

λ -Types on the λ -Calculus with Abbreviations

Ferruccio Guidi

Department of Computer Science
University of Bologna, Italy

fguidi@cs.unibo.it

Informal presentation of the paper available at
<http://arxiv.org/cs.LO/0611040>

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

The standard desired properties hold

It recalls Church simply typed λ -calculus

A possible use case

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

Sort hierarchy parameter

$$g \in G \equiv \{\text{next} \in \mathbb{N} \rightarrow \mathbb{N}; \text{next_lt} \in h < \text{next}(h)\}$$

Constructions

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

Constructors

Terms and types (in prefix notation)

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

Contexts (in prefix notation)

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

Reduction steps

β -contraction

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

δ -expansion

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

ζ -contraction

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

(not allowed if the redex is a context)

ϵ -contraction

$$\langle V \rangle.T \quad \rightarrow_{\epsilon} \quad T$$

ν -conversion

$$(W).\delta x \leftarrow V.T \quad \rightarrow_{\nu} \quad \delta x \leftarrow V.(W).T$$

contextual δ -expansion

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

Type assignment

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

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

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

$$\frac{C \vdash_g V : T \quad C.\delta x \leftarrow V \vdash_g T_1 : T_2 \quad C.\delta x \leftarrow V \vdash_g T_2 : T_0}{C \vdash_g \delta x \leftarrow V.T_1 : \delta x \leftarrow V.T_2} \text{ abbr}$$

$$\frac{C \vdash_g V : T \quad C.\lambda x : V \vdash_g T_1 : T_2 \quad C.\lambda x : V \vdash_g T_2 : T_0}{C \vdash_g \lambda x : V.T_1 : \lambda x : V.T_2} \text{ abst}$$

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

$$\frac{C \vdash_g T_1 : T_2 \quad C \vdash_g T_2 : T_0}{C \vdash_g \langle T_2 \rangle.T_1 : T_2} \text{ cast}$$

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

Main certified properties

The proofs of all properties (592 up to now) are formalised and are certified by the proof assistants *coq* and *matita*

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 $C \not\vdash U \Leftrightarrow^* V$

The terms are not typed with themselves

If $C \vdash_g T : U$ then $C \not\vdash U \Leftrightarrow^* T$

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 normalizable

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

Type inference is decidable

$C \not\vdash_g T_1 : T_2$ or there exists T_2 such that $C \vdash_g T_1 : T_2$