Higher mathematics via type theory

Paige Randall North

University of Pennsylvania

15 March 2022

Outline

1. Background on type theory and univalent foundations
2. The univalence principle
3. Directed homotopy type theory
4. Summary

Outline

1 Background on type theory and univalent foundations

2 The univalence principle

3 Directed homotopy type theory

4 Summary

Type theory's beginnings

- 1970s: Martin-Löf introduces his type theory
  - As a self-sufficient foundation of mathematics
  - Well-suited for machine verification

Type theory's beginnings

- 1970s: Martin-Löf introduces his type theory
  - As a **self-sufficient** foundation of mathematics
  - Well-suited for machine verification

Type theory's beginnings

- 1970s: Martin-Löf introduces his type theory
  - As a **self-sufficient** foundation of mathematics
  - Well-suited for machine verification

Classical mathematics:

Categorical notions

Set theory

First-order logic

Type theory's beginnings

- 1970s: Martin-Löf introduces his type theory
  - As a **self-sufficient** foundation of mathematics
  - Well-suited for machine verification

Classical mathematics:

Categorical notions

Set theory

First-order logic

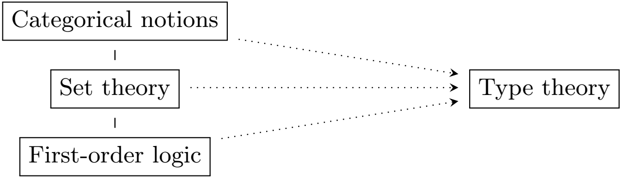
Mathematics à la Martin-Löf:

Type theory

Type theory's beginnings

- 1970s: Martin-Löf introduces his type theory
  - As a **self-sufficient** foundation of mathematics
  - Well-suited for machine verification

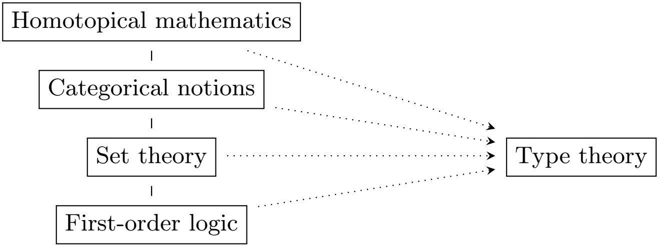
Classical mathematics: **Mathematics à la Martin-Löf:**



Type theory's beginnings

- 1970s: Martin-Löf introduces his type theory
  - As a **self-sufficient** foundation of mathematics
  - Well-suited for machine verification

Classical mathematics: **Mathematics à la Martin-Löf:**

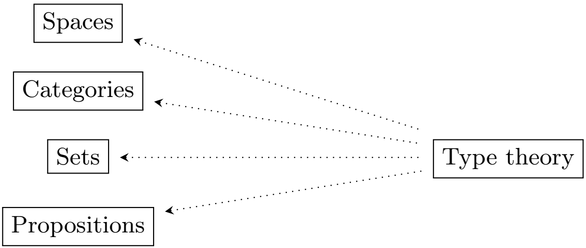


What does type theory look like?

- In mathematics, statements look like the following:
  - Consider a natural number n. The sum $n + n$ is even.
  - Consider a space X. The cone on X is contractible.
- In type theory, we write this as
  - $n : \mathbb{N} \vdash e(n) : \mathsf{isEven}(n + n)$
  - $X : \mathsf{Spaces} \vdash c(X) : \mathsf{isContr}(CX)$
- Type theory provides:
  - natural numbers type $\mathbb{N}$
  - product type $A \times B$
  - sum type $A + B$
  - function type $A \to B$
  - a universe type Type
  - a type (!) of equalities $a =_A b$
  - etc

Interpretations of type theory into classical mathematics

Classical mathematics: **Mathematics à la Martin-Löf:**



Interpretations of type theory into classical mathematics

Types	**Terms**	**Product**	**Equality**
Propositions	proofs	$\wedge$	$=$
Sets	elements	$\times$	$=$
Categories	objects	$\times$	$\cong$
Spaces	points	$\times$	$\sim$

Interpretations of type theory into classical mathematics

Types	**Terms**	**Product**	**Equality**
Propositions	proofs	$\wedge$	$=$
Sets	elements	$\times$	$=$
Categories	objects	$\times$	$\cong$
Spaces	points	$\times$	$\sim$

Interpretations of type theory into classical mathematics

Types	Terms	Product	Equality
Propositions	proofs	$\wedge$	$=$
Sets	elements	$\times$	$=$
Categories	objects	$\times$	$\cong$
Spaces	points	$\times$	$\sim$

- Everything in type theory respects equality
  - This is the definition of equality in type theory (roughly)

Interpretations of type theory into classical mathematics

Types	**Terms**	**Product**	**Equality**
Propositions	proofs	$\wedge$	$=$
Sets	elements	$\times$	$=$
Categories	objects	$\times$	$\cong$
Spaces	points	$\times$	$\sim$

- Everything in type theory respects equality
  - This is the definition of equality in type theory (roughly)
- Mathematics in type theory
  $+$ Interpretation into Xs where equality is interpreted by Y
  $\rightsquigarrow$ Mathematics in Xs up to Y

Interpretations of type theory into classical mathematics

Types	Terms	Product	Equality
Propositions	proofs	$\wedge$	$=$
Sets	elements	$\times$	$=$
Categories	objects	$\times$	$\cong$
Spaces	points	$\times$	$\sim$

- Everything in type theory respects equality
  - This is the definition of equality in type theory (roughly)
- Mathematics in type theory
  $+$ Interpretation into Xs where equality is interpreted by Y
  $\rightsquigarrow$ Mathematics in Xs up to Y
  - Our goal in the *Univalence Principle* project[1] is to develop (higher) categorical mathematics in type theory.

