Undecidability of Equality in the Free Locally Cartesian Closed Category

Simon Castellan, Pierre Clairambault and Peter Dybjer

3rd July 2015
TLCA 2015, Warsaw

Introduction — Equivalences

Well-known correspondence:

$$\text{Simply-typed } \lambda\text{-calculus} \simeq \text{Cartesian Closed Categories}$$

Extension to Martin-Löf (1979) **extensional** type theory (TT):

$$\mathsf{ETT}_{\Sigma,\Pi,\mathsf{N}_1} \simeq \text{Locally Cartesian Closed Categories}$$

(Seely, Curien, Hofmann, and later Clairambault&Dybjer)

Introduction — Equivalences

Well-known correspondence:

$$\text{Simply-typed } \lambda\text{-calculus} \simeq \text{Cartesian Closed Categories (1)}$$

Extension to Martin-Löf (1979) **extensional** type theory (TT):

$$\mathrm{ETT}_{\Sigma,\Pi,N_1} \simeq \text{Locally Cartesian Closed Categories}$$

(Seely, Curien, Hofmann, and later Clairambault&Dybjer)

By (1) normalization of STLC implies decidability of equality in the free CCC.

The question. Can we do something similar for LCCCs?

Problems to address:

- *ETT is undecidable.* Which fragment?
- *The free LCCC:* Does it arise as the term model of ETT?

I. Undecidability of equality – the syntax side

Undecidability in type theory

Undecidability of equality for MLTT with

- Π-types
- *extensional* identity types
- **a universe closed under Π**

Indeed, the context Γ_λ encodes the problem of conversion in untyped λ-calculus:

$$\Gamma_\lambda = X : U, p : \mathsf{I}(U, X, X \Rightarrow X)$$

Undecidability in type theory

Undecidability of equality for MLTT with

- Π-types
- *extensional* identity types
- **a universe closed under Π**

Indeed, the context Γ_λ encodes the problem of conversion in untyped λ-calculus:

$$\Gamma_\lambda = X : U, p : \mathsf{I}(U, X, X \Rightarrow X)$$

But LCCCs do *not* support universes. We prove a stronger result:

Theorem. Judgemental equality in MLTT with extensional identity types, Π-types and a base type is undecidable.

Proof of the undecidability

Proof technique. Reduction to equality in combinatory logic (CL).

CL can be encoded in the following context Γ_{CL} over a base type o:

$$
\begin{array}{ll}
K : o, S : o, \cdot : o \to o \to o, & (\textit{term signature of } CL) \\
p : \Pi xy : o.\, \mathsf{I}(k \cdot x \cdot y, x), & (k \cdot x \cdot y =_{CL} x) \\
q : \Pi xyz : o.\, \mathsf{I}(s \cdot x \cdot y \cdot z, x \cdot z \cdot (y \cdot z)) & (s \cdot x \cdot y \cdot z =_{CL} x \cdot z \cdot (y \cdot z))
\end{array}
$$

Then, for terms of CL t and u,

$$t =_{CL} u \quad \text{iff} \quad \Gamma_{CL} \vdash \mathsf{I}(t, u) \text{ is inhabited} \quad \text{iff} \quad \Gamma \vdash t = u \quad \square$$

II. The freeness problem

But, wait, what does free mean?

A model is *free over a set* X if it is **initial** among models with a chosen interpretation for every element of X.

In our work, X is reduced to a single type o.

Our goal: Build a pair $(\mathcal{C}, o)$ where $o \in |\mathcal{C}|$, initial in $\mathbf{LCCC}^o$:

- *Objects:* Pairs $(\mathcal{D}, o')$ where $o' \in |\mathcal{D}|$.
- *Morphisms*: preserving the structure and *o up to iso*.

But, wait, what does free mean?

A model is *free over a set* X if it is **initial** among models with a chosen interpretation for every element of X.

In our work, X is reduced to a single type o.

Our goal: Build a pair $(\mathcal{C}, o)$ where $o \in |\mathcal{C}|$, initial in $\mathbf{LCCC}^o$:

- *Objects:* Pairs $(\mathcal{D}, o')$ where $o' \in |\mathcal{D}|$.
- *Morphisms*: preserving the structure and *o up to iso*.

