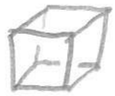
Models of HoTT

(simplicial & cubical)

Christian Sattler

EPIT 2021

0. Introduction & Overview
1. What is a model of type theory?
2. Presheaf models

——— Exercises

3. The simplicial model of HoTT
4. Cubical models and open questions

——— Exercises

0. Introduction & overview

Model of type theory:

- Interpretation of language that turns judgmental equality into actual equality. Slogan

There are many frameworks for models, sometimes equivalent, other times differing in some aspects:

- collection of types over context Γ seen as set or groupoid or category.
- split vs. non-split
- contextual / democratic
- algebraic or categorical notion
  - algebraic: category of models
  - categorical: bicategory of models
- terms or context projections (display maps) as primitive

We will work with categories with families.

Model of HoTT: Just a model of type theory with elements witnessing function extensionality + univalence.

History of models of HoTT:

Set model
- univalent universe of propositions

Groupoid model — Hofmann-Streicher
- univalence for (strict) sets

Simplicial model — Voevodsky
- full univalence
- non-constructive
- models spaces/∞-groupoids

⋮

Inverse diagram models — Shulman

⋮

Model for injective fibrations in simplicial presheaves — Shulman

Cubical models — Coquand + coworkers
- full univalence
- often does not model spaces/∞-groupoids
- computational

- BCH model
- connection-based models
  - unknown if can model spaces
- Cartesian-based models
  - can model spaces (equivariant model)

HoTT and ∞-toposes

We want to "interpret" HoTT in ∞-toposes.

But models of HoTT are not (directly) ∞-categories!

↳ They are 1-categories with homotopical structure, presenting an ∞-category.

Like model categories!

$$\begin{tikzcd} \text{models of HoTT} \arrow[r, "\text{present}^*"] & \infty\text{-toposes} \\ \text{HoTT} \arrow[u, "\text{interprets into}"'] & \end{tikzcd}$$

So to interpret HoTT into a given ∞-topos $\mathcal{X}$, first need to find suitable presentation of $\mathcal{X}$.

Shulman found such a presentation for every ∞-topos!

So far, this is a classical (non-constructive) story.

1. What is a model of type theory?

Don't want to bother with modelling "named variables".

↳ Abstract over contexts as lists of hypotheses $x_1 : A_1, \ldots, x_n : A_n$ by modelling contexts + substitutions as a category $\mathcal{C}$.

$$\Delta \xrightarrow{\ \sigma\ } \Gamma$$

$$[y_1 : B_1, \ldots, y_m : B_m] \qquad [x_1 : A_1, \ldots, x_n : A_n]$$

$$\begin{aligned} x_1 &= t_1[y_1, \ldots, y_m] \\ &\vdots \\ x_n &= t_n[y_1, \ldots, y_m] \end{aligned}$$

Every context Γ has a set of types $Ty(\Gamma)$.
We can substitute types: $A \in Ty(\Gamma) \rightsquigarrow A[\sigma] \in Ty(\Delta)$.
↳ Ty is a presheaf over $\mathcal{C}$.

Same for terms Tm, but they additionally depend on a type.

Excursion: Presheaves and discrete fibrations

Def Presheaves over category $\mathcal{C}$ are functors $\mathcal{C}^{op} \to Set$.

Notation: $\widehat{\mathcal{C}} = \mathrm{Presheaf}(\mathcal{C})$

Traditional definition

Grothendieck construction*

*restricted version

$$\int : [\mathcal{C}^{op}, Set] \xrightarrow[\text{ff}]{\text{fully faithful}} \{\text{category over } \mathcal{C}\}$$

$$F \longmapsto \begin{array}{c} \int F \\ \downarrow \\ \mathcal{C} \end{array}$$

"category of elements"

$$\begin{array}{c} \mathcal{E} \\ \downarrow \\ \mathcal{C} \end{array}$$

We regard $\mathcal{E}$ as "displayed" over $\mathcal{C}$:

- $\mathcal{E}_0(A)$ set for $A \in \mathcal{C}_0$
- $\mathcal{E}_1(x, y, f)$ set for $x \in \mathcal{E}_0(A)$, $y \in \mathcal{E}_0(B)$, $f \in \mathcal{C}_1(A, B)$

$(\int F)_0(A) = F(A)$

$(\int F)_1(x, y, f) = [F(f)(y) = x]$

That way, we can strictly reindex categories over a base.

