XTT: Cubical Syntax for Extensional Equality

(without equality reflection)

June 11, 2019

Jonathan Sterling[1] Carlo Angiuli[1] Daniel Gratzer[2]

[1]Carnegie Mellon University

[2]Aarhus University

Equality in type theory

a thorny and controversial subject! here are some words that all type theorists fear:

Equality in type theory

a thorny and controversial subject! here are some words that all type theorists fear: definitional equality,

Equality in type theory

a thorny and controversial subject! here are some words that all type theorists fear: definitional equality, conversion (???),

Equality in type theory

a thorny and controversial subject! here are some words that all type theorists fear: definitional equality, conversion (???), judgmental equality,

Equality in type theory

a thorny and controversial subject! here are some words that all type theorists fear: definitional equality, conversion (???), judgmental equality, propositional equality, ...

Equality in type theory

a thorny and controversial subject! here are some words that all type theorists fear: definitional equality, conversion (???), judgmental equality, propositional equality, ...

the main scientific distinctions that can be made are in fact:

- what equations can the machine take responsiblity for? $(\alpha, \delta, \beta, \eta, \xi, \nu, \ldots)$
- what equations induce coercions in terms (silent vs. non-silent)? are they (weakly, strictly) coherent?

these considerations are *dialectically linked*

Equality in type theory

a thorny and controversial subject! here are some words that all type theorists fear: definitional equality, conversion (???), judgmental equality, propositional equality, ...

the main scientific distinctions that can be made are in fact:

- what equations can the machine take responsiblity for? $(\alpha, \delta, \beta, \eta, \xi, \nu, \ldots)$
- what equations induce coercions in terms (silent vs. non-silent)? are they (weakly, strictly) coherent?

these considerations are *dialectically linked*

Nuprl and **Andromeda** make all equations “silent”: semantically advantageous, but unfortunate side effect is that only α, δ can be fully automated (*).

Equality in type theory

a thorny and controversial subject! here are some words that all type theorists fear: definitional equality, conversion (???), judgmental equality, propositional equality, ...

the main scientific distinctions that can be made are in fact:

- what equations can the machine take responsiblity for? $(\alpha, \delta, \beta, \eta, \xi, \nu, \ldots)$
- what equations induce coercions in terms (silent vs. non-silent)? are they (weakly, strictly) coherent?

these considerations are *dialectically linked*

Nuprl and **Andromeda** make all equations “silent”: semantically advantageous, but unfortunate side effect is that only α, δ can be fully automated (*).

formalisms based on **ITT** maximize automatic equations, at the cost of some coercions appearing in terms. developing user-friendly **ITT**-style formalisms with well-behaved extensionality principles (**OTT**,**HoTT**,**CuTT**) has been a challenge.

Equality in type theory

a thorny and controversial subject! here are some words that all type theorists fear: definitional equality, conversion (???), judgmental equality, propositional equality, ...

the main scientific distinctions that can be made are in fact:

- what equations can the machine take responsiblity for? $(\alpha, \delta, \beta, \eta, \xi, \nu, \ldots)$
- what equations induce coercions in terms (silent vs. non-silent)? are they (weakly, strictly) coherent?

these considerations are *dialectically linked*

Nuprl and **Andromeda** make all equations "silent": semantically advantageous, but unfortunate side effect is that only α, δ can be fully automated (*).

formalisms based on **ITT** maximize automatic equations, at the cost of some coercions appearing in terms. developing user-friendly **ITT**-style formalisms with well-behaved extensionality principles (**OTT**,**HoTT**,**CuTT**) has been a challenge.

today, we examine XTT: a new take on OTT, using cubes.

Observational Type Theory

a big inspiration for me to get into type theory:

Altenkirch and McBride [AM06]. *Towards Observational Type Theory*.

Altenkirch, McBride, and Swierstra [AMS07]. "Observational Equality, Now!"

Observational Type Theory

a big inspiration for me to get into type theory:

Altenkirch and McBride [AM06]. *Towards Observational Type Theory*.

Altenkirch, McBride, and Swierstra [AMS07]. "Observational Equality, Now!"

- hierarchy of *closed*/*inductive* universes of Bishop sets, props

Observational Type Theory

a big inspiration for me to get into type theory:

Altenkirch and McBride [AM06]. *Towards Observational Type Theory*.

Altenkirch, McBride, and Swierstra [AMS07]. “Observational Equality, Now!”

