A Restricted Version of Reflection Compatible with Homotopy Type Theory

Théo Winterhalter,
supervised by Andrej Bauer and Matthieu Sozeau

11 September, 2017

Abstract

In this report, I present my work over the year on the relation between the different notion of equalities, particularly in the setting of homotpy type theory. We present a new notion of restricted reflection that is compatible with that setting, while still providing useful applications.

Acknowledgements

Although I do not mention them in the header, I would like to mention Philip Haselwarter and Nicolas Tabareau that also worked with me on this topic.

Introduction

When one wants to make sure their mathematical developements actually hold formally, they can rely on the power of proof assistants such as Coq [?] and Agda [?]. These allow users to write their proofs in a manner that is *understood* by the computer so that the machine can actually check if the statements written down make sense. As useful as it might be, these tools can sometimes be annoying to use as they force the user to go into extreme details.

The job of the people working on these tools is—in part—to make them easier to use and ever stronger. One such way is to allow an intuitive use of the notion of equality that is often required in mathematical reasonings. This is however not a trivial problem, and many notions of equalities have been (and still are being) investigated in type theory (the core foundation of the main proof assistants).

Another part of this work is to make sure that the type theories we use actually make sense (e.g. are consistent). This can in turn be proven using the very tools we develop. Part of my focus during this internship has been dedicated to the developpement of tools in the Coq proof assistant to prove meta-theorems of type theory. They are already available on github [?].

Our main contribution is still a new formulation of the reflection rule that is used to bridge the gap between the two main notion of equality in type theory that is compatible with the now famous Homotopy Type Theory (HoTT) and its Univalence axiom.

1 Source theory

The source type theory comes with a peculiar rule for reflection that brings a change in the usual conversion relation. Our reflection rule is stated as follows:

$$\frac{\Gamma \vdash e : \mathsf{Id}_{\mathsf{Bool}}(b_1, b_2)}{\Delta \vdash b_1 \equiv_\Gamma b_2 : \mathsf{Bool}} \quad (\Gamma \subseteq \Delta)$$

While conversion is now:

$$\frac{\Gamma \vdash t : A \qquad \Gamma \vdash A \equiv_\Gamma B}{\Gamma \vdash t : B}$$

The idea is that we can only use reflection on equality that already made sense in the context used for the conversion. This allows us to do a transport on that very equality at top-level.

2 Idea of the translation

For the translation, we assume we are only eliminating one application of the reflection rule involving $e : \mathsf{Id}_{Bool}(b_1, b_2)$. This means we can define the following notions, only by mentionning these e, b_1 and b_2.

2.1 Transports

In Oury's work, definitional equality gets translated to propositional equality but this alone implies the need of UIP. Instead, we refine this idea to have *transports* over e (the global equality). This means the only definitional equalities involved are on booleans that verify UIP.

Definition 1 (Transports). *We define mutually transports between contexts, between types, and between terms.*

- *For contexts* Γ_1 *and* Γ_2 *we define* $\gamma : \Gamma_1 \Longrightarrow \Gamma_2$ *as pointwise* $\Longrightarrow$ *on the types that compose the two contexts (it means that they must be of the same length).*

- *We define* $\Gamma \vdash \tau : A \Longrightarrow B$ *as two predicates* C_1 *and* C_2 *over two booleans and an equality between them, such that* $\Gamma \vdash C_1[b_1, b_2, e] \equiv_? A$ *and* $\Gamma \vdash C_2[b_1, b_2, e] \equiv_? B$ *and for all* b*,* $\Gamma \vdash C_1[b, b, \mathsf{refl}_b\,] \equiv_? C_2[b, b, \mathsf{refl}_b\,]$.

- *We define* $\Gamma \vdash \rho : u \Longrightarrow v : A$ *as two term-predicates* t_1 *and* t_2 *such that* $\Gamma \vdash t_1[b_1, b_2, e] \equiv_? u : A$ *and* $\Gamma \vdash t_2[b_1, b_2, e] \equiv_? v : A$ *and for all* b*,* $\Gamma \vdash t_1[b, b, \mathsf{refl}_b\,] \equiv_? t_2[b, b, \mathsf{refl}_b\,] : A$*.* Should it be A everywhere? Or should A depend on the equality as well?

