A Higher Structure Identity Principle

Benedikt Ahrens

j.w.w. Paige R North, Michael Shulman, Dimitris Tsementzis

Indiscernability of identicals

Indiscernability of identicals

$$x = y \quad \rightarrow \quad \forall P \, (P(x) \leftrightarrow P(y))$$

- Reasoning **in logic** is invariant under equality
- **In mathematics**, reasoning should be invariant under weaker notion of sameness!

Equivalence principle

Reasoning in mathematics should be **invariant under** the appropriate notion of **sameness**.

Aczel's Structure Identity Principle

Notion of sameness depends on the objects under consideration:

An equivalence principle for group theorists

$$G \cong H \to \forall \text{ group-theoretic properties } P, (P(G) \leftrightarrow P(H))$$

An equivalence principle for category theorists

$$A \simeq B \to \forall \text{ category-theoretic properties } P, (P(A) \leftrightarrow P(B))$$

Aczel's Structure Identity Principle (SIP)

Isomorphic mathematical structures are structurally identical; i.e. have the same structural properties.

Aczel's Structure Identity Principle

Notion of sameness depends on the objects under consideration:

An equivalence principle for group theorists

$$G \cong H \to \forall \text{ group-theoretic properties } P, (P(G) \leftrightarrow P(H))$$

An equivalence principle for category theorists

$$A \simeq B \to \forall \text{ category-theoretic properties } P, (P(A) \leftrightarrow P(B))$$

Aczel's Structure Identity Principle (SIP)

Isomorphic mathematical structures are structurally identical; i.e. have the same structural properties.

What are "structural" properties?

Violating the equivalence principle

What is **not** a structural property?

Exercise

Find a statement about categories that is not invariant under the equivalence of categories

Violating the equivalence principle

What is **not** a structural property?

Exercise

Find a statement about categories that is not invariant under the equivalence of categories

A solution

"The category $\mathscr{C}$ has exactly one object."

Violating the equivalence principle

What is **not** a structural property?

Exercise

Find a statement about categories that is not invariant under the equivalence of categories

A solution

"The category $\mathscr{C}$ has exactly one object."

Can we rule out such "non-structural" statements?

A language for invariant properties

Michael Makkai, *Towards a Categorical Foundation of Mathematics*:

> *The basic character of the Principle of Isomorphism is that of a* ***constraint on the language*** *of Abstract Mathematics; a welcome one, since it provides for the separation of sense from nonsense.*

A language for invariant properties

Michael Makkai, *Towards a Categorical Foundation of Mathematics*:

> *The basic character of the Principle of Isomorphism is that of a* ***constraint on the language*** *of Abstract Mathematics; a welcome one, since it provides for the separation of sense from nonsense.*

Makkai's FOLDS (First Order Logic with Dependent Sorts)

A language for categorical structures in which only invariant properties can be expressed

- FOLDS is not a foundation of mathematics
- Invariance only for properties, not for constructions

Univalent foundations and the equivalence principle

Voevodsky's goals

- Univalent foundations as an "invariant language"
- Invariance not only for statements, but also for constructions: **constructions on things can be transported along equivalences of things**

Essential ingredients for EP in UF

- Martin-Löf identity type
- Voevodsky's univalence axiom

Transport along equivalence of types

$$x =_{\mathscr{U}} y \to \prod_{(P:\mathscr{U}\to\mathscr{U})} (P(x) \simeq P(y))$$

$$\text{univalence} : \prod_{(x,y:\mathscr{U})} (x =_{\mathscr{U}} y \xrightarrow{\sim} x \simeq y)$$

$$x \simeq y \to \prod_{(P:\mathscr{U}\to\mathscr{U})} (P(x) \simeq P(y))$$

Transport along isomorphism of groups

$$x =_{\mathsf{Grp}} y \to \prod_{(P:\mathsf{Grp}\to\mathcal{U})} (P(x) \simeq P(y))$$

$$\text{univalence}: \prod_{(x,y:\mathsf{Grp})} (x =_{\mathsf{Grp}} y \xrightarrow{\sim} x \cong y)$$

$$x \cong y \to \prod_{(P:\mathsf{Grp}\to\mathcal{U})} (P(x) \simeq P(y))$$

Transport along isomorphism of groups

$$x =_{\mathsf{Grp}} y \to \prod_{(P : \mathsf{Grp} \to \mathscr{U})} (P(x) \simeq P(y))$$

$$\text{univalence} : \prod_{(x,y : \mathsf{Grp})} (x =_{\mathsf{Grp}} y \xrightarrow{\sim} x \cong y)$$

$$x \cong y \to \prod_{(P : \mathsf{Grp} \to \mathscr{U})} (P(x) \simeq P(y))$$

Structure Identity Principle

- Works similarly for many other structures that form a 1-category/groupoid
- Coquand & Danielsson, "Isomorphism is equality"
- HoTT book, Section 9.9

