

An introduction to univalent foundations and the equivalence principle

Paige Randall North

University of Pennsylvania

28 March 2022

Outline

- ① Background on type theory and univalent foundations
- ② The univalence principle¹

¹jww Ahrens, Shulman, Tsementzis

Outline

- 1 Background on type theory and univalent foundations
- 2 The univalence principle²

²jww Ahrens, Shulman, Tsementzis

Type theory's beginnings

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

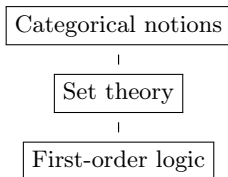
Type theory's beginnings

- 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:



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:

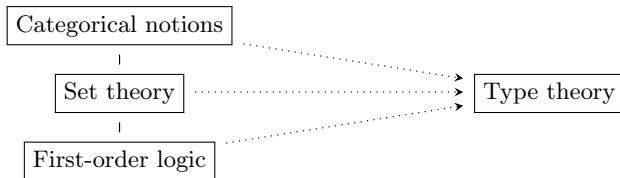


Type theory's beginnings

- 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:

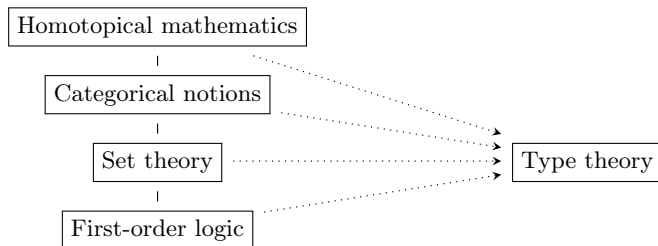


Type theory's beginnings

- 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:



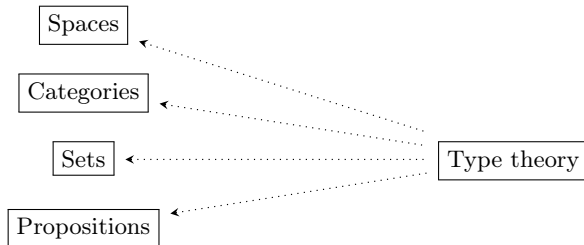
What does type theory look like?

- 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
 - $n : \mathbb{N} \vdash e(n) : \text{isEven}(n + n)$
 - $X : \mathbf{Spaces} \vdash c(X) : \text{isContr}(CX)$
- Type theory provides:
 - natural numbers type \mathbb{N}
 - product type $A \times B$
 - sum type $A + B$
 - function type $A \rightarrow B$
 - a universe type \mathbf{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	\simeq

Interpretations of type theory into classical mathematics

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

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 X s where equality is interpreted by Y
 - \rightsquigarrow Mathematics in X s 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 X s where equality is interpreted by Y
 - \rightsquigarrow Mathematics in X s 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 X s where equality is interpreted by Y
 - \rightsquigarrow Mathematics in X s up to Y

Interpretations of type theory into classical mathematics

Types	Terms	Product	Equality
Propositions	proofs	\wedge	$=$
Sets	elements	\times	$=$
Groupoids	objects	\times	\approx
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 X s where equality is interpreted by Y
 - \rightsquigarrow Mathematics in X s up to Y

Different notions of equality

Synthetic vs. analytic equalities

In type theory with the equality type, we always have a (“synthetic”) equality type between $a, b : D$

$$a =_D b.$$

Depending on the type D , we might also have a type of “analytic” equalities

$$a \simeq_D b.$$

A *univalence principle* for this D and this \simeq_D states that

$$(a =_D b) \rightarrow (a \simeq_D b)$$

is an equivalence.

Identicals and indiscernibilites

Identity of indiscernibles

Leibniz: two things are equal when they are *indiscernible* (have the same properties).

$$(a = b) \leftarrow (\forall P. P(a) \leftrightarrow P(b))$$

Identicals and indiscernibilites

Identity of indiscernibles

Leibniz: two things are equal when they are *indiscernible* (have the same properties).

$$(a = b) \leftrightarrow (\forall P. P(a) \leftrightarrow P(b))$$

Identicals and indiscernibles

Identity of indiscernibles

Leibniz: two things are equal when they are *indiscernible* (have the same properties).

$$(a =_D b) \leftrightarrow \left(\prod_{P:D \rightarrow \mathbf{Type}} P(a) \simeq P(b) \right)$$

Identicals and indiscernibles

Identity of indiscernibles

Leibniz: two things are equal when they are *indiscernible* (have the same properties).

$$(a =_D b) \leftrightarrow \left(\prod_{P:D \rightarrow \mathbf{Type}} P(a) \simeq P(b) \right)$$

- This holds in type theory.