From any meta-transport, we can deduce a real one as the action using the J eliminator on e (on terms, it is "only" under some predicate P, while on contexts, it comes from application of the respective transports on types through substitutions).

2.2 Relations for the translation

Before defining our translation, we define a relation between the expressions of the source type theory and those of the target stating that they have the same shape modulo the application of *meta-transports*. We write $t \triangleleft t'$ when t' (target) has the same shape as t (source).

Following this definition, we also define $t_1 \bowtie t_2$ between two expressions of the source to state that there is some t in the source such that $t \triangleleft t_1$ and $t \triangleleft t_2$, but this is merely implied by it and rather defined inductively (so that it's easier to work with).

Note: These relations are purely syntactic!

Now, we want to relate the notion of shape and the notion of meta-transport. This is the purpose of the following lemma that we believe is crucial to the main theorem as it allows to prove coherence between translations.

Lemma 2 (Coherence of translation)**.**

- *If* Γ_1 ctx *and* Γ_2 ctx *and* $\Gamma_1 \bowtie \Gamma_2$*, then there exists* $\gamma : \Gamma_1 \Longrightarrow \Gamma_2$.

- *If* $\gamma : \Gamma_1 \Longrightarrow \Gamma_2$ *and* $\Gamma_1 \vdash A_1$ type *and* $\Gamma_2 \vdash A_2$ type *and* $A_1 \bowtie A_2$ *then there exists* $\Gamma_2 \vdash \tau : \gamma.A_1 \Longrightarrow A_2$.

- *If* $\gamma : \Gamma_1 \Longrightarrow \Gamma_2$ *and* $\Gamma_1 \vdash u_1 : A_1$ *and* $\Gamma_2 \vdash u_2 : A_2$ *and* $u_1 \bowtie u_2$ *then there exists* $\Gamma_2 \vdash \tau : \gamma.A_1 \Longrightarrow A_2$ *and* $\Gamma_2 \vdash \rho : \tau.\gamma.u_1 \Longrightarrow u_2 : A_2$.

Proof. We focus on a few specific cases:

- Case Π: $\gamma : \Gamma_1 \Longrightarrow \Gamma_2$ and $\Gamma_1 \vdash \prod_{A_1} B_1$ type and $\Gamma_2 \vdash \prod_{A_2} B_2$ type and

$$\frac{A_1 \bowtie A_2 \qquad B_1 \bowtie B_2}{\prod_{A_1} B_1 \bowtie \prod_{A_2} B_2}$$

  By inversion we have $\Gamma_1 \vdash A_1$ type, $\Gamma_2 \vdash A_2$ type, $\Gamma_1, A_1 \vdash B_1$ type and $\Gamma_2, A_2 \vdash B_2$ type. So, by IH we have $\Gamma_2 \vdash \tau_1 : \gamma.A_1 \Longrightarrow A_2$, and thus $\gamma, \tau_A : \Gamma_1, A_1 \Longrightarrow \Gamma_2, A_2$.

  By definition we have the corresponding predicates C_1 and C_2 for As and D_1 and D_2 for Bs. We deduce $P_1 := \gamma.(\prod_{\gamma^{-1}.C_1}(\gamma, \tau_A)^{-1}.D_1)$ that works for $\gamma.(\prod_{A_1} B_1)$, and $P_2 := \prod_{C_2} D_2$ (noting that even γ and τ_A do depend on $[b_1, b_2, e]$, it is crucial to remark that the action of a meta-transport disappears when e is eliminated).

