

Univalent Foundations and the Structure Identity Principle

Ulrik Buchholtz

Logic Seminar, Stanford, January 8, 2013

- 1 Outline
- 2 Motivation for Univalent Foundations
- 3 Review of Constructive Type Theory
- 4 The Homotopy-theoretical Interpretation
- 5 Mathematics in Homotopy Type Theory
- 6 Open Problems

The material in this talk is mostly gathered from informal writings and talks by

- Peter Aczel
- Thierry Coquand
- Mike Shulman
- Thomas Streicher
- Vladimir Voevodsky
- *and many others!*

Most references can be found through the Homotopy Type Theory website, <http://homotopytypetheory.org/>, and blog posts at The n -Category Café, <http://golem.ph.utexas.edu/category/>.

- 1 Outline
- 2 Motivation for Univalent Foundations**
- 3 Review of Constructive Type Theory
- 4 The Homotopy-theoretical Interpretation
- 5 Mathematics in Homotopy Type Theory
- 6 Open Problems

Motivation for Univalent Foundations

For the purposes of this talk, I'm going to motivate Univalent Foundations through a basic observation of mathematical practice:

The Structure Identity Principle

Isomorphic structures are structurally identical in that they share the same *structural properties*.

And in mathematical practice, isomorphic structures are used *interchangeably*.

Therefore, isomorphic structures should be *identical* (since that's what identity is for: substitution).

Motivation for Univalent Foundations

For the purposes of this talk, I'm going to motivate Univalent Foundations through a basic observation of mathematical practice:

The Structure Identity Principle

Isomorphic structures are structurally identical in that they share the same *structural properties*.

And in mathematical practice, isomorphic structures are used *interchangeably*.

Therefore, isomorphic structures should be *identical* (since that's what identity is for: substitution).

However, what exactly are the *structural properties*? And how can we achieve that isomorphic structures are actually identical?

Failure of the SIP in material set theory

The von Neumann naturals: $\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}, \dots$

The Zermelo naturals: $\emptyset, \{\emptyset\}, \{\{\emptyset\}\}, \dots$

These give rise to isomorphic Peano structures, but differ on whether, e.g.,

$$0 \in 2?$$

Failure of the SIP in material set theory

The von Neumann naturals: $\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}, \dots$

The Zermelo naturals: $\emptyset, \{\emptyset\}, \{\{\emptyset\}\}, \dots$

These give rise to isomorphic Peano structures, but differ on whether, e.g.,

$$0 \in 2?$$

That is, therefore, not a structural property.

Homotopy type theory (HoTT) is intensional dependent type theory with the Univalence Axiom (to be defined in the sequel).

SIP in HoTT

Isomorphic structures are identical.

If $A, B: C$ where C is a type of structures of some kind, then

$$A \cong_C B \rightarrow \text{Id}_C(A, B)$$

Obviously, if this approach is to work, an identity claim cannot be “proof-irrelevant” in the usual sense, because there are two isomorphisms of a two-element set with itself, and we’re in trouble if we don’t know which one we’re using when we’re substituting using such an identity.

Therefore, in the Univalent Foundations we need to be able to recover information from proofs. This is well-known from constructive mathematics, where for instance an inequality between real numbers carries important information (at what precision the numbers differ).

Relevant propositions in mathematical practice

Even in ordinary mathematical practice, examples of proof-relevance abounds.

This is particularly true in higher algebra (algebraic topology and algebraic geometry) where we find writing like “by the proof of . . .” or “the map constructed in the proof of . . .”

Higher dimensional identity

It is now well known that the correct notion of structural isomorphism varies according to the “dimensionality” of the concepts involved.

Dimension	Structure	Criterion of identity
0	Sets: elements/objects	equality
1	Categories: plus arrows	isomorphism
2	2-categories: plus 2-arrows	equivalence
⋮	⋮	⋮

In Homotopy Type Theory, the notion of *identity* will capture all these.

Structuralism versus materialism

We have two competing notions of set theory:

- Material set theories, such as ZFC.
- Structural set theories, such as ETCS.