Problem. $\mathbf{LCCC}^o$ does not have an initial object (no chosen structure) !

Bi-initiality in a 2-category

Our category $\mathbf{LCCC}^o$ is more adequately described as a 2-category.

Taking that into account we aim for a weaker notion of initiality:

Definition (Bi-initiality)

An object $\mathcal{A}$ is **bi-initial** in a 2-category $\mathscr{C}$ if for each object $\mathcal{B}$:

- there is a map $\mathcal{A} \to \mathcal{B}$
- for parallel maps $F, G : \mathcal{A} \to \mathcal{B}$, there is a unique 2-cell $F \Rightarrow G$.

$\to$ There is the unique map up to unique isomorphism from $\mathcal{A}$ to $\mathcal{B}$.

Proving this directly in our setting is combinatorially involved.
(Several problems have to dealt with at the same time.)

Categories with families to the rescue

Categories with families (cwfs) capture the kernel of dependent type theory **before the introduction of type constructors.**

For our problem, we make a détour through cwfs:

- Closer to syntax of type theory (Chosen structure)
- Support 1-categorical (*strict*) and 2-categorical (*weak*) structures.
- The biequivalence of **2-categories** of Clairambault&Dybjer

$$\mathbf{LCCC} \simeq \mathbf{CwF}^{\Sigma,\Pi,\mathsf{I},\mathsf{N}_1}_{\mathsf{weak}}.$$

III. Constructing the free category with families

Categories with families

Categories with families axiomatize the structure of dependency in contexts and types of MLTT.

Definition (A concrete definition of cwfs)

A *category with families* is a pair $(\mathcal{C}, T)$ where

- $\mathcal{C}$ is a category of contexts and substitutions with a terminal object 1.
- $T : \mathcal{C}^{\mathrm{op}} \to \mathbf{Fam}$ maps Γ to $(Tm_{\mathcal{C}}(\Gamma, A))_{(A \in Ty_{\mathcal{C}}(\Gamma))}$.

along with a context comprehension operator associating to each $\Gamma \in |\mathcal{C}|$ and $A \in Ty_{\mathcal{C}}(\Gamma)$ a context $\Gamma \cdot A$ coming with

- $\Gamma \cdot A \xrightarrow{p} \Gamma$
- $\Gamma \cdot A \vdash q : A[p]$

satisfying a universal property.

The syntax of type theory – Raw terms

Four syntactic classes: contexts, substitutions, types, terms:

$$
\begin{array}{rcl}
\Gamma & ::= & 1 \mid \Gamma.A \\
A & ::= & o \mid A[\gamma] \mid \Pi(A, A) \mid \Sigma(A, A) \mid \mathsf{I}(a, a) \\
\gamma & ::= & \gamma \circ \gamma \mid \mathsf{id}_{\Gamma} \mid \langle\rangle_{\Gamma} \mid \mathsf{p}_{A} \mid \langle \gamma, a \rangle_{A} \\
a & ::= & a[\gamma] \mid \mathsf{q}_{A} \\
 & \mid & \lambda(A, a) \mid \mathsf{app}(A, a, a) \\
 & \mid & \mathsf{pair}(A, a, a) \mid \mathsf{fst}(A, a) \mid \mathsf{snd}(A, A, a) \\
 & \mid & \mathsf{refl}(a)
\end{array}
$$

Exactly the language of cwfs (*explicit substitutions* and *de Bruijn indices*).

Syntax of type theory – Partial Equivalence Relations

On top of the raw terms, we define four pers:

$$\Gamma = \Gamma' \vdash \qquad \Gamma \vdash A = A' \quad \Gamma \vdash f = g : \Delta \quad \Gamma \vdash t = t' : A$$