- Case application: $\gamma : \Gamma_1 \Longrightarrow \Gamma_2$ and $\Gamma_1 \vdash u_1 \mathbin{@}^{A_1.B_1} v_1 : T_1$ and $\Gamma_2 \vdash u_2 \mathbin{@}^{A_2.B_2} v_2 : T_2$ and

$$\frac{u_1 \bowtie u_2 \qquad A_1 \bowtie A_2 \qquad B_1 \bowtie B_2 \qquad v_1 \bowtie v_2}{u_1 \mathbin{@}^{A_1.B_1} v_1 \bowtie u_2 \mathbin{@}^{A_2.B_2} v_2}$$

  By inversion we have $\Gamma_i \vdash A_i$ type and $\Gamma_i, A_i \vdash B_i$ type and $\Gamma_i \vdash u_i : \prod_{A_i} B_i$ and $\Gamma_i \vdash v_i : A_i$ and $\Gamma_i \vdash B_i[v_i] \equiv_? T_i$.

  By IH we have $\Gamma_2 \vdash \tau_A : A_1 \Longrightarrow A_2$ and thus $\gamma, \tau_A : \Gamma_1, A_1 \Longrightarrow \Gamma_2, A_2$, so by IH, $\Gamma_2, A_2 \vdash \tau_B : \gamma, \tau_A.B_1 \Longrightarrow B_2$.

  By IH we get as well $\Gamma_2 \vdash \tau_\pi : \gamma.(\prod_{A_1} B_1) \Longrightarrow \prod_{A_2} B_2$ together with $\Gamma_2 \vdash \rho_u : \tau_\pi.\gamma.u_1 \Longrightarrow u_2 : \prod_{A_2} B_2$.

  Likewise, $\Gamma_2 \vdash \tau'_A : \gamma.A_1 \Longrightarrow A_2$ and $\Gamma_2 \vdash \rho_v : \tau'_A.\gamma.v_1 \Longrightarrow v_2 : A_2$.

  We first want $\Gamma_2 \vdash \_ : \gamma.B_1[v_1] \Longrightarrow B_2[v_2]$.

  We have $\Gamma_2 \vdash \_ : B_2[\tau'_A.\gamma.v_1] \Longrightarrow B_2[v_2]$ from ρ_v, so we need $\Gamma_2 \vdash \_ : \gamma.B_1[v_1] \Longrightarrow B_2[\tau'_A.\gamma.v_1]$.

  Adapating τ_B, we get $\Gamma_2 \vdash \_ : \gamma.B_1[v_1] \Longrightarrow B_2[\tau_A.\gamma.v_1]$, meaning there remains $\Gamma_2 \vdash \_ : \gamma.B_2[\tau_A.\gamma.v_1] \Longrightarrow B_2[\tau'_A.\gamma.v_1]$ which should simply hold.

  The whole conclusion should follow similarly.

- Case context transport on types: $\gamma : \Gamma_1 \Longrightarrow \Gamma_2$ and $\Gamma_i \vdash A_i$ type and

$$\frac{A_1 \bowtie A_2}{A_1 \bowtie \delta.A_2}$$

Although δ isn't technically a term and thus cannot be given a "type" directly from inversion of typing, we argue that the required information can be recovered from the inversion of the substitutions with J involved. This means we still get $\Delta \vdash A_2$ type and $\delta : \Delta \implies \Gamma_2$. From this we get $\delta^{-1} \circ \gamma : \Gamma_1 \implies \Delta$. Then, by induction hypothesis we have $\Delta \vdash \tau : (\delta^{-1} \circ \gamma).A_1 \implies A_2$. Now, we expect some $\Delta \vdash \_ : \gamma.A_1 \implies \delta.A_2$ instead.

Composing on both sides by δ should be harmless, and $\delta \circ \delta^{-1}$ should cancel each other.

- *TODO*

□

Note that the variable case is the one responsible for the need of $\gamma : \Gamma_1 \implies \Gamma_2$. *What about UIP and funext?? When do we need them?*

We will actually use the following corollary (the lemma was stated as such in order for the proof to go through).

