Models for Axiomatic Type Theory

Daniël Otten and Matteo Spadetto

Contents

We present Axiomatic Type Theory (type theory without reductions).

Then we compare two semantics for ATT:

- comprehension categories: more traditional and well-studied, closely follows the syntax and intricasies of type theory.
- path categories (Van den Berg, Moerdijk 2017): more concise, taking inspiration from homotopy theory.

Both model a minimal version of ATT: only $=$-types, but weakly because they only specify substitutions up to isomorphism.

However, we can turn comprehension categories into actual models.

Our Contributions

Path categories are equivalent to certain comprehension categories. This allows us to turn path categories into actual models as well.

We introduce a more fine-grained notion: display path categories, and show a similar equivalence.

We obtain the following diagram of 2-categories:

$$\begin{array}{ccc}
\mathrm{PathCat} & \xrightarrow{\sim} & \mathrm{ComprehensionCat}_{\mathrm{Contextual},=,\Sigma_{\beta\eta}} \\
U \uparrow \dashv \downarrow C & & F \uparrow \dashv \downarrow U \\
\mathrm{DisplayPathCat} & \xrightarrow[\sim]{} & \mathrm{ComprehensionCat}_{\mathrm{Contextual},=}
\end{array}$$

Axiomatic Type Theory

Equality

Intensional type theory (ITT) has two notions of equality:

definitional ($\equiv$)	external	reductions	decidable
propositional ($=$)	internal	proofs	undecidable

So, definitional eq forms a decidable fragment of propositional eq:

the fragment that the computer checks for us.

There are reasons to consider larger or smaller fragments:

- larger makes it easier to work inside of the system,
- smaller makes is easier to find models for the system.

Other Fragments

Other fragments come up in practice.

Larger fragments:

- We can have more definitional eq and remain decidable.
- Proof assistants like `Agda` allow you to add reductions.

Smaller fragments:

- Cubical Type Theory only has a propositional β-rule for $=$-types.
- Coinductive types only have a propositional β-rule because the definitional version makes type checking undecidable.

Extremes

We consider the two extremes:

- Extensional type theory (ETT), where every eq is definitional.
- Axiomatic type theory (ATT), without any definitional eq.

These form the min and max of a lattice with ITT in the middle:

$$\begin{array}{c} \text{ETT} \\ \uparrow \\ \text{ITT} \\ \uparrow \\ \text{ATT} \end{array}$$

Complexity and Conservativity

The complexity of type checking:

- ETT: undecidable,
- ITT: nonelementary,
- ATT: quadratic.

So, does ETT prove more than ATT? Yes, namely:

- binder extensionality (bindext),
- uniqueness of identity proofs (uip).

However, these are the only additional things we can prove.

(Winterhalter 2019)

Semantics

We show an equivalence between two semantics for ATT:

comprehension categories and path categories.

Comprehension categories closely follow the syntax.

Path categories simplify $=$-types using the intuition of paths:

We explicitly model formation and introduction, however, instead of elimination and β-axiom, we require that the introduction is an homotopy equivalence.

Minimal Dependent Type Theory

We consider a minimal dependent type theory: only $=$-types.

$$\frac{}{\Gamma, x : A, x' : A \vdash x =_A x' : \mathsf{Type}}(= \text{form})$$

$$\frac{}{\Gamma, x : A \vdash \mathsf{refl}_x : x =_A x}(= \text{intro})$$

$$\frac{\begin{array}{l}\Gamma, x : A, x' : A, p : x =_A x' \vdash C[x, x', p] : \mathsf{Type} \\ \Gamma, x : A \vdash c[x] : C[x, x, \mathsf{refl}_x]\end{array}}{\Gamma, x : A, x' : A, p : x =_A x' \vdash \mathsf{ind}^{=}_{C,c,p} : C[x, x', p]}(= \text{elim})$$

$$\frac{\begin{array}{l}\Gamma, x : A, x' : A, p : x =_A x' \vdash C[x, x', p] : \mathsf{Type} \\ \Gamma, x : A \vdash c[x] : C[x, x, \mathsf{refl}_x]\end{array}}{\Gamma, x : A \vdash \mathsf{ind}^{=}_{C,c,\mathsf{refl}_x} \equiv_{C[x,x,\mathsf{refl}_x]} c[x]}(= \beta\text{-reduction})$$