[1] jww Ahrens, Shulman, Tsementzis

Interpretations of type theory into classical mathematics

Types	Terms	Product	Equality
Propositions	proofs	$\wedge$	$=$
Sets	elements	$\times$	$=$
Categories	objects	$\times$	$\cong$
Spaces	points	$\times$	$\sim$

- Everything in type theory respects equality
  - This is the definition of equality in type theory (roughly)
- Mathematics in type theory
  $+$ Interpretation into Xs where equality is interpreted by Y
  $\rightsquigarrow$ Mathematics in Xs up to Y
  - Our goal in the *Univalence Principle* project[1] is to develop (higher) categorical mathematics in type theory.
  - My goal in my *Directed homotopy type theory* project is to modify type theory so that there is an interpretation of categories where (directed) equality is interpreted by *morphisms*

[1] jww Ahrens, Shulman, Tsementzis

Interpretations of type theory into classical mathematics

Types	**Terms**	**Product**	**Equality**
Propositions	proofs	$\wedge$	$=$
Sets	elements	$\times$	$=$
Groupoids	objects	$\times$	$\cong$
Spaces	points	$\times$	$\sim$

- Everything in type theory respects equality
  - This is the definition of equality in type theory (roughly)
- Mathematics in type theory
  $+$ Interpretation into Xs where equality is interpreted by Y
  $\rightsquigarrow$ Mathematics in Xs up to Y
  - Our goal in the *Univalence Principle* project[1] is to develop (higher) categorical mathematics in type theory.
  - My goal in my *Directed homotopy type theory* project is to modify type theory so that there is an interpretation of categories where (directed) equality is interpreted by *morphisms*

[1] jww Ahrens, Shulman, Tsementzis

Univalence

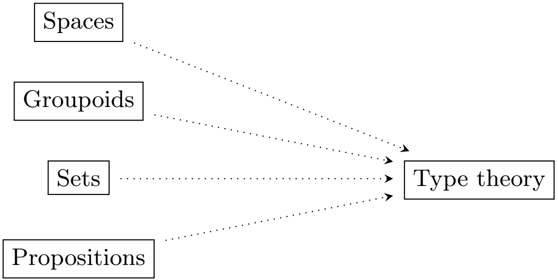
- We've seen that equality in type theory can be interpreted as notions weaker than classical equality (e.g. isomorphism, paths).
- Voevodsky imported weakness for equality from the interpretation in spaces into type theory by imposing the *Univalence Axiom* (UA):

The canonical function $(A =_U B) \to (A \simeq B)$ is an equivalence of types, for any types A and B.

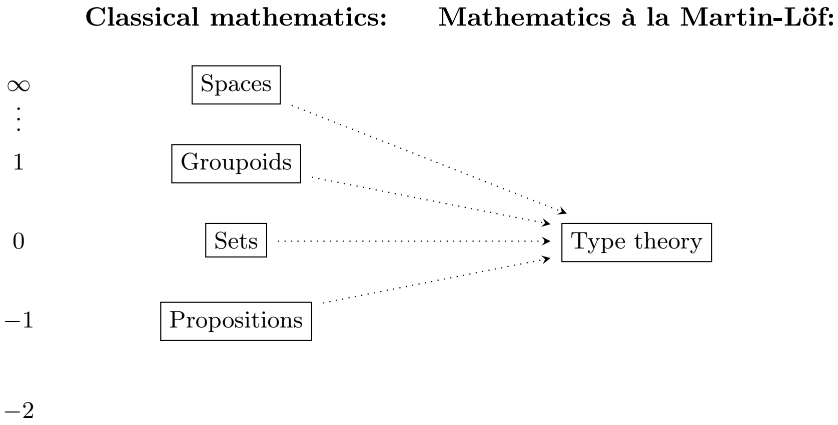
- UA is validated by the interpretation into spaces, but not into propositions, sets, or groupoids.
- Instead we **internalize** these notions.

Internalization of classical mathematics into type theory

Classical mathematics: **Mathematics à la Martin-Löf:**



Internalization of classical mathematics into type theory



Univalent mathematics

- If two types A, B are propositions,

$$(A \simeq B) \simeq (A \leftrightarrow B)$$

Univalent mathematics

- If two types A, B are propositions,

$$(A =_{\mathrm{Prop}} B) \overset{UA}{\simeq} (A \simeq B) \simeq (A \leftrightarrow B)$$

  so everything respects bi-implication of propositions.

Univalent mathematics

- If two types A, B are propositions,

$$(A =_{\mathrm{Prop}} B) \overset{UA}{\simeq} (A \simeq B) \simeq (A \leftrightarrow B)$$

  so everything respects bi-implication of propositions.
- If A, B are sets,

$$(A =_{\mathrm{Set}} B) \overset{UA}{\simeq} (A \simeq B) \simeq (A \cong B)$$

  so everything respects bijection of sets.

Univalent mathematics

- If two types A, B are propositions,

$$(A =_{\text{Prop}} B) \overset{UA}{\simeq} (A \simeq B) \simeq (A \leftrightarrow B)$$

so everything respects bi-implication of propositions.

- If A, B are sets,

$$(A =_{\text{Set}} B) \overset{UA}{\simeq} (A \simeq B) \simeq (A \cong B)$$

so everything respects bijection of sets.

- For types A, B which are structured sets (groups, rings, etc),

$$(A =_{\text{Grp}} B) \overset{UA}{\simeq} (A \simeq B) \simeq (A \cong B)$$

so everything respects isomorphism of groups (or rings, etc).[2]

[2] Coquand-Danielsson 2013

Univalent mathematics

- If two types A, B are propositions,

$$(A =_{\mathrm{Prop}} B) \overset{UA}{\simeq} (A \simeq B) \simeq (A \leftrightarrow B)$$

so everything respects bi-implication of propositions.

- If A, B are sets,

$$(A =_{\mathrm{Set}} B) \overset{UA}{\simeq} (A \simeq B) \simeq (A \cong B)$$

so everything respects bijection of sets.