Transport along equivalence of **univalent** categories

$$x =_{\mathsf{uCat}} y \to \prod_{(P:\mathsf{uCat}\to\mathcal{U})} (P(x) \simeq P(y))$$

$$\text{univalence} : \prod_{(x,y:\mathsf{uCat})} (x =_{\mathsf{uCat}} y \xrightarrow{\sim} x \simeq y)$$

$$x \simeq y \to \prod_{(P:\mathsf{uCat}\to\mathcal{U})} (P(x) \simeq P(y))$$

Transport along equivalence of **univalent** categories

$$x =_{\mathsf{uCat}} y \to \prod_{(P:\mathsf{uCat}\to\mathcal{U})} (P(x) \simeq P(y))$$

$$\text{univalence} : \prod_{(x,y:\mathsf{uCat})} (x =_{\mathsf{uCat}} y \xrightarrow{\sim} x \simeq y)$$

$$x \simeq y \to \prod_{(P:\mathsf{uCat}\to\mathcal{U})} (P(x) \simeq P(y))$$

SIP for categories

- Holds only for categories that satisfy themselves a univalence condition: **local univalence implies global univalence**
- Univalent categories are the right notion of categories in univalent foundations

Our work: Transport along equivalence of higher structures

- Define a notion of "signature" for higher-categorical structures
- Given signature $\mathscr{L}$, define
  - $\mathscr{L}$-structures
  - Univalence of $\mathscr{L}$-structures
  - Equivalence between $\mathscr{L}$-structures
- Prove a univalence result for univalent $\mathscr{L}$-structures:

$$\mathsf{univalence} : \prod_{(x,y:\mathbf{Struc}_{\mathscr{L}})} \left((x =_{\mathbf{uStruc}_{\mathscr{L}}} y) \xrightarrow{\sim} (x \simeq_{\mathbf{uStruc}_{\mathscr{L}}} y) \right)$$

Overview

In this talk

- Review of univalent foundations
- Univalence for set-level structures (Structure Identity Principle)
- Univalence for **univalent** categories
- Univalence for other higher-categorical structures (Higher Structure Identity Principle)

Overview

1. Primer: Martin-Löf type theory
2. Some important concepts in univalent foundations
3. The Structure Identity Principle for set-level structures
4. Equivalence principle for categories
5. FOLDS signature and structures for categories
6. Digression: equality and theories
7. Isomorphisms and univalence
8. Equivalence principle for FOLDS structures

Outline

Martin-Löf's dependent type theory

Summary: Martin-Löf type theory

- A language for mathematics and functional programming
- Consists of types and terms
- Every term has a unique type, written $a : A$
- **Type dependency:** type families with parameter(s)

Example (Dependent types)

- $\mathsf{Vector}(A, n)$ — type of vectors over A of length n
- $\mathsf{isEven}(n)$ — propositions are types
- $a =_A b$ — identity type for $a, b : A$

Overview of types in type theory

Type former	Notation	(special case)	canonical term
Inhabitant	$a : A$		
Dependent type	$x : A \vdash B(x)$		
Sigma type	$\sum_{x:A} B(x)$	$A \times B$	(a, b)
Product type	$\prod_{x:A} B(x)$	$A \to B$	$\lambda(x : A).b$
Coproduct type	$A + B$		$\mathsf{inl}(a)$, $\mathsf{inr}(b)$
Identity type	$a = b$		$\mathsf{refl}(a) : a = a$
Universe	$\mathscr{U}$		
Base types	Nat, Bool, 1, 0		

Universes

Universes

- There is also a type $\mathscr{U}$. Its elements are types, $A : \mathscr{U}$.
- The dependent type $x : A \vdash B$ is equivalently a function

$$\lambda x.B : A \to \mathscr{U}$$

What is the type of $\mathscr{U}$?

- Actually, hierarchy $(\mathscr{U}_i)_{i \in I}$ to avoid paradoxes.
- But we ignore this for the most part, and only write $\mathscr{U}$.

$$(n : \mathsf{Nat}), (A : \mathscr{U}) \vdash \mathsf{Vect}(A, n) : \mathscr{U}$$

Identity vs equality

Inhabitants of $x = y$ behave like equality in many ways

- sym : $\prod_{x,y:A} x = y \to y = x$
- trans : $\prod_{x,y,z:A} x = y \times y = z \to x = z$
- transport^A : $\prod_{x,y:A} x = y \to \prod_{B:A\to\mathcal{U}} \big(B(x) \simeq B(y)\big)$

Inhabitants of $x = y$ behave **un**like equality

- Can iterate identity type
- Cannot show that any two identities are identical

Outline

The important features of univalent foundations

Homotopy levels

- Stratification of types according to “complexity” of their identity types
- Logic: notion of propositions given by one layer of this hierarchy

Univalence axiom

Specifies the identity type of a universe, which is unspecified in pure Martin-Löf type theory

