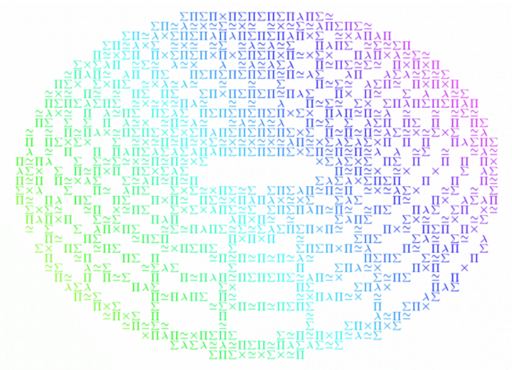
Homotopy type theory



Ingo Blechschmidt

November 25th, 2014

Outline

1. Foundations
   - What are foundations?
   - What's problematic with set-based foundations?

Homotopy type theory is a new branch of mathematics that combines aspects of several different fields in a surprising way. It is part of Voevodsky's *univalent foundations* program and based on a recently discovered connection between homotopy theory and type theory, a branch of mathematical logic and theoretical computer science.

In homotopy type theory, any set (really: *type*) behaves like a topological space, or more precisely, a homotopy type. The basic notion of equality is reimagined in an interesting way: Analogous to how two given points in a space may be joined by more than one path, two elements of a set can be equal in many ways. A new axiom, the *univalence axiom*, posits that equivalent structures really are the same, thus formalizing a widespread notational practice.

Besides explaining how working in homotopy type theory feels like, the talk will give answers to the listed questions. The talk does not assume any background in formal logic or type theory.

- What are logical foundations for mathematics and why should we care?
- What are the disadvantages of traditional set-based approaches to foundations?
- Why is the development of homotopy theory radically simplified in homotopy type theory?
- How are the seemingly diverse activities of *proving propositions* and *exhibiting constructions* identified?
- How do inductive definitions of important spaces concisely capture their homotopy-theoretic content?
- Why is homotopy type theory a major step towards practically useful and easily applicable proof assistants?

What are foundations?

- Foundations set the logical context for doing maths.
- Their details don't matter in everyday work (mostly).
- But their main concepts do.

What are foundations?

- Foundations set the logical context for doing maths.
- Their details don't matter in everyday work (mostly).
- But their main concepts do.

- Classical foundations are *set-based* (ZF, ZFC, ...): **Everything is a set.**
- $0 := \emptyset, \quad 1 := \{0\}, \quad 2 := \{0,1\}, \quad \ldots$
- $(x, y) := \{\{x\}, \{x, y\}\}$ (Kuratowski pairing)
- $(x, y, z) := (x, (y, z))$
- maps: (X, Y, R) with $R \subseteq X \times Y$ such that ...

- Foundations allow us to be maximally precise.
- A *proof* as commonly understood is really a shorthand for a (never spelled out) fully formal proof.
- Unlike informal proofs, the correctness of a formal proof can be checked mechanically.

Logicomix: An Epic Search for Truth

What's wrong with set-based foundations?

Set-based foundations ...

- allow to formulate nonsensical questions,
- do not reflect typed mathematical practice,
- require complex encoding of "higher-level" subjects, complicating interactive proof environments.

- Examples for questions which can be formulated:
  - Is $2 = (0, 0)$? (No, when using my definitions.)
  - Is $\sin \in \pi$? (Depends on your definitions.)
- In ordinary practice, these questions would be deemed as nonsensical, since they disrespect the *types* of mathematical objects and are not invariant under isomorphisms of the involved structures.

- Fully unravel the definition of "manifold" in set-theoretical language to get a grasp of the complex encodings needed.
- This is no problem for humans, but it is for machines.

- Note: Set theory is perfectly fine for studying *sets*.