- For types A, B which are structured sets (groups, rings, etc),

$$(A =_{\mathrm{Grp}} B) \overset{UA}{\simeq} (A \simeq B) \simeq (A \cong B)$$

so everything respects isomorphism of groups (or rings, etc).[2]

- For *univalent* categories A, B,

$$(A =_{\mathrm{UCat}} B) \overset{UA}{\simeq} (A \simeq B) \simeq (A \simeq B)$$

so everything respects equivalence of univalent categories.[3]

[2] Coquand-Danielsson 2013

[3] Ahrens-Kapulkin-Shulman 2015

Univalent mathematics

- Voevodsky dreamt of ‘univalent mathematics’ in which

$$(A =_{\mathrm{D}} B) \simeq (A \simeq_{\mathrm{D}} B) \tag{$*$}$$

  where D is any type of mathematical object (propositions, sets, groups, categories, ∞-categories, etc) and $\simeq_{\mathrm{D}}$ is the appropriate notion of ‘sameness’ for that type of objects.
- This would give us an appropriate language in which to study D.

Univalent mathematics

- Voevodsky dreamt of 'univalent mathematics' in which

$$(A =_{\mathrm{D}} B) \simeq (A \simeq_{\mathrm{D}} B) \tag{$*$}$$

  where D is any type of mathematical object (propositions, sets, groups, categories, ∞-categories, etc) and $\simeq_{\mathrm{D}}$ is the appropriate notion of 'sameness' for that type of objects.
- This would give us an appropriate language in which to study D.
- In our *Univalence Principle* project, we developed machinery which automatically produces $(*)$ for a all D described by a signature (e.g. groups, categories, bicategories, opetopic categories, ...)

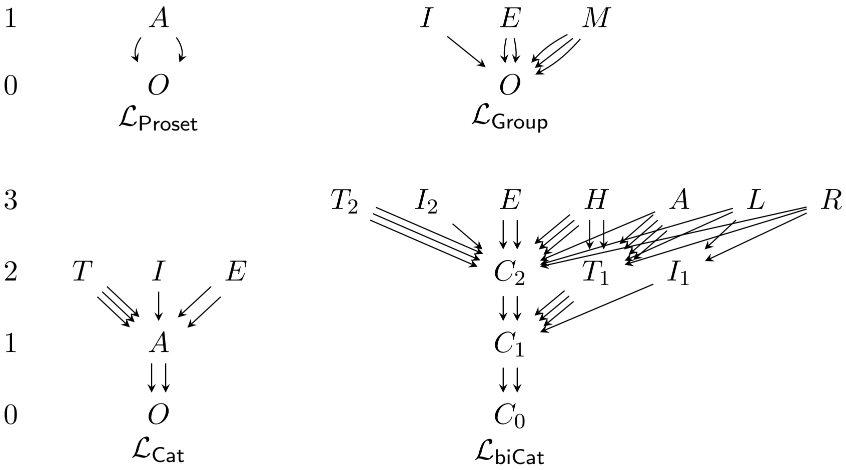
Univalent mathematics

- Voevodsky dreamt of ‘univalent mathematics’ in which
$$(A =_{\mathrm{D}} B) \simeq (A \simeq_{\mathrm{D}} B) \tag{$*$}$$
where D is any type of mathematical object (propositions, sets, groups, categories, ∞-categories, etc) and $\simeq_{\mathrm{D}}$ is the appropriate notion of ‘sameness’ for that type of objects.
- This would give us an appropriate language in which to study D.
- In our *Univalence Principle* project, we developed machinery which automatically produces ($*$) for a all D described by a signature (e.g. groups, categories, bicategories, opetopic categories, ...)
- In my *Directed homotopy type theory* project, I am developing a synthetic notion of directed equality that satisfies directed univalence.
$$(A \Rightarrow_{\mathrm{D}} B) \simeq (A \rightarrow_{\mathrm{D}} B)$$

Outline

1. Background on type theory and univalent foundations
2. The univalence principle
3. Directed homotopy type theory
4. Summary

Signatures



1 A
0 O
$\mathcal{L}_{\mathsf{Proset}}$
I E M
O
$\mathcal{L}_{\mathsf{Group}}$
3
2 T I E
1 A
0 O
$\mathcal{L}_{\mathsf{Cat}}$
T_2 I_2 E H A L R
C_2 T_1 I_1
C_1
C_0
$\mathcal{L}_{\mathsf{biCat}}$

Structures

- Morally, an **$\mathcal{L}$-structure** for a signature $\mathcal{L}$ is a Reedy fibrant diagram $\mathcal{L} \to \mathrm{Type}$.

Structures

- Morally, an **$\mathcal{L}$-structure** for a signature $\mathcal{L}$ is a Reedy fibrant diagram $\mathcal{L} \to \text{Type}$.
- In type theory, we define an **$\mathcal{L}$-structure** fiberwise.

Structures

- Morally, an **$\mathcal{L}$-structure** for a signature $\mathcal{L}$ is a Reedy fibrant diagram $\mathcal{L} \to \text{Type}$.
- In type theory, we define an **$\mathcal{L}$-structure** fiberwise.
- An $\mathcal{L}_{\mathsf{Cat}}$-structure $\mathcal{C}$ consists of:

$$
\begin{tikzcd}
T \arrow[dr, shift left] \arrow[dr] \arrow[dr, shift right] & I \arrow[d] & E \arrow[dl, shift left] \arrow[dl, shift right] \\
& A \arrow[d, shift left] \arrow[d, shift right] & \\
& O &
\end{tikzcd}
$$

$\mathcal{L}_{\mathsf{Cat}}$

- $\mathcal{C}O : \mathsf{Type}$
- $x, y : \mathcal{C}O \vdash \mathcal{C}A(x, y) : \mathsf{Type}$
- $x : \mathcal{C}O, f : \mathcal{C}A(x, x) \vdash \mathcal{C}I_x(f) : \mathsf{Type}$
- $x, y, z : \mathcal{C}O, f : \mathcal{C}A(x, y), g : \mathcal{C}A(y, z), h : \mathcal{C}A(x, z) \vdash \mathcal{C}T_{x,y,z}(f, g, h) : \mathsf{Type}$
- $x, y : \mathcal{C}O, f, g : \mathcal{C}A(x, y) \vdash \mathcal{C}E_{x,y}(f, g) : \mathsf{Type}$

