An invitation to formalising mathematics

Michael B. Rothgang (he/him)

Symplectic geometry group
Humboldt-Universität zu Berlin

Berlin Mathematical School BMS
Berlin Mathematics Research Center MATH+
Humboldt-Universität zu Berlin
erc European Research Council
Established by the European Commission

BMS Student Conference 2024
February 21, 2024

Slides at `www2.mathematik.hu-berlin.de/~rothganm/`

Outline of today's talk

What is a proof?

Proof: formal definition

A mathematical proof is a sequence of *formal* logical deductions, starting from a set of axioms.

Proof: practical definition

A mathematical proof is a sequence of arguments convincing an educated reader. *In principle*, all details can be filled in.

Proof correctness is a social convention!

What is a proof: practical issues

- proof correctness is a social convention
- folklore results: believed true but no written proof
- most papers have errors: most are minor and fixable, some errors are grave

Example (Poincaré's, stability of the solar system)

Every single issue of Acta Mathematica retracted and reprinted.

Example (Four-colour theorem)

Proofs by Kempe and Tait (around 1880) each believed correct — for 11 years.

Example (Classification of finite simple groups)

Gap (quasi-thin case), only closed after 21 years

Some papers are wrong

Example (Baker's theorem, 1970)

- key lemma is false (Rempe-Sixsmith 2019)
- many papers using it can be fixed; another bunch is now open
- five much-cited papers "generalised" the argument

Example (Hilbert's 21st problem)

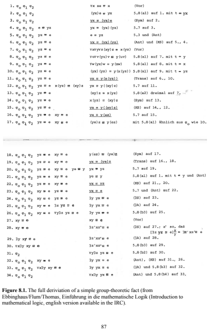
"Proof" by Plemelj (1908) found wrong in 1970s
solved in 1990 with different answer

Example (Hilbert's 16th problem, part 2)

Solution by Dulac (1923), found wrong in 1981

What does formalisation mean?