Logic in univalent foundations

Propositions

Type A is a proposition when we can define a term of type

$$\mathsf{isProp}(A) \;:\equiv\; \prod_{x,y:A} x = y$$

$$\mathsf{Prop} \;:\equiv\; \sum_{X:\mathcal{U}} \mathsf{isProp}(X) \qquad \text{world of propositions}$$

- implication $A \to B$
- conjunction $A \times B$
- predicate $B : A \to \mathsf{Prop}$
- universal quantification $\prod_{a:A} B(a)$
- disjunction and existential need extra care

Contractible types, propositions and sets

- A is **contractible**

$$\mathsf{isContr}(A) \;:\equiv\; \sum_{x:A} \prod_{y:A} y = x$$

- A is a **proposition**

$$\mathsf{isProp}(A) \;:\equiv\; \prod_{x,y:A} x = y$$

- A is a **set**

$$\mathsf{isSet}(A) \;:\equiv\; \prod_{x,y:A} \mathsf{isProp}(x = y)$$

$$\mathsf{Prop} \;:\equiv\; \sum_{X:\mathcal{U}} \mathsf{isProp}(X) \qquad \mathsf{Set} \;:\equiv\; \sum_{X:\mathcal{U}} \mathsf{isSet}(X)$$

Contractible types, propositions and sets

- A is **contractible**

$$\mathsf{isContr}(A) \quad :\equiv \quad \sum_{x:A} \prod_{y:A} y = x$$

- A is a **proposition**

$$\mathsf{isProp}(A) \quad :\equiv \quad \prod_{x,y:A} \mathsf{isContr}(x = y)$$

- A is a **set**

$$\mathsf{isSet}(A) \quad :\equiv \quad \prod_{x,y:A} \mathsf{isProp}(x = y)$$

$$\mathsf{Prop} \quad :\equiv \quad \sum_{X:\mathcal{U}} \mathsf{isProp}(X) \qquad \mathsf{Set} \quad :\equiv \quad \sum_{X:\mathcal{U}} \mathsf{isSet}(X)$$

Equivalences

Definition

A map $f : A \to B$ is an **equivalence** if it has contractible fibers, i.e.,

$$\mathsf{isequiv}(f) \;:\equiv\; \prod_{b:B} \mathsf{isContr}\left(\sum_{a:A} f(a) = b\right)$$

The type of equivalences:

$$A \simeq B \;:\equiv\; \sum_{f:A\to B} \mathsf{isequiv}(f)$$

Outline

Monoids in type theory

In type theory, a monoid is a tuple $(M, \mu, e, \alpha, \lambda, \rho)$ where

1. $M : \mathbf{Set}$
2. $\mu : M \times M \to M$
3. $e : M$
4. $\alpha : \Pi_{(a,b,c:M)} \mu(\mu(a,b),c) = \mu(a,\mu(b,c))$
5. $\lambda : \Pi_{(a:M)} \mu(e,a) = a$
6. $\rho : \Pi_{(a:M)} \mu(a,e) = a$

Monoids in type theory

In type theory, a monoid is a tuple $(M, \mu, e, \alpha, \lambda, \rho)$ where

1. $M : \mathbf{Set}$
2. $\mu : M \times M \to M$
3. $e : M$
4. $\alpha : \Pi_{(a,b,c:M)} \mu(\mu(a,b),c) = \mu(a,\mu(b,c))$
5. $\lambda : \Pi_{(a:M)} \mu(e,a) = a$
6. $\rho : \Pi_{(a:M)} \mu(a,e) = a$

Why $M : \mathbf{Set}$?

Monoids in type theory

In type theory, a monoid is a tuple $(M, \mu, e, \alpha, \lambda, \rho)$ where

1. $M : \mathbf{Set}$
2. $\mu : M \times M \to M$
3. $e : M$
4. $\alpha : \Pi_{(a,b,c:M)} \mu(\mu(a,b),c) = \mu(a,\mu(b,c))$
5. $\lambda : \Pi_{(a:M)} \mu(e,a) = a$
6. $\rho : \Pi_{(a:M)} \mu(a,e) = a$

Why $M : \mathbf{Set}$?

Abstractly, a monoid is a (dependent) pair $(data, proof)$ where

- *data* is 1.–3.
- *proof* is 4.–6.

The type of monoids

- We want two monoids $(data, proof)$ and $(data', proof')$ to be the same if $data$ is the same as $data'$.
- This is guaranteed when the types of $proof$ and $proof'$ are **propositions**.
- This in turn is guaranteed when M is a **set**.

