On the ingredients for Fermat

Kevin Buzzard, Imperial College London

VaNTAGe seminar, 30th April 2024

Before we start

Thank you very much to the organisers for giving me this opportunity to speak.

Overview of the talk

I'm going to talk about a project whose ultimate goal is a Lean formalisation of a proof of Fermat's Last Theorem.

The talk is in four parts:

- Introduction to/overview of the project.
- Grand opening!
- Mathematical details of the proposed proof.
- How to get involved.

Overview of the key players

Lean is a free and open source programming language expressive enough to understand mathematical theorems and proofs.

But the core language and its standard library don't contain much mathematics itself.

`mathlib` is a free, open source, and very fast-moving, formally-verified mathematics library written in Lean.

Right now it contains most of a typical undergraduate mathematics degree, and some Masters/early PhD level stuff.

Formalisation of mathematics

One possible application of this library is a realisation of Tom Hales' "Formal Abstracts" idea.

The idea: a database (perhaps integrated into Math Reviews/ZBMath) of publishing *formal statements* of the main result(s) of new mathematics papers.

Hales proposed this in 2017 at the "Big Proof" conference at the Newton Institute in Cambridge UK.

One possible use: AI training.

One issue: bridging the gap between Masters level mathematics and modern reserach.

Difficulties with this idea

Unfortunately, "go and make definitions in Lean of the key concepts in your subject area" is hard to sell.

Example: we've had schemes in `mathlib` for years now but as far as I know we still don't have the definition of a proper morphism of schemes, or of a curve over a field (see forthcoming AIM workshop though...).

Contributing a definition also means contributing some "API" for that definition (basic theorems about it) to prove that the definition is usable, so the work is not as easy as it sounds.

So how and why do definitions end up in `mathlib`?

It's because people need them for projects.

The Langlands Program

I attend the London Number Theory Seminar, and week in week out I see people proving theorems that it is not even possible to *state* in Lean.

Example: many of the key objects used in the Langlands Program are not formalised.

Note: this is in stark contrast to some other areas, e.g. combinatorics, where some important modern results can be formalised in real time (Mehta, Dillies, Tao...).

Formalising Fermat

For this, and several other reasons, I proposed a project to begin a Lean formalisation of a modern proof of Fermat's Last Theorem.

I got funding from the EPSRC, and I am extremely grateful to them for funding such a non-standard project.

I will be working on this project full time for five years starting in October.

It's not "my project" – it's a free and open source collaborative project, and I just happen to be "leading" it right now.

Later on I will explain how you can join in and help with the project.

Which proof?

I have given some general audience talks about the project already.

But this VaNTAGe seminar gives me an opportunity to give a very different talk, where I explain some of the details of the route we're taking.

tl;dr: we're not formalising the original Wiles/Taylor–Wiles proof, we're taking a more modern approach, following a route basically designed by Taylor.

There is no written reference for this approach right now, and I apologise in advance that part of this talk will be technical.

I just need to say one more thing before we start on this.

What I am not claiming

I am not claiming that we will have finished the formalisation within 5 years.

What I claimed in the proposal would be that, at the very least, I would *reduce the problem* of formalising FLT to theorems which were known in the 1980s.

Pithy summary: "I'll start by formalising Wiles' paper but not the references" (although we are not formalising Wiles' route.)

As you'll see in this talk, a full formalisation of FLT will be a *tremendous* amount of work.

However, benefits such as the Formal Abstracts applications will come much sooner (definition of an automorphic representation, for example).

The act of formalisation

Preparing for the task of formalising FLT using `mathlib` is similar to preparing a graduate course.

You have a coherent story which you want to tell, and you have to try and explain it to a system which has a solid background in undergraduate mathematics and knows some graduate level material.

I've written the first lecture or two of this course; so let's launch the project!

[Kevin now spends several minutes clicking various buttons]

The proof we'll formalise

For reasons I do not fully understand, `mathlib` has had more success in the algebraic side of things than the analytic side.

This might be sociological. Or it might not be.

One thing I'm sure about is that I *personally* would rather be doing algebra than analysis.

And part of the reason I'm embarking upon this project is that I *personally* am going to have a large amount of *fun* running it.

The question I asked Richard Taylor: What is the cleanest route to a proof of FLT in the 2020s, ideally with as little analysis as possible?

This section is a summary of his detailed response.

A proof of FLT

I'll start slow :-)

The claim is that there are no solutions to $a^n + b^n = c^n$ in positive integers, for $n \geq 3$.

Any $n \geq 3$ is either divisible by an odd prime, or is a power of 2 and hence a multiple of 4.

So it suffices to prove the claim for $n = 4$ and n an odd prime.

