Sets in Homotopy type theory

Egbert Rijke, Bas Spitters
jww Univalent Foundations Program

Sept 26th, 2013

Supported by EU FP7 STREP FET-open ForMATH

Homotopy type theory

Towards a new practical foundation for mathematics.
Closer to mathematical practice, inherent treatment of equivalences.

Towards a new design of proof assistants:
Proof assistant with a clear (denotational) semantics, guiding the addition of new features.

Concise computer proofs.

Challenges

Sets in Coq setoids (no quotients), no unique choice (quasi-topos), ...

Coq in Sets somewhat tricky, not fully abstract (UIP,...)

Towards a more symmetric treatment.

Two generalizations of Sets

To keep track of isomorphisms we want to generalize sets to
groupoids (proof relevant equivalence relations)
2-groupoids (add coherence conditions for associativity),
$\ldots$, ∞-groupoids

Topos theory

Wikimedia Commons

Topos theory

A topos is like:

- a semantics for intuitionistic formal systems
  model of intuitionistic higher order logic.
- a category of sheaves on a site
- a category with finite limits and power-objects
- a generalized space

Higher topos theory

Combine these two generalizations.

A higher topos is like:

- a model category which is Quillen equivalent to simplicial $Sh(C)_S$ for some model ∞-site (C, S).
- a generalized space (presented by homotopy types)
- a place for abstract homotopy theory
- a place for abstract algebraic topology
- a semantics for Martin-Löf type theory with univalence and higher inductive types.

Higher topos theory

Shulman/Cisinski: HoTT+univalence for h-Tarski universes can be interpreted in any Grothendieck ∞-topos.

h=Hofmann, homotopy

Type U of codes. Coercion $\mathrm{El} : U \to$ Type, plus operations like

$$\mathrm{Pi} : \Pi a : U, (\mathrm{El} a \to U) \to U$$

El only respects these operations up to propositional equality:

$$\mathrm{El}(\mathrm{Pi} ab) = \Pi x : \mathrm{El} a, \mathrm{El}(bx)$$

Q: higher topos from the syntax of type theory? (Kapulkin).

Envisioned applications

Type theory with univalence and higher inductive types as the internal language for higher topos theory??

- higher categorical foundation of mathematics
- framework for large scale formalization of mathematics
- expressive programming language
- synthetic pre-quantum physics
  (Schreiber/Shulman, cf. Bohr toposes)

Towards elementary ∞-topos theory.
Effective ∞-topos?, glueing (Shulman),...
Here: Develop mathematics in this framework
Coq formalization[2]

[2] https://github.com/HoTT/HoTT/

The hierarchy of complexity

Definition

We say that a type A is contractible if there is an element of type

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

Contractible types are said to be of level -2.

Definition

We say that a type A is a mere proposition if there is an element of type

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

Mere propositions are said to be of level -1.

The hierarchy of complexity

Definition

We say that a type A is a set if there is an element of type

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

Sets are said to be of level 0.

Definition

Let A be a type. We define

$$\begin{aligned} \text{is-}(-2)\text{-type}(A) &:\equiv \mathsf{isContr}(A) \\ \text{is-}(n+1)\text{-type}(A) &:\equiv \prod_{x,y:A} \text{is-}n\text{-type}(x =_A y) \end{aligned}$$

Equivalence

A good (homotopical) definition of equivalence is:

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

This is a mere proposition.

Direct consequences of Univalence

Univalence implies:

- functional extensionality
  Lemma ap10 {A B} (f g : A → B): (f=g → f == g).
  Lemma FunExt {A B}: forall f g, IsEquiv (ap10 f g).
- logically equivalent propositions are equal:
  Lemma uahp '{ua:Univalence}: forall P P': hProp, (P ↔ P')→ P = P'.
- isomorphic Sets are equal
  all definable type theoretical constructions respect isomorphisms

Theorem (Structure invariance principle)

Isomorphic structures (monoids, groups,...) may be identified.

Informal in Bourbaki. Formalized in agda (Coquand, Danielsson).

The classes of n-types are closed under

- dependent products
- dependent sums
- idenity types
- W-types, when $n \geq -1$
- equivalences

Thus, besides 'propositions as types' we also get propositions as n-types for every $n \geq -2$. Often, we will stick to 'propositions as types', but some mathematical concepts (e.g. the axiom of choice) are better interpreted using 'propositions as (-1)-types'.
Concise formal proofs

Higher inductive types

Higher inductive types internalize colimits.

Higher inductive types generalize inductive types by freely adding higher structure (equalities).

Bertot's Coq implementation of Licata's agda trick.

The implementation by Barras should suffice for the present work.

Squash

NuPrl's squash equates all terms in a type

Higher inductive definition:

```
Inductive minus1Trunc (A : Type) : Type :=
  | min1 : A → minus1Trunc A
  | min1_path : forall (x y: minus1Trunc A), x = y
```

Reflection into the mere propositions

Awodey, Bauer [ ]-types.