The essential image of $\int$ are the discrete fibrations:

$$\begin{array}{ccc} ? \overset{u}{\dashrightarrow} y & & \mathcal{E} \\ \downarrow & & \\ A \xrightarrow{f} B & & \mathcal{C} \end{array}$$

$\forall f \in \mathcal{C}_1(A, B)$, $y \in \mathcal{E}_0(B)$ have unique lift $x \in \mathcal{E}_0(A)$, $u \in \mathcal{E}_1(f, x, y)$.

Often, discrete fibrations are a better model for presheaves $\widehat{\mathcal{C}}$, than $\mathcal{C}^{op} \to Set$!

We use both interchangeably.

Categories with families

Definition

A <u>category with families</u> consists of: cwf

- Category $\mathcal{C}$
- Terminal object $1 \in \mathcal{C}$
- $\mathrm{Ty} \in \widehat{\mathcal{C}}$
- $\mathrm{Tm} \in \widehat{\int \mathrm{Ty}}$
- <u>Context extension:</u>
  for $\Gamma \in \mathcal{C}$ and $A \in \mathrm{Ty}(\Gamma)$, a representation of:

  $$(\mathcal{C} \downarrow \Gamma)^{op} \longrightarrow \mathrm{Set}$$
  $$\Delta \xrightarrow{\sigma} \Gamma \longmapsto \mathrm{Tm}(\Delta, A[\sigma])$$

  $\Leftrightarrow$ a terminal object $(\Gamma.A, p_A, q_A)$ in the category of tuples (Δ, σ, t) where:
  - $\Delta \in \mathcal{C}$
  - $\Delta \xrightarrow{\sigma} \Gamma$
  - $t \in \mathrm{Tm}(\Delta, A[\sigma])$

$\square$

$$\begin{tikzcd} \Gamma.A \arrow[d] \\ \Gamma \end{tikzcd}$$

$\Gamma.A$ context extension

context projection / display map for A

$q_A \in \mathrm{Tm}(\Gamma.A, A[p_A])$
generic term, 'last variable'

$$\begin{tikzcd} \Delta \arrow[rr, dashed, "{\langle \sigma, t \rangle}"] \arrow[dr, "\sigma"'] & & \Gamma.A \arrow[dl, "p_A"] \\ & \Gamma & \end{tikzcd}$$

For algebraic notion of model:
- context extension is structure, strictly preserved by morphisms of models
- obtain <u>category</u> of models
- "Syntax" $=_{\mathrm{def}}$ initial object

Context extension is really a property of the presheaf Tm!

Alternative definition (cwf)

discrete fibrations

$$\begin{tikzcd} \mathrm{Tm} \arrow[d] \arrow[d, phantom, shift left=2, "\dashv"] \\ \mathrm{Ty} \arrow[u, bend right=60, "\text{context extension}"'] \arrow[d] \\ \mathcal{C} \arrow[d] \arrow[d, phantom, shift left=2, "\dashv"] \\ 1 \arrow[u, bend right=60, "\text{terminal object}"'] \end{tikzcd}$$

Have right adjoints to indicated projections

$\square$

Notation for adjunctions:

$$\begin{tikzcd} \mathcal{C} \arrow[r, bend right, "\text{right adjoint}"'] \arrow[r, phantom, "\perp"] & \mathcal{D} \arrow[l, bend right, "\text{left adjoint}"'] \end{tikzcd}$$

Exercise: show that this means the same as the first definition.

Type formers

Dependent sums

For $\Gamma \in \mathcal{C}$, $A \in Ty(\Gamma)$, $B \in Ty(\Gamma . A)$, have $\Sigma(A,B) \in Ty(\Gamma)$ with a bijection

$$\begin{tikzcd} Tm(\Gamma, \Sigma(A,B)) \arrow[r, bend right, "\pi_1, \pi_2"'] \arrow[loop below, phantom, "\eta"] & \left\{ \left(\begin{array}{l} a \in Tm(\Gamma, A), \\ b \in Tm(\Gamma, B[\langle id, a \rangle]) \end{array} \right) \right\} \arrow[l, bend right, "pair"'] \arrow[loop below, phantom, "\beta"] \end{tikzcd},$$

natural in Γ.