Cases $n = 4$ and $n = 3$ can be resolved via elementary arguments (Fermat, Euler). The observation is that $x^4 + 1 = z^2$ and $x^3 + y^3 = 1$ are rank 0 elliptic curves.

So WLOG $n = \ell \geq 5$ is prime.

The Frey Curve

We will actually prove the more general claim that if a, b, c are nonzero (but possibly negative) integers and $\ell \geq 5$ is prime then $a^\ell + b^\ell \neq c^\ell$.

So assume we have a counterexample and let's seek a contradiction.

We can assume that a, b, c are pairwise coprime (divide out common factors).

One of them is thus even, and we can assume it's b (apply $(a, b, c) \mapsto (b, a, c)$ if it's a, and $(a, b, c) \mapsto (a, -c, -b)$ if it's c).

Now a is odd and, applying $(a, b, c) \mapsto (-a, -b, -c)$ if necessary, we can assume that $a = 3 \bmod 4$.

Under these hypotheses, the *Frey–Hellegouarch curve* $E : Y^2 = X(X - a^\ell)(X + b^\ell)$ can be checked to be semistable (even at 2).

The Frey curve

Now $E : Y^2 = X(X - a^\ell)(X + b^\ell)$ has j-invariant $2^8(c^{2\ell} - a^\ell b^\ell)^2/(abc)^{2\ell}$.

The ℓ-torsion $E(\overline{\mathbb{Q}})[\ell]$ is 2-dimensional and admits an action $\overline{\rho}$ of $\mathrm{Gal}(\overline{\mathbb{Q}}/\mathbb{Q})$.

If $p \nmid \ell abc$ then the curve, and hence this Galois representation, is unramified at p.

If $p \mid abc$ then the curve is semistable at p so one can analyse the ℓ-torsion as a Galois module by the theory of the Tate curve.

More precisely, we have $E(\overline{\mathbb{Q}}_p) = \overline{\mathbb{Q}}_p^\times/\langle q \rangle$ possibly up to a quadratic twist, and $v_p(q) = -v_p(j)$, which, if $p \neq 2$, is a multiple of ℓ.

Hence if $p \mid abc$ with $p \neq 2$ and $p \neq \ell$ then $\overline{\rho}$ is unramified at p, and if $p = \ell$ then $\overline{\rho}$ is flat at p.

The Frey curve

Upshot: the ℓ-torsion in the Frey curve E associated to a counterexample to FLT with $\ell \geq 5$ is unramified outside 2ℓ, flat at ℓ, and tamely ramified at 2.

Call such a $\overline{\rho} : \mathrm{Gal}(\overline{\mathbb{Q}}/\mathbb{Q}) \to \mathrm{GL}_2(\mathbb{Z}/\ell\mathbb{Z})$ *hardly ramified*.

Idea: this plus semistability of E is too good to be true.

Wiles made this rigorous: Mazur's torsion theorem implies (after a little work) that ρ can't be reducible, and a theorem of Wiles and Ribet shows it can't be irreducible either, so contradiction.

This is a point in the proof which I'm referring to as a *bifurcation*.

Mazur's theorem is 100+ pages of hard arithmetic geometry, but was known in the 1980s, so right now I'm skipping it.

Skipping Mazur's theorem

I'm skipping Mazur for now because I want to work on the modularity lifting theorem.

If anyone wants to start another project (or get another grant) formalising a proof of Mazur's theorem in Lean, that would be great.

I've already formalised the statement :-)

```
theorem Mazur (E : EllipticCurve ℚ) :
    {P : E.toWeierstrassCurve (ℚ) | ∃ n ≥ 1, n • P = 0}.ncard ≤ 16 := sorry
```

If nobody starts on Mazur then probably I'll start on it once I've proved the modularity lifting theorem we need.

But I fully expect the modularity lifting theorem to take several years.

Wiles/Ribet

Assuming Mazur, our task now is to show that if $\ell \geq 5$ and $\overline{\rho} : \mathrm{Gal}(\overline{\mathbb{Q}}/\mathbb{Q}) \to \mathrm{GL}_2(\mathbb{Z}/\ell\mathbb{Z})$ is hardly ramified and irreducible, then we have a contradiction.

At this point we diverge from the original Wiles/Taylor–Wiles proof.

W/TW used that $\overline{\rho} = E[\ell]$, proved E was modular by arguing at 3 and 5, and then used Ribet to deduce $\overline{\rho}$ was modular of weight 2 and level 2 and thus couldn't be irreducible.,

The argument at 3 used cyclic base change and Langlands–Tunnell (which uses non-Galois cubic base change).

Avoiding some analysis

We're going to argue at ℓ directly, meaning we don't need Langlands–Tunnell or Ribet.

We do however still need cyclic base change (hard analysis).

