Formalisation

or: How I Learned to Stop Worrying and Love the Computer

Clive Newstead

Graduate Student Seminar

Tuesday 30th January 2018

What is a proof?

A proof of a proposition P is...

- an argument that convinces yourself that P is correct;
- an argument that convinces others that P is correct;
- a logically coherent sequence of statements, beginning with axioms or known results, and ending with P;
- all of the above?
- something else entirely?

Observation #1: the literature

Mathematicians are only human. (Most of us, anyway. . .)

- We make lots of errors.
- We're lazy.
- We rely heavily on our intuition.
- We have no idea what we're doing.

$\rightsquigarrow$ We cannot be fully confident in the mathematical literature.

Observation #2: the difficulty

Mathematics is really really really hard!

- It takes a long time to get to grips with the theory.
- Proof techniques are never guaranteed to work.
- Lots of what we do is very tedious.

$\rightsquigarrow$ We waste a lot of time and effort.

Observation #3: the isolation

We're not very good at speaking to each other.

- There is little communication between fields.
- It's often hard to tell if a result has already been proved.
- It is difficult to read papers in other areas.
- We alienate non-mathematicians.

$\rightsquigarrow$ The mathematical community is isolated and disjointed.

Computers to the rescue!

Using computers might help with some of these issues.

- They can verify the correctness of proofs.
- They can assist with the process of proving a result.
- They can provide extensive databases of mechanised results.

. . . in theory. . .

What is an interactive theorem prover?

An *interactive theorem prover* (ITP) typically consists of:

- An underlying **logical system**.
- A trusted **kernel**.
- An **elaborator**.
- One or more **libraries**.

Examples of ITPs

Proof assistants have been around for a while.

Examples: Coq, Agda, HOL (& variants), Isabelle, NuPRL, **Lean**

Proof that $\sqrt{2}$ is irrational in Isabelle

```
theorem sqrt2_not_rational:
  "sqrt (real 2) ∉ ℚ"
proof
  let ?x = "sqrt (real 2)"
  assume "?x ∈ ℚ"
  then obtain m n :: nat where
    sqrt_rat: "¦?x¦ = real m / real n" and lowest_terms: "coprime m n"
    by (rule Rats_abs_nat_div_natE)
  hence "real (m^2) = ?x^2 * real (n^2)" by (auto simp add: power2_eq_square)
  hence eq: "m^2 = 2 * n^2" using of_nat_eq_iff power2_eq_square by fastforce
  hence "2 dvd m^2" by simp
  hence "2 dvd m" by simp
  have "2 dvd n" proof-
    from ‹2 dvd m› obtain k where "m = 2 * k" ..
    with eq have "2 * n^2 = 2^2 * k^2" by simp
    hence "2 dvd n^2" by simp
    thus "2 dvd n" by simp
  qed
  with ‹2 dvd m› have "2 dvd gcd m n" by (rule gcd_greatest)
  with lowest_terms have "2 dvd 1" by simp
  thus False using odd_one by blast
qed
```

Source: Wikipedia

Some verified results

- **Four colour theorem** (B. Werner & G. Gonthier, 2005, Coq)
- **Dirichlet's theorem** (J. Harrison, 2010, HOL Light)
- **Feit–Thompson theorem** (G. Gonthier, 2012, Coq)
- **Kepler conjecture** (T. Hales *et al.*, 2014, HOL Light & Isabelle)
- **Green's theorem** (M. Abdulaziz & L. Paulson, 2016, Isabelle)

Many results are still up for grabs!

Formalisation projects in Pittsburgh

Lots of formalisation is being done right under our noses!

At CMU:

- **Lean standard library** (J. Avigad, R. Lewis, . . .)
- **Homotopy theory** (S. Awodey, F. van Doorn, E. Rijke, J. Frey, F. Wellen, . . .)
- **RedPRL** (R. Harper, J. Sterling, C. Angiuli, E. Cavallo, Favonia, D. Gratzer, . . .)

At Pitt:

- **Formal Abstracts in Mathematics** (T. Hales, . . .)

Set theory?

Most 'formal' mathematics is done in ZFC set theory.

This is not ideal for formalisation.

- $\forall x, y, z\ (x \cdot y) \cdot z = x \cdot (y \cdot z)$
- $(x+y)^n = \sum_{k=0}^{n} \binom{n}{k} x^k y^{n-k}$ is true in any commutative ring.
- What does '2' refer to?

Problem: The **role** of an object is not inherent to the object.

Type theory!

In type theory, every object (or **term**) has a unique **type**.

$$2_{\mathbb{N}} : \mathbb{N} \qquad 2_{\mathbb{Z}} : \mathbb{Z} \qquad (x \mapsto x^2) : \mathbb{R} \to \mathbb{R} \qquad \mathbb{N} : \text{Type}$$

We can interpret types as **propositions**, whose terms are **proofs**.

Example. If a is a proof of A and b is a proof of B, how do we get a proof of 'A and B'?

Solution. Paste together a and b!

In type theory:

$$a : A, b : B \Rightarrow \langle a, b \rangle : A \times B$$

Some more analogies

type	**as a set**	**as a proposition**
$A \times B$	cartesian product	conjunction
$A + B$	disjoint union	disjunction
$A \to B$	function set	implication
$\sum_{x:A} B(x)$	indexed disjoint union	existential quantification
$\prod_{x:A} B(x)$	indexed product	universal quantification

term	**as an element**	**as a proof**
$\langle a, b \rangle : A \times B$	ordered pair	proof of A and proof of B
$c : A + B$	element of A or of B	proof of A or of B
$f : A \to B$	function from A to B	proof of B from proof of A
$\langle a, b \rangle : \sum_{x:A} B(x)$	ordered pair	witness & proof
$f : \prod_{x:A} B(x)$	dependent function	proof for arbitrary $x : A$

Suitability of type theory

Type theory is also useful because. . .

- It mirrors programming.
- It is constructive. (Non-constructive mathematics is still possible!)
- It is proof-relevant.
- It makes heavy use of induction and recursion.

Most mathematicians wouldn't detect a foundational shift.

What's good

Some things are going well in the world of formalisation.

- It's fun.
- There's lots of interest right now.
- The technology is getting better by the day.
- Non-mathematicians are becoming interested in mathematics.
- Libraries are getting bigger.

What's bad

Some things are going less well in the world of formalisation.

- It's a steep learning curve.
- It feels like programming.
- There is little consensus.
- There's not much money in it.
- Mathematicians are skeptics.

Considerations for the future

There are lots of issues affecting the future of ITPs.

- One foundation to rule them all?
- User-friendliness.
- The right level of interactivity.
- Incorporation in mainstream mathematical education.
- Automatic generation of human-readable proofs.

Thanks for listening!

These slides are available at
`http://math.cmu.edu/~cnewstea/talks/20180130.pdf`