Summarily:

$$\mathsf{Monoid} \;:\equiv \sum_{(M:\mathsf{Set})} \sum_{(\mu,e):\mathsf{MonoidStr}(M)} \mathsf{MonoidAxioms}(M,(\mu,e))$$

Can show

$$\mathsf{isProp}(\mathsf{MonoidAxioms}(M,(\mu,e)))$$

Monoid isomorphisms

Given $\mathbf{M} \equiv (M, \mu, e, \alpha, \lambda, \rho)$ and $\mathbf{M}' \equiv (M', \mu', e', \alpha', \lambda', \rho')$, a **monoid isomorphism** is a bijection $f : M \cong M'$ preserving μ and e.

Monoid isomorphisms

Given $\mathbf{M} \equiv (M, \mu, e, \alpha, \lambda, \rho)$ and $\mathbf{M}' \equiv (M', \mu', e', \alpha', \lambda', \rho')$, a **monoid isomorphism** is a bijection $f : M \cong M'$ preserving μ and e.

$$
\begin{aligned}
\mathbf{M} = \mathbf{M}' &\simeq (M, \mu, e) = (M', \mu', e') \\
&\simeq \sum_{p : M = M'} (\mathsf{transport}^{Y \mapsto (Y \times Y \to Y)}(p, \mu) = \mu') \\
&\qquad\qquad \times (\mathsf{transport}^{Y \mapsto Y}(p, e) = e') \\
&\simeq \sum_{f : M \cong M'} \left(f \circ \mu \circ (f^{-1} \times f^{-1}) = \mu'\right) \\
&\qquad\qquad \times (f \circ e = e') \\
&\simeq \mathbf{M} \cong \mathbf{M}'
\end{aligned}
$$

Transport along monoid isomorphism

We now have two ingredients:

1.
$$\mathsf{transport}_{\mathbf{M},\mathbf{M}'} : (\mathbf{M} = \mathbf{M}') \to \prod_{B:\mathsf{Monoid}\to\mathcal{U}} \big(B(\mathbf{M}) \simeq B(\mathbf{M}')\big)$$

2.
$$(\mathbf{M} = \mathbf{M}') \quad \simeq \quad (\mathbf{M} \cong \mathbf{M}')$$

Composing these, we get

$$\mathsf{transport}_{\mathbf{M},\mathbf{M}'} : (\mathbf{M} \cong \mathbf{M}') \to \prod_{B:\mathsf{Monoid}\to\mathcal{U}} \big(B(\mathbf{M}) \simeq B(\mathbf{M}')\big)$$

I.e., any structure on monoids that can be expressed in univalent type theory can be transported along isomorphism of monoids.

Structure Identity Principle

Structure Identity Principle (Aczel, Coquand&Danielsson)

For many set-level structures in univalent foundations, paths are isomorphisms.

Examples include:

- monoids, groups, rings
- posets
- discrete fields
- sets with fixpoint operator

Structure Identity Principle

Structure Identity Principle (Aczel, Coquand&Danielsson)

For many set-level structures in univalent foundations, paths are isomorphisms.

Examples include:

- monoids, groups, rings
- posets
- discrete fields
- sets with fixpoint operator

What about **categories**?

Outline

Categories in type theory

A **category** $\mathscr{C}$ is given by

- a type $\mathscr{C}_0 : \mathscr{U}$ of **objects**
- for any $a, b : \mathscr{C}_0$, a set $\mathscr{C}(a, b) : \mathscr{U}$ of **morphisms**
- operations: identity & composition

$$1_a : \mathscr{C}(a, a)$$
$$(\circ)_{a,b,c} : \mathscr{C}(b, c) \to \mathscr{C}(a, b) \to \mathscr{C}(a, c)$$

- axioms: unitality & associativity

$$1 \circ f = f \qquad f \circ 1 = f \qquad (h \circ g) \circ f = h \circ (g \circ f)$$

Categories in type theory

A **category** $\mathscr{C}$ is given by

- a type $\mathscr{C}_0 : \mathscr{U}$ of **objects**
- for any $a, b : \mathscr{C}_0$, a set $\mathscr{C}(a,b) : \mathscr{U}$ of **morphisms**
- operations: identity & composition

$$1_a : \mathscr{C}(a,a)$$
$$(\circ)_{a,b,c} : \mathscr{C}(b,c) \to \mathscr{C}(a,b) \to \mathscr{C}(a,c)$$

- axioms: unitality & associativity

$$1 \circ f = f \qquad f \circ 1 = f \qquad (h \circ g) \circ f = h \circ (g \circ f)$$

A **univalent category** is a category $\mathscr{C}$ such that

$$(a = b) \to (a \cong b)$$

is an equivalence for all $a, b : \mathscr{C}_0$.

Local univalence implies global univalence

Theorem

For categories A and B, let $A \simeq B$ denote the type of equivalences from A to B. If A and B are univalent, we have

$$(A =_{\mathsf{uCat}} B) \simeq (A \simeq B).$$

Corollary

If A and B are equivalent univalent categories, then they share the same properties.
For any $X : \mathsf{uCat} \vdash P(X) : \mathcal{U}$,

$$(A \simeq B) \to (P(A) \simeq P(B)).$$

Goal

Prove a similar theorem for other categorical structures.