I know of no proof of FLT which avoids Mazur, and I know of no proof which avoids cyclic base change.

We also need to characterise the image of cyclic base change, so we need multiplicity 1 for GL_2 over totally real fields, and also Jacquet-Langlands (more hard analysis).

Again I will *state* the results and then just park them and use them, hoping others will take them on.

Lean lets you modularise the proof.

A modularity lifting theorem

Let me now work on stating the modularity lifting theorem that we will use (twice) in order to prove that there are no irreducible hardly ramified representations $\overline{\rho} : \mathrm{Gal}(\overline{\mathbb{Q}}/\mathbb{Q}) \to \mathrm{GL}_2(\mathbb{Z}/\ell\mathbb{Z})$.

In 1993 this machinery only worked for classical modular forms.

But now it works for automorphic forms on quaternion algebras over totally real fields.

So let me say something about these.

Automorphic forms

A modularity theorem attaches an automorphic form to a Galois representation.

But let's start by going the "easy way" – attaching Galois representations to automorphic forms.

(There is an argument saying that we've got so good at modularity lifting theorems that actually this is now the hard way.)

Galois reps attached to auto reps.

Say F is a totally real number field and D/F is a totally definite quaternion algebra.

"Totally definite" means $D \otimes_F \mathbb{R}$ is the quaternions for all $F \to \mathbb{R}$.

So the Shimura variety is 0-dimensional and there is no analysis in the definition of an automorphic representation for $D^\times$.

If $U \subseteq (D \otimes_F \mathbb{A}_F^f)^\times$ is a compact open subgroup, then we define the weight 2 automorphic forms of level U for D to be

$$\{\phi : D^\times \backslash (D \otimes_F \mathbb{A}_F^f)^\times / (\mathbb{A}_F^f)^\times U \to \mathbb{C}\}.$$

Space of automorphic forms

$$S_D(U) := \{\phi : D^\times \backslash (D \otimes_F \mathbb{A}_F^f)^\times / (\mathbb{A}_F^f)^\times U \to \mathbb{C}\}.$$

This is a finite-dimensional complex vector space (the double coset space is finite).

It comes with natural Hecke operators $T_\mathfrak{p}$ for all but finitely many primes $\mathfrak{p}$ of $\mathcal{O}_F$.

If ϕ is a (nonzero) eigenform for these Hecke operators then the subfield E of $\mathbb{C}$ generated by the eigenvalues is finite over $\mathbb{Q}$.

I claim that if we choose $E \to \overline{\mathbb{Q}}_p$ then there's a Galois representation $\rho_\phi : \mathrm{Gal}(\overline{F}/F) \to \mathrm{GL}_2(\overline{\mathbb{Q}}_\ell)$ associated to ϕ, such that $\rho_\phi(\mathrm{Frob}_\mathfrak{p})$ has trace equal to the eigenvalue of $T_\mathfrak{p}$ for all good primes $\mathfrak{p} \nmid \ell$.

Galois representation attached to an eigenform

Just to restate the miracle: if we have an eigenvector for a competely combinatorial set-up coming from a totally definite quaternion algebra then there's some kind of Galois representation associated to it.

The proof: use Jacquet-Langlands to find a corresponding automorphic representation attached to a quaternion algebra which is split at one or two infinite places, and then "find the representation in the cohomology of the corresponding Shimura variety".

So here we need all of class field theory, moduli spaces of abelian varieties, canonical models of Shimura varieties, étale cohomology, and the computation of étale cohomology of a Shimura variety.

This was all known in the 1980s so I'm skipping it for now.

Alternative approach

AI is rubbish at research maths right now.

But AI is going to get better at maths.

Some very intelligent people think that within 5 years we might be able to put Deligne/Carayol/Taylor's work on Shimura varieties and Galois representations into a machine and get Lean code out which pretty much works.

As far as I am concerned this is a complete unknown variable.

I am skeptical.

But I'm not an expert in AI.

And AI is *definitely* going to get better at maths.

The modularity lifting theorem

For the modularity lifting theorem I only care about levels of the form $\Gamma_1(\mathfrak{n})$ at for $\mathfrak{n}$ a squarefree product of primes.

We need to know that the Galois representation obeys local-global at $\mathfrak{p}$ if $\mathfrak{p}$ doesn't divide ℓ.

And we also need to know it's flat at ℓ if we have full level at ℓ.

Again, all known in the 1980s.

And a local analysis of the moduli problem at $\Gamma_1(\mathfrak{p})$ level is much easier than the general case.

The modularity lifting theorem

Suppose F is totally real of even degree, $\ell \geq 5$ is unramified in F, S is a finite set of finite places of F not containing any primes above ℓ, and say $\rho : \mathrm{Gal}(\overline{F}/F) \to \mathrm{GL}_2(\overline{\mathbb{Q}}_\ell)$ is unramified outside $S \cup \{\ell\}$, is flat at ℓ, and has cyclotomic determinant.

