Is HoTT the way to do mathematics?

K. Buzzard

1st April 2020, OWLS

Thank you to the organisers for the invitation!

Definition (David M Roberts, `hott.zulipchat.com`)

A *generic mathematician*, or more precisely a *generic pure mathematician*, is a mathematician working in the areas of algebra, analysis, geometry, topology or number theory, using classical logic and the axiom of choice.

In most mathematics departments, most of the pure mathematicians who work there are generic mathematicians.

- Me: a generic mathematician for 25 years.
- Now interested in formalising mathematics on a computer.
- Subject of the talk: **Which system to use?**

Summary of talk:

1) Formalising mathematics: why, and how?

2) Univalence: pros and cons.

3) Formalising algebraic geometry.

Why formalise mathematics on a computer?

- Computer scientists say: because then it will definitely be right!
- Generic mathematicians respond: Don't be paranoid – it's already definitely right.
- Computer scientists say: because what happens when your experts die out?
- Generic mathematicians respond: our understanding gets better quicker than the expert death rate.
- Computer scientists say: would it not be intrinsically fascinating to have a fully formalised proof of Fermat's Last Theorem?
- Generic mathematicians respond: No.

Generic mathematicians are *unconvinced* by the arguments above.
Empirical observation: generic mathematicians have a firm grip on decisions such as what is fashionable, who gets funding, and who gets major prizes in pure mathematics.

Goals which are *feasible* and which might start to change the opinions of generic mathematicians:

- Create a reliable digital graduate student who will grind out tedious calculations.
- Create a searchable database of known mathematics.
- Create training data for an AI.

Digitising things is a good idea.

We do not even know how much mathematics it is feasible to formalise.

We also *cannot predict* what will happen if we try it.

Which system?

Propaganda now over – let's talk details.

Claim: to get generic mathematicians interested in formalising *modern generic mathematics*, our system *must* allow

- Classical logic;
- The axiom of choice;
- Dependent types;
- Serious automation.

This is the world in which they already operate, and they have no desire to do anything different.
This has consequences.

- Serious automation: seems to *currently* rule out the popular set theory systems (Mizar, MetaMath).
- Dependent types: seems to rule out the Higher Order Logic systems (Isabelle/HOL, HOL4, HOL Light).

Examples of systems which fit the bill:

- Lean, and "vanilla" Coq;
- UniMath (a Coq library), and the Coq HoTT library (another one).

UniMath and HoTT/Coq have access to the *univalence axiom*. Lean and vanilla Coq do not. Univalence is an axiom proposed by Voevodsky, following ideas of Awodey and Warren and others. Definition to follow on next slide!

Open problem: do generic mathematicians want univalence? Or can they do without it?

The univalence axiom

- Lean does *not* have the univalence axiom.
- In Lean's type theory, propositions are "proof-irrelevant". At most one proof of $A = B$.
- Lean: any two proofs of $A = B$ are *equal by definition*.
- Equivalence $A \simeq B$: an apparently weaker notion. Data!
- $A \simeq B$ means $f : A \to B$ and $g : B \to A$ with fg and gf the identity function. "A bijection".
- Classically, if A is a type/set with n terms/elements, then $A \simeq A$ has $n!$ terms/elements.
- In Lean, $A = A$ is a type with only one term (a theorem with only one proof).
- Univalence: $(A = B) \simeq (A \simeq B)$. "Equivalence is the same as equality".
- In Lean's type theory this immediately leads to a contradiction (no bijection because $1 \neq n!$ in general).

Interesting (to me) empirical observations about univalence and its consequences ("if two objects are equivalent, they are equal").

- Voevodsky (key proponent) was a generic mathematician.
- Consequences of the axiom are very natural in structural mathematics (a big part of generic mathematics).
- Example. Say A and B are isomorphic Huber rings, and A is strongly Noetherian. Is B strongly Noetherian?
- In Lean we will prove this with the `equiv_rw` tactic. Basic tactic is in Lean (as of yesterday) but still much work needed.
- In a univalent system we get the proof for free.
- $A \simeq B$ so $A = B$ so $P(A) \implies P(B)$.

Localising rings

Example of where this mattered to me.

Reminder: a *commutative ring* is a set or type R equipped with addition, subtraction and multiplication, satisfying the usual axioms.

Examples: $\mathbb{Z}$, $\mathbb{Q}$, $\mathbb{R}$, $\mathbb{C}$.

- Schemes (Grothendieck, 1960).
- Basic fact: every commutative ring R gives rise to a scheme $\mathrm{Spec}(R)$.

Some mathematics students and I formalised this construction in Lean.