- hierarchy of *closed*/*inductive* universes of Bishop sets, props
- heterogeneous equality type $\mathbf{Eq}(M : A, N : B)$ defined as *generic program*, by recursion on type codes A, B

$$\begin{aligned} &\mathbf{Eq}(F_0 : A_0 \to B_0, F_1 : A_1 \to B_1) = \\ &\quad (x_0 : A_0)(x_1 : A_1)(\widetilde{x} : \mathbf{Eq}(x_0 : A_0, x_1 : A_1)) \\ &\quad \to \mathbf{Eq}(F_0(x_0) : B_0, F_1(x_1) : B_1) \end{aligned} \qquad \text{(funext)}$$

Observational Type Theory

a big inspiration for me to get into type theory:

Altenkirch and McBride [AM06]. *Towards Observational Type Theory*.
Altenkirch, McBride, and Swierstra [AMS07]. "Observational Equality, Now!"

- hierarchy of *closed*/*inductive* universes of Bishop sets, props
- heterogeneous equality type $\mathbf{Eq}(M : A, N : B)$ defined as *generic program*, by recursion on type codes A, B

$$\begin{aligned}&\mathbf{Eq}(F_0 : A_0 \to B_0, F_1 : A_1 \to B_1) = \\ &\quad (x_0 : A_0)(x_1 : A_1)(\widetilde{x} : \mathbf{Eq}(x_0 : A_0, x_1 : A_1)) \\ &\quad \to \mathbf{Eq}(F_0(x_0) : B_0, F_1(x_1) : B_1)\end{aligned} \qquad \text{(funext)}$$

- judgmental UIP (proof irrelevance)

Observational Type Theory

a big inspiration for me to get into type theory:

Altenkirch and McBride [AM06]. *Towards Observational Type Theory*.
Altenkirch, McBride, and Swierstra [AMS07]. "Observational Equality, Now!"

- hierarchy of *closed*/*inductive* universes of Bishop sets, props
- heterogeneous equality type $\mathbf{Eq}(M : A, N : B)$ defined as *generic program*, by recursion on type codes A, B

$$\begin{aligned} &\mathbf{Eq}(F_0 : A_0 \to B_0, F_1 : A_1 \to B_1) = \\ &\quad (x_0 : A_0)(x_1 : A_1)(\tilde{x} : \mathbf{Eq}(x_0 : A_0, x_1 : A_1)) \\ &\quad \to \mathbf{Eq}(F_0(x_0) : B_0, F_1(x_1) : B_1) \end{aligned} \qquad \text{(funext)}$$

- judgmental UIP (proof irrelevance)
- many primitives: reflexivity, respect, coercion, coherence, heterogeneous irrelevance (see Altenkirch, McBride, and Swierstra [AMS07])

Observational Type Theory

a big inspiration for me to get into type theory:

Altenkirch and McBride [AM06]. *Towards Observational Type Theory*.
Altenkirch, McBride, and Swierstra [AMS07]. "Observational Equality, Now!"

- hierarchy of *closed*/*inductive* universes of Bishop sets, props
- heterogeneous equality type $\mathbf{Eq}(M : A, N : B)$ defined as *generic program*, by recursion on type codes A, B

$$\begin{aligned}&\mathbf{Eq}(F_0 : A_0 \to B_0, F_1 : A_1 \to B_1) = \\ &(x_0 : A_0)(x_1 : A_1)(\tilde{x} : \mathbf{Eq}(x_0 : A_0, x_1 : A_1)) \\ &\to \mathbf{Eq}(F_0(x_0) : B_0, F_1(x_1) : B_1)\end{aligned} \qquad \text{(funext)}$$

- judgmental UIP (proof irrelevance)
- many primitives: reflexivity, respect, coercion, coherence, heterogeneous irrelevance (see Altenkirch, McBride, and Swierstra [AMS07])
- metatheory: canonicity, decidability of type checking

the primitives of OTT

- reflexivity, symmetry, transitivity

the primitives of OTT

- reflexivity, symmetry, transitivity
- respect

$$\frac{A : \mathbf{U} \qquad x : A \vdash B[x] : \mathbf{U} \qquad M_0, M_1 : A \qquad \widetilde{M} : \mathbf{Eq}(M_0 : A, M_1 : A)}{\mathbf{resp}_{x:A.B[x]}(M_0, M_1, \widetilde{M}) : \mathbf{Eq}(B[M_0] : \mathbf{U}, B[M_1] : \mathbf{U})}$$

the primitives of OTT

- reflexivity, symmetry, transitivity
- respect

$$\frac{A : \mathbf{U} \qquad x : A \vdash B[x] : \mathbf{U} \qquad M_0, M_1 : A \qquad \widetilde{M} : \mathbf{Eq}(M_0 : A, M_1 : A)}{\mathbf{resp}_{x:A.B[x]}(M_0, M_1, \widetilde{M}) : \mathbf{Eq}(B[M_0] : \mathbf{U}, B[M_1] : \mathbf{U})}$$