Some (boring?) rules:

$$\frac{\Gamma' = \Gamma \vdash}{\Gamma = \Gamma' \vdash} \qquad \frac{\Gamma = \Gamma' \vdash \qquad \Gamma' = \Gamma'' \vdash}{\Gamma = \Gamma'' \vdash}$$

$$\frac{\Gamma \vdash A = A' \qquad \Gamma \cdot A \vdash B = B'}{\Gamma \vdash \Pi(A, B) = \Pi(A', B')} \quad \frac{\Gamma \vdash a = b : A \qquad \Gamma \vdash a' = b' : A}{\Gamma \vdash \mathsf{I}(a, a') = \mathsf{I}(b, b')}$$

$$\frac{\Gamma \vdash t = t' : A \qquad \Gamma = \Gamma' \vdash \qquad \Gamma \vdash A = A'}{\Gamma' \vdash t = t' : A'}$$

$$\frac{\Gamma \vdash f = f : \Delta}{\Gamma \vdash f \circ \mathsf{id}_\Gamma = f : \Delta} \qquad \frac{\Gamma \vdash f = f : \Delta \cdot A}{\Gamma \vdash f = \langle \mathsf{p}_A \circ f, \mathsf{q}_{A[f]} \rangle_A : \Delta \cdot A}$$

Typed terms. $\Gamma \vdash$ is a short-hand for $\Gamma = \Gamma' \vdash$ and so on.

The term model $\mathcal{T}$

The cwf $\mathcal{T}$. The quotient of raw terms by the pers is a cwf:

- $|\mathcal{T}| = \{\Gamma \mid \Gamma = \Gamma \vdash\}/(\_ = \_ \vdash)$
- $\mathcal{T}([\Gamma], [\Delta]) = \{\Gamma \vdash f : \Delta \mid \Gamma \vdash f = f : \Delta\}/(\Gamma \vdash \_ = \_ : \Delta)$
- $\mathrm{Ty}_{\mathcal{T}}([\Gamma]) = \ldots$
- $\mathrm{Tm}_{\mathcal{T}}([\Gamma], [A]) = \ldots$

Initiality of $(\mathcal{T}, [o])$ with strict morphisms. Following Streicher:

- A *partial interpretation* is defined on the raw terms
- Its domain is proved to contain the typable terms

This yields $[\![\cdot]\!] : \mathcal{T} \to \mathcal{C}$ for any cwf $\mathcal{C}$.
Unicity is a simple induction over the syntax.
Extends to type constructors: $\mathcal{T}^{\Sigma}, \mathcal{T}^{\Sigma,\Pi}, \ldots, \mathcal{T}^{\Sigma,\Pi,\mathrm{I},\mathrm{N}_1}$.

IV. The bifree locally cartesian closed category

$\mathcal{T}$ in bi-initial in $\mathbf{CwF}^o_{\text{weak}}$

$\mathcal{T}$ is initial in the category of cwfs with *strict* morphisms.

Proof of bi-initiality of $(\mathcal{T}, o)$ in $\mathbf{CwF}^o_{\text{weak}}$.

- We already have a (strict) morphism $[\![\cdot]\!] : \mathcal{T} \to \mathcal{C}$ to any cwf $\mathcal{C}$.
- Let $F : \mathcal{T} \to \mathcal{C}$ be another morphism.
  1. We build an iso $\varphi : [\![\cdot]\!] \Rightarrow F$ by induction on the syntax (**technically involved**)
  2. We prove that any other *2-cell* $\varphi' : [\![\cdot]\!] \to F$ is equal to φ.

This implies that $\mathcal{T}$ is bi-initial.

Also extends to type constructor: $\mathcal{T}^{\Sigma,\Pi,\mathsf{I},\mathsf{N}_1}$ is bi-initial.

$U\mathcal{T}^{\Sigma,\Pi,I,N_1}$ is initial in $\mathbf{LCCC}^o$

Recall the biequivalence between cwfs and lcccs:

$$\mathbf{CwF}_{\Sigma,\Pi,I} \underset{T}{\overset{U}{\rightleftarrows}} \mathbf{LCCC}$$

Biequivalences preserve bi-initial objects:
$\Rightarrow U\mathcal{T}^{\Sigma,\Pi,\mathrm{I},\mathrm{N}_1}$ is bi-initial in $\mathbf{LCCC}^o$ and the bifree LCCC on a base type.

Conclusion

Summary.

- Constructed the syntax of type theory as the *initial* and *bi-initial* cwfs.
- Generalized the result of undecidability of equality to a weaker type theory.
- The proof can also be carried out in Burroni's equational presentation of LCCCs. (Remark by T. Coquand)
- A biproduct of our paper is that we construct the free cwf in a very economical way.

Perspectives and further work

- A tension between chosen structure and universal property also exists in CCCs.
- Prove correctness of nbe for intensional type theory through an initiality result?