Localisation

- A ring R has $+$ and $-$ and $\times$ but what about division?
- I cannot find "the true $2/3$" in the ring $\mathbb{Z}$.
- Basic idea: enlarge $\mathbb{Z}$ to get $\mathbb{Q}$.
- More refined idea:

$$\mathbb{Z}[1/3] := \left\{ \frac{a}{b} \mid a \in \mathbb{Z} \text{ and } b = 3^n \right\}.$$

One can think of $\mathbb{Q} = \mathbb{Z}[1/2, 1/3, 1/4, 1/5, \ldots]$.
Or $\mathbb{Q} = \mathbb{Z}[1/S]$ with $S = \{1, 2, 3, 4, 5, \ldots\}$.
What is the formal story?

- R: our original commutative ring.
- S: the elements of R we want to divide by.
- Goal: new ring $R[1/S]$ where we can.
- First step: modify S to ensure that $1 \in S$ and if $a, b \in S$ then $ab \in S$.
- Naive guess for $R[1/S]$: the set of pairs $R \times S$, with $(r, s) \in R \times S$ corresponding to $r/s \in R[1/S]$.
- No good: $1/2 = 2/4$ in the rationals.
- Fix: $R[1/S]$ is a quotient of $R \times S$ by a certain equivalence relation.

Note standard abuse of notation:
$R[1/3] = R[1/\{3\}] = R[1/\{1, 3, 3^2, 3^3, 3^4, \ldots\}]$.

Say A is a ring, and $1/2 \in A$ and $1/3 \in A$.

Then their product $1/6 \in A$.

Conversely, if $1/6 \in A$, then $2 \times 1/6 = 1/3$ and $3 \times 1/6 = 1/2 \in A$.

So as any generic mathematician will tell you, this means that for any ring R, we have $R[1/2][1/3] = R[1/6]$.

This kind of equality is *explicitly written* in Grothendieck's work.

However, this equality is *not, strictly speaking, true*. The sets and equivalence relations used to form these things are *not literally equal*.

However, $R[1/2][1/3]$ and $R[1/6]$ are equivalent. Indeed, they satisfy the same universal property, so they are *canonically isomorphic*, an informal but stronger notion of equivalence.

When formalising Grothendieck's construction in Lean (no univalence), we ran into arguments where our sources replaced $R[1/f][1/g]$ by $R[1/fg]$ without comment.

Without univalence, we had to pay. We had to rewrite some proofs so that they applied not just to $R[1/f]$ but to any ring satisfying the same universal property as $R[1/f]$.

This turned out to be a delicate argument in API extraction, which was ultimately solved in this case by Neil Strickland.

Theorem (Strickland)

Let R be a commutative ring, let $S \subseteq R$ be a multiplicative subset (i.e., a submonoid), and let $f : R \to A$ be a morphism of commutative rings. Then $R \to A$ is isomorphic to $R \to R[1/S]$ in the category of R-algebras if and only if the following three things hold:

1. *For all $s \in S$, $f(s)$ is invertible in A;*
2. *For all $a \in A$ there exists $r \in R$ and $s \in S$ such that $f(r)/f(s) = a$;*
3. *The kernel of f is precisely the elements of R annihilated by an element of S.*

The proof is trivial. The clever part is spotting the statement. In the relevant proof we were formalising (in the Stacks Project), these were the only facts about $R[1/S]$ used in the proof.

With univalence, would this all have been much easier? Or would there have been other problems instead? **Nobody knows because nobody tried.**

Univalence might be the secret sauce which makes formalisation easier for mathematicians.

However, although there has been *lots of mathematics* done in these univalent/HoTT systems, there has been very little *generic mathematics*.

This has to change. We need to know which system we should be using.

The new chatroom `hott.zulipchat.com` is a place where formalising generic mathematics gets discussed. There is also plenty of talk about type theory and constructivism.

Lean and Coq both have a huge amount of basic undergraduate mathematics. We need to get such mathematics into one or more of the HoTT systems, to *see if the HoTT systems are suitable for generic mathematics*.

F. Wiedijk, in "The QED manifesto revisited" (2007), points out that any system aiming to formalise a substantial corpus of mathematics needs to be able to "integrate work by multiple people into a nice coherent whole".

Lean's maths library is doing precisely this, because of an internet chatroom and GitHub.

It is time that the HoTT theories caught up, and started formalising generic mathematics. Will generic maths be better with univalence? *Nobody knows because we didn't try yet.*

As far as I know, *no HoTT system even has localisation of rings*, and certainly none of them have schemes.

`hott.zulipchat.com`

Thank you for your attention.

Notes (written after talk).
People asked about Agda and Arend. I have no problems with these systems! Let's see some generic maths done in Agda and Arend! Valery Isaev told me that in fact he *had* done localisation of rings in Arend, but there are no topological spaces yet. Someone should make topological spaces in Arend, following the exposition in Lean.
Thorsten Altenkirch pointed out to me that my definition of equivalence was not the standard one, but was merely equivalent to the standard one. Sorry. My understanding of these nuances is poor.

Finally, a quote:
"A canonical isomorphism is [denoted by] $=$". J. S. Milne, "Etale cohomology", terminology and conventions.

Prove a theorem. Write a function. @XenaProject