Identicals and indiscernibilites

Identity of indiscernibles

Leibniz: two things are equal when they are *indiscernible* (have the same properties).

$$(a =_D b) \leftrightarrow \left(\prod_{P:D \rightarrow \mathbf{Type}} P(a) \simeq P(b) \right)$$

- This holds in type theory.
- Given a univalence principle $(a =_D b) \simeq (a \simeq_D b)$, we find an *equivalence principle*:

$$(a \simeq_D b) \rightarrow \left(\prod_{P:D \rightarrow \mathbf{Type}} P(a) \simeq P(b) \right).$$

Univalence

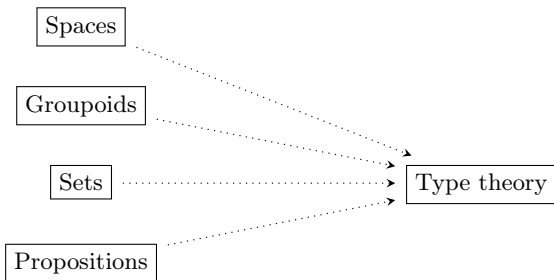
- We've seen that equality in type theory can be interpreted as notions weaker than classical equality (e.g. isomorphism, paths).
- Voevodsky imported weakness for equality from the interpretation in spaces into type theory by imposing the *Univalence Axiom* (UA):

The canonical function $(A =_{\text{Type}} B) \rightarrow (A \simeq B)$ is an equivalence of types, for any types A and B .

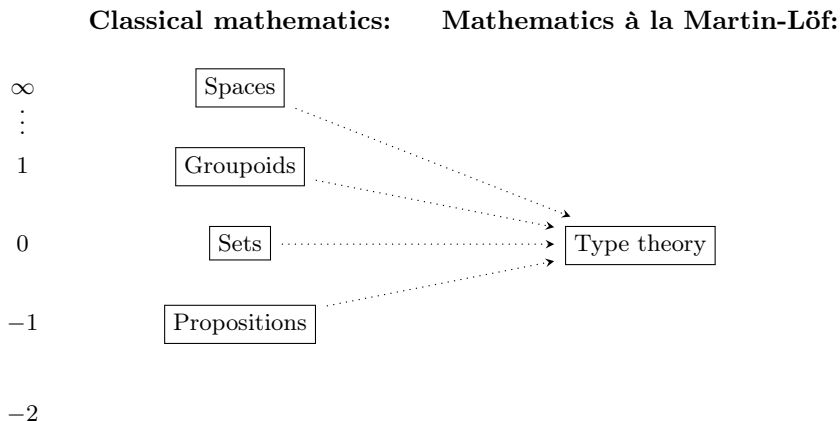
- UA is validated by the interpretation into spaces, but not into propositions, sets, or groupoids.
- Instead we **internalize** these notions.

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,

$$(A =_{\text{Prop}} B) \stackrel{U^A}{\simeq} (A \simeq B) \simeq (A \leftrightarrow B)$$

so everything respects bi-implication of propositions.

Univalent mathematics

- If two types A, B are propositions,

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

so everything respects bi-implication of propositions.

- If A, B are sets,

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

so everything respects bijection of sets.

Univalent mathematics

- If two types A, B are propositions,

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

so everything respects bi-implication of propositions.

- If A, B are sets,

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

so everything respects bijection of sets.

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

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

so everything respects isomorphism of groups (or rings, etc).³

³Coquand-Danielsson 2013

Univalent mathematics

- If two types A, B are propositions,

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

so everything respects bi-implication of propositions.

- If A, B are sets,

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

so everything respects bijection of sets.

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

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

so everything respects isomorphism of groups (or rings, etc).³

- For *univalent* categories A, B ,

$$(A =_{\text{UCat}} B) \stackrel{UA}{\simeq} (A \simeq B) \simeq (A \simeq B)$$

so everything respects equivalence of univalent categories.⁴

³Coquand-Danielsson 2013

⁴Ahrens-Kapulkin-Shulman 2015

Univalent mathematics

- Voevodsky dreamt of ‘univalent mathematics’ in which

$$(A =_D B) \simeq (A \simeq_D B)$$

where D is any type of mathematical object (propositions, sets, groups, categories, ∞ -categories, etc) and \simeq_D is the appropriate notion of ‘sameness’ for that type of objects.

