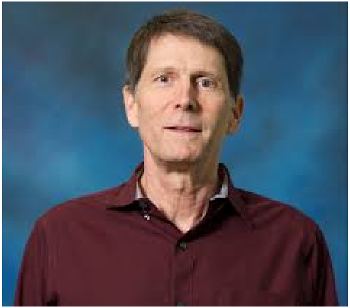
Formal Abstracts

Presenter: Floris van Doorn

August 24, 2018

Disclaimers



- The main force behind Formal Abstracts is Thomas Hales.
  - Other people involved: Jeremy Avigad, Rob Lewis, Mario Carneiro, Johannes Hölzl, ...
- I have not worked within the project yet, and do not know all plans in detail.
- Formal Abstracts is as of now vaporware.

Automation in Mathematics

- Proving
- Proof checking
- Conjecture generation
- Example generation
- Finding counterexamples
- Error detection
- Computing
- Generalizing
- Searching
- Transforming the literature
- Machine translation
- Teaching
- Collaborating
- Exploration

Motivation

For many of these activities, it is important or even essential to have a machine-readable semantic representation of mathematical objects.

The goal of Formal Abstracts is to link human readable mathematical statements with machine-readable statements.

This does not include formal proofs.

Applications

- Searching: We can make theorems more easily searchable by human and machine.
- Translation: It can be used to translate mathematics between natural languages.
- Analysis: We will build a big data set of formal and informal definitions and theorems for machine-learning projects that analyze the entire mathematical corpus.
- Exploration: We could make a tool such as a Google Earth for mathematics, providing an intuitive visual map of the entire world of mathematics.
- We make formal methods more relevant for mathematicians.

$\vdots$

The Dream

We want Formal Abstracts to

- give a statement of the main theorem(s) of each published mathematical paper in a language that is both human and machine readable;
- link each term in theorem statements to a precise definition of that term (again in human/machine readable form);
- ground every statement and definition in the system of some foundational system for doing mathematics.
- connect this database to other tools and programs.

Problems

If we omit proofs, there are various issues that come up.

- It is well-known in the formalization community that it is extremely difficult to get definitions correct until theorems are proved about the definitions.
- Mathematicians are notorious bad at giving the complete context needed for definitions.
  - Example: one definition in the formal proof of the Kepler conjecture took nearly 40 revisions to get right.
- Various definitions are specifications that take a theorem asserting the existence of something, and that thing that exists is given a name. Definition building cannot take place without theorem proving.
- Building definitions requires non-obvious identifications.
  - Example: "Define X to be any of the following equivalent concepts."

Solution: Do Formal Abstracts anyway

Concrete First Steps

Tom Hales was awarded a grant from the Sloane Foundation earlier this year. Next steps:

- Devise a formal language for Formal Abstracts which can be parsed into various proof assistants. Initially we will use Lean.
- In a small team, build a library of many definitions used in mathematics.
  - This list will go beyond the traditional topics covered by formalization projects.
  - The initial list of theorems will be chosen based on popularity and critical acclaim.
  - We carefully curate this library which will set the standard for outside contributions, when we open the project to outsiders.
- Create a functional service for mathematicians to search and contribute formal abstracts.

Organisation

- There will be a web submissions page for new formal abstracts that can be navigated without specialized training in dependent type theory.
- These submission will be transformed to polished formal scripts by a technical support team in Hanoi, Vietnam.
- Contributions will be managed by editors and reviewed by referees. The board includes experts in mathematics, formalization, and proof automation.
- The research group in Pittsburgh will be responsible for the mathematical end of the project.

Connection to Other Tools

- Theorem provers.
- Algebra software.
  - Example: Rob Lewis has constructed a two-way bridge between Lean and Mathematica.
- MathSciNet, zbMath, Wikipedia, ...
- LaTeX

  Let M be a \formal{banach manifold}[fabstracts.org/10.1109/5.771073].

Pre-empt some Questions

- Will you use classical logic? **Yes.**
- Will the formalization happen in homotopy type theory? **No.**
- Can I contribute my own theorems? **Not yet.**

- How will you set up algebraic structures?
- How will you deal with slightly different definitions of concepts in the literature?
- How will you define X?

We don't know yet!