Structures

- Morally, an **$\mathcal{L}$-structure** for a signature $\mathcal{L}$ is a Reedy fibrant diagram $\mathcal{L} \to$ Type.
- In type theory, we define an **$\mathcal{L}$-structure** fiberwise.
- An $\mathcal{L}_{\mathsf{Cat}}$-structure $\mathcal{C}$ consists of:

$$
\begin{tikzcd}
T \arrow[dr, shift left=2] \arrow[dr] \arrow[dr, shift right=2] & I \arrow[d] & E \arrow[dl, shift left=1] \arrow[dl, shift right=1] \\
& A \arrow[d, shift left=1] \arrow[d, shift right=1] & \\
& O &
\end{tikzcd}
$$

$\mathcal{L}_{\mathsf{Cat}}$

  - $\mathcal{C}O : \mathsf{Type}$
  - $x, y : \mathcal{C}O \vdash \mathcal{C}A(x, y) : \mathsf{Type}$
  - $x : \mathcal{C}O, f : \mathcal{C}A(x, x) \vdash \mathcal{C}I_x(f) : \mathsf{Type}$
  - $x, y, z : \mathcal{C}O, f : \mathcal{C}A(x, y), g : \mathcal{C}A(y, z), h : \mathcal{C}A(x, z) \vdash \mathcal{C}T_{x,y,z}(f, g, h) : \mathsf{Type}$
  - $x, y : \mathcal{C}O, f, g : \mathcal{C}A(x, y) \vdash \mathcal{C}E_{x,y}(f, g) : \mathsf{Type}$

- Then we add axioms.

Level-wise equivalence

Proposition

For two $\mathcal{L}$-structures S, T,

$$(S =_{\mathcal{L}-\mathsf{Str}} T) \simeq (S \cong_{\mathcal{L}-\mathsf{Str}} T)$$

where $\cong_{\mathcal{L}-\mathsf{Str}}$ denotes levelwise equivalence.

Level-wise equivalence

Proposition

For two $\mathcal{L}$-structures S, T,

$$(S =_{\mathcal{L}-\mathsf{Str}} T) \simeq (S \cong_{\mathcal{L}-\mathsf{Str}} T)$$

where $\cong_{\mathcal{L}-\mathsf{Str}}$ denotes levelwise equivalence.

A levelwise equivalence $\mathcal{C} \cong_{\mathcal{L}_{\mathsf{Cat}}-\mathsf{Str}} \mathcal{D}$ consists of:

- $e_O : \mathcal{C}O \xrightarrow{\sim} \mathcal{D}O$
- $x, y : \mathcal{C}O \vdash e_A : \mathcal{C}A(x, y) \xrightarrow{\sim} \mathcal{D}(e_O x, e_O y)$
- $x : \mathcal{C}O, f : \mathcal{C}A(x, x) \vdash e_I : \mathcal{C}I_x(f) \xrightarrow{\sim} \mathcal{D}I_{e_O x}(e_A f)$
- $x, y, z : \mathcal{C}O, f : \mathcal{C}A(x, y), g : \mathcal{C}A(y, z), h : \mathcal{C}A(x, z) \vdash$ $\mathcal{C}T_{x,y,z}(f, g, h) \xrightarrow{\sim} \mathcal{D}T_{e_O x, e_O y, e_O z}(e_A f, e_A g, e_A h)$
- $x, y : \mathcal{C}O, f, g : \mathcal{C}A(x, y) \vdash \mathcal{C}E_{x,y}(f, g) \xrightarrow{\sim} \mathcal{C}E_{e_O x, e_O y}(e_A f, e_A g)$

Level-wise equivalence

Proposition

For two $\mathcal{L}$-structures S, T,

$$(S =_{\mathcal{L}-\mathsf{Str}} T) \simeq (S \cong_{\mathcal{L}-\mathsf{Str}} T)$$

where $\cong_{\mathcal{L}-\mathsf{Str}}$ denotes levelwise equivalence.

A levelwise equivalence $\mathcal{C} \cong_{\mathcal{L}_{\mathsf{Cat}}-\mathsf{Str}} \mathcal{D}$ consists of:

- $e_O : \mathcal{C}O \xrightarrow{\sim} \mathcal{D}O$
- $x, y : \mathcal{C}O \vdash e_A : \mathcal{C}A(x, y) \xrightarrow{\sim} \mathcal{D}(e_O x, e_O y)$
- $x : \mathcal{C}O, f : \mathcal{C}A(x, x) \vdash e_I : \mathcal{C}I_x(f) \xrightarrow{\sim} \mathcal{D}I_{e_O x}(e_A f)$
- $x, y, z : \mathcal{C}O, f : \mathcal{C}A(x, y), g : \mathcal{C}A(y, z), h : \mathcal{C}A(x, z) \vdash$ $\mathcal{C}T_{x,y,z}(f, g, h) \xrightarrow{\sim} \mathcal{D}T_{e_O x, e_O y, e_O z}(e_A f, e_A g, e_A h)$
- $x, y : \mathcal{C}O, f, g : \mathcal{C}A(x, y) \vdash \mathcal{C}E_{x,y}(f, g) \xrightarrow{\sim} \mathcal{C}E_{e_O x, e_O y}(e_A f, e_A g)$

But this is not an equivalence of categories.

Level-wise equivalence

Proposition

For two $\mathcal{L}$-structures S, T,

$$(S =_{\mathcal{L}-\mathsf{Str}} T) \simeq (S \cong_{\mathcal{L}-\mathsf{Str}} T)$$

where $\cong_{\mathcal{L}-\mathsf{Str}}$ denotes levelwise equivalence.

A levelwise equivalence $\mathcal{C} \cong_{\mathcal{L}_{\mathsf{Cat}}-\mathsf{Str}} \mathcal{D}$ consists of:

- $e_O : \mathcal{C}O \xrightarrow{\sim} \mathcal{D}O$
- $x, y : \mathcal{C}O \vdash e_A : \mathcal{C}A(x, y) \xrightarrow{\sim} \mathcal{D}(e_O x, e_O y)$
- $x : \mathcal{C}O, f : \mathcal{C}A(x, x) \vdash e_I : \mathcal{C}I_x(f) \xrightarrow{\sim} \mathcal{D}I_{e_O x}(e_A f)$
- $x, y, z : \mathcal{C}O, f : \mathcal{C}A(x, y), g : \mathcal{C}A(y, z), h : \mathcal{C}A(x, z) \vdash$ $\mathcal{C}T_{x,y,z}(f, g, h) \xrightarrow{\sim} \mathcal{D}T_{e_O x, e_O y, e_O z}(e_A f, e_A g, e_A h)$
- $x, y : \mathcal{C}O, f, g : \mathcal{C}A(x, y) \vdash \mathcal{C}E_{x,y}(f, g) \xrightarrow{\sim} \mathcal{C}E_{e_O x, e_O y}(e_A f, e_A g)$

But this is not an equivalence of categories.
And is it appropriate to call $\mathcal{C}, \mathcal{D}$ categories?