- coercion

$$\frac{A, B : \mathbf{U} \qquad Q : \mathbf{Eq}(A : \mathbf{U}, B : \mathbf{U}) \qquad M : A}{[Q] \downarrow_B^A M : B}$$

the primitives of OTT

- reflexivity, symmetry, transitivity
- respect

$$\frac{A : \mathbf{U} \qquad x : A \vdash B[x] : \mathbf{U} \qquad M_0, M_1 : A \qquad \widetilde{M} : \mathbf{Eq}(M_0 : A, M_1 : A)}{\mathbf{resp}_{x:A.B[x]}(M_0, M_1, \widetilde{M}) : \mathbf{Eq}(B[M_0] : \mathbf{U}, B[M_1] : \mathbf{U})}$$

- coercion

$$\frac{A, B : \mathbf{U} \qquad Q : \mathbf{Eq}(A : \mathbf{U}, B : \mathbf{U}) \qquad M : A}{[Q] \downarrow_B^A M : B}$$

- coherence

$$\frac{A, B : \mathbf{U} \qquad Q : \mathbf{Eq}(A : \mathbf{U}, B : \mathbf{U}) \qquad M : A}{[\![Q]\!] \downarrow_B^A M : \mathbf{Eq}(A : M, B : [Q] \downarrow_B^A M)}$$

the primitives of OTT

- reflexivity, symmetry, transitivity
- respect

$$\frac{A : \mathbf{U} \qquad x : A \vdash B[x] : \mathbf{U} \qquad M_0, M_1 : A \qquad \widetilde{M} : \mathbf{Eq}(M_0 : A, M_1 : A)}{\mathbf{resp}_{x:A.B[x]}(M_0, M_1, \widetilde{M}) : \mathbf{Eq}(B[M_0] : \mathbf{U}, B[M_1] : \mathbf{U})}$$

- coercion

$$\frac{A, B : \mathbf{U} \qquad Q : \mathbf{Eq}(A : \mathbf{U}, B : \mathbf{U}) \qquad M : A}{[Q] \downarrow_B^A M : B}$$

- coherence

$$\frac{A, B : \mathbf{U} \qquad Q : \mathbf{Eq}(A : \mathbf{U}, B : \mathbf{U}) \qquad M : A}{[\![Q]\!] \downarrow_B^A M : \mathbf{Eq}(A : M, B : [Q] \downarrow_B^A M)}$$

(many of these can be *defined* in the Agda model of **OTT**, but must be *primitive operations* in “real” **OTT**.)

cubical reconstruction: XTT

goal: find smaller set of primitives which systematically generate (something in the spirit of) **OTT**

idea: start with Cartesian cubical type theory [ABCFHL], restrict to *Bishop sets* à la Coquand [Coq17]

the XTT paper

Sterling, Angiuli, and Gratzer [SAG19]. "Cubical Syntax for Reflection-Free Extensional Equality". *Formal Structures for Computation and Deduction (FSCD 2019).*

see also Chapman, Forsberg, and McBride [CFM18] ("The Box of Delights (Cubical Observational Type Theory)") for the beginnings of a different account of Cubical **OTT**.

(we won't talk about propositions or quotients today. but talk to me about it after! there is a strictness mismatch in both **OTT**,**XTT**.)

XTT: equality using the interval

rather than defining heterogeneous equality by recursion on type structure, define *dependent equality* all at once using a formal interval:

$$\frac{}{0, 1 : \mathbb{I}}$$

EQ FORMATION

$$\frac{i : \mathbb{I} \vdash A : \mathbf{U} \qquad M : A[0] \qquad N : A[1]}{\mathbf{Eq}_{i.A[i]}(M, N) : \mathbf{U}}$$

EQ INTRODUCTION

$$\frac{i : \mathbb{I} \vdash M[i] : A[i] \qquad M[0] = N_0 : A[0] \qquad M[1] = N_1 : A[1]}{\lambda i.M[i] : \mathbf{Eq}_{i.A[i]}(N_0, N_1)}$$

EQ ELIMINATION

$$\frac{M : \mathbf{Eq}_{i.A[i]}(N_0, N_1) \qquad r : \mathbb{I}}{M(r) : A[r] \qquad M(0) = N_0 : A[0] \qquad M(1) = N_1 : A[1]}$$

(along with more β, η rules, etc.)

function extensionality in XTT

we have function extensionality by swapping quantifiers:

$$\frac{F_0, F_1 : A \to B \qquad Q : (x : A) \to \mathbf{Eq}_{\_.B}(F_0(x), F_1(x))}{\lambda i.\lambda x.Q(x)(i) : \mathbf{Eq}_{\_.A\to B}(F_0, F_1)}$$