Suppose also that for all $v \in S$ and all $g \in \ker(I_v \to k(v)^\times)$, we have $tr(\rho(g)) = 2$.

Theorem: If $\overline{\rho}$ is modular of level $U_1(S)$ and absolutely irreducible even when restricted to $F(\zeta_\ell)$, then ρ is also modular of level $U_1(S)$.

Proof: see Taylor 2018 Stanford graduate course (although the proof was known before this).

Overview of the rest of the argument

Say $\overline{\rho} : \mathrm{Gal}(\overline{\mathbb{Q}}/\mathbb{Q}) \to \mathrm{GL}_2(\mathbb{Z}/\ell\mathbb{Z})$ is the ℓ-torsion in the Frey curve associated to a counterexample.

One now writes down a certain modular curve parametrising elliptic curves whose ℓ-torsion looks like $\overline{\rho}$ and whose p-torsion is induced from a character ψ, for some auxiliary prime p.

A theorem of Moret-Bailly guarantees that this curve has a global point over some totally real F disjoint from ψ and the number field cut out by $\overline{\rho}$, satisfying some local conditions.

This point corresponds to an elliptic curve A/F.

The modularity lifting theorem applied at p, after possibly making F larger, implies that A is modular.

Hence $\overline{\rho}$ is potentially modular.

Overview of the rest of the argument

One can lift $\overline{\rho}$ to a characteristic zero representation ρ unramified outside 2ℓ and flat at ℓ.

The modularity lifting theorem applied again, gives that ρ is potentially modular.

A trick due to Taylor using Brauer induction shows that ρ is part of a compatible family of level 2.

One now specialises this family at 3.

Overview of the rest of the argument

So far: given our "unlikely" $\overline{\rho}$ we have lifted it, put it in a compatible family, and now specialising at 3 gives us a continuous irreducible 3-adic representation ρ_3 of $\mathrm{Gal}(\overline{\mathbb{Q}}/\mathbb{Q})$ with cyclotomic determinant, unramified outside 2 and 3, flat at 3, and whose semisimplification at 2 is trivial plus cyclotomic on inertia.

Choose a Galois-stable lattice and reduce mod 3, and let K be the number field cut out by this extension.

A careful argument using Fontaine's local analysis in his paper about nonexistence of abelian varieties over $\mathbb{Z}$ (also known in the 1980s) bounds the root discriminant of K.

Using the Odlyzko bounds gives us bounds for the degree of K over $\mathbb{Q}$, which contradict irreducibility of ρ_3. Done!

Stating the modularity lifting theorem

`mathlib` has adeles and quaternion algebras and number fields (and totally real fields will be easy to define).

Undergraduate and masters students at Imperial have defined Galois representations and Frobenius elements, so they are on the way.

So formalising the *statement* that an automorphic representation for totally definite D/F gives a Galois representation is viable.

But the proof will be a huge amount of work.

A warm-up

However, the "easy" reductions at the beginning of the maths part of this talk are very much ready to be formalised.

Example: if E has good reduction at p then $E[\ell]$ is unramified at p.

Example: the theory of the Tate curve.

Example: the determinant of $E[\ell]$ is the mod ℓ cyclotomic character.

Example: the j-invariant of the Frey curve.

Example: the Frey curve is semistable.

All standard stuff from Silverman I / Silverman II, and none of it is in mathlib yet.

Managing the formalisation

The formalisation, like all other nontrivial Lean mathematics formalisation projects, will use Patrick Massot's `leanblueprint` software.

The most famous component of this software is the *blueprint graph*.

But the component I want to mention here is the detailed LaTeX exposition of the argument.

Managing the formalisation

The LaTeX exposition of the argument is what powers the project.

The workflow might look like the following:

I spend a lot of time writing detailed LaTeX proofs of intermediate results.

An arbitrary person from around the world who knows something about Lean can choose a node which is "ready" (i.e., blue), read the LaTeX, translate it into Lean, and make a pull request to the repository.

If the Lean proof compiles, I can merge the PR without worrying that the person is a crank or has a misunderstanding of the mathematics.

`leanblueprint` enables collaboration at scale in mathematics.

Managing the formalisation

I'll be managing the formalisation on the `#FLT` stream on the Lean Zulip.

I'll post fortnightly updates explaining what needs doing.

If you need to learn Lean, then I would recommend searching "Mathematics in Lean" and working through this.

As is probably clear from this talk, I am going to need a *lot* of help if we're going to make this happen.

But I fully also intend to have a lot of fun doing it.

Thanks a lot for your time!