Minimal Dependent Type Theory

Without Π-types, we have to strengthen the rules:

$$\frac{}{\Gamma, x : A, x' : A \vdash x =_A x' : \mathsf{Type}}(= \mathsf{form})$$

$$\frac{}{\Gamma, x : A \vdash \mathsf{refl}_x : x =_A x}(= \mathsf{intro})$$

$$\frac{\begin{array}{l}\Gamma, x : A, x' : A, p : x =_A x', \Delta[x, x', p] \vdash C[x, x', p] : \mathsf{Type} \\ \Gamma, x : A, \Delta[x, x, \mathsf{refl}_x] \vdash c[x] : C[x, x, \mathsf{refl}_x]\end{array}}{\Gamma, x : A, x' : A, p : x =_A x', \Delta[x, x', p] \vdash \mathsf{ind}^{=}_{C,c,p} : C[x, x', p]}(= \mathsf{elim})$$

$$\frac{\begin{array}{l}\Gamma, x : A, x' : A, p : x =_A x', \Delta[x, x', p] \vdash C[x, x', p] : \mathsf{Type} \\ \Gamma, x : A, \Delta[x, x, \mathsf{refl}_x] \vdash c[x] : C[x, x, \mathsf{refl}_x]\end{array}}{\Gamma, x : A, \Delta[x, x, \mathsf{refl}_x] \vdash \mathsf{ind}^{=}_{C,c,\mathsf{refl}_x} \equiv_{C[x,x,\mathsf{refl}_x]} c[x]}(= \beta\text{-reduction})$$

Minimal Dependent Type Theory

In ATT, we have β-axioms instead of β-reductions:

$$\frac{}{\Gamma, x : A, x' : A \vdash x =_A x' : \mathsf{Type}}\,(= \mathsf{form})$$

$$\frac{}{\Gamma, x : A \vdash \mathsf{refl}_x : x =_A x}\,(= \mathsf{intro})$$

$$\frac{\Gamma, x : A, x' : A, p : x =_A x', \Delta[x, x', p] \vdash C[x, x', p] : \mathsf{Type} \qquad \Gamma, x : A, \Delta[x, x, \mathsf{refl}_x] \vdash c[x] : C[x, x, \mathsf{refl}_x]}{\Gamma, x : A, x' : A, p : x =_A x', \Delta[x, x', p] \vdash \mathsf{ind}^{=}_{C,c,p} : C[x, x', p]}\,(= \mathsf{elim})$$

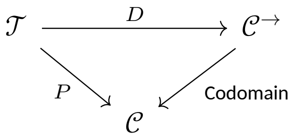
$$\frac{\Gamma, x : A, x' : A, p : x =_A x', \Delta[x, x', p] \vdash C[x, x', p] : \mathsf{Type} \qquad \Gamma, x : A, \Delta[x, x, \mathsf{refl}_x] \vdash c[x] : C[x, x, \mathsf{refl}_x]}{\Gamma, x : A, \Delta[x, x, \mathsf{refl}_x] \vdash \beta^{=}_{C,c,x} : \mathsf{ind}^{=}_{C,c,\mathsf{refl}_x} =_{C[x,x,\mathsf{refl}_x]} c[x]}\,(= \beta\text{-}\mathsf{ax})$$

Comprehension Categories

Comprehension Categories

A comprehension category consists of:

- a category $\mathcal{C}$ of contexts,
- a category $\mathcal{T}$ of types,
- a fibration $P : \mathcal{T} \to \mathcal{C}$ sending every type to its context,
- a full and faithful functor $D : \mathcal{T} \to \mathcal{C}^{\to}$ sending every type A in context Γ to the display map $D_A : \Gamma.A \to \Gamma$.



A term of type A in context Γ is an $a : \Gamma \to \Gamma.A$ s.t. $D_A \circ a = \mathrm{id}_\Gamma$.

Substitution

That $P : \mathcal{T} \to \mathcal{C}$ is a fibration means that we can do substitution:

for a type A in context Γ and a context morphism $\sigma : \Delta \to \Gamma$, there exists a type $A[\sigma]$ in context Δ and a pullback square:

$$\begin{array}{ccc} \Delta.A[\sigma] & \xrightarrow{\sigma.A} & \Gamma.A \\ {\scriptstyle D_{A[\sigma]}}\downarrow & \lrcorner & \downarrow{\scriptstyle D_A} \\ \Delta & \xrightarrow[\sigma]{} & \Gamma \end{array}$$

However, in general, we cannot pick every $A[\sigma]$ such that:

$$A[\mathrm{id}] = A,$$

$$A[\tau \circ \sigma] = A[\sigma][\tau].$$

A comprehension category with compatible choices is called split.

Splitting

A comprehension category has to be split to model type theory.