“respect” is just function application

given $A : \mathbf{U}$ and $x : A \vdash B[x] : \mathbf{U}$ and $Q : \mathbf{Eq}_{\_.A}(M_0, M_1)$, we have:

$$\lambda i.B[Q(i)] : \mathbf{Eq}_{\_.\mathbf{U}}(B[M_0], B[M_1])$$

judgmental UIP via *boundary separation*

in **OTT**, we always have $Q_0 = Q_1 : \mathbf{Eq}(M : A, N : B)$; we achieve this modularly using a *boundary separation*[1] rule:

$$\frac{r : \mathbb{I} \qquad r = 0 \vdash M = N : A \qquad r = 1 \vdash M = N : A}{M = N : A}$$

(does not mention equality type!!)

given $Q_0, Q_1 : \mathbf{Eq}_{i.A}(M, N)$, we have $Q_0 = Q_1 : \mathbf{Eq}_{i.A}(M, N)$ by the β, η, ξ rules of the equality type, together with boundary separation.

[1](it is a presheaf separation condition for a certain coverage on the category of contexts)

generalized coercion: coercion, coherence, and more

we generalize **OTT**'s coercion $[Q] \downarrow_B^A M$ and coherence $[\![Q]\!] \downarrow_B^A M$ with a single operator to coerce between parts of a cube [ABCFHL]:

$$\frac{r, r' : \mathbb{I} \qquad i : \mathbb{I} \vdash A[i] : \mathbf{U} \qquad M : A[r]}{[i.A[i]] \downarrow_{r'}^{r} M : A[r']}$$

given $Q : \mathbf{Eq}_{\_.\mathbf{U}}(A, B)$, we define:

$$[Q] \downarrow_B^A M = [i.Q(i)] \downarrow_1^0 M$$
$$[\![Q]\!] \downarrow_B^A M = \lambda i.[j.Q(j)] \downarrow_i^0 M$$

slogan: coherence is just coercion from a point to a line

like in **OTT** (but unlike **CuTT**), coercion must be calculated by recursion on A, B rather than Q; requires closed universe. ask me why!

subjective metatheory: counting grains of sand

we used to study the metatheory of *presentations* of type theories, not of type theories.

subjective metatheory: counting grains of sand

we used to study the metatheory of *presentations* of type theories, not of type theories.

1. “raw” terms, “raw” substitution, insufficient annotations (*a priori* no determinate notion of model, nor interpretation)
2. ???
3. interpretation into models???

subjective metatheory: counting grains of sand

we used to study the metatheory of *presentations* of type theories, not of type theories.

1. “raw” terms, “raw” substitution, insufficient annotations (*a priori* no determinate notion of model, nor interpretation)
2. prove normalization for raw syntax (but without using model theory!)
3. interpretation into models???

subjective metatheory: counting grains of sand

we used to study the metatheory of *presentations* of type theories, not of type theories.

1. "raw" terms, "raw" substitution, insufficient annotations (*a priori* no determinate notion of model, nor interpretation)
2. prove normalization for raw syntax (but without using model theory!)
   2.1 operational semantics
   2.2 PER "model" of type theory
   2.3 logical relation between syntax and PER "model"

   (~ 200 pages of work)
3. interpretation into models???

subjective metatheory: counting grains of sand

we used to study the metatheory of *presentations* of type theories, not of type theories.

1. "raw" terms, "raw" substitution, insufficient annotations (*a priori* no determinate notion of model, nor interpretation)
2. prove normalization for raw syntax (but without using model theory!)
   - 2.1 operational semantics
   - 2.2 PER "model" of type theory
   - 2.3 logical relation between syntax and PER "model"

   (~ 200 pages of work)
3. sound & complete interpretation (~ 100 more pages of work)

subjective metatheory: counting grains of sand

we used to study the metatheory of *presentations* of type theories, not of type theories.

1. "raw" terms, "raw" substitution, insufficient annotations (*a priori* no determinate notion of model, nor interpretation)
2. prove normalization for raw syntax (but without using model theory!)
   - 2.1 operational semantics
   - 2.2 PER "model" of type theory
   - 2.3 logical relation between syntax and PER "model"

   (~ 200 pages of work)
3. sound & complete interpretation (~ 100 more pages of work)

actually this is totally intractable to do more than once! let's bootstrap it a different way.

objective metatheory and categorical gluing

a new (old) syntax-invariant approach to metatheory

[2] See also Coquand, Huber, and Sattler [CHS19], Kaposi, Huber, and Sattler [KHS19], and Shulman [Shu15].