Indiscernibility

Definition

Given an $\mathcal{L}$-structure M, and an object S of $\mathcal{L}$, we say that two elements $x, y : MS$ are *indiscernible* if substituting x for y in any object of $\mathcal{L}$ that depends on (i.e. object with a morphism to) S produces equivalent types.

Definition

An $\mathcal{L}$-structure M is *univalent* if for any object S of $\mathcal{L}$, and any $x, y : MS$, the type of indiscernibilities between x and y is equivalent to the type of equalities between x and y.

Univalent $\mathcal{L}_{\text{cat}}$ structures

Let $\mathcal{C}$ be a univalent $\mathcal{L}_{\text{cat}}$ structure.

$$\begin{tikzcd}
T \arrow[dr, shift left=1.5] \arrow[dr] \arrow[dr, shift right=1.5] & I \arrow[d] & E \arrow[dl, shift left] \arrow[dl, shift right] \\
& A \arrow[d, shift left] \arrow[d, shift right] & \\
& O &
\end{tikzcd}$$

Univalent $\mathcal{L}_{\text{cat}}$ structures

Let $\mathcal{C}$ be a univalent $\mathcal{L}_{\text{cat}}$ structure.

$$
\begin{tikzcd}
T \arrow[dr, shift left=1.5] \arrow[dr] \arrow[dr, shift right=1.5] & I \arrow[d] & E \arrow[dl, shift left=1] \arrow[dl, shift right=1] \\
& A \arrow[d, shift left=1] \arrow[d, shift right=1] & \\
& O &
\end{tikzcd}
$$

- Any two terms $x : \mathcal{C}O, f : \mathcal{C}A(x,x) \vdash i, j : \mathcal{C}I_x(f)$ are indiscernible.

$\rightarrow$ Each $\mathcal{C}I_x(f)$ is a proposition.

$\rightarrow$ Similarly, each $\mathcal{C}T_{x,y,z}(f,g,h)$, $\mathcal{C}E_{x,y}(f,g)$ is a proposition.

Univalent $\mathcal{L}_{\text{cat}}$ structures

Let $\mathcal{C}$ be a univalent $\mathcal{L}_{\text{cat}}$ structure.

$$
\begin{tikzcd}
T \arrow[dr, shift left] \arrow[dr] \arrow[dr, shift right] & I \arrow[d] & E \arrow[dl, shift left] \arrow[dl, shift right] \\
& A \arrow[d, shift left] \arrow[d, shift right] & \\
& O &
\end{tikzcd}
$$

- Any two terms $x : \mathcal{C}O, f : \mathcal{C}A(x,x) \vdash i, j : \mathcal{C}I_x(f)$ are indiscernible.

→ Each $\mathcal{C}I_x(f)$ is a proposition.

→ Similarly, each $\mathcal{C}T_{x,y,z}(f,g,h)$, $\mathcal{C}E_{x,y}(f,g)$ is a proposition.

- In the axioms for a category, we have that E behaves like equality (is reflexive and a congruence for T, I, E.)

→ Univalence at A means that $f = g$ is equivalent to $\mathcal{C}E_{x,y}(f,g)$.

→ $\mathcal{C}A(x,y)$ is a set.

Univalent $\mathcal{L}_{\text{cat}}$ structures

- The indiscernibilities between $a, b : \mathcal{C}O$ consist of
  - $\phi_{x\bullet} : \mathcal{C}A(x,a) \cong \mathcal{C}A(x,b)$ for each $x : \mathcal{C}O$
  - $\phi_{\bullet z} : \mathcal{C}A(a,z) \cong \mathcal{C}A(b,z)$ for each $z : \mathcal{C}O$
  - $\phi_{\bullet\bullet} : \mathcal{C}A(a,a) \cong \mathcal{C}A(b,b)$
  - The following for all appropriate w, x, y, z, f, g, h:

$$\mathcal{C}T_{x,y,a}(f,g,h) \leftrightarrow \mathcal{C}T_{x,y,b}(f,\phi_{y\bullet}(g),\phi_{x\bullet}(h))$$
$$\mathcal{C}T_{x,a,z}(f,g,h) \leftrightarrow \mathcal{C}T_{x,b,z}(\phi_{x\bullet}(f),\phi_{\bullet z}(g),h)$$
$$\mathcal{C}T_{a,z,w}(f,g,h) \leftrightarrow \mathcal{C}T_{b,z,w}(\phi_{\bullet z}(f),g,\phi_{\bullet w}(h))$$
$$\mathcal{C}T_{x,a,a}(f,g,h) \leftrightarrow \mathcal{C}T_{x,b,b}(\phi_{x\bullet}(f),\phi_{\bullet\bullet}(g),\phi_{x\bullet}(h))$$
$$\mathcal{C}T_{a,x,a}(f,g,h) \leftrightarrow \mathcal{C}T_{b,x,b}(\phi_{\bullet x}(f),\phi_{x\bullet}(g),\phi_{\bullet\bullet}(h))$$
$$\mathcal{C}T_{a,a,x}(f,g,h) \leftrightarrow \mathcal{C}T_{b,b,x}(\phi_{\bullet\bullet}(f),\phi_{\bullet x}(g),\phi_{\bullet x}(h))$$
$$\mathcal{C}T_{a,a,a}(f,g,h) \leftrightarrow \mathcal{C}T_{b,b,b}(\phi_{\bullet\bullet}(f),\phi_{\bullet\bullet}(g),\phi_{\bullet\bullet}(h))$$

$$\mathcal{C}I_a(f) \leftrightarrow \mathcal{C}I_b(\phi_{\bullet\bullet}(f))$$
$$\mathcal{C}E_{x,a}(f,g) \leftrightarrow \mathcal{C}E_{x,b}(\phi_{x\bullet}(f),\phi_{x\bullet}(g$$
$$\mathcal{C}E_{a,x}(f,g) \leftrightarrow \mathcal{C}E_{b,x}(\phi_{\bullet x}(f),\phi_{\bullet x}(g$$
$$\mathcal{C}E_{a,a}(f,g) \leftrightarrow \mathcal{C}E_{b,b}(\phi_{\bullet\bullet}(f),\phi_{\bullet\bullet}(g$$