answer 1: humans write more detailed proofs



1.	$\varphi_0 \varphi_1 \varphi_2$	$\forall x\ xe \equiv x$	(Vor)
2.	$\varphi_0 \varphi_1 \varphi_2$	$(yx)e \equiv yx$	5.8(a1) auf 1. mit $t \equiv yx$
3.	$\varphi_0 \varphi_1 \varphi_2$	$\underline{yx \equiv (yx)e}$	(Sym) auf 2.
4.	$\varphi_0 \varphi_1 \varphi_2\ e \equiv yz$	$yx \equiv (yx)(yz)$	5.7 auf 3.
5.	$\varphi_0 \varphi_1 \varphi_2\ yz \equiv e$	$e \equiv yz$	5.3 und (Ant)
6.	$\varphi_0 \varphi_1 \varphi_2\ yz \equiv e$	$\underline{yx \equiv (yx)(yz)}$	(Ant) und (KS) auf 5., 4.
7.	$\varphi_0 \varphi_1 \varphi_2\ yz \equiv e$	$\forall x \forall y \forall z (xy)z \equiv x(yz)$	(Vor)
8.	$\varphi_0 \varphi_1 \varphi_2\ yz \equiv e$	$\forall u \forall v (yu)v \equiv y(uv)$	5.8(a1) auf 7. mit $t \equiv y$
9.	$\varphi_0 \varphi_1 \varphi_2\ yz \equiv e$	$\forall w (yx)w \equiv y(xw)$	5.8(a1) auf 8. mit $t \equiv x$
10.	$\varphi_0 \varphi_1 \varphi_2\ yz \equiv e$	$(yx)(yz) \equiv y(x(yz))$	5.8(a1) auf 9. mit $t \equiv yz$
11.	$\varphi_0 \varphi_1 \varphi_2\ yz \equiv e$	$\underline{yx \equiv y(x(yz))}$	(Trans) auf 6., 10.
12.	$\varphi_0 \varphi_1 \varphi_2\ yz \equiv e\ \ x(yz) \equiv (xy)z$	$yx \equiv y((xy)z)$	5.7 auf 11.
13.	$\varphi_0 \varphi_1 \varphi_2\ yz \equiv e$	$(xy)z \equiv x(yz)$	5.8(a2) dreimal auf 7.
14.	$\varphi_0 \varphi_1 \varphi_2\ yz \equiv e$	$x(yz) \equiv (xy)z$	(Sym) auf 13.
15.	$\varphi_0 \varphi_1 \varphi_2\ yz \equiv e$	$\underline{yx \equiv y((xy)z)}$	(KS) auf 14., 12.
16.	$\varphi_0 \varphi_1 \varphi_2\ yz \equiv e\ \ xy \equiv e$	$\underline{yx \equiv y(ez)}$	5.7 auf 15.
17.	$\varphi_0 \varphi_1 \varphi_2\ yz \equiv e\ \ xy \equiv e$	$(ye)z \equiv y(ez)$	mit 5.8(a1) ähnlich aus φ_0 wie 10.
18.	$\varphi_0 \varphi_1 \varphi_2\ yz \equiv e\ \ xy \equiv e$	$y(ez) \equiv (ye)z$	(Sym) auf 17.
19.	$\varphi_0 \varphi_1 \varphi_2\ yz \equiv e\ \ xy \equiv e$	$\underline{yx \equiv (ye)z}$	(Trans) auf 16., 18.
20.	$\varphi_0 \varphi_1 \varphi_2\ yz \equiv e\ \ xy \equiv e\ \ ye \equiv y$	$yx \equiv yz$	5.7 auf 19.
21.	$\varphi_0 \varphi_1 \varphi_2\ yz \equiv e\ \ xy \equiv e$	$ye \equiv y$	5.8(a1) auf 1. mit $t \equiv y$ und (Ant)
22.	$\varphi_0 \varphi_1 \varphi_2\ yz \equiv e\ \ xy \equiv e$	$\underline{yx \equiv yz}$	(KS) auf 21., 20.
23.	$\varphi_0 \varphi_1 \varphi_2\ xy \equiv e\ \ yz \equiv e$	$\underline{yx \equiv e}$	5.7 und (Ant) auf 22.
24.	$\varphi_0 \varphi_1 \varphi_2\ xy \equiv e\ \ yz \equiv e$	$\exists y\ yx \equiv e$	(∃S) auf 23.
25.	$\varphi_0 \varphi_1 \varphi_2\ xy \equiv e\ \ \exists z\ yz \equiv e$	$\exists y\ yx \equiv e$	(∃A) auf 24.
26.	$\varphi_0 \varphi_1 \varphi_2\ xy \equiv e\ \ \forall y \exists z\ yz \equiv e$	$\exists y\ yx \equiv e$	5.8(b3) auf 25.
27.	$xy \equiv e$	$xy \equiv e$	(Vor)
28.	$xy \equiv e$	$\exists z' xz' \equiv e$	(∃S) auf 27.; z' so, daß $[\exists z\ yz \equiv e]\frac{x}{y} = \exists z' xz' \equiv e$
29.	$\exists y\ xy \equiv e$	$\exists z' xz' \equiv e$	(∃A) auf 28.
30.	$\forall x \exists y\ xy \equiv e$	$\exists z' xz' \equiv e$	5.8(b3) auf 29.
31.	φ_2	$\forall y \exists z\ yz \equiv e$	5.8(b2) auf 30.
32.	$\varphi_0 \varphi_1 \varphi_2\ xy \equiv e$	$\exists y\ yx \equiv e$	(Ant), (KS) auf 31., 26.
33.	$\varphi_0 \varphi_1 \varphi_2\ \forall x \exists y\ xy \equiv e$	$\exists y\ yx \equiv e$	(∃A) und 5.8(b3) auf 32.
34.	$\varphi_0 \varphi_1 \varphi_2$	$\forall x \exists y\ yx \equiv e$	(Ant) und 5.8(b4) auf 33.