objective metatheory and categorical gluing

a new (old) syntax-invariant approach to metatheory

1. type theory is essentially algebraic (insist on it!) [Car86; ACD08; Awo18; Uem19]; presentations considered up to isomorphism

[2] See also Coquand, Huber, and Sattler [CHS19], Kaposi, Huber, and Sattler [KHS19], and Shulman [Shu15].

objective metatheory and categorical gluing

a new (old) syntax-invariant approach to metatheory

1. type theory is essentially algebraic (insist on it!) [Car86; ACD08; Awo18; Uem19]; presentations considered up to isomorphism
2. each type theory $\mathbb{T}$ *automatically* induces a category of algebras with initial object (soundness and completeness); initial algebra is covered by *fully-annotated* De Bruijn syntax (but this doesn't matter)

[2] See also Coquand, Huber, and Sattler [CHS19], Kaposi, Huber, and Sattler [KHS19], and Shulman [Shu15].

objective metatheory and categorical gluing

a new (old) syntax-invariant approach to metatheory

1. type theory is essentially algebraic (insist on it!) [Car86; ACD08; Awo18; Uem19]; presentations considered up to isomorphism
2. each type theory $\mathbb{T}$ *automatically* induces a category of algebras with initial object (soundness and completeness); initial algebra is covered by *fully-annotated* De Bruijn syntax (but this doesn't matter)
3. easily prove canonicity, normalization, decidability of type checking for initial $\mathbb{T}$-algebra using categorical gluing/logical families [Coq18][2]

[2] See also Coquand, Huber, and Sattler [CHS19], Kaposi, Huber, and Sattler [KHS19], and Shulman [Shu15].

objective metatheory and categorical gluing

a new (old) syntax-invariant approach to metatheory

1. type theory is essentially algebraic (insist on it!) [Car86; ACD08; Awo18; Uem19]; presentations considered up to isomorphism
2. each type theory $\mathbb{T}$ *automatically* induces a category of algebras with initial object (soundness and completeness); initial algebra is covered by *fully-annotated* De Bruijn syntax (but this doesn't matter)
3. easily prove canonicity, normalization, decidability of type checking for initial $\mathbb{T}$-algebra using categorical gluing/logical families [Coq18][2]
4. relate "informal" & unannotated syntax to initial $\mathbb{T}$-algebra by elaboration (using the above)

[2] See also Coquand, Huber, and Sattler [CHS19], Kaposi, Huber, and Sattler [KHS19], and Shulman [Shu15].

objective metatheory and categorical gluing

a new (old) syntax-invariant approach to metatheory

1. type theory is essentially algebraic (insist on it!) [Car86; ACD08; Awo18; Uem19]; presentations considered up to isomorphism
2. each type theory $\mathbb{T}$ *automatically* induces a category of algebras with initial object (soundness and completeness); initial algebra is covered by *fully-annotated* De Bruijn syntax (but this doesn't matter)
3. easily prove canonicity, normalization, decidability of type checking for initial $\mathbb{T}$-algebra using categorical gluing/logical families [Coq18][2]
4. relate "informal" & unannotated syntax to initial $\mathbb{T}$-algebra by elaboration (using the above)

the language of category theory makes each of the preceding steps "easy", and independent of syntax / representation details. no raw terms, no PERs.

[2] See also Coquand, Huber, and Sattler [CHS19], Kaposi, Huber, and Sattler [KHS19], and Shulman [Shu15].

cubical gluing: canonicity for XTT

to warm up, we proved canonicity for **XTT** using a cubical gluing technique (independently proposed by Awodey).

[3] Circulated by S. Awodey in 2015.

cubical gluing: canonicity for XTT

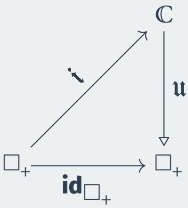
to warm up, we proved canonicity for **XTT** using a cubical gluing technique (independently proposed by Awodey). Let $\square_+$ be the completion of $\square$ with an initial object (i.e. constrained dimension contexts); $\mathbb{C}$ is the (fibered) category of **XTT**-contexts.

$$\begin{array}{c} \mathbb{C} \\ \Big\downarrow \mathfrak{U} \\ \square_+ \end{array}$$

[3] Circulated by S. Awodey in 2015.

cubical gluing: canonicity for XTT

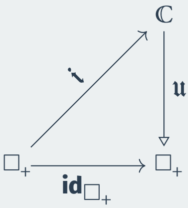
to warm up, we proved canonicity for **XTT** using a cubical gluing technique (independently proposed by Awodey). Let $\square_+$ be the completion of $\square$ with an initial object (i.e. constrained dimension contexts); $\mathbb{C}$ is the (fibered) category of **XTT**-contexts.