ETCS goes a long way towards realizing our structuralist goals, but fails for two reasons:

- 1 Too many things are “coded”: subsets, functions, pairs. (Note that at least in ZFC, subsets *are just* subsets.)
- 2 Even though it only gives structural properties, we can't reap the benefits, i.e., the Strong Structure Identity Principle.

Dependent type theory with the Axiom of Univalence solves these problems, with a language that's *more* natural.

Aside: Structuralist elements in Cantor

Cantor, *Beiträge zur Begründung der transfiniten Mengenlehre*, 1895:

Every set M has a definite “power,” which we will also call its “cardinal number.”

We will call by the name “power” or “cardinal number” of M the general concept by which, by means of our active faculty of thought, arises from the set M when we make abstraction of the nature of its various elements m and of the order in which they are given.

We denote the result of this double act of abstraction, the cardinal number or power of M , by \overline{M} .

Since every single element m , if we abstract from its nature, becomes a “unit,” the cardinal number \overline{M} is a definite set composed of units, and this number has existence in our mind as an intellectual image or projection of the given set M .

- 1 Outline
- 2 Motivation for Univalent Foundations
- 3 Review of Constructive Type Theory**
- 4 The Homotopy-theoretical Interpretation
- 5 Mathematics in Homotopy Type Theory
- 6 Open Problems

Review of Constructive Type Theory

Usually motivated through the Brouwer-Heyting-Kolmogorov interpretation and Martin-Löf's meaning explanations.

However, we can also motivate dependent type theory directly as the natural way to build a language containing only *structural properties*.

To capture the idea that some mathematical objects are what they are “by their very nature” (essentially, not accidentally), we build the language on *typing judgements*, $\Gamma \vdash a : A$.

This should be a decidable relation (“we recognize a proof when we see it”).

A judgement has the form $\Gamma \vdash \mathcal{B}$ where Γ is a context

$$x_1: A_1, x_2: A_2, \dots, x_n: A_n$$

where each A_i may reference the previous variables x_1, \dots, x_{i-1} , and \mathcal{B} has one of the forms

- A Type
- $a: A$
- $A = B$ Type
- $a = b: A$

(and all these terms may reference the variables x_i in the context.)

As an example we give the rules for dependent products:

$$\frac{\Gamma \vdash A \text{ Type} \quad \Gamma, x: A \vdash B \text{ Type}}{\Gamma \vdash \Pi_{x:A} B \text{ Type}}$$

$$\frac{\Gamma, x: A \vdash b: B}{\Gamma \vdash \lambda x: A. b: \Pi_{x:A} B}$$

$$\frac{\Gamma \vdash f: \Pi_{x:A} B \quad \Gamma \vdash a: A}{\Gamma \vdash f(a): B[x := a]}$$

$$\frac{\Gamma, x: A \vdash b: B \quad \Gamma \vdash a: A}{\Gamma \vdash (\lambda x: A. b)(a) = b[x := a]: B[x := a]}$$

It a minimum we need

- $0, 1, \mathbb{B}, \mathbb{N}$
- $\Pi_{x:A} B$
- $\Sigma_{x:A} B$

Then we can define

- $A \rightarrow B := \Pi_{-:A} B$
- $A \times B := \Sigma_{-:A} B$

With *inductive types*, we only need dependent product.

Propositions as Types

Proposition	Type
\perp	0
\top	1

Propositions as Types

Proposition	Type
\perp	0
\top	1
$A \wedge B$	$A \times B$
$A \vee B$	$A + B$
$A \Rightarrow B$	$A \rightarrow B$

Propositions as Types

Proposition	Type
\perp	0
\top	1
$A \wedge B$	$A \times B$
$A \vee B$	$A + B$
$A \Rightarrow B$	$A \rightarrow B$
$\forall x: A. B(x)$	$\prod_{x:A} B(x)$
$\exists x: A. B(x)$	$\sum_{x:A} B(x)$