Univalent $\mathcal{L}_{\mathrm{cat}}$ structures

- The indiscernibilities between $a, b : \mathcal{C}O$ consist of
  - $\phi_{x\bullet} : \mathcal{C}A(x,a) \cong \mathcal{C}A(x,b)$ for each $x : \mathcal{C}O$
  - $\phi_{\bullet z} : \mathcal{C}A(a,z) \cong \mathcal{C}A(b,z)$ for each $z : \mathcal{C}O$
  - $\phi_{\bullet\bullet} : \mathcal{C}A(a,a) \cong \mathcal{C}A(b,b)$
  - The following for all appropriate w, x, y, z, f, g, h:

$$\begin{aligned}
&\mathcal{C}T_{x,y,a}(f,g,h) \leftrightarrow \mathcal{C}T_{x,y,b}(f, \phi_{y\bullet}(g), \phi_{x\bullet}(h))\\
&\mathcal{C}T_{x,a,z}(f,g,h) \leftrightarrow \mathcal{C}T_{x,b,z}(\phi_{x\bullet}(f), \phi_{\bullet z}(g), h)\\
&\mathcal{C}T_{a,z,w}(f,g,h) \leftrightarrow \mathcal{C}T_{b,z,w}(\phi_{\bullet z}(f), g, \phi_{\bullet w}(h))\\
&\mathcal{C}T_{x,a,a}(f,g,h) \leftrightarrow \mathcal{C}T_{x,b,b}(\phi_{x\bullet}(f), \phi_{\bullet\bullet}(g), \phi_{x\bullet}(h))\\
&\mathcal{C}T_{a,x,a}(f,g,h) \leftrightarrow \mathcal{C}T_{b,x,b}(\phi_{\bullet x}(f), \phi_{x\bullet}(g), \phi_{\bullet\bullet}(h))\\
&\mathcal{C}T_{a,a,x}(f,g,h) \leftrightarrow \mathcal{C}T_{b,b,x}(\phi_{\bullet\bullet}(f), \phi_{\bullet x}(g), \phi_{\bullet x}(h))\\
&\mathcal{C}T_{a,a,a}(f,g,h) \leftrightarrow \mathcal{C}T_{b,b,b}(\phi_{\bullet\bullet}(f), \phi_{\bullet\bullet}(g), \phi_{\bullet\bullet}(h))
\end{aligned}$$

$$\begin{aligned}
&\mathcal{C}I_a(f) \leftrightarrow \mathcal{C}I_b(\phi_{\bullet\bullet}(f))\\
&\mathcal{C}E_{x,a}(f,g) \leftrightarrow \mathcal{C}E_{x,b}(\phi_{x\bullet}(f), \phi_{x\bullet}(g\\
&\mathcal{C}E_{a,x}(f,g) \leftrightarrow \mathcal{C}E_{b,x}(\phi_{\bullet x}(f), \phi_{\bullet x}(g\\
&\mathcal{C}E_{a,a}(f,g) \leftrightarrow \mathcal{C}E_{b,b}(\phi_{\bullet\bullet}(f), \phi_{\bullet\bullet}(g
\end{aligned}$$

- But this an isomorphism in the usual categorical sense.

$\to$ Univalence at O means that $x = y$ is equivalent to $x \cong y$.

The right notion of equivalence

Main theorem

For two *univalent* $\mathcal{L}$-structures S, T,

$$(S =_{\mathcal{L}-\mathsf{Str}} T) \simeq (S \cong_{\mathcal{L}-\mathsf{Str}} T) \simeq (S \cong^*_{\mathcal{L}-\mathsf{Str}} T) \simeq (S \twoheadrightarrow T)$$

where $\cong^*_{\mathcal{L}-\mathsf{Str}}$ denotes levelwise equivalence up to indiscernbility and $\twoheadrightarrow$ denotes a very split surjective morphism.

The right notion of equivalence

Main theorem

For two *univalent* $\mathcal{L}$-structures S, T,

$$(S =_{\mathcal{L}-\mathsf{Str}} T) \simeq (S \cong_{\mathcal{L}-\mathsf{Str}} T) \simeq (S \cong^*_{\mathcal{L}-\mathsf{Str}} T) \simeq (S \twoheadrightarrow T)$$

where $\cong^*_{\mathcal{L}-\mathsf{Str}}$ denotes levelwise equivalence up to indiscernbility and $\twoheadrightarrow$ denotes a very split surjective morphism.

Very surjective morphisms of $\mathcal{L}_{\mathrm{cat}}$-structures

A *very surjective morphism* or *equivalence* $F : \mathcal{C} \simeq \mathcal{D}$ of $\mathcal{L}_{\mathrm{cat}}$-structures consists of surjections

- $FO : \mathcal{C}O \twoheadrightarrow \mathcal{D}O$
- $FA : \mathcal{C}A(x,y) \twoheadrightarrow \mathcal{D}A(Fx, Fy)$ for every $x, y : \mathcal{C}O$
- $FT : \mathcal{C}T(f,g,h) \twoheadrightarrow \mathcal{D}T(Ff, Fg, Fh)$ for all $f : \mathcal{C}A(x,y), g : \mathcal{C}A(y,z), h : \mathcal{C}A(x,z)$
- $FE : \mathcal{C}E(f,g) \twoheadrightarrow \mathcal{D}E(Ff, Fg)$ for all $f, g : \mathcal{C}A(x,y)$
- $FI : \mathcal{C}I(f) \twoheadrightarrow \mathcal{D}I(Ff)$ for all $f : \mathcal{C}A(x,x)$

The right notion of equivalence

Main theorem

For two *univalent* $\mathcal{L}$-structures S, T,

$$(S =_{\mathcal{L}-\mathsf{Str}} T) \simeq (S \cong_{\mathcal{L}-\mathsf{Str}} T) \simeq (S \cong^*_{\mathcal{L}-\mathsf{Str}} T) \simeq (S \twoheadrightarrow T)$$

where $\cong^*_{\mathcal{L}-\mathsf{Str}}$ denotes levelwise equivalence up to indiscernbility and $\twoheadrightarrow$ denotes a very split surjective morphism.