Dependent products

For $\Gamma \in \mathcal{C}$, $A \in Ty(\Gamma)$, $B \in Ty(\Gamma . A)$, have $\Pi(A,B) \in Ty(\Gamma)$ with a bijection

$$\begin{tikzcd} Tm(\Gamma, \Pi(A,B)) \arrow[r, bend right, "app"'] \arrow[loop below, phantom, "\eta"] & Tm(\Gamma . A, B) \arrow[l, bend right, "\lambda"'] \arrow[loop below, phantom, "\beta"] \end{tikzcd},$$

natural in Γ.

Exercise: identity types

Adjoint perspective on Σ / Π

$$\begin{tikzcd}[column sep=small] \Gamma.\Sigma(A,B) \simeq \Gamma.A.B \arrow[dr] \arrow[rr, phantom] & & \\ & \Gamma & \end{tikzcd} \qquad \begin{tikzcd} \Gamma.A.B \arrow[d] \\ \Gamma.A \arrow[d] \\ \Gamma \end{tikzcd}$$

$$\begin{tikzcd} \Gamma.\Sigma(A,B) \arrow[rr, dashed] \arrow[dr] & & \Delta \arrow[dl, "\sigma"] \\ & \Gamma & \end{tikzcd} \underset{\text{natural}}{\simeq} \begin{tikzcd} \Gamma.A.B \arrow[rr, dashed] \arrow[dr] & & \Delta.A[\sigma] \arrow[dl] \\ & \Gamma.A & \end{tikzcd}$$

$$\begin{tikzcd} \Delta \arrow[rr, dashed] \arrow[dr] & & \Gamma.\Pi(A,B) \arrow[dl] \\ & \Gamma & \end{tikzcd} \underset{\text{natural}}{\simeq} \begin{tikzcd} \Delta.A[\sigma] \arrow[rr, dashed] \arrow[dr] & & \Gamma.A.B \arrow[dl] \\ & \Gamma.A & \end{tikzcd}$$

$$\Sigma_A \dashv p_A^* \dashv \Pi_A$$

after switching to "display map" presentation.

Caveat: Σ_A / Π_A only defined on types / display maps!

$$\mathrm{Tm}(\Gamma, A) \simeq \left\{ \begin{tikzcd} \Gamma.A \arrow[d] \\ \Gamma \arrow[u, dashed, bend left] \end{tikzcd} \text{ section} \right\}$$

Lifting perspective on Id

$$\begin{tikzcd} \Gamma.A \arrow[r, "d"] \arrow[d, "\mathrm{refl}_A"'] & \Gamma.A.A.\mathrm{Id}_A.C \arrow[d, "p_C"] \\ \Gamma.A.A.\mathrm{Id}_A \arrow[r, equal] \arrow[ur, dashed, "J_{C,d}"'] & \Gamma.A.A.\mathrm{Id}_A \end{tikzcd}$$

$C \in \mathrm{Ty}(\Gamma.A.A.\mathrm{Id}_A)$
motive

$d \in \mathrm{Tm}(\Gamma.A, \cancel{C[\langle p_A, q}$ $C[\mathrm{refl}_A])$
witness

Universe

$\mathcal{U} \in Ty(\Gamma)$, $El \in Ty(\Gamma.\mathcal{U})$
natural in Γ

Equivalently:
$\mathcal{U} \in Ty(1)$, $El \in Ty(1.\mathcal{U})$

$A \in Tm(\Gamma, \mathcal{U}) \longmapsto El[\langle id_\Gamma, A\rangle]$
decodes elements of $\mathcal{U}$ into types.