Luckily, there are ways to split comprehension categories:

$$\begin{array}{c} \mathsf{SplitCompCat} \\ \big\uparrow \dashv \big\downarrow \dashv \big\uparrow \\ \mathsf{CompCat} \end{array}$$

We are mostly interested in the left adjoint:

- (Lumsdaine, Warren 2014): Local Universe Construction.
- (Bocquet 2021): Generic Contexts.

Our equivalence will show that we can split path categories.

Adding Structure

Comprehension categories only model the basic structure of dependent type theory.

Each type former gives more requirements.

In this talk we focus on one type former: $=$-types.

Identity Types

The requirements are translated from the inference rules:

- The formation rule is:

$$\frac{}{\Gamma, x : A, x' : A \vdash x =_A x' : \text{Type}}(= \text{form})$$

  So, we require a type Id_A in context $\Gamma.A.A[D_A]$.

- The introduction rules is:

$$\frac{}{\Gamma, x : A \vdash \text{refl}_x : x =_A x}(= \text{intro})$$

  So, we require a term refl_A of type $\text{Id}_A[\delta_A]$ in context $\Gamma.A$ where $\delta_A : \Gamma.A \to \Gamma.A.A[D_A]$ duplicates the last variable.

- We omit the elimination and β rules ($\text{ind}^{=}_{A,C,c}$ and $\beta^{=}_{A,C,c}$).

Stability

In addition, we need choices that are stable under substitution:

$$\begin{aligned} \mathsf{Id}_A[\sigma] &= \mathsf{Id}_{A[\sigma]}, \\ \mathsf{refl}_A[\sigma] &= \mathsf{refl}_{A[\sigma]}, \\ \mathsf{ind}^{=}_{A,C,c}[\sigma] &= \mathsf{ind}^{=}_{A[\sigma],C[\sigma],c[\sigma]}, \\ \beta^{=}_{A,C,c}[\sigma] &= \beta^{=}_{A[\sigma],C[\sigma],c[\sigma]}. \end{aligned}$$

Fortunately, when we split the comprehension category, we also turn weakly stable structure into stable structure.

We have weakly stable $=$-types if for every type A there exist an $=$-type $(\mathsf{Id}_A, \mathsf{refl}_A, \mathsf{ind}^{=}_A, \beta^{=}_A)$ s.t. for every σ:

there exist i and b s.t. $(\mathsf{Id}_A[\sigma], \mathsf{refl}_A[\sigma], i, b)$ is an $=$-type.

Path Categories

Path Categories

A path category is a category $\mathcal{C}$ with two classes of maps:

- fibrations: closed under pullbacks and compositions,
- (weak) equivalences, satisfying 2-out-of-6: if we have

$$A \xrightarrow{f} B \xrightarrow{g} C \xrightarrow{h} D$$

  where $g \circ f$ and $h \circ g$ are weak equivalences,
  then f, g, h, and $h \circ g \circ h$ are weak equivalences.

If a map is both then we call it an acyclic fibration:

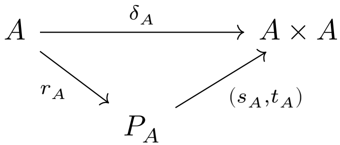
- every isomorphism is an acyclic fibration,
- pullbacks of acyclic fibrations are acyclic fibrations,
- every acyclic fibration has a section.

$\mathcal{C}$ has a terminal object 1 and every map $A \to 1$ is a fibration.

Path Objects

Lastly, a path category has a path object for every object A:

- a factorisation of the diagonal $\delta_A = (\mathrm{id}_A, \mathrm{id}_A)$:



  into a weak equivalence r_A followed by a fibration (s_A, t_A).

We can use path objects to show that every morphism factors as a weak equivalence followed by a fibration. (mapping path space)

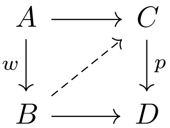
Homotopy Theory

We call two maps $f, g : A \to B$ homotopic, written $f \simeq g$, if there exists a map $h : A \to P_B$ such that $s_B \circ h = f$ and $t_B \circ h = g$.

We call $f : A \to B$ an homotopy equivalence, if there exists a map $g : B \to A$ such that $g \circ f \simeq \mathrm{id}_A$ and $f \circ g \simeq \mathrm{id}_B$.

The homotopy equivalences are precisely the weak equivalences.

In addition, we have a lifting theorem: for a commutative square



where w is an equivalence and p is a fibration, there is a $d : B \to C$ unique up to homotopy such that the lower triangle commutes and the upper triangle commutes up to homotopy.

Equivalences

From a Path Category to a Comprehension Category

Suppose that we have a path category $\mathcal{C}$.