Envisioned result

Given a signature $\mathscr{L}$, and two $\mathscr{L}$-univalent $\mathscr{L}$-structures M and N, then

$$(M = N) = (M \simeq_{\mathscr{L}} N)$$

Need notions of

- signatures $\mathscr{L}$
- $\mathscr{L}$-structures
- $\mathscr{L}$-equivalence of $\mathscr{L}$-structures
- $\mathscr{L}$-univalence of $\mathscr{L}$-structures

Outline

Two-level type theory

Working in the two-level type theory of Annenkov-Capriotti-Kraus.

- Universes $\mathscr{U} \hookrightarrow \mathscr{U}^s$
- $\mathscr{U}$ implements univalent type theory.
- Every type $T : \mathscr{U}^s$ is equipped with a strict equality type $a \equiv_T b$ with the usual rules for the identity type, but which also satisfies UIP.

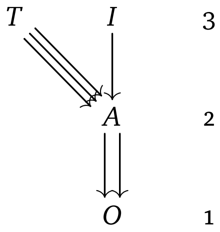
FOLDS-signatures

Definition

A signature is a graded one-way semicategory of finite height:

- all arrows go "downwards"
- the height is the minimum natural number with non-empty type of objects

$\mathscr{L}_{\text{cat}} :=$

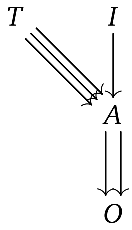


with some strict equalities between morphisms

Overview: $\mathscr{L}_{\text{cat}}$-structures

We can define the data of a category $\mathscr{C}$ to be

- A type $\mathscr{C}O : \mathscr{U}$
- A family $\mathscr{C}A : \mathscr{C}O \times \mathscr{C}O \to \mathscr{U}$
- A family $\mathscr{C}I : \prod_{(x:\mathscr{C}O)} \mathscr{C}A(x,x) \to \mathscr{U}$
- A family $\mathscr{C}T : \prod_{(x,y,z:\mathscr{C}O)} \mathscr{C}A(x,y) \to \mathscr{C}A(y,z) \to \mathscr{C}A(x,z) \to \mathscr{U}$



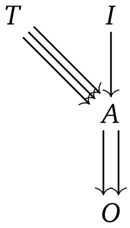
Here:

- Think of $\mathscr{C}I$, $\mathscr{C}T$ as the *predicates* ‘is an identity’, ‘is a composite’.
- $\mathscr{L}_{\text{cat}}$-*univalence* will imply that $\mathscr{C}I$, $\mathscr{C}T$ are pointwise propositions.
- $\mathscr{L}_{\text{cat}}$-*univalence* will imply that $\mathscr{C}A$ is pointwise a set.
- $\mathscr{L}_{\text{cat}}$-*univalence* will imply that $\mathscr{C}O$ is a 1-type.

Overview: $\mathscr{L}_{\text{cat}}$-structures

We can define the data of a category $\mathscr{C}$ to be

- A type $\mathscr{C}O : \mathscr{U}$
- A family $\mathscr{C}A : \mathscr{C}O \times \mathscr{C}O \to \mathscr{U}$
- A family $\mathscr{C}I : \prod_{(x:\mathscr{C}O)} \mathscr{C}A(x,x) \to \mathscr{U}$
- A family $\mathscr{C}T : \prod_{(x,y,z:\mathscr{C}O)} \mathscr{C}A(x,y) \to \mathscr{C}A(y,z) \to \mathscr{C}A(x,z) \to \mathscr{U}$



Remark: these $\mathscr{L}_{\text{cat}}$-structures do not satisfy any axioms, e.g.,

- composition is a functional relation
- left and right unitality for composition

Such axioms will be discussed now.

Outline

Equality

To the data, we add axioms such as

- "There is a composite of every composable pair of arrows."

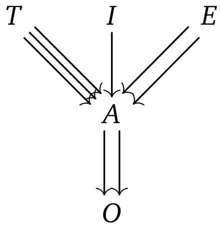
$$\forall(x,y,z:O).\forall(f:A(x,y)).\forall(g:A(y,z)).\exists(h:A(x,z)).T_{x,y,z}(f,g,h)$$

- "Composites are unique."

$$\forall(x,y,z:O).\forall(f:A(x,y)).\forall(g:A(y,z)).\forall(h,h':A(x,z)).$$
$$T_{x,y,z}(f,g,h)\to T_{x,y,z}(f,g,h')\to(h=h')$$

So we need to add an equality 'predicate':