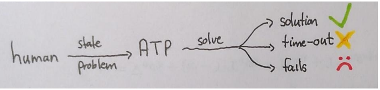
Figure 8.1. The full deriviation of a simple group-theoretic fact (from Ebbinghaus/Flum/Thomas, Einführung in die mathematische Logik (Introduction to mathematical logic, english version available in the IRC).

87

problem: impractical in the large
how to formalise "draw a picture"?

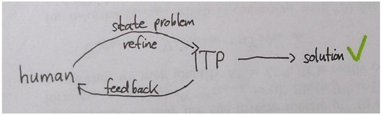
What does formalisation mean? (cont.)

answer 2: automated theorem proving



problems: hit or miss; opaque

answer 3: interactive theorem proving



Why formalise?

Verification

Understanding

Creation

Collaboration

Why formalise?

- verification: peer reviewer's dream
  only check definitions and theorems make sense
- understanding: reader chooses amount of detail
  Demo by Patrick Massot and Kyle Miller:
  `https://www.imo.universite-paris-saclay.fr/~patrick.massot/Examples/ContinuousFrom.html`
- database of theorems: searching known and related results
  only requires *statements* of main results
- creation: can this lemma be generalised? unused assumptions?
- collaboration: less trust required

What has been formalised already: let's guess

- Banach–Schauder open mapping theorem
- Birkhoff Ergodic Theorem
- Mandelbrot set is connected
- Cauchy-Kovalevskaya Theorem on existence of an analytical solution of an analytical PDE.
- Denjoy's theorem: a C^2 orientation-preserving diffeomorphism of the circle with an irrational rotation number is conjugate to a rotation.
- Sphere eversion
- Existence of Haar measure
- Existence of a smooth partition of unity
- Feit–Thompson theorem/odd order theorem
- Fermat's Last Theorem
- Four colour theorem
- Galois correspondence
- Herman-Yoccoz theorem on linearization of a circle diffeomorphism
- Jordan curve theorem
- Liouville theorem: an entire holomorphic function is a constant
- Hilbert's Nullstellensatz
- Picard-Lindelöf theorem (existence and uniqueness of solutions of ODEs)
- Poincaré-Bendixson Theorem
- Poincaré recurrence theorem
- Sard's Theorem
- The continuum hypothesis is independent of ZFC.

Let's guess: the answer

Only 5 are not formalised yet (AFAIK)

- Cauchy-Kovalevskaya Theorem on existence of an analytic solution of an analytic PDE
- Denjoy's theorem on rotation number
- Herman-Yoccoz theorem on linearization of a circle diffeomorphism
- Fermat's Last Theorem (in progress)
- Sard's Theorem (in progress)

Notable formalisation projects

- 2005 Four colour theorem
- 2012 Odd Order Theorem
- 2014 Kepler's conjecture (Hales et al)
- 2019 Ellenberg-Gijswijt's result on the cap set conjecture
- 2022 Liquid Tensor Experiment (Commelin et al): fundamental lemma about condensed mathematics
- 2022 unit fractions project before referee report
- 2023 upper bound on diagonal Ramsey numbers before referee report
- 2023 polynomial Freeman-Rusza conjecture (Tao et al) took 3 weeks; complete before paper submitted

Some ongoing projects

- Almost Periodicity in Arithmetic Progressions
- Existence of an aperiodic monotile
- Prime Number Theorem (Kontorovich-Tao et al)
- Fermat's Last Theorem (Buzzard)

A zoo of interactive theorem provers

- four widely used interactive theorem provers: Coq, Isabelle/HOL, Mizar and Lean
- large mathematics libraries: *mathcomp*, *Archive of formal proofs*, *Mizar Mathematical Library*, *mathlib*
- Coq: standard tool for software verification
- Isabelle: simple foundations, powerful automation
- Mizar: huge library
- Lean: newest (<10 years old), fast-growing

Formalising research mathematics

- need a large library of mathematics
- need an integrated library: connecting different fields, in a compatible way
- Why mathlib?
  - large integrated library
  - growing *fast*
  - system and tools are improving quickly
  - friendly and diverse community (github, zulip)

Short demo

What is formalisation like?