Propositions as Types

Proposition	Type
\perp	0
\top	1
$A \wedge B$	$A \times B$
$A \vee B$	$A + B$
$A \Rightarrow B$	$A \rightarrow B$
$\forall x: A. B(x)$	$\prod_{x:A} B(x)$
$\exists x: A. B(x)$	$\sum_{x:A} B(x)$
$x =_A y$	$\text{Id}_A(x, y)$

$$\frac{\Gamma \vdash a: A \quad \Gamma \vdash b: A}{\Gamma \vdash \text{Id}_A(a, b) \text{ Type}}$$

$$\frac{\Gamma \vdash a: A}{\Gamma \vdash \text{r}(a): \text{Id}_A(a, a)}$$

$$\frac{\Gamma, x: A, y: A, p: \text{Id}_A(x, y) \vdash C \text{ Type} \quad \Gamma, x: A \vdash b: C[y := x, p := \text{r}(x)]}{\Gamma, x: A, y: A, p: \text{Id}_A(x, y) \vdash J(C, \lambda x. b): C}$$

$$\frac{\Gamma, x: A, y: A, p: \text{Id}_A(x, y) \vdash C \text{ Type} \quad \Gamma, x: A \vdash b: C[y := x, p := \text{r}(x)]}{\Gamma, x: A \vdash J(C, \lambda x. b)[y := x, p := \text{r}(x)] = b: C[y := x, p := \text{r}(x)]}$$

Identity Types – induction principle

In fact, $\text{Id}_A(a, b)$ is an inductive family with one constructor $r(x)$, so the elimination rule is an induction principle.

Note

To use induction on $p: \text{Id}_A(a, b)$ the type must be fully general. So we can't use induction on an assumption $p: \text{Id}_A(a, a)$ to conclude that $p = r(a)$.

- 1 Outline
- 2 Motivation for Univalent Foundations
- 3 Review of Constructive Type Theory
- 4 The Homotopy-theoretical Interpretation**
- 5 Mathematics in Homotopy Type Theory
- 6 Open Problems

The Hofmann-Streicher Groupoid Model

In the 1990's Martin Hofmann and Thomas Streicher noted that the identity types satisfy conditions similar to the laws of groupoids:

Indeed, $r(a)$ gives unit elements in $\text{Id}_A(a, a)$, and we can derive inverses $f^{-1}: \text{Id}_A(b, a)$ and composites $g \circ f: \text{Id}_A(a, c)$ for $f: \text{Id}_A(a, b)$ and $g: \text{Id}_A(b, c)$ by identity induction.

These satisfy the groupoid laws *up to higher propositional identities*. They conjectured that identity thus should be equipped with the structure of *weak higher dimensional groupoids*.

This was later verified by Benno van den Berg and Richard Garner, and Peter LeFanu Lumsdaine.

The Model in Kan simplicial sets

Around 2006, Vladimir Voevodsky noted that type theory has a homotopical model in simplicial sets, using Kan fibrations to model dependent types, and the usual path-objects from the homotopy theory of simplicial sets to model identity types.

This model verifies (and perhaps inspired?) the Univalence Axiom.

Slogan

We can interpret

- types as spaces
- identity types as path spaces

Models in Abstract Homotopy Theory

Weak factorization systems form one half of Quillen's *model categories* which are an abstract approach to homotopy theory.

Intentional dependent type theory is sound and complete for this class of categorical models following work by Steve Awodey, Michael Warren, Nicola Gambino and Richard Garner.

- 1 Outline
- 2 Motivation for Univalent Foundations
- 3 Review of Constructive Type Theory
- 4 The Homotopy-theoretical Interpretation
- 5 Mathematics in Homotopy Type Theory**
- 6 Open Problems

Can we give some glimpse of what mathematics in Homotopy Type Theory looks like?

Stratification of Types into Homotopy Levels

Some types in Homotopy Type Theory have degenerate identity types. This gives rise to a whole hierarchy of homotopy levels (hLevels) analogous to the notion of truncated spaces in topology. A type represents an n -truncated space if and only if it has homotopy level $n + 2$.

hLevel	Name	Definition
0	contractible	$\text{isContr}(X) := \Sigma_{x:X} \Pi_{y:X} \text{Id}_X(y, x)$
1	hProp	$\Pi_{x:X} \Pi_{y:X} \text{isContr}(\text{Id}_X(x, y))$
2	hSet	$\Pi_{x:X} \Pi_{y:X} \text{isProp}(\text{Id}_X(x, y))$
3	hGroupoid	\vdots
\vdots	\vdots	\vdots
n	h- $(n - 2)$ -groupoid	\vdots
\vdots	\vdots	\vdots

By default, reasoning in homotopy type theory is fully “proof-relevant” via Propositions-as-Types.