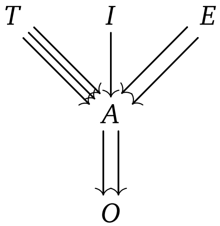
$\mathscr{L}_{\text{cat+E}} :=$



$\mathscr{L}_{\text{cat+E}}$-structures

We can define the data of a category $\mathscr{C}$ to be

- A type $\mathscr{C}O : \mathscr{U}$
- A family $\mathscr{C}A : \mathscr{C}O \times \mathscr{C}O \to \mathscr{U}$
- A family $\mathscr{C}I : \prod_{(x:\mathscr{C}O)} \mathscr{C}A(x,x) \to \mathscr{U}$
- A family $\mathscr{C}T : \prod_{(x,y,z:\mathscr{C}O)} \mathscr{C}A(x,y) \to \mathscr{C}A(y,z) \to \mathscr{C}A(x,z) \to \mathscr{U}$
- A family $\mathscr{C}E : \prod_{(x,y:\mathscr{C}O)} \mathscr{C}A(x,y) \to \mathscr{C}A(x,y) \to \mathscr{U}$



Here:

- $\mathscr{L}_{\text{cat+E}}$-*univalence* will imply that $\mathscr{C}E$ is a proposition.
- $\mathscr{L}_{\text{cat+E}}$-*univalence* + axioms making E into an equivalence relation and congruence will imply that $(f = g) = \mathscr{C}E(f,g)$.

1-univalent FOLDS-categories

A 1-univalent FOLDS-category consists of an $\mathscr{L}_{\mathrm{cat+E}}$-structure

- $\mathscr{C}O : \mathscr{U}$
- $\mathscr{C}A : \mathscr{C}O \times \mathscr{C}O \to \mathscr{U}$
- $\mathscr{C}I : \prod_{(x:\mathscr{C}O)} \mathscr{C}A(x,x) \to \mathscr{U}$
- $\mathscr{C}T : \prod_{(x,y,z:\mathscr{C}O)} \mathscr{C}A(x,y) \to \mathscr{C}A(y,z) \to \mathscr{C}A(x,z) \to \mathscr{U}$
- $\mathscr{C}E : \prod_{(x,y:\mathscr{C}O)} \mathscr{C}A(x,y) \to \mathscr{C}A(x,y) \to \mathscr{U}$

such that

- $\mathscr{C}I_x(f)$, $\mathscr{C}T_{x,y,z}(f,g,h)$, and $\mathscr{C}E_{x,y}(f,g)$ are propositions
- $\mathscr{C}A(x,y)$ is a set,
- $\mathscr{C}E_{x,y}(f,g) = (f = g)$,

and the axioms of a category are satisfied.

Lemma

The type of 1-univalent FOLDS-cats is equivalent to the type of (pre)categories.

Summary: equality and axioms

Summary on equality

- Can add equality predicate
- Typically add equality on top-level (e.g., for A, but not for O)
- Imposing suitable axioms on equality predicate ensures it is equivalent to actual equality

Remark

The notion of isomorphism and univalence we give does not depend on axioms for structures, but only on the shape of structures, i.e., on the signature

Outline

Univalent FOLDS-categories

Goal

To state the univalence condition

$$(a = b) \simeq (a \cong b)$$

for categories in terms of the the FOLDS structure.

Given $a, b : \mathscr{C}O$, we can define an isomorphism $a \cong b$ using the Yoneda Lemma:

- For each $x : \mathscr{C}O$, an equality $\phi_{x\bullet} : \mathscr{C}A(x, a) = \mathscr{C}A(x, b)$.
- For each $x, y : \mathscr{C}O$, $f : \mathscr{C}A(x, y)$, $g : \mathscr{C}A(y, a)$, and $h : \mathscr{C}A(x, a)$, we have

$$\mathscr{C}T_{x,y,a}(f, g, h) = \mathscr{C}T_{x,y,b}(f, \phi_{y\bullet}(g), \phi_{x\bullet}(h))$$

  with $\phi_{y\bullet}(g) \circ f = \phi_{x\bullet}(g \circ f)$

This is a bit ad hoc and not symmetric.

FOLDS isomorphism for categories

Instead, can define $a \cong b$ to consist of the following equalities between all the types of our signature with a and b substituted in *all* possible ways:

- For each $x : \mathscr{C}O$, an equality $\phi_{x\bullet} : \mathscr{C}A(x,a) = \mathscr{C}A(x,b)$.
- For each $z : \mathscr{C}O$, an equality $\phi_{\bullet z} : \mathscr{C}A(a,z) = \mathscr{C}A(b,z)$.
- An equality $\phi_{\bullet\bullet} : \mathscr{C}A(a,a) = \mathscr{C}A(b,b)$.
- The following equalities for all appropriate w, x, y, z, f, g, h:

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

$$I_{a,a}(f) = I_{b,b}(\phi_{\bullet\bullet}(f))$$
$$E_{x,a}(f,g) = E_{x,b}(\phi_{x\bullet}(f), \phi_{x\bullet}(g))$$
$$E_{a,x}(f,g) = E_{b,x}(\phi_{\bullet x}(f), \phi_{\bullet x}(g))$$
$$E_{a,a}(f,g) = E_{b,b}(\phi_{\bullet\bullet}(f), \phi_{\bullet\bullet}(g))$$

"Everything above a, b thinks that a and b are the same."

