throws}\ \chi_2 && \text{(function type)}
\end{aligned}
$$

The completion procedure as a set of inference rules:
The completion procedure as an algorithm:

$$
\begin{aligned}
&\mathit{complete} :: \mathbf{Env} \times \mathbf{Ty} \to \mathbf{ExnTy} \times \mathbf{Exn} \times \mathbf{Env}\\
&\mathit{complete}\ \overline{e_i :: \kappa_i}\ \text{bool} =\\
&\quad \mathbf{let}\ e\ \mathit{be\ fresh}\\
&\quad \mathbf{in}\ \ \langle \widehat{\text{bool}}; e\ \overline{e_i}; e :: \overline{\kappa_i \Rightarrow}\ \textsc{exn} \rangle
\end{aligned}
$$

$$\frac{}{\overline{e_i :: \kappa_i} \vdash \mathrm{bool} : \widehat{\mathrm{bool}} \ \&\ e\ \overline{e_i} \triangleright e :: \overline{\kappa_i} \Rightarrow \textsc{exn}}\ [\textsc{C-Bool}]$$

$$\frac{\overline{e_i :: \kappa_i} \vdash \tau : \widehat{\tau} \ \&\ \chi \triangleright \overline{e_j :: \kappa_j}}{\overline{e_i :: \kappa_i} \vdash [\tau] : [\widehat{\tau}\ \mathbf{throws}\ \chi] \ \&\ e\ \overline{e_i} \triangleright e :: \overline{\kappa_i} \Rightarrow \textsc{exn}, \overline{e_j :: \kappa_j}}\ [\textsc{C-List}]$$

$$\frac{\vdash \tau_1 : \widehat{\tau}_1 \ \&\ \chi_1 \triangleright \overline{e_j :: \kappa_j} \quad \overline{e_i :: \kappa_i}, \overline{e_j :: \kappa_j} \vdash \tau_2 : \widehat{\tau}_2 \ \&\ \chi_2 \triangleright \overline{e_j :: \kappa_j}}{\overline{e_i :: \kappa_i} \vdash \tau_1 \to \tau_2 : \forall \overline{e_j :: \kappa_j}. \left(\widehat{\tau}_1\ \mathbf{throws}\ \chi_1 \to \widehat{\tau}_2\ \mathbf{throws}\ \chi_2\right) \ \&\ e\ \overline{e_i} \triangleright e :: \overline{\kappa_j} \Rightarrow \textsc{exn}, \overline{e_k :: \kappa_k}}\ [\textsc{C-Arr}]$$

Figure 1: Type completion ($\Gamma \vdash \tau : \widehat{\tau} \ \&\ \chi \triangleright \Gamma'$)