Thus to “prove as proposition” means to give an element of the type, and to say two propositions “are equivalent” means to construct a homotopy equivalence between them.

This fits mathematical practice because, after all, most propositions are h-propositions. And for those that aren’t, we need the extra information.

That said, there are many cases where we want to reason differently, and many of these are captured by *modalities*. In analogy with the well-known modalities, *necessarily* and *possibly* (which are not formalized in HoTT), we might have

Adverb	Modality
<i>classically</i> (or <i>surely</i> ?)	double-negation
<i>anonymously</i> (or <i>merely</i> ?)	(-1) -truncation (giving hProps)
⋮	⋮
<i>locally</i>	sheafification for a subtopos
<i>codiscretely</i>	$\#$ (sharp; in a cohesive topos)

Homotopy Equivalences

There are several ways to define that $f: A \rightarrow B$ is an equivalence internally in homotopy type theory:

- There is a function $g: B \rightarrow A$, homotopies $\rho: f \circ g \rightarrow 1_B$ and $\sigma: g \circ f \rightarrow 1_A$ and *one* higher homotopy $f(\sigma) \rightarrow \rho f$.
- Every homotopy fiber of f is contractible.
- f is both surjective and injective (read correctly).

Whichever of these we choose, the rest are homotopy equivalent to it!

To set up the univalence axiom we introduce *type universes*, U , reflecting the judgement $A \text{ Type}$.

The Univalence Axiom (Voevodsky)

For types A and B , the identity type $\text{Id}_U(A, B)$ is equivalent to the type $\text{Eq}(A, B)$ of (homotopy) equivalences between A and B . More precisely, the canonical map:

$$A: \text{Type}, B: \text{Type} \vdash \text{Id}_U(A, B) \rightarrow \text{Eq}(A, B)$$

is itself an equivalence.

- The real numbers,

- The real numbers,
- Ordinals,

- The real numbers,
- Ordinals,
- Higher inductive types (combinatorial geometry, localization),

- The real numbers,
- Ordinals,
- Higher inductive types (combinatorial geometry, localization),
- Ontology: do all types come with a cell decomposition?

- The real numbers,
- Ordinals,
- Higher inductive types (combinatorial geometry, localization),
- Ontology: do all types come with a cell decomposition?
- Higher coinductive types (Postnikov systems?),

- The real numbers,
- Ordinals,
- Higher inductive types (combinatorial geometry, localization),
- Ontology: do all types come with a cell decomposition?
- Higher coinductive types (Postnikov systems?),
- Computational interpretation coexisting with the homotopy-theoretical.

- The real numbers,
- Ordinals,
- Higher inductive types (combinatorial geometry, localization),
- Ontology: do all types come with a cell decomposition?
- Higher coinductive types (Postnikov systems?),
- Computational interpretation coexisting with the homotopy-theoretical.
- Generalized predicativity forced because of higher topos theory.

Conclusion from a working mathematician

Urs Schreiber, *What is Homotopy Type Theory Good For?*

That says: not only is there *some* formal language to capture homotopy theory. No, moreover: it's a *natural* language, potentially more natural than the language you have been using so far, and speaking this language may help to make more transparent phenomena in homotopy theory that are less transparent otherwise.

- 1 Outline
- 2 Motivation for Univalent Foundations
- 3 Review of Constructive Type Theory
- 4 The Homotopy-theoretical Interpretation
- 5 Mathematics in Homotopy Type Theory
- 6 Open Problems**

- Proof-theoretic strength of the univalence axiom? (Conjecture: conservative over MLTT, in the sense that an internal model construction is possible).
- Computational interpretation of univalence (seems easy enough but details are tricky): Harper-Licata's two-dimensional type theory is a beginning.
- Formalization of higher-dimensional inductive and coinductive types.
- Analog of cumulative hierarchy to facilitate set-theoretical research. (E.g.: *reflection principles* or *induction principles for universes* – problem: seems to clash with univalence!)