Univalent FOLDS categories

Theorem

In any 1-univalent FOLDS category, the type of isomorphisms $a \cong b$ just defined is equivalent to the type of ordinary isomorphisms $a \cong b$.

Definition

A univalent FOLDS category is a 1-univalent FOLDS category such that for all $a, b : \mathscr{C} O$, the canonical map

$$(a = b) \to (a \cong b)$$

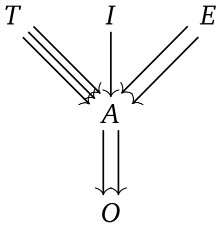
is an equivalence.

Theorem

A 1-univalent FOLDS category is univalent if and only if its corresponding precategory is a univalent category.

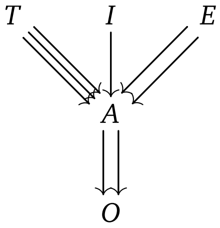
Univalence at O, at A, at T, I, and E

- For an $\mathscr{L}_{\mathrm{cat+E}}$-structure $\mathscr{C}$, we have given a definition of isomorphism for two objects $a, b : \mathscr{C}O$.
- This definition does not depend on the axioms of a category (e.g., composition is a function), but only on the signature.



Univalence at O, at A, at T, I, and E

- For an $\mathscr{L}_{\text{cat+E}}$-structure $\mathscr{C}$, we have given a definition of isomorphism for two objects $a, b : \mathscr{C}O$.
- This definition does not depend on the axioms of a category (e.g., composition is a function), but only on the signature.

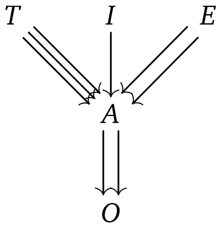


The same principle gives a definition of isomorphism between

- two morphisms $f, g : \mathscr{C}A$ between same objects
- two triangles $c, d : \mathscr{C}T$ in the same fiber, etc.

Univalence at T

- For any $c, d : \mathscr{C}T_{x,y,a}(f, g, h)$, everything above c, d "thinks" c and d are the same, trivially.
- So $(c \cong d) = 1$, and $\mathscr{C}T$ being univalent means that $(c = d) = (c \cong d)$.
- Thus, $\mathscr{C}$ being univalent at T means that each $\mathscr{C}T_{x,y,a}(f, g, h)$ is a proposition.
- Similar for I and E



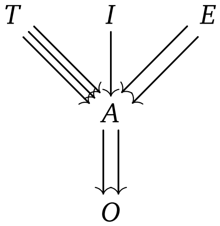
Univalence at A

Let $\mathscr{C}$ be a $\mathscr{L}_{\mathrm{cat}}$-structure that is univalent at T, I, and E.

- For any $f, g : \mathscr{C}A(a,b)$, the type $f \cong g$ is a product of equivalences

$$(f \cong g) = \Big(T(f,k,h) = T(g,k,h)\Big) \times \dots$$

- These types of equivalences are propositions.
- Thus, $\mathscr{C}$ being univalent at A means that each $\mathscr{C}A(a,b)$ is a set.



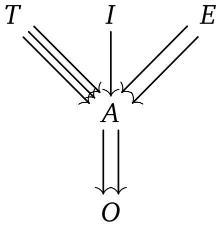
Univalence at A

Let $\mathscr{C}$ be a $\mathscr{L}_{\text{cat}}$-structure that is univalent at T, I, and E.

- For any $f, g : \mathscr{C}A(a,b)$, the type $f \cong g$ is a product of equivalences

$$(f \cong g) = \Big(T(f,k,h) = T(g,k,h)\Big) \times \dots$$

- These types of equivalences are propositions.
- Thus, $\mathscr{C}$ being univalent at A means that each $\mathscr{C}A(a,b)$ is a set.
    - Also, $f \cong g$ implies $E_{a,b}(f,g)$ and conversely, hence univalence at A means $(f = g) = E_{a,b}(f,g)$.



Categorical equivalences

For univalent FOLDS categories $\mathscr{C}, \mathscr{D}$, we had an equivalence.

$$(\mathscr{C} = \mathscr{D}) \simeq (\mathscr{C} \simeq \mathscr{D})$$

We can also FOLDS-ify categorical equivalences:

- A *very split surjective morphism* $F : \mathscr{C} \twoheadrightarrow \mathscr{D}$ of $\mathscr{L}_{\mathrm{cat+E}}$-structures consists of split surjections
  - $F_O : \mathscr{C}O \twoheadrightarrow \mathscr{D}O$
  - $F_A : \prod_{x,y:\mathscr{C}O} \mathscr{C}A(x,y) \twoheadrightarrow \mathscr{D}A(F_O x, F_O y)$
  - $F_T : \prod_{x,y,z:\mathscr{C}O, f:\mathscr{C}A(x,y), g:\mathscr{C}A(y,z), h:\mathscr{C}A(x,z)} \mathscr{C}T(f,g,h) \twoheadrightarrow \mathscr{D}T(F_A f, F_A g, F_A h)$
  - ...