Theorem

epi-mono factorization. Set is a regular category.

Logic

Set theoretic foundation is formulated in first order logic.
In type theory logic can be defined, propositions as (-1)-types:

$$
\begin{aligned}
\top &:\equiv \mathbf{1} \\
\bot &:\equiv \mathbf{0} \\
P \wedge Q &:\equiv P \times Q \\
P \Rightarrow Q &:\equiv P \to Q \\
P \Leftrightarrow Q &:\equiv P = Q \\
\neg P &:\equiv P \to \mathbf{0} \\
P \vee Q &:\equiv \| P + Q \| \\
\forall (x : A).\, P(x) &:\equiv \prod_{x:A} P(x) \\
\exists (x : A).\, P(x) &:\equiv \left\| \sum_{x:A} P(x) \right\|
\end{aligned}
$$

models constructive logic, not axiom of choice.

Unique choice

```
Definition hexists {X} (P:X→ Type):=(minus1Trunc (sigT P) ).

Definition atmost1P {X} (P:X→ Type):=
    (forall x₁ x₂ :X, P x₁ → P x₂ → (x₁  = x₂ )).

Definition hunique {X} (P:X→ Type):=(hexists P) * (atmost1P P).

Lemma iota {X} (P:X→  Type):
  (forall x, IsHProp (P x)) → (hunique P) → sigT P.
```

In Coq we cannot escape Prop.
Exact completion: add quotients to a category.
Similarly: Consider setoids $(T, \equiv)$.
Spiwack: Setoids in Coq give a quasi-topos.

Quotients

Towards sets in homotopy type theory.

Voevodsky: univalence provides (impredicative) quotients.

Quotients can also be defined as a higher inductive type

```
Inductive Quot (A : Type) (R:rel A) : hSet :=
  | quot : A → Quot A
  | quot_path : forall x y, (R x y), quot x = quot y
(* | _ :isset (Quot A).*)
```

Truncated colimit.

We verified the universal properties of quotients.

Modelling set theory

pretopos: extensive exact category
ΠW-pretopos: pretopos with Π and W-types.

Theorem

0-Type is a ΠW-pretopos (constructive set theory).

Assuming AC, we have a well-pointed boolean elementary topos with choice (Lawvere set theory).
Define the cumulative hierarchy $\emptyset, P(\emptyset), \ldots, P(V_\omega), \ldots,$
by higher induction. Then V is a model of constructive set theory.

Theorem (Awodey)

Assuming AC, V models ZFC.

We have retrieved the old foundation.

Predicativity

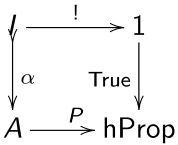
In predicative topos theory: no subobject classifier/power set.
AST provides a framework for defining various predicative toposes.
Joyal/Moerdijk/Awodey/...: Algebraic Set Theory (AST).
Categorical treatment of set and class theories.
Two challenges:

- From pure HoTT we do not (seem to) obtain the collection axiom from AST.
  Idea: add such an axiom based on a sheaf-stable version of the presentation axiom:
  Every type is covered by a projective type.
  Should hold in the cubical sets model.
- The universe is not a set, but a groupoid!

Higher categorical version of AST?
Perhaps HoTT already provides this...

Large subobject classifier

The subobject classifier lives in a higher universe.
Use universe polymorphism.



With propositional univalence, hProp classifies monos into A.
Equivalence between predicates and subsets.
This correspondence is the crucial property of a topos.
Sanity check: epis are surjective (by universe polymorphism).

Object classifier

$Fam(A) := \{(I, \alpha) \mid I : Type, \alpha : I \to A\}$ (slice cat)
$Fam(A) \cong A \to Type$
(Grothendieck construction, using univalence)

$$\begin{array}{ccc} I & \xrightarrow{i} & \text{Type}_\bullet \\ \Big\downarrow{\scriptstyle \alpha} & & \Big\downarrow{\scriptstyle \pi_1} \\ A & \xrightarrow{P} & \text{Type} \end{array}$$

$\text{Type}_\bullet = \{(B, x) \mid B : \text{Type}, x : B\}$
Classifies *all* maps into A + group action of isomorphisms.
Crucial construction in ∞-toposes.
Proper treatment of Grothendieck universes from set theory.
Formalized in Coq.
Improved treatment of universe polymorphism (h/t Sozeau).
Object classifier equivalent to univalence, assuming funext.

Towards elementary higher topos theory

We are developing internal higher topos theory.

- Factorization systems for *n*-levels, generalizing epi-mono factorization.
- Modal type theory for reflective subtoposes, sheafification.
- Descent theorem: Homotopy colimits defined by higher inductive types behave well.
  We use an internal model construction:
  graph presheaf model of type theory.

Conclusion

- Practical foundation for mathematics
- HoTT generalizes the old foundation
- Towards a proof assistant with a clear denotational semantics
- Towards elementary higher topos theory