Corollary 3 (Coherence)**.**

- *If* Γ_1 ctx *and* Γ_2 ctx *and* $\Gamma_1 \bowtie \Gamma_2$*, then there exists* $\gamma : \Gamma_1 \implies \Gamma_2$.
- *If* $\Gamma \vdash A_1$ type *and* $\Gamma \vdash A_2$ type *and* $A_1 \bowtie A_2$ *then there exists* $\Gamma \vdash \tau : A_1 \implies A_2$.
- *If* $\Gamma \vdash u_1 : A$ *and* $\Gamma \vdash u_2 : A$ *and* $u_1 \bowtie u_2$ *then there exists* $\Gamma \vdash \rho : u_1 \implies u_2 : A$.

Proof. The proof basically requires the meta-transport relation to be reflexive which it is trivially. In the term case, we also need to know that any $\Gamma \vdash \tau : A \implies A$ has an action that amounts to the identity, this is the case by axiom K. □

2.3 Translation

Here we mainly lay out the idea of the theorem we want to prove, but first we need to define the notion of translation of a judgment.

Definition 4 (Translation)**.** *We define the relation* $\Gamma' \vdash \mathcal{J}' \in [\![\Gamma \vdash \mathcal{J}]\!]$ *on judgments of the target, and judgments of the source as roughly* $\Gamma \triangleleft \Gamma'$ *and* $\mathcal{J} \triangleleft \mathcal{J}'$ *and* $\Gamma' \vdash \mathcal{J}'$*. This however does not apply to equality judgments. For those we instead produce a meta-transport.*

Remember that $J \lhd J'$ and $J \lhd J''$ implies $J' \bowtie J''$, meaning that two translations of the same term fall under lemma 3.

Lemma 5. *For any type T, we can always choose a translation that has the same head constructor as T.*

Proof. Assume we have $\Gamma' \vdash t' : T' \in [\![\Gamma \vdash t : T]\!]$. By definition we have $T \lhd T'$ meaning that T' is some T'' with the same head constructor as T decorated by meta-transports.

As in corollary 3, we get a transport that we apply to t' to get some $\Gamma'' \vdash t'' : T'' \in [\![\Gamma \vdash t : T]\!]$. (Note that we generally get a different Γ'' from Γ' in the process.) □

Theorem 6 (Translation)**.**

- *If* $\Gamma\ \mathsf{ctx}$ *then, there exists* $\Gamma'\ \mathsf{ctx} \in [\![\Gamma\ \mathsf{ctx}]\!]$.
- *If* $\Gamma \vdash A\ \mathsf{type}$ *then, there exists* $\Gamma' \vdash A'\ \mathsf{type} \in [\![\Gamma \vdash A\ \mathsf{type}]\!]$.
- *If* $\Gamma \vdash u : A$ *then, there exists* $\Gamma' \vdash u' : A' \in [\![\Gamma \vdash u : A]\!]$.
- What about substitutions, should they be explicit?
- *If* $\Gamma \equiv \Delta$ *then, there exists* $\Gamma'\ \mathsf{ctx} \in [\![\Gamma\ \mathsf{ctx}]\!]$ *and* $\Delta'\ \mathsf{ctx} \in [\![\Delta\ \mathsf{ctx}]\!]$ *as well as* $\gamma : \Gamma' \Longrightarrow \Delta'$.
- *If* $\Gamma \vdash A \equiv_? B$ *then, there exists* $\Gamma' \vdash A'\ \mathsf{type} \in [\![\Gamma \vdash A\ \mathsf{type}]\!]$ *and* $\Gamma' \vdash B'\ \mathsf{type} \in [\![\Gamma \vdash B\ \mathsf{type}]\!]$ *(with the same* Γ'*) as well as* $\Gamma' \vdash \tau : A' \Longrightarrow B'$.
- *If* $\Gamma \vdash u \equiv_? v : A$ *then, there exists* $\Gamma' \vdash u' : A' \in [\![\Gamma \vdash u : A]\!]$ *and* $\Gamma' \vdash v' : A' \in [\![\Gamma \vdash v : A]\!]$ *(with same* Γ' *and* A'*) as well as* $\Gamma' \vdash \rho : u' \Longrightarrow v' : A'$.

Proof. TODO □