Towards Higher Observational Type Theory

Ambrus Kaposi
Eötvös Loránd University, Budapest

j.w.w. Thorsten Altenkirch and Mike Shulman

TYPES 2022
Nantes
20 June 2022
How is $\text{Id}_A : A \rightarrow A \rightarrow \text{Type}$ defined?

- Ordinary type theory: inductively by
  \[ \text{refl} : (a : A) \rightarrow \text{Id}_A a a \]

- Cubical type theory:
  \[ \text{Id}_A a_0 a_1 := (e : \mathbb{I} \rightarrow A) \times (e_0 = a_0) \times (e_1 = a_1) \]

- Observational type theory:
  \[ \text{Id}_{A \times B} (a_0, b_0) (a_1, b_1) = \text{Id}_A a_0 a_1 \times \text{Id}_B b_0 b_1 \]
  \[ \text{Id}_{A \rightarrow B} f g = (x : A) \rightarrow \text{Id}_B (f x) (g x) \]
  \[ \text{Id}_{\text{Bool}} a b = \text{if } a \text{ then } (\text{if } b \text{ then } \top \text{ else } \bot) \text{ else } (\text{if } b \text{ then } \bot \text{ else } \top) \]
  \[ \text{Id}_{\text{Type}} A B = (A \simeq B) \]

1. funext for free from the definition of $\text{Id}$ for $\Pi$
2. definitional injectivity and disjointness of constructors of inductive types
3. univalence by definition (hopefully)
4. no need for interval and higher dimensions
Observational type theory: a problem

\[
\text{Id}_{\Sigma(x:A).B \times (a_0, b_0)} (a_1, b_1) = \\
\sum (e : \text{Id}_A a_0 a_1). \text{Id}_B b_0 b_1 \\
\sum (e : \text{Id}_A a_0 a_1). \text{Id}_B b_0 (\text{transport}_B e b_0) b_1 \\
\sum (e : \text{Id}_A a_0 a_1). \text{Id}_B b_0 (\text{transport}_B e^{-1} b_1)
\]

Instead:

1. type dependency
2. transports: asymmetry, we don’t want to mention transport when specifying Id, we might only want parametricity
3. parametricity: preservation of correspondances (relations); equality: preservation of equivalences

- Altenkirch–McBride–Swierstra 2007: John Major equality
  - incompatible with univalence
  - a model construction / syntactic translation
1. syntactic translation on contexts, types, terms or constructing a displayed model from any model (and a section if we start with the syntax)
2. for experts: context should better be mapped to a context with projections, but I use the indexed version for conciseness
3. we tried adding all the \( R \) operations and their equations as new syntax expressing \( \text{Id} \) for \( \text{Con} \), \( \text{Id} \) for \( \text{Ty} \), cong/ap
4. refl adds new normal forms (it can’t be defined, there are non-parametric models)
The external parametricity translation can specify internal parametricity!

We just need to change from an external viewpoint to an internal.
In the presheaf model over the syntax of type theory, we have

\[
\begin{align*}
  &\text{Ty}^\circ : \text{Set} \\
  &\text{Tm}^\circ : \text{Ty}^\circ \to \text{Set} \\
  &\Sigma^\circ : (A : \text{Ty}^\circ) \to (\text{Tm}^\circ A \to \text{Ty}^\circ) \to \text{Ty}^\circ
\end{align*}
\]

We define the standard model of type theory internally to presheaves over the syntax.

\[
\begin{align*}
  \text{Con} &:= \text{Ty}^\circ \\
  \text{Ty} \Gamma &:= \text{Tm}^\circ \Gamma \to \text{Ty}^\circ \\
  \text{Tm} \Gamma A &:= (\gamma : \text{Tm}^\circ \Gamma) \to \text{Tm}^\circ (A \gamma) \\
  (\Gamma, A) &:= \Sigma^\circ \Gamma A
\end{align*}
\]

1. syntax of type theory forms a category
2. two-level type theory (\circ notation), HOAS
3. translate everything to external in words
4. model = CwF + extra
5. standard model = set model = type model
Internal parametricity

\[
\Gamma : \text{Con} \\
\frac{}{\Gamma^R : \text{Ty}(\Gamma, \Gamma)}
\]

\[
A : \text{Ty}\Gamma \\
\frac{}{A^R : \text{Ty}(\gamma_0 : \Gamma, \gamma_1 : \Gamma, \Gamma^R, A[\gamma_0], A[\gamma_1])}
\]

\[
a : \text{Tm}\Gamma A \\
\frac{}{a^R : \text{Tm}(\gamma_0 : \Gamma, \gamma_1 : \Gamma, \Gamma^R) (A^R[a[\gamma_0], a[\gamma_1]])}
\]

\[
(\Gamma, A)^R[(\gamma_0, a_0), (\gamma_1, a_1)] = \Sigma(\gamma_2 : \Gamma^R[\gamma_0, \gamma_1]).A^R[\gamma_0, \gamma_1, \gamma_2, a_0, a_1]
\]
Internal parametricity

\[ \Gamma : \text{Ty}^\circ \]
\[ \Gamma^R : \text{Tm}^\circ \Gamma \to \text{Tm}^\circ \Gamma \to \text{Ty}^\circ \]

\[ A : \text{Tm}^\circ \Gamma \to \text{Ty}^\circ \]
\[ A^R : \text{Tm}^\circ (\Gamma^R \gamma_0 \gamma_1) \to \text{Tm}^\circ (A \gamma_0) \to \text{Tm}^\circ (A \gamma_1) \to \text{Ty}^\circ \]

\[ a : (\gamma : \text{Tm}^\circ \Gamma) \to \text{Tm}^\circ (A \gamma) \]
\[ a^R : (\gamma_2 : \text{Tm}^\circ (\Gamma^R \gamma_0 \gamma_1)) \to \text{Tm}^\circ (A^R \gamma_2 (a \gamma_0) (a \gamma_1)) \]

\[ (\Sigma^\circ A)^R (\gamma_0, a_0) (\gamma_1, a_1) = \Sigma^\circ (\gamma_2 : \Gamma^R \gamma_0 \gamma_1). A^R \gamma_2 a_0 a_1 \]

1. We replace Con, Ty, \ldots by the standard model
1. We rename the operations.
2. This is the core of the syntax of H.O.T.T.
Summary

- The syntax for internal parametricity is the internal Bernardy logical relation interpretation.
  - Internal to presheaves over the syntax a.k.a. two level type theory, HOAS, logical framework.
  - Logical relation over the internal standard model.

- Work in progress!

- To get H.O.T.T., we need: transport, symmetries.
  - See Mike’s talks at the CMU HoTT seminar (click!)

- Compared to cubical type theory, cubical internal parametricity:
  - To specify the syntax, we don’t need an interval or talk about dimensions
  - Stricter, e.g. univalence computes better

1. More precisely, section of the logical relation displayed model over the standard model.