Understanding, formal verification, and the philosophy of mathematics

Jeremy Avigad

Department of Philosophy and Department of Mathematical Sciences
Carnegie Mellon University

February, 2013

The new epistemology of mathematics

Since Plato, the philosophy of mathematics has been concerned with:

- the nature of mathematical objects, and
- the appropriate justification for mathematical knowledge.

But we employ other normative judgments as well:

- some theorems are interesting
- some questions are natural
- some concepts are fruitful, or powerful
- some proofs provide better explanations than others
- some historical developments are important
- some observations are insightful

. . . and so on.

The problem of multiple proofs

On the standard account, the value of a mathematical proof is that it warrants the truth of the resulting theorem.

Why, then, do we often value a new proof of a previous established theorem?

For example, Gauss published six proofs of the law of quadratic reciprocity in his lifetime, and left us two unpublished versions as well.

Franz Lemmermeyer has documented 233 proofs (available online, with references).

The problem of multiple proofs

These observations are not new. For example:

> *It might be said: "—that every proof, even of a proposition which has already been proved, is a contribution to mathematics". But why is it a contribution if its only point was to prove the proposition? Well, one can say: "the new proof shews (or makes) a new connexion". — Wittgenstein, Remarks on the Foundations of Mathematics, III–60*

Indeed, it is *not* a great mystery. There is a lot we can say about what we learn from different proofs.

But the philosophy of mathematics has had relatively little to say about the matter.

The problem of conceptual possibility

It is often said that some mathematical advance was "made possible" by a prior conceptual development.

For example, Riemann's introduction of the complex zeta function and the use of complex analysis made it possible for Hadamard and de la Vallée Poussin to prove the prime number theorem in 1896.

What is the sense of "possibility" here?

Intuition: a certain "understanding" guides us. (But let's focus on the phenomena, not the word.)

Outline

- Understanding
  - The problem of multiple proofs
  - The problem of conceptual possibility
  - Some vague intuitions
- Formal verification
  - Understanding mathematical assertions
  - Understanding mathematical proofs
  - Understanding mathematical types
  - Understanding mathematical inference
  - Understanding mathematical diagrams
- The philosophy of mathematics
  - Mathematical methods and abilities
  - Mathematical concepts
  - Mathematical ease and difficulty

Vague intuitions

Mathematics is hard.

Mathematical solutions, proofs, and calculations involve long sequences of steps, that have to be chosen and composed in precise ways.

To compound matters, there are too many options; among the many steps we may plausibly take, most will get us absolutely nowhere.

And we have limited cognitive capacities — we can only keep track of so much data, anticipate the result of a few small steps, remember so many background facts.

We rely on our understanding to help us and to guide us.

Vague intuitions

Does understanding the demonstration of a theorem consist in examining each of the syllogisms of which it is composed in succession, and being convinced that it is correct and conforms to the rules of the game? In the same way, does understanding a definition consist simply in recognizing that the meaning of all the terms employed is already known, and being convinced that it involves no contradiction?

. . . Almost all are more exacting; they want to know not only whether all the syllogisms of a demonstration are correct, but why they are linked together in one order rather than in another. As long as they appear to them engendered by caprice, and not by an intelligence constantly conscious of the end to be attained, they do not think they have understood.

(Poincaré, *Science et méthod*)

Vague intuitions

Logic teaches us that on such and such a road we are sure of not meeting an obstacle; it does not tell us which is the road that leads to the desired end. (Ibid.)

Discovery consists precisely in not constructing useless combinations, but in constructing those that are useful, which are an infinitely small minority. Discovery is discernment, selection. (Ibid.)

Vague intuitions

But not yet have we solved the incantation of this whiteness, and learned why it appeals with such power to the soul; and more strange and far more portentous. . . and yet should be as it is, the intensifying agent in things the most appalling to mankind.

Is it that by its indefiniteness it shadows forth the heartless voids and immensities of the universe, and thus stabs us from behind with the thought of annihilation, when beholding the white depths of the milky way? Or is it, that as in essence whiteness is not so much a colour as the visible absence of colour; and at the same time the concrete of all colours; is it for these reasons that there is such a dumb blankness, full of meaning, in a wide landscape of snows–a colourless, all-colour of atheism from which we shrink?

(Meville, *Moby Dick*, Chapter 42).

Vague intuitions

The sea had jeeringly kept his finite body up, but drowned the infinite of his soul. Not drowned entirely, though. Rather carried down alive to wondrous depths, where strange shapes of the unwarped primal world glided to and fro before his passive eyes; and the miser-merman, Wisdom, revealed his hoarded heaps; and among the joyous, heartless, ever-juvenile eternities, Pip saw the multitudinous, God-omnipresent, coral insects, that out of the