$Ty_{\mathcal{U}} \in \widehat{\mathcal{C}}$ $\qquad Tm_{\mathcal{U}} \in \widehat{\int Ty_{\mathcal{U}}}$

$Ty_{\mathcal{U}}(\Gamma) = Tm(\Gamma, \mathcal{U})$ $\qquad Tm_{\mathcal{U}}(A) = Tm(\Gamma, El[\langle id_\Gamma, A\rangle])$

defines <u>cwf structure on $\mathcal{C}$ induced by $\mathcal{U}$.</u>

$\mathcal{U}$ closed under type formers means:
(1) induced cwf structure has type formers, $\qquad Ty_{\mathcal{U}}$
(2) map of cwf structures preserves type formers. $\qquad Ty_{\mathcal{U}} \to Ty$

Can abstractly define cumulative hierachy using morphisms of cwf structures.

Axioms (FunExt + univalence)

Witnessed by a term of the type for the axiom (naturally in the context).

Examples

- Set as cwf:

  $Ty(\Gamma) = \Gamma \to Set$

  $Tm(\Gamma, A) = (x : \Gamma) \to A(x)$

- Groupoids as cwf:

  $Ty(\Gamma) = \Gamma \to Gpd$

  $$Tm(\Gamma, A) = \left\{ \begin{tikzcd} \int A \arrow[d] \\ \Gamma \arrow[u, dashed, bend left] \end{tikzcd} \right\}$$

  Split fibration model.
  There is also the (cloven) fibration model: pseudofunctors $\Gamma \to Gpd$.

- Cube category $\square$:

  $Ty(X) = 1$

  $Tm(X, *) = \square(X, I)$

  for interval I generating $\square$

- For a cwf $\mathcal{C}$, the core $\mathcal{C}_{fib}$ has
  - objects $Ty(1)$
  - maps $1.A \dashrightarrow 1.B$
  - $Ty_{fib}(A) = Ty(1.A)$

- For a cwf $\mathcal{C}$, and $\Gamma \in \mathcal{C}$, the slice $\mathcal{C} \downarrow \Gamma$ inherits a cwf structure.

...

2. Presheaf models

Goal: For category $\mathcal{C}$, make presheaves $\widehat{\mathcal{C}}$ into a model of "extensional" type theory
equality reflection

Extraordinarily useful:

- Can use ETT as internal language for presheaves.
- Can use presheaves to express naturality of type formers.
  natural models
  HOAS
  LF
- Bootstrapping basis for all known semantic models of HoTT.

Let $\Gamma \in \widehat{\mathcal{C}}$.

In displayed form

$\mathrm{Ty}(\Gamma) = \{\text{discrete fibration over } \textstyle\int \Gamma\}$

$A \in \mathrm{Ty}(\Gamma)$ written
$$\begin{array}{c} A \\ \downarrow \\ \Gamma \end{array} \quad \text{or} \quad \begin{array}{c} \Gamma.A \\ \downarrow \\ \Gamma \end{array}$$

$\mathrm{Tm}(\Gamma, A) = \{\text{section of } A\}$

Context extension given by taking the "total space"/ dependent sum of presheaves.

Simplification

Build a cwf on categories with discrete fibrations as types. The presheaf model over C is the slice model over C.

Σ Given by composition of discrete fibrations.

Id + equality reflection

Given by levelwise equality.

For Π and U, we revisit the Grothendieck construction:

$$\begin{tikzcd}
 & \{\text{category over } \mathcal{C}\} \arrow[dl, bend right] \arrow[dl, bend left] \arrow[dd, shift left, "\text{forget}"] \\
{[\mathcal{C}^{op}, \text{Set}]} \arrow[ur, "\text{ff}"] \arrow[dr, "\text{ff}"'] & \\
 & \{\text{graph over } \mathcal{C}\} \arrow[ul, bend left] \arrow[ul, bend right] \arrow[uu, shift left, "\dashv"]
\end{tikzcd}$$

$\mathcal{E}_0, \mathcal{E}_1$ with no operations

Π-types are easy in graphs.

presheaves over $[0] \rightrightarrows [1]$

Compute Π-types in $\widehat{\mathcal{C}}$ by moving to graphs over $\mathcal{C}$, doing the Π-type there, and then applying the right adjoint to go back to presheaves.

Universe in categories over $\mathcal{C}$: just Set!

$$\begin{array}{c} \mathcal{C} \times \text{Set} \\ \downarrow \\ \mathcal{C} \end{array}$$

Transport it to $\widehat{\mathcal{C}}$ using the right adjoint

Exercises: work out the details.