Theorem

For univalent FOLDS categories $\mathscr{C}$ and $\mathscr{D}$ we have

$$(\mathscr{C} \twoheadrightarrow \mathscr{D}) \simeq (\mathscr{C} \simeq \mathscr{D})$$

Outline

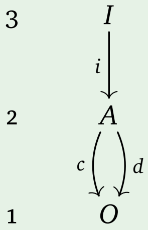
Reminder: signatures

Definition

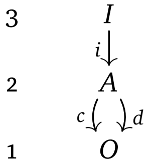
A signature is a graded one-way semicategory of finite height:

- all arrows go “downwards”
- the height is the minimum natural number with non-empty type of objects

Example (Signature for reflexive graphs)

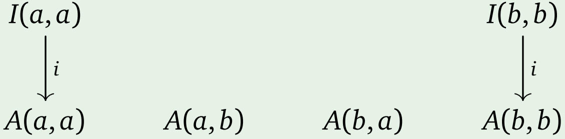


Derivation by example



Example

In $\mathscr{L}_{\mathrm{rg}}$ we have $(\mathscr{L}_{\mathrm{rg}})_0 \equiv \{O\}$. Let M_0 be a (a function picking out) the two-element set $\{a, b\}$. Then $(\mathscr{L}_{\mathrm{rg}})'_{M_0}$ is the following signature, with four sorts of rank 0 and two sorts of rank 1:



Structures of a signature

Example

An $\mathscr{L}_{\mathbf{rg}}$**-structure** is given by

- a type M_o
- recursively, a structure for the signature $(\mathscr{L}_{\mathrm{rg}})'_{M_o}$

Definition

An $\mathscr{L}$**-structure** M is given by

- a function $M_o : \mathscr{L}_o \to \mathscr{U}$
- recursively, a structure M' for the signature $(\mathscr{L})'_M$

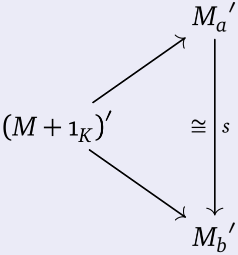
There are also suitable notions of

- morphism of structures
- isomorphism of structures

$\mathscr{L}$-isomorphism

Definition

Let M be an $\mathscr{L}$-structure, $K : \mathscr{L}_0$, and $a, b : MK$. The type $a \cong_M b$ of $\mathscr{L}$-isomorphisms from a to b is defined to be the type of levelwise equivalences $s : M_a \cong {M_b}'$ in $\mathbf{Struc}_{\mathscr{L}'_{M_0+1_K}}$ under $(M + 1_K)'$, i.e. those s such that the following triangle commutes:



$\mathscr{L}$-univalence

Definition ($\mathscr{L}$-univalence)

Let K be an object of $\mathscr{L}_0$. We say that M is **univalent at** K if

$$\prod_{x,y\,:\,MK} \mathsf{isequiv}(\mathsf{idtoiso}^{MK}_{x,y})$$

We say that M is **$\mathscr{L}$-univalent** if

- M is K-univalent for every $K : \mathscr{L}_0$ and
- M' is univalent.

Results

Theorem

Let $\mathscr{L} : \mathsf{Sig}(n+1)$*, M a univalent* $\mathscr{L}$*-structure,* $K : \mathscr{L}_0$*. Then* $M_0(K)$ *is of h-level* $n+1$.

Theorem

For a signature $L : \mathsf{Sig}(n)$*, the type of univalent L-structures is of h-level* $n+1$.

Theorem (Higher Structure Identity Principle)

Consider $\mathscr{L}$*-structures* M, N *for some signature* $\mathscr{L}$ *such that M is univalent. Then*

$$(M = N) = (M \twoheadrightarrow N)$$

Examples

- First-order logic (with equality)
- Categories
- Dagger categories
- (Ana)functors
- Profunctors
- Displayed categories / Fibrations

Remarks

Preprint to be on the arXiv soon

- Two notions of signature: FOLDS-signatures and axiomatic signatures
- Translation from FOLDS- to axiomatic signatures
- Examples in terms of FOLDS-signatures
- Abstract reasoning about axiomatic signatures

Remarks

Preprint to be on the arXiv soon

- Two notions of signature: FOLDS-signatures and axiomatic signatures
- Translation from FOLDS- to axiomatic signatures
- Examples in terms of FOLDS-signatures
- Abstract reasoning about axiomatic signatures

Thanks for your attention!