[3] Circulated by S. Awodey in 2015.

cubical gluing: canonicity for XTT

to warm up, we proved canonicity for **XTT** using a cubical gluing technique (independently proposed by Awodey). Let $\square_+$ be the completion of $\square$ with an initial object (i.e. constrained dimension contexts); $\mathbb{C}$ is the (fibered) category of **XTT**-contexts.

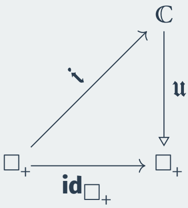


the splitting of $\mathfrak{u}$ interprets dimension substitutions, as well as “relatively terminal” contexts $\mathfrak{i}(\Psi) : \mathbb{C}$ for each $\Psi : \square_+$.

[3] Circulated by S. Awodey in 2015.

cubical gluing: canonicity for XTT

to warm up, we proved canonicity for **XTT** using a cubical gluing technique (independently proposed by Awodey). Let $\square_+$ be the completion of $\square$ with an initial object (i.e. constrained dimension contexts); $\mathbb{C}$ is the (fibered) category of **XTT**-contexts.



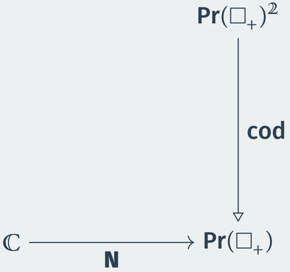
the splitting of $\mathfrak{u}$ interprets dimension substitutions, as well as "relatively terminal" contexts $\mathfrak{i}(\Psi) : \mathbb{C}$ for each $\Psi : \square_+$. we further obtain a "nerve":[3]

$$\mathbf{N} : \mathbb{C} \longrightarrow \mathbf{Pr}(\square_+)$$
$$\mathbf{N}(\Gamma) = \mathbb{C}[\mathfrak{i}(-), \Gamma]$$

[3] Circulated by S. Awodey in 2015.

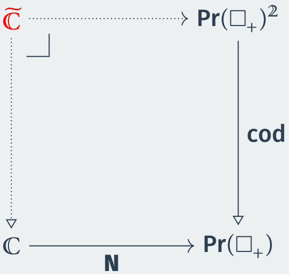
gluing along the cubical nerve

by gluing the codomain fibration along $\mathbb{C} \xrightarrow{\mathbf{N}} \mathbf{Pr}(\square_+)$, we obtain a category of *cubical logical families* (proof-relevant Kripke logical predicates):



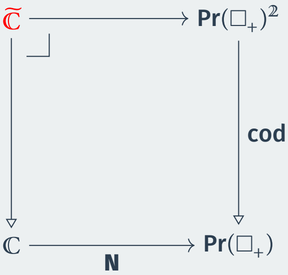
gluing along the cubical nerve

by gluing the codomain fibration along $\mathbb{C} \xrightarrow{\mathbf{N}} \mathbf{Pr}(\square_+)$, we obtain a category of *cubical logical families* (proof-relevant Kripke logical predicates):



gluing along the cubical nerve

by gluing the codomain fibration along $\mathbb{C} \xrightarrow{\mathbf{N}} \mathbf{Pr}(\square_+)$, we obtain a category of *cubical logical families* (proof-relevant Kripke logical predicates):



idea: lift the **XTT**-algebra structure from $\mathbb{C}$ to $\widetilde{\mathbb{C}}$, yielding canonicity at base type for *any* representative of the initial **XTT**-algebra $\mathbb{C}$.

summary of contributions

- (Cartesian) cubical reconstruction of **OTT**
- first steps in objective metatheory for cubical type theory
    - algebraic model theory
    - (strict) canonicity by gluing
- next: normalization, decidability of type checking, elaboration!

References I

[ABCFHL] Carlo Angiuli, Guillaume Brunerie, Thierry Coquand, Kuen-Bang Hou (Favonia), Robert Harper, and Daniel R. Licata. "Syntax and Models of Cartesian Cubical Type Theory". Preprint. Feb. 2019. URL: https://github.com/dlicata335/cart-cube (cit. on pp. 22, 27).

[ACD08] Andreas Abel, Thierry Coquand, and Peter Dybjer. "On the Algebraic Foundation of Proof Assistants for Intuitionistic Type Theory". In: *Functional and Logic Programming*. Ed. by Jacques Garrigue and Manuel V. Hermenegildo. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 3–13. ISBN: 978-3-540-78969-7 (cit. on pp. 34–39).

[AFH17] Carlo Angiuli, Kuen-Bang Hou (Favonia), and Robert Harper. *Computational Higher Type theory III: Univalent Universes and Exact Equality*. 2017. arXiv: 1712.01800.