Very surjective morphisms of $\mathcal{L}_{\mathrm{cat}}$-structures

A *very surjective morphism* or *equivalence* $F : \mathcal{C} \simeq \mathcal{D}$ of univalent $\mathcal{L}_{\mathrm{cat}}$-structures consists of surjections

- $FO : \mathcal{C}O \twoheadrightarrow \mathcal{D}O$
- $FA : \mathcal{C}A(x, y) \twoheadrightarrow \mathcal{D}A(Fx, Fy)$ for every $x, y : \mathcal{C}O$
- $FT : \mathcal{C}T(f, g, h) \twoheadrightarrow \mathcal{D}T(Ff, Fg, Fh)$ for all $f : \mathcal{C}A(x, y), g : \mathcal{C}A(y, z), h : \mathcal{C}A(x, z)$
- $FE : \mathcal{C}E(f, g) \twoheadrightarrow \mathcal{D}E(Ff, Fg)$ for all $f, g : \mathcal{C}A(x, y)$
- $FI : \mathcal{C}I(f) \twoheadrightarrow \mathcal{D}I(Ff)$ for all $f : \mathcal{C}A(x, x)$

The right notion of equivalence

Main theorem

For two *univalent* $\mathcal{L}$-structures S, T,

$$(S =_{\mathcal{L}-\mathsf{Str}} T) \simeq (S \cong_{\mathcal{L}-\mathsf{Str}} T) \simeq (S \cong^*_{\mathcal{L}-\mathsf{Str}} T) \simeq (S \twoheadrightarrow T)$$

where $\cong^*_{\mathcal{L}-\mathsf{Str}}$ denotes levelwise equivalence up to indiscernbility and $\twoheadrightarrow$ denotes a very split surjective morphism.

Very surjective morphisms of $\mathcal{L}_{\text{cat}}$-structures

A *very surjective morphism* or *equivalence* $F : \mathcal{C} \simeq \mathcal{D}$ of univalent $\mathcal{L}_{\text{cat}}$-structures consists of surjections

- $FO : \mathcal{C}O \twoheadrightarrow \mathcal{D}O$
- $FA : \mathcal{C}A(x, y) \twoheadrightarrow \mathcal{D}A(Fx, Fy)$ for every $x, y : \mathcal{C}O$
- $FT : \mathcal{C}T(f, g, h) \leftrightarrow \mathcal{D}T(Ff, Fg, Fh)$ for all $f : \mathcal{C}A(x, y), g : \mathcal{C}A(y, z), h : \mathcal{C}A(x, z)$
- $FE : \mathcal{C}E(f, g) \leftrightarrow \mathcal{D}E(Ff, Fg)$ for all $f, g : \mathcal{C}A(x, y)$
- $FI : \mathcal{C}I(f) \leftrightarrow \mathcal{D}I(Ff)$ for all $f : \mathcal{C}A(x, x)$

The right notion of equivalence

Main theorem

For two *univalent* $\mathcal{L}$-structures S, T,

$$(S =_{\mathcal{L}-\mathsf{Str}} T) \simeq (S \cong_{\mathcal{L}-\mathsf{Str}} T) \simeq (S \cong^*_{\mathcal{L}-\mathsf{Str}} T) \simeq (S \twoheadrightarrow T)$$

where $\cong^*_{\mathcal{L}-\mathsf{Str}}$ denotes levelwise equivalence up to indiscernbility and $\twoheadrightarrow$ denotes a very split surjective morphism.

Very surjective morphisms of $\mathcal{L}_{\text{cat}}$-structures

A *very surjective morphism* or *equivalence* $F : \mathcal{C} \simeq \mathcal{D}$ of univalent $\mathcal{L}_{\text{cat}}$-structures consists of surjections

- $FO : \mathcal{C}O \twoheadrightarrow \mathcal{D}O$
- $FA : \mathcal{C}A(x, y) \twoheadrightarrow \mathcal{D}A(Fx, Fy)$ for every $x, y : \mathcal{C}O$
- $FT : \mathcal{C}T(f, g, h) \leftrightarrow \mathcal{D}T(Ff, Fg, Fh)$ for all $f : \mathcal{C}A(x, y), g : \mathcal{C}A(y, z), h : \mathcal{C}A(x, z)$
- $FE : (f = g) \leftrightarrow (Ff = Fg)$ for all $f, g : \mathcal{C}A(x, y)$
- $FI : \mathcal{C}I(f) \leftrightarrow \mathcal{D}I(Ff)$ for all $f : \mathcal{C}A(x, x)$

The right notion of equivalence

Main theorem

For two *univalent* $\mathcal{L}$-structures S, T,

$$(S =_{\mathcal{L}-\mathsf{Str}} T) \simeq (S \cong_{\mathcal{L}-\mathsf{Str}} T) \simeq (S \cong^*_{\mathcal{L}-\mathsf{Str}} T) \simeq (S \twoheadrightarrow T)$$

where $\cong^*_{\mathcal{L}-\mathsf{Str}}$ denotes levelwise equivalence up to indiscernbility and $\twoheadrightarrow$ denotes a very split surjective morphism.

