Towards the Unification of Terms, Types and Contexts

Ferruccio Guidi

Department of Computer Science
University of Bologna, Italy

fguidi@cs.unibo.it

Short presentation of the paper available at
http://arxiv.org/abs/cs/0611040

Certified proofs of all presented results are available at
http://helm.cs.unibo.it/lambda-delta

Leading conjecture

There is a typed λ-calculus satisfying the standard desired properties, whose terms, types and contexts use the same constructions

The calculus $\lambda\delta$

Terms and types share the same constructions

Every context is also a term

There is an infinite sequence of sorts
but they are not universes

The standard desired properties hold

It recalls Church simply typed λ-calculus
but has uniformly dependent types

A possible use case

Foundation for predicative type theories like
CTT (Martin-Löf), mTT (Maietti & Sambin)

Sort hierarchy parameter

$$g \in G \equiv \{\mathrm{next} \in \mathbb{N} \to \mathbb{N};\ \mathit{next\_lt} \in h < \mathrm{next}(h)\}$$

Constructions

Sort of index k, variable reference,
Church-style abstraction, abbreviation,
application, explicit type cast

Constructors

Terms and types (in prefix notation)

$$\mathrm{Sort}_k \quad x \quad \lambda x{:}V.T \quad \delta x{\leftarrow}V.T \quad (V).T \quad \langle V\rangle.T$$

Contexts (in prefix notation)

$$\mathrm{Sort}_k \quad \lambda x{:}V.C \quad \delta x{\leftarrow}V.C \quad (V).C \quad \langle V\rangle.C$$

Why these constructors?

The sequence of sorts allows
to type all legal terms

Abbreviations allow subject reduction

Casts reduce type checking to type inference

Strict substitution

$[x^{+}\!\leftarrow\! W]T$ replaces one or more occurrences of x in T with W (is undefined if $x \notin \mathrm{FV}(T)$)

Reduction steps (no critical pairs)

β-contraction

$$(W).\lambda x{:}V.T \quad \to_\beta \quad \delta x\!\leftarrow\! W.T$$

δ-expansion

$$\delta x\!\leftarrow\! V.T \quad \to_\delta \quad \delta x\!\leftarrow\! V.[x^{+}\!\leftarrow\! V]T$$

ζ-contraction

$$\delta x\!\leftarrow\! V.T \quad \to_\zeta \quad T \quad \text{if } x \notin \mathrm{FV}(T)$$

(not allowed if the redex is a context)

τ-contraction

$$\langle V\rangle.T \quad \to_\tau \quad T$$

υ-conversion

$$(W).\delta x\!\leftarrow\! V.T \quad \to_\upsilon \quad \delta x\!\leftarrow\! V.(W).T$$

contextual δ-expansion

$$D.\delta x\!\leftarrow\! V.D' \vdash \quad T \quad \to_\delta \quad [x^{+}\!\leftarrow\! V]T$$

Parallel reduction and conversion

$C \vdash T_1 \Rightarrow^* T_2$ and $C \vdash T_1 \Leftrightarrow^* T_2$

Type assignment

$$\frac{}{C \vdash_g \mathrm{Sort}_h : \mathrm{Sort}_{\mathrm{next}_g(h)}}\ \mathrm{sort}$$

$$\frac{C = D.\delta x{\leftarrow}V.D' \quad D \vdash_g V : T}{C \vdash_g x : T}\ \mathrm{def}$$

$$\frac{C = D.\lambda x{:}V.D' \quad D \vdash_g V : T}{C \vdash_g x : V}\ \mathrm{decl}$$

$$\frac{C \vdash_g V : T \quad C.\delta x{\leftarrow}V \vdash_g T_1 : T_2}{C \vdash_g \delta x{\leftarrow}V.T_1 : \delta x{\leftarrow}V.T_2}\ \mathrm{abbr}$$

$$\frac{C \vdash_g V : T \quad C.\lambda x{:}V \vdash_g T_1 : T_2}{C \vdash_g \lambda x{:}V.T_1 : \lambda x{:}V.T_2}\ \mathrm{abst}$$

$$\frac{C \vdash_g W : V \quad C \vdash_g U : \lambda x{:}V.T}{C \vdash_g (W).U : (W).\lambda x{:}V.T}\ \mathrm{appl}$$

$$\frac{C \vdash_g T_1 : T_2 \quad C \vdash_g T_2 : T_3}{C \vdash_g \langle T_2 \rangle.T_1 : \langle T_3 \rangle.T_2}\ \mathrm{cast}$$

$$\frac{C \vdash_g T_2 : T \quad C \vdash_g V : T_1 \quad C \vdash T_1 \Leftrightarrow^* T_2}{C \vdash_g V : T_2}\ \mathrm{conv}$$

Main certified properties

Reduction is confluent

If $C \vdash T_0 \Rightarrow^* T_1$ and $C \vdash T_0 \Rightarrow^* T_2$, there exists T such that $C \vdash T_1 \Rightarrow^* T$ and $C \vdash T_2 \Rightarrow^* T$

Types are typable

If $C \vdash_g T_1 : T_2$, there exists T_3 such that $C \vdash_g T_2 : T_3$

The types of a term are convertible

If $C \vdash_g T : T_1$ and $C \vdash_g T : T_2$ then $C \vdash T_1 \Leftrightarrow^* T_2$

The λ-abstraction is predicative

If $C \vdash_g \lambda x{:}V.T : U$ then $C \nvdash U \Leftrightarrow^* V$

The λ-abstraction is not absorbent $(x \notin \mathrm{FV}(U_2))$

If $C \vdash_g \lambda x{:}V.T : U_1$ and $C.\lambda x{:}V \vdash_g T : U_2$ then $C \nvdash U_1 \Leftrightarrow^* U_2$

The terms are not typed with themselves

If $C \vdash_g T : U$ then $C \nvdash U \Leftrightarrow^* T$

Reduction preserves the type

If $C \vdash T \Rightarrow^* T_1$ and $C \vdash_g T : T_2$ then $C \vdash_g T_1 : T_2$

The typable terms are strongly normalisable

If $C \vdash_g T : U$ then $C \vdash \mathrm{sn}(T)$

Type inference is decidable

$C \nvdash_g T_1 : T_2$ or there exists T_2 such that $C \vdash_g T_1 : T_2$

Type checking reduces to type inference

$C \vdash_g T : V$ iff there exists U such that $C \vdash_g \langle V \rangle.T : U$