Formalizing type theory through transport hell

Szumi Xie Viktor Bense

Eötvös Loránd University (ELTE)

SSTT 2026, Ljubljana

The result

Formalization of canonicity of type theory (with Π, Bool, and a universe)

The result

Formalization of canonicity of type theory (with Π, Bool, and a universe)

- Canonicity: $(t : \mathsf{Tm} \diamond \mathsf{Bool}) \to (t = \mathsf{true}) + (t = \mathsf{false})$

The result

Formalization of canonicity of type theory (with Π, Bool, and a universe)

- Canonicity: $(t : \mathsf{Tm} \diamond \mathsf{Bool}) \to (t = \mathsf{true}) + (t = \mathsf{false})$
- Intrinsic quotiented syntax

The result

Formalization of canonicity of type theory (with Π, Bool, and a universe)

- Canonicity: $(t : \mathsf{Tm} \diamond \mathsf{Bool}) \to (t = \mathsf{true}) + (t = \mathsf{false})$
- Intrinsic quotiented syntax
- “Gluing-style” proof / proof-relevant logical predicates

$$[\![A]\!] : \mathsf{Tm} \diamond A \to \mathsf{Set}$$
$$[\![\mathsf{Bool}]\!]\, t \coloneqq (t = \mathsf{true}) + (t = \mathsf{false})$$

The result

Formalization of canonicity of type theory (with Π, Bool, and a universe)

- Canonicity: $(t : \mathsf{Tm} \diamond \mathsf{Bool}) \to (t = \mathsf{true}) + (t = \mathsf{false})$
- Intrinsic quotiented syntax
- “Gluing-style” proof / proof-relevant logical predicates

$$[\![A]\!] : \mathsf{Tm} \diamond A \to \mathsf{Set}$$
$$[\![\mathsf{Bool}]\!]\, t \coloneqq (t = \mathsf{true}) + (t = \mathsf{false})$$

- The proof computes

The result

Formalization of canonicity of type theory (with Π, Bool, and a universe)

- Canonicity: $(t : \mathsf{Tm} \diamond \mathsf{Bool}) \to (t = \mathsf{true}) + (t = \mathsf{false})$
- Intrinsic quotiented syntax
- “Gluing-style” proof / proof-relevant logical predicates

$$[\![A]\!] : \mathsf{Tm} \diamond A \to \mathsf{Set}$$
$$[\![\mathsf{Bool}]\!]\, t \coloneqq (t = \mathsf{true}) + (t = \mathsf{false})$$

- The proof computes

Challenge: transport hell

The result

Formalization of canonicity of type theory (with Π, Bool, and a universe)

- Canonicity: $(t : \mathsf{Tm} \diamond \mathsf{Bool}) \to (t = \mathsf{true}) + (t = \mathsf{false})$
- Intrinsic quotiented syntax
- “Gluing-style” proof / proof-relevant logical predicates

$$[\![A]\!] : \mathsf{Tm} \diamond A \to \mathsf{Set}$$
$$[\![\mathsf{Bool}]\!]\, t \coloneqq (t = \mathsf{true}) + (t = \mathsf{false})$$

- The proof computes

Challenge: transport hell

Previous work: Kaposi & Pujet (2025) using strictification, too slow to compute

The result

Formalization of canonicity of type theory (with Π, Bool, and a universe)

- Canonicity: $(t : \mathsf{Tm} \diamond \mathsf{Bool}) \to (t = \mathsf{true}) + (t = \mathsf{false})$
- Intrinsic quotiented syntax
- “Gluing-style” proof / proof-relevant logical predicates

$$[\![A]\!] : \mathsf{Tm} \diamond A \to \mathsf{Set}$$
$$[\![\mathsf{Bool}]\!]\, t \coloneqq (t = \mathsf{true}) + (t = \mathsf{false})$$

- The proof computes

Challenge: transport hell

Previous work: Kaposi & Pujet (2025) using strictification, too slow to compute

Main contribution: A technique to work inside transport hell

Metatheory / proof assistant

Supports quotient inductive-inductive types (QIIT)

Has canonicity

Proof irrelevant equality

Metatheory / proof assistant

Supports quotient inductive-inductive types (QIIT)

Has canonicity

Proof irrelevant equality

→ Observational type theory (OTT)

Metatheory / proof assistant

Supports quotient inductive-inductive types (QIIT)

Has canonicity

Proof irrelevant equality

→ Observational type theory (OTT)

Proof assistant needs SProp and rewrite rules: *Agda* or Rocq

Transport hell
(live coding)
[link]

Other tricks

Controlling OTT computation

Records without η-expansion

Other tricks

Controlling OTT computation

Records without η-expansion

Ehrhard-style categories with families

Fibered instead of indexed logical predicates

$$[\![A]\!] : (S : \mathsf{Set}) \times (S \to \mathsf{Tm} \diamond A)$$

Demo
[link]

Drawbacks

It’s still transport hell

Boilerplate

Explicit equality arguments

Slow typechecking

Conclusion

A technique to work inside transport hell using heterogeneous equality and the combinator

$$- \blacktriangleright - : f \cong f' \to a \cong a' \to f\ a \cong f'\ a'$$

Intrinsic formalization of canonicity of type theory which computes on examples

Future work:

- Normalization of type theory
- A dedicated OTT proof assistant

Formalization: `https://github.com/szumixie/mltt`