References II

[AK16a] Thorsten Altenkirch and Ambrus Kaposi. “Normalisation by Evaluation for Dependent Types”. In: *1st International Conference on Formal Structures for Computation and Deduction (FSCD 2016)*. Ed. by Delia Kesner and Brigitte Pientka. Vol. 52. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany: Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2016, 6:1–6:16. ISBN: 978-3-95977-010-1. DOI: `10.4230/LIPIcs.FSCD.2016.6`.

[AK16b] Thorsten Altenkirch and Ambrus Kaposi. “Type Theory in Type Theory Using Quotient Inductive Types”. In: *Proceedings of the 43rd Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*. POPL ’16. St. Petersburg, FL, USA: ACM, 2016, pp. 18–29. ISBN: 978-1-4503-3549-2. DOI: `10.1145/2837614.2837638`.

References III

[AM06] Thorsten Altenkirch and Conor McBride. *Towards Observational Type Theory*. 2006. URL: www.strictlypositive.org/ott.pdf (cit. on pp. 11–16).

[AMB13] Guillaume Allais, Conor McBride, and Pierre Boutillier. "New Equations for Neutral Terms: A Sound and Complete Decision Procedure, Formalized". In: *Proceedings of the 2013 ACM SIGPLAN Workshop on Dependently-typed Programming*. DTP '13. Boston, Massachusetts, USA: ACM, 2013, pp. 13–24. ISBN: 978-1-4503-2384-0. DOI: 10.1145/2502409.2502411.

[AMS07] Thorsten Altenkirch, Conor McBride, and Wouter Swierstra. "Observational Equality, Now!" In: *Proceedings of the 2007 Workshop on Programming Languages Meets Program Verification*. PLPV '07. Freiburg, Germany: ACM, 2007, pp. 57–68. ISBN: 978-1-59593-677-6 (cit. on pp. 11–16).

References IV

[Awo18] Steve Awodey. “Natural models of homotopy type theory”. In: *Mathematical Structures in Computer Science* 28.2 (2018), pp. 241–286. DOI: 10.1017/S0960129516000268 (cit. on pp. 34–39).

[BD08] Alexandre Buisse and Peter Dybjer. “Towards formalizing categorical models of type theory in type theory”. In: *Electronic Notes in Theoretical Computer Science* 196 (2008), pp. 137–151.

[Car86] John Cartmell. “Generalised algebraic theories and contextual categories”. In: *Annals of Pure and Applied Logic* 32 (1986), pp. 209–243. ISSN: 0168-0072 (cit. on pp. 34–39).

[CCD17] Simon Castellan, Pierre Clairambault, and Peter Dybjer. “Undecidability of Equality in the Free Locally Cartesian Closed Category (Extended version)”. In: *Logical Methods in Computer Science* 13.4 (2017).

References V

[CCHM17] Cyril Cohen, Thierry Coquand, Simon Huber, and Anders Mörtberg. "Cubical Type Theory: a constructive interpretation of the univalence axiom". In: *IfCoLog Journal of Logics and their Applications* 4.10 (Nov. 2017), pp. 3127–3169. URL: http://www.collegepublications.co.uk/journals/ifcolog/?00019.

[CFM18] James Chapman, Fredrik Nordvall Forsberg, and Conor McBride. "The Box of Delights (Cubical Observational Type Theory)". Unpublished note. Jan. 2018. URL: https://github.com/msp-strath/platypus/blob/master/January18/doc/CubicalOTT/CubicalOTT.pdf (cit. on p. 22).

References VI

[CHS19] Thierry Coquand, Simon Huber, and Christian Sattler. “Homotopy canonicity for cubical type theory”. In: *Proceedings of the 4th International Conference on Formal Structures for Computation and Deduction (FSCD 2019)*. Ed. by Herman Geuvers. Vol. 131. 2019 (cit. on pp. 34–39).

[Coq17] Thierry Coquand. *Universe of Bishop sets*. Feb. 2017. URL: http://www.cse.chalmers.se/~coquand/bishop.pdf (cit. on p. 22).

[Coq18] Thierry Coquand. *Canonicity and normalization for Dependent Type Theory*. Oct. 2018. arXiv: 1810.09367 (cit. on pp. 34–39).

[Fio02] Marcelo Fiore. “Semantic Analysis of Normalisation by Evaluation for Typed Lambda Calculus”. In: *Proceedings of the 4th ACM SIGPLAN International Conference on Principles and Practice of Declarative Programming*. PPDP ’02. Pittsburgh, PA, USA: ACM, 2002, pp. 26–37. ISBN: 1-58113-528-9. DOI: 10.1145/571157.571161.