HOAS

Higher-order abstract syntax

For cwf $\mathcal{C}$, can describe type formers using the internal language of $\hat{\mathcal{C}}$.

$$\varepsilon \vdash Ty \text{ type}$$

$$A : Ty \vdash Tm(A) \text{ type}$$

Π-types

Given $A : Ty$, $B : Tm(A) \to Ty$,
have $\Pi(A, B) : Ty$ and

$$Tm(\Pi(A, B)) \simeq \prod_{a : Tm(A)} Tm(B(a)).$$

Don't have to mention context Γ anymore!

2. Presheaf models

Goal: For category $\mathcal{C}$, make presheaves $\hat{\mathcal{C}}$ into a model of "extensional" type theory
equality reflection

Extraordinarily useful:
- Can use ETT as internal language for presheaves.
- Can use presheaves to express naturality of type formers. | natural models, HOAS, LF
- Bootstrapping basis for <u>all known semantic models</u> of HoTT.

Let $\Gamma \in \hat{\mathcal{C}}$.

$Ty(\Gamma) = \{\text{discrete fibration over } \int\Gamma\}$
In displayed form

$A \in Ty(\Gamma)$ written $\begin{array}{c} A \\ \downarrow \\ \Gamma \end{array}$ or $\begin{array}{c} \Gamma.A \\ \downarrow \\ \Gamma \end{array}$

$Tm(\Gamma, A) = \{\text{section of } A\}$

Context extension given by taking the "total space"/ dependent sum of presheaves.

<u>Simplification</u>

Build a cwf on <u>categories</u> with discrete fibrations as types. The presheaf model over $\mathcal{C}$ is the <u>slice model</u> over $\mathcal{C}$.

$\underline{\Sigma}$ Given by composition of discrete fibrations.

$\underline{\mathrm{Id}}$ + equality reflection

Given by levelwise equality.

For Π and $\mathcal{U}$, we revisit the Grothendieck construction:

$$
\begin{tikzcd}
 & \{\text{category over } \mathcal{C}\} \arrow[dl, bend right=40] \arrow[dl, bend left=20] \arrow[d, "\text{forget}"] \arrow[d, phantom, "\dashv"', shift right=3] \\
{[\mathcal{C}^{op}, \mathrm{Set}]} \arrow[ur, "\text{ff}"] \arrow[ur, phantom, "\perp", shift left=3] \arrow[ur, phantom, "\perp"', shift right=3] \arrow[dr, "\text{ff}"'] \arrow[dr, phantom, "\perp", shift left=3] \arrow[dr, phantom, "\perp"', shift right=3] & \\
 & \{\text{graph over } \mathcal{C}\} \arrow[ul, bend left=40] \arrow[ul, bend right=20] \arrow[u, bend left=40]
\end{tikzcd}
$$

$\mathcal{E}_0, \mathcal{E}_1$ with no operations

Π-types are easy in graphs.

presheaves over $[0] \rightrightarrows [1]$

Compute Π-types in $\widehat{\mathcal{C}}$ by moving to graphs over $\mathcal{C}$, doing the Π-type there, and then applying the right adjoint to go back to presheaves.

Universe in categories over $\mathcal{C}$: just Set! $\quad \begin{array}{c} \mathcal{C} \times \mathrm{Set} \\ \downarrow \\ \mathcal{C} \end{array}$

Transport it to $\widehat{\mathcal{C}}$ using the right adjoint

Exercises: work out the details.

HOAS

Higher-order abstract syntax

For cwf $\mathcal{C}$, can describe type formers using the internal language of $\hat{\mathcal{C}}$.

$$\varepsilon \vdash \mathrm{Ty} \text{ type}$$

$$A : \mathrm{Ty} \vdash \mathrm{Tm}(A) \text{ type}$$

Π-types

Given $A : \mathrm{Ty}$, $B : \mathrm{Tm}(A) \to \mathrm{Ty}$, have $\Pi(A, B) : \mathrm{Ty}$ and

$$\mathrm{Tm}(\Pi(A, B)) \simeq \prod_{a : \mathrm{Tm}(A)} \mathrm{Tm}(B(a)).$$

Don't have to mention context Γ anymore!