Very surjective morphisms of $\mathcal{L}_{\text{cat}}$-structures

A *very surjective morphism* or *equivalence* $F : \mathcal{C} \simeq \mathcal{D}$ of univalent $\mathcal{L}_{\text{cat}}$-structures consists of surjections

- $FO : \mathcal{C}O \twoheadrightarrow \mathcal{D}O$
- $FA : \mathcal{C}A(x, y) \cong \mathcal{D}A(Fx, Fy)$ for every $x, y : \mathcal{C}O$
- $FT : \mathcal{C}T(f, g, h) \leftrightarrow \mathcal{D}T(Ff, Fg, Fh)$ for all $f : \mathcal{C}A(x, y), g : \mathcal{C}A(y, z), h : \mathcal{C}A(x, z)$
- $FE : (f = g) \leftrightarrow (Ff = Fg)$ for all $f, g : \mathcal{C}A(x, y)$
- $FI : \mathcal{C}I(f) \leftrightarrow \mathcal{D}I(Ff)$ for all $f : \mathcal{C}A(x, x)$

Summary

For every signature $\mathcal{L}$, we have

- a notion of structure,
- a notion of indiscernibility within each sort,
- a notion of univalent structures,
- a notion of equivalence,
- a univalence theorem.

Summary

For every signature $\mathcal{L}$, we have

- a notion of structure,
- a notion of indiscernibility within each sort,
- a notion of univalent structures,
- a notion of equivalence,
- a univalence theorem.

The paper includes examples of

- †-categories,
- profunctors,
- bicategories,
- opetopic bicategories,
- ...

Current and future work

- Drop the splitness condition for certain structures.
- Extend to infinite structures.
- Formulate an analogue to the Rezk completion.
- Translate the theory into one about structures which can include explicit functions.
- Explore mathematics within examples.
- Give a model-theoretic account.

Outline

1. Background on type theory and univalent foundations
2. The univalence principle
3. Directed homotopy type theory
4. Summary

Directed type theory

- The **identity type** of type theory captures the paths of spaces, or the isomorphisms in a higher groupoid
- At the core of directed type theory is a **homomorphism type** which captures **directed** paths of **directed** spaces, or the morphisms of a higher category
- We are motivated by applications of directed topology to:
  - concurrent processes
  - robotics
  - rewriting
  - neuroscience

What does directed mean?

higher groupoids

What does directed mean?

higher groupoids

higher categories

(undirected paths $\subseteq$ directed paths)

What does directed mean?

higher groupoids

higher categories
(undirected paths $\subseteq$ directed paths)

directed spaces
(directed paths $\subseteq$ undirected paths)

The homomorphism type

- The identity type $x =_A y$ is the smallest reflexive relation $A \times A \to \mathsf{Type}$ on A

The homomorphism type

- The identity type $x =_A y$ is the smallest reflexive relation $A \times A \to \mathsf{Type}$ on A
- The homomorphism type $x \to_A y$ is the smallest reflexive relation $A \times A \to \mathsf{Type}$ on A

The homomorphism type

- The identity type $x =_A y$ is the smallest reflexive relation $A \times A \to \mathsf{Type}$ on A where $\to$ is functorial in identity terms of $A \times A$.
- The homomorphism type $x \to_A y$ is the smallest reflexive relation $A^{\mathsf{op}} \times A \to \mathsf{Type}$ on A where $\to$ is functorial in homomorphism terms in $A^{\mathsf{op}} \times A$.

The homomorphism type

- The identity type $x =_A y$ is the smallest reflexive relation $A \times A \to \mathsf{Type}$ on A where $\to$ is functorial in identity terms of $A \times A$.
- The homomorphism type $x \to_A y$ is the smallest reflexive relation $A^{\mathsf{op}} \times A \to \mathsf{Type}$ on A where $\to$ is functorial in homomorphism terms in $A^{\mathsf{op}} \times A$.

- Distinguishing different sorts of functoriality, or giving certain variables **modalities**, leads us to **modal type theory**

The homomorphism type

- The identity type $x =_A y$ is the smallest reflexive relation $A \times A \to \mathsf{Type}$ on A where $\to$ is functorial in identity terms of $A \times A$.
- The homomorphism type $x \to_A y$ is the smallest reflexive relation $A^{\mathsf{op}} \times A \to \mathsf{Type}$ on A where $\to$ is functorial in homomorphism terms in $A^{\mathsf{op}} \times A$.

- Distinguishing different sorts of functoriality, or giving certain variables **modalities**, leads us to **modal type theory**
- We must also develop a directed version of homotopy theory, in particular a notion of **directed weak factorization system**

The homomorphism type

Past work:

- The identity type $x =_A y$ is the smallest reflexive relation $A \times A \to \mathsf{Type}$ on A where $\to$ is functorial in identity terms of $A \times A$.
- The homomorphism type $x \to_A y$ is the smallest reflexive relation $A^{\mathsf{op}} \times A \to \mathsf{Type}$ on A where $\to$ is functorial in homomorphism terms in $A^{\mathsf{op}} \times A$.

Current work:

- Distinguishing different sorts of functoriality, or giving certain variables **modalities**, leads us to **modal type theory**
- We must also develop a directed version of homotopy theory, in particular a notion of **directed weak factorization system**

Future work:

- Develop notion of directed homotopy colimit, corresponding to higher inductive types of univalent foundations.
- Show that directed type theory is a conservative extension of type theory.
- Develop mathematics within directed type theory, oriented towards both theory and applications.

Outline

1. Background on type theory and univalent foundations
2. The univalence principle
3. Directed homotopy type theory
4. Summary

Summary

- In the **Univalence principle** project, we develop higher structures within type theory.

Summary

- In the **Univalence principle** project, we develop higher structures within type theory.
- In the **Directed type theory** project, we modify type theory itself to encode higher categories and directed spaces.

Summary

- In the **Univalence principle** project, we develop higher structures within type theory.
- In the **Directed type theory** project, we modify type theory itself to encode higher categories and directed spaces.
- We envision a **Directed** univalence principle project one day.

Summary

- In the **Univalence principle** project, we develop higher structures within type theory.
- In the **Directed type theory** project, we modify type theory itself to encode higher categories and directed spaces.
- We envision a **Directed** univalence principle project one day.
- My research program focuses on
  - developing the most natural formalisms for homotopical and higher mathematics, and then
  - doing homotopical and higher mathematics therein.

Thank you!