References VII

[Hub18] Simon Huber. “Canonicity for Cubical Type Theory”. In: *Journal of Automated Reasoning* (June 13, 2018). ISSN: 1573-0670. DOI: 10.1007/s10817-018-9469-1.

[JT93] Achim Jung and Jerzy Tiuryn. “A new characterization of lambda definability”. In: *Typed Lambda Calculi and Applications*. Ed. by Marc Bezem and Jan Friso Groote. Berlin, Heidelberg: Springer Berlin Heidelberg, 1993, pp. 245–257. ISBN: 978-3-540-47586-6.

[KHS19] Ambrus Kaposi, Simon Huber, and Christian Sattler. “Gluing for type theory”. In: *Proceedings of the 4th International Conference on Formal Structures for Computation and Deduction (FSCD 2019)*. Ed. by Herman Geuvers. Vol. 131. 2019 (cit. on pp. 34–39).

[KKA19] Ambrus Kaposi, András Kovács, and Thorsten Altenkirch. “Constructing Quotient Inductive-inductive Types”. In: *Proc. ACM Program. Lang.* 3.POPL (Jan. 2019), 2:1–2:24. ISSN: 2475-1421. DOI: 10.1145/3290315.

References VIII

[ML75a] Per Martin-Löf. “About Models for Intuitionistic Type Theories and the Notion of Definitional Equality”. In: *Proceedings of the Third Scandinavian Logic Symposium*. Ed. by Stig Kanger. Vol. 82. Studies in Logic and the Foundations of Mathematics. Elsevier, 1975, pp. 81–109.

[ML75b] Per Martin-Löf. “An Intuitionistic Theory of Types: Predicative Part”. In: *Logic Colloquium ’73*. Ed. by H. E. Rose and J. C. Shepherdson. Vol. 80. Studies in Logic and the Foundations of Mathematics. Elsevier, 1975, pp. 73–118. DOI: 10.1016/S0049-237X(08)71945-1.

[MS93] John C. Mitchell and Andre Scedrov. “Notes on sconing and relators”. In: *Computer Science Logic*. Ed. by E. Börger, G. Jäger, H. Kleine Büning, S. Martini, and M. M. Richter. Berlin, Heidelberg: Springer Berlin Heidelberg, 1993, pp. 352–378. ISBN: 978-3-540-47890-4.

References IX

[SAG19] Jonathan Sterling, Carlo Angiuli, and Daniel Gratzer. "Cubical Syntax for Reflection-Free Extensional Equality". In: *Proceedings of the 4th International Conference on Formal Structures for Computation and Deduction (FSCD 2019)*. Ed. by Herman Geuvers. Vol. 131. 2019. DOI: 10.4230/LIPIcs.FSCD.2019.32. arXiv: 1904.08562 (cit. on p. 22).

[Shu06] Michael Shulman. *Scones, Logical Relations, and Parametricity*. Blog. 2006. URL: https://golem.ph.utexas.edu/category/2013/04/scones_logical_relations_and_p.html.

[Shu15] Michael Shulman. "Univalence for inverse diagrams and homotopy canonicity". In: *Mathematical Structures in Computer Science* 25.5 (2015), pp. 1203–1277. DOI: 10.1017/S0960129514000565 (cit. on pp. 34–39).

[SS18] Jonathan Sterling and Bas Spitters. *Normalization by gluing for free λ-theories*. Sept. 2018. arXiv: 1809.08646 [cs.LO].

References X

[Ste18] Jonathan Sterling. *Algebraic Type Theory and Universe Hierarchies.* Dec. 2018. arXiv: `1902.08848 [math.LO]`.

[Str91] Thomas Streicher. *Semantics of Type Theory: Correctness, Completeness, and Independence Results.* Cambridge, MA, USA: Birkhauser Boston Inc., 1991. ISBN: 0-8176-3594-7.

[Str94] Thomas Streicher. *Investigations Into Intensional Type Theory.* Habilitationsschrift, Universität München. 1994.

[Str98] Thomas Streicher. "Categorical intuitions underlying semantic normalisation proofs". In: *Preliminary Proceedings of the APPSEM Workshop on Normalisation by Evaluation*. Ed. by O. Danvy and P. Dybjer. Department of Computer Science, Aarhus University, 1998.

[Uem19] Taichi Uemura. *A General Framework for the Semantics of Type Theory*. 2019. arXiv: `1904.04097` (cit. on pp. 34–39).

References XI

[Voe16] Vladimir Voevodsky. *Mathematical theory of type theories and the initiality conjecture.* Research proposal to the Templeton Foundation for 2016-2019, project description. Apr. 2016. URL: http://www.math.ias.edu/Voevodsky/other/Voevodsky%20Templeton%20proposal.pdf.