From setoid hell to homotopy heaven?

The role of extensionality in Type Theory

Thorsten Altenkirch

Functional Programming Laboratory
School of Computer Science
University of Nottingham

June 1, 2017

intensional Mathematical objects are equal iff they have the same definition.

extensional Mathematical objects are equal if they have the same behaviour.

The case for extensionality

- Mathematics at large: building towers of abstraction.
- To do this we need to hide implementation details.
- Hence we need **extensionality**.
- In an extensional system we can always decide to model intensional aspects.
- But if we don't have extensionality from the beginning we can't make it up.

The paradox of intensional type theory (ITT)

- We can only observe extensional aspects of our objects.
- In this sense ITT is more extensional than set theory.
- On the other hand ITT only identifies objects that have the same definitions (intensional equality type)
- In particular it lacks:

  functional extensionality Two functions that are pointwise equal are equal.

  propositional extensionality Two propositions that are logically equivalent are equal.

  Quotients We can quotient a type by an equivalence relation

Setoids

- To overcome this weakness we used setoids.
- A setoid is a type with an equivalence relation.
- We make extensional equality explicit.

Setoid Hell

- Where exactly do we use setoids and where types?
- We have to introduce a lot of boilerplate, e.g. we define List as an operation on types but now we have to lift this also to setoids.
- This gets even worse when we consider families of setoids, as for example in categories where the objects have a non-trivial equality.
- We never actually hide the implementation, any user of a setoid may still depend on the implementation details.

Observational Type Theory

- Make explicit the type theory of setoids.
- Types are given by:
  - Elements
  - A propositional equality type
- A proposition is a type with no information.
- We obtain:
  - functional extensionality
  - propositional extensionality
  - quotients

More extensionality

- A set is a type with a propositional equality
- New principle:

  Set extensionality Two sets are equal if they are in a one-to-one correspondence.
- Equality can no longer be propositional because there is more than one way sets can be in a one-to-one correspondence.
- We can model types as groupoids (= glorified setoids).

The paradox of extensional type theory

- Extensional type theory features the equality reflection rule, identifying judgemental and propositional equality.
- We obtain:

  functional extensionality
  propositional extensionality
  quotients
- We cannot have set extensionality because equality reflection means that we cannot have non-propositional equalities.

Going on until the cows come home

- Why stop at groupoids?
- We can define a general notion of equivalence taking proof-relevant equality into account (cf. Ian Orton's talk).
- More extensionality:

  Univalence Two types that are equivalent are equal.
- We model types as weak ω-groupoids (infinitely glorified setoids)

State of the art

- Weak ω-groupoids are difficult!
- Recent progress: cubical set model and `cubical`.
- Models univalence.
- Still problems with modelling HITs.
- Most constructions don't need higher dimensions.
- We can work with setoids or groupoids.