We obtain a comprehension category:

- $\mathcal{C}$ forms the category of contexts,
- the full $\mathcal{C}^{\text{fib}} \subseteq \mathcal{C}^{\to}$ of fibrations forms the category of types,
- the codomain functor $\mathcal{C}^{\text{fib}} \to \mathcal{C}$ sends a type to its context.

We will show that it has additional structure:

- weakly stable $=$-types,
- weakly stable Σ-types with β and η reductions,
- contextuality (contexts are finite).

Weakly Stable =-Types

For a type A we take

$$\mathsf{Id}_A := (s_A, t_A) : P_A \to A \times A, \qquad \text{(formation)}$$

$$\mathsf{refl}_A := r_A : A \to PA. \qquad \text{(introduction)}$$

The elimination and β-axiom follow from our lifting theorem and the fact that r_A is an equivalence.

We get weak stability because we can show that path objects are preserved by taking pullbacks.

Weakly Stable Σ-Types with β and η

The intuitive reason that we obtain Σ-types is that path categories do not distinguish between a single extension $\Gamma.A$ of Γ and arbitrary extensions $\Gamma.A_0.\ldots.A_{n-1}$.

The requirements on a comprehension category can be simplified to:

> for every type $\Gamma.A.B$ we have a type $\Sigma_A B$ in context Γ and an isomorphism $\Gamma.A.B \cong \Gamma.\Sigma_A B$ making the following commute:

$$
\begin{array}{ccc}
\Gamma.A.B & \overset{\sim}{=\!=\!=} & \Gamma.\Sigma_A B \\
\downarrow & & \downarrow \\
\Gamma.A & \longrightarrow & \Gamma
\end{array}
$$

Holds in path categories: compositions of fibrations are fibrations.

Contextuality

A comprehension category is called contextual if the category of contexts has a terminal object 1 and for every context Γ there exist:

- a type A_0 in context 1,
- a type A_1 in context $1.A_0$,
- a type A_2 in context $1.A_0.A_1$,

  $\vdots$

- a type A_{n-1} in context $1.A_0.\ldots.A_{n-2}$,

such that $\Gamma \cong 1.A_0.\ldots.A_{n-1}$.

Holds in path categories: every map $\Gamma \to 1$ is a fibration.

From a Comprehension Category to a Path Category

Suppose that we have a comprehension category $\mathcal{C}$ with $=$, $\Sigma_{\beta,\eta}$, and contextuality.

Then we can turn $\mathcal{C}$ into a path category by taking:

- the fibrations as the class of display maps and isomorphisms closed under composition,
- the weak equivalences as the homotopy equivalences.

Then $\mathcal{C}$ has path objects because it has $=$-types.

Display Path Categories

In a display path category we distinguish $\Gamma.A$ and $\Gamma.A_0.\dots.A_{n-1}$.

Instead of fibrations we use display maps as a primitive notion.

Fibrations are compositions of display maps and isomorphisms.

In addition, we replace path objects for objects Γ with a seemingly weaker notion: path objects for display maps $A \to \Gamma$.

This is sufficient: we can use a lifting theorem and transport to inductively construct path objects for objects.

Equivalence

We obtain the following diagram of 2-categories:

$$
\begin{array}{ccc}
\mathrm{PathCat} & \xrightarrow{\sim} & \mathrm{ComprehensionCat}_{\mathrm{Contextual},=,\Sigma_{\beta\eta}} \\
U \big\uparrow \;\dashv\; \big\downarrow C & & F \big\uparrow \;\dashv\; \big\downarrow U \\
\mathrm{DisplayPathCat} & \xrightarrow[\sim]{} & \mathrm{ComprehensionCat}_{\mathrm{Contextual},=}
\end{array}
$$

Here the U's are forgetful, F is a free, and C is a cofree.

We end this talk with some open questions:

- Can we simplify other type formers as we did with $=$-types?
- In particular, are propositional Σ-types and Π-types homotopical weakenings of left and right adjoints of the pullback functor.

References

Benno van den Berg and Ieke Moerdijk (2017): Exact completion of path categories.

Benno van den Berg (2018): Path categories and propositional identity types.

Rafaël Bocquet (2020): Coherence of strict equalities in dependent type theories.

Rafaël Bocquet (2021): Strictification of weakly stable type-theoretic structures using generic contexts.

Martin Hofmann (1995): On the interpretation of type theory in locally Cartesian closed categories.

Peter Lumsdaine and Michael Warren (2014): The local universes model, an overlooked coherence construction for dependent type theories.

Nicolas Oury (2005): Extensionality in the calculus of constructions.

Matteo Spadetto (2023): A conservativity result for homotopy elementary types in dependent type theory.

Theo Winterhalter (2019): Formalisation and meta-theory of type theory.