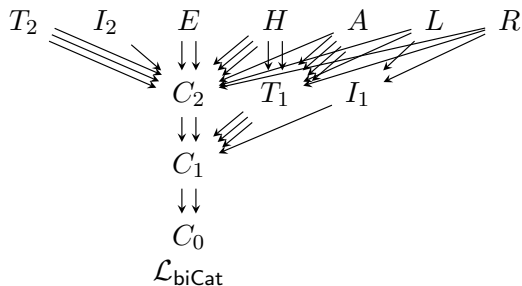
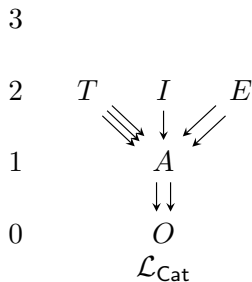
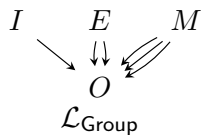
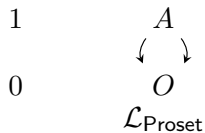
- This would give us an appropriate language in which to study D .

Outline

- ① Background on type theory and univalent foundations
- ② The univalence principle⁵

⁵jww Ahrens, Shulman, Tsementzis

Signatures



Structures

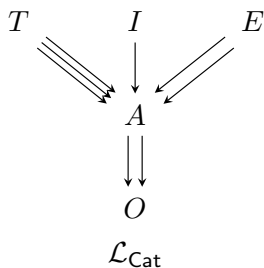
- Morally, an \mathcal{L} -**structure** for a signature \mathcal{L} is a Reedy fibrant diagram $\mathcal{L} \rightarrow \mathbf{Type}$.

Structures

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

Structures

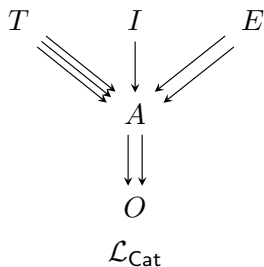
- Morally, an \mathcal{L} -**structure** for a signature \mathcal{L} is a Reedy fibrant diagram $\mathcal{L} \rightarrow \mathbf{Type}$.
- In type theory, we define an \mathcal{L} -**structure** fiberwise.
- An \mathcal{L}_{Cat} -structure \mathcal{C} consists of:



- $CO : \mathbf{Type}$
- $x, y : CO \vdash CA(x, y) : \mathbf{Type}$
- $x : CO, f : CA(x, x) \vdash CI_x(f) : \mathbf{Type}$
- $x, y, z : CO, f : CA(x, y), g : CA(y, z), h : CA(x, z) \vdash CT_{x,y,z}(f, g, h) : \mathbf{Type}$
- $x, y : CO, f, g : CA(x, y) \vdash CE_{x,y}(f, g) : \mathbf{Type}$

Structures

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



- $CO : \mathbf{Type}$
 - $x, y : CO \vdash CA(x, y) : \mathbf{Type}$
 - $x : CO, f : CA(x, x) \vdash CI_x(f) : \mathbf{Type}$
 - $x, y, z : CO, f : CA(x, y), g : CA(y, z), h : CA(x, z) \vdash CT_{x,y,z}(f, g, h) : \mathbf{Type}$
 - $x, y : CO, f, g : CA(x, y) \vdash CE_{x,y}(f, g) : \mathbf{Type}$
- Then we add axioms.

Level-wise equivalence

Proposition

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

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

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

Level-wise equivalence

Proposition

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

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

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

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

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

Level-wise equivalence

Proposition

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

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

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

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

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

But this is not an equivalence of categories.

Level-wise equivalence

Proposition

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

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

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

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

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

But this is not an equivalence of categories.

And is it appropriate to call \mathcal{C}, \mathcal{D} categories?

Indiscernibility

Definition

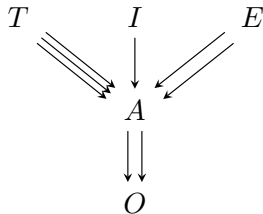
Given an \mathcal{L} -structure M , and an object S of \mathcal{L} , we say that two elements $x, y : MS$ are *indiscernible* if substituting x for y in any object of \mathcal{L} that depends on (i.e. object with a morphism to) S produces equivalent types.

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 .