- fussy; has learning curve
- it's fun — like a video game or programming
- makes you understand mathematics better

Demo: backup in case of technical issues

Compact.lean ×

Mathlib > Topology > UniformSpace > Compact.lean

```
    intro y hxy
    simp [comap_const_of_not_mem (compl_singleton_mem_nhds hxy) (Classical.not_not.2 rfl)]
#align uniform_space_of_compact_t2 uniformSpaceOfCompactT2

/-!
### Heine-Cantor theorem
-/


/-- Heine-Cantor: a continuous function on a compact uniform space is uniformly
continuous. -/
theorem CompactSpace.uniformContinuous_of_continuous [CompactSpace α] {f : α → β}
    (h : Continuous f) : UniformContinuous f :=
calc map (Prod.map f f) (𝓤 α)
   = map (Prod.map f f) (𝓝ˢ (diagonal α)) := by rw [nhdsSet_diagonal_eq_uniformity]
  _ ≤ 𝓝ˢ (diagonal β)                       := (h.prod_map h).tendsto_nhdsSet mapsTo_prod_map_diagonal
  _ ≤ 𝓤 β                                  := nhdsSet_diagonal_le_uniformity
#align compact_space.uniform_continuous_of_continuous CompactSpace.uniformContinuous_of_continuous

/-- Heine-Cantor: a continuous function on a compact set of a uniform space is uniformly
continuous. -/
theorem IsCompact.uniformContinuousOn_of_continuous {s : Set α} {f : α → β} (hs : IsCompact s)
    (hf : ContinuousOn f s) : UniformContinuousOn f s := by
  rw [uniformContinuousOn_iff_restrict]
  rw [isCompact_iff_compactSpace] at hs
  rw [continuousOn_iff_continuous_restrict] at hf
  exact CompactSpace.uniformContinuous_of_continuous hf
#align is_compact.uniform_continuous_on_of_continuous IsCompact.uniformContinuousOn_of_continuous
```

Lean Infoview ×

▼ Compact.lean:168:0

▼ Expected type

```
α : Type u_1
β : Type u_2
γ : Type u_3
inst✝² : UniformSp
inst✝¹ : UniformSp
inst✝ : CompactSpa
f : α → β
h : Continuous f
⊢ UniformContinuou
```

▶ All Messages (0)

Learning Lean

- play the natural number game:
- textbook: mathematics in Lean
  `https://leanprover-community.github.io/mathematics_in_lean/index.html`
- further resources:
  `https://leanprover-community.github.io/learn.html`
- questions? ask on zulip

Lean tutorials

- upcoming: Edinburgh, May 27-31 (women and mathematicians of minority gender); registration by Feb 26
- past/registration closed:
  - Düsseldorf (September 2023)
  - Regensburg (September 2023)
  - Rome (Jan 2024)
  - Marseille (March 2024)
  - Bonn (May 2024)
- up-to-date list: `https://leanprover-community.github.io/events.html`

Lean in Berlin

- Lean study group, summer 2024 (email me if interested)
- Sebastian Pokutta, Tibor Szabó: Lean-related project
- Marc Kegel had a student using Lean
- ask your master's thesis advisor :-)

Thanks for listening! Any questions?

Comparing mathematical libraries: a closer look

- Archive of formal proofs: 4.1 million lines
  not integrated, articles are re-developing theory
  about half is "computer science" (e.g., properties of algorithms and programs)
- Coq's library: different focus from standard mathematics (e.g., care about constructivism)
- MML: large and integrated; no statistics on size
- mathlib: 1.5 million lines, integrated

My contributions to mathlib

Sard's theorem: prerequisites and reduction to normed spaces

- measure zero subsets of a manifold
- locally Lipschitz maps
- nowhere dense, meagre and sigma-compact sets
- local diffeomorphisms

Other mathematics

- interior and boundary of a manifold
- inverse function theorem for manifolds
- immersions, submersions and embeddings

Sphere eversion project: cleaning up, moving code into mathlib

Long-term vision: formalising the foundations of symplectic geometry