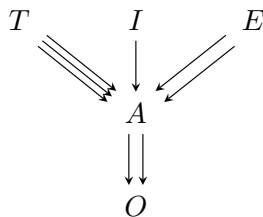
Univalent \mathcal{L}_{cat} structures

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



Univalent \mathcal{L}_{cat} structures

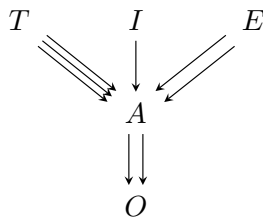
Let \mathcal{C} be a univalent \mathcal{L}_{cat} structure.



- 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.

Univalent \mathcal{L}_{cat} structures

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



- 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}_{cat} structures

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

$$\mathcal{CT}_{x,y,a}(f, g, h) \leftrightarrow \mathcal{CT}_{x,y,b}(f, \phi_{y\bullet}(g), \phi_{x\bullet}(h))$$

$$\mathcal{CI}_a(f) \leftrightarrow \mathcal{CI}_b(\phi_{\bullet\bullet}(f))$$

$$\mathcal{CT}_{x,a,z}(f, g, h) \leftrightarrow \mathcal{CT}_{x,b,z}(\phi_{x\bullet}(f), \phi_{\bullet z}(g), h)$$

$$\mathcal{CE}_{x,a}(f, g) \leftrightarrow \mathcal{CE}_{x,b}(\phi_{x\bullet}(f), \phi_{x\bullet}(g))$$

$$\mathcal{CT}_{a,z,w}(f, g, h) \leftrightarrow \mathcal{CT}_{b,z,w}(\phi_{\bullet z}(f), g, \phi_{\bullet w}(h))$$

$$\mathcal{CE}_{a,x}(f, g) \leftrightarrow \mathcal{CE}_{b,x}(\phi_{\bullet x}(f), \phi_{\bullet x}(g))$$

$$\mathcal{CT}_{x,a,a}(f, g, h) \leftrightarrow \mathcal{CT}_{x,b,b}(\phi_{x\bullet}(f), \phi_{\bullet\bullet}(g), \phi_{x\bullet}(h))$$

$$\mathcal{CE}_{a,a}(f, g) \leftrightarrow \mathcal{CE}_{b,b}(\phi_{\bullet\bullet}(f), \phi_{\bullet\bullet}(g))$$

$$\mathcal{CT}_{a,x,a}(f, g, h) \leftrightarrow \mathcal{CT}_{b,x,b}(\phi_{\bullet x}(f), \phi_{x\bullet}(g), \phi_{\bullet\bullet}(h))$$

$$\mathcal{CT}_{a,a,x}(f, g, h) \leftrightarrow \mathcal{CT}_{b,b,x}(\phi_{\bullet\bullet}(f), \phi_{\bullet x}(g), \phi_{\bullet x}(h))$$

$$\mathcal{CT}_{a,a,a}(f, g, h) \leftrightarrow \mathcal{CT}_{b,b,b}(\phi_{\bullet\bullet}(f), \phi_{\bullet\bullet}(g), \phi_{\bullet\bullet}(h))$$

Univalent \mathcal{L}_{cat} structures

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

$$\mathcal{CT}_{x,y,a}(f, g, h) \leftrightarrow \mathcal{CT}_{x,y,b}(f, \phi_{y\bullet}(g), \phi_{x\bullet}(h))$$

$$\mathcal{CI}_a(f) \leftrightarrow \mathcal{CI}_b(\phi_{\bullet\bullet}(f))$$

$$\mathcal{CT}_{x,a,z}(f, g, h) \leftrightarrow \mathcal{CT}_{x,b,z}(\phi_{x\bullet}(f), \phi_{\bullet z}(g), h)$$

$$\mathcal{CE}_{x,a}(f, g) \leftrightarrow \mathcal{CE}_{x,b}(\phi_{x\bullet}(f), \phi_{x\bullet}(g))$$

$$\mathcal{CT}_{a,z,w}(f, g, h) \leftrightarrow \mathcal{CT}_{b,z,w}(\phi_{\bullet z}(f), g, \phi_{\bullet w}(h))$$

$$\mathcal{CE}_{a,x}(f, g) \leftrightarrow \mathcal{CE}_{b,x}(\phi_{\bullet x}(f), \phi_{\bullet x}(g))$$

$$\mathcal{CT}_{x,a,a}(f, g, h) \leftrightarrow \mathcal{CT}_{x,b,b}(\phi_{x\bullet}(f), \phi_{\bullet\bullet}(g), \phi_{x\bullet}(h))$$

$$\mathcal{CE}_{a,a}(f, g) \leftrightarrow \mathcal{CE}_{b,b}(\phi_{\bullet\bullet}(f), \phi_{\bullet\bullet}(g))$$

$$\mathcal{CT}_{a,x,a}(f, g, h) \leftrightarrow \mathcal{CT}_{b,x,b}(\phi_{\bullet x}(f), \phi_{x\bullet}(g), \phi_{\bullet\bullet}(h))$$

$$\mathcal{CT}_{a,a,x}(f, g, h) \leftrightarrow \mathcal{CT}_{b,b,x}(\phi_{\bullet\bullet}(f), \phi_{\bullet x}(g), \phi_{\bullet x}(h))$$

$$\mathcal{CT}_{a,a,a}(f, g, h) \leftrightarrow \mathcal{CT}_{b,b,b}(\phi_{\bullet\bullet}(f), \phi_{\bullet\bullet}(g), \phi_{\bullet\bullet}(h))$$

- But this an isomorphism in the usual categorical sense.

→ 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 ,

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

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

The right notion of equivalence

Main theorem

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

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

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

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

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

- $FO : \mathcal{CO} \twoheadrightarrow \mathcal{DO}$
- $FA : \mathcal{CA}(x, y) \twoheadrightarrow \mathcal{DA}(Fx, Fy)$ for every $x, y : \mathcal{CO}$
- $FT : \mathcal{CT}(f, g, h) \twoheadrightarrow \mathcal{DT}(Ff, Fg, Fh)$ for all $f : \mathcal{CA}(x, y), g : \mathcal{CA}(y, z), h : \mathcal{CA}(x, z)$
- $FE : \mathcal{CE}(f, g) \twoheadrightarrow \mathcal{DE}(Ff, Fg)$ for all $f, g : \mathcal{CA}(x, y)$
- $FI : \mathcal{CI}(f) \twoheadrightarrow \mathcal{DI}(Ff)$ for all $f : \mathcal{CA}(x, x)$

The right notion of equivalence

Main theorem

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

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

where $\cong_{\mathcal{L}\text{-Str}}^*$ denotes levelwise equivalence up to indiscernibility and \twoheadrightarrow denotes a very split surjective morphism.

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

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

- $FO : \mathcal{CO} \twoheadrightarrow \mathcal{DO}$
- $FA : \mathcal{CA}(x, y) \twoheadrightarrow \mathcal{DA}(Fx, Fy)$ for every $x, y : \mathcal{CO}$
- $FT : \mathcal{CT}(f, g, h) \twoheadrightarrow \mathcal{DT}(Ff, Fg, Fh)$ for all $f : \mathcal{CA}(x, y), g : \mathcal{CA}(y, z), h : \mathcal{CA}(x, z)$
- $FE : \mathcal{CE}(f, g) \twoheadrightarrow \mathcal{DE}(Ff, Fg)$ for all $f, g : \mathcal{CA}(x, y)$
- $FI : \mathcal{CI}(f) \twoheadrightarrow \mathcal{DI}(Ff)$ for all $f : \mathcal{CA}(x, x)$

The right notion of equivalence

Main theorem

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

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

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

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

A *very surjective morphism* or *equivalence* $F : \mathcal{C} \simeq \mathcal{D}$ of **univalent** \mathcal{L}_{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 ,

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

where $\cong_{\mathcal{L}\text{-Str}}^*$ denotes levelwise equivalence up to indiscernibility and \twoheadrightarrow denotes a very split surjective morphism.

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

A *very surjective morphism* or *equivalence* $F : \mathcal{C} \simeq \mathcal{D}$ of **univalent** \mathcal{L}_{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 ,

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

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

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

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

- $FO : \mathcal{CO} \twoheadrightarrow \mathcal{DO}$
- $FA : \mathcal{CA}(x, y) \cong \mathcal{DA}(Fx, Fy)$ for every $x, y : \mathcal{CO}$
- $FT : \mathcal{CT}(f, g, h) \leftrightarrow \mathcal{DT}(Ff, Fg, Fh)$ for all $f : \mathcal{CA}(x, y), g : \mathcal{CA}(y, z), h : \mathcal{CA}(x, z)$
- $FE : (f = g) \leftrightarrow (Ff = Fg)$ for all $f, g : \mathcal{CA}(x, y)$
- $FI : \mathcal{CI}(f) \leftrightarrow \mathcal{DI}(Ff)$ for all $f : \mathcal{CA}(x, x)$

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

- \dagger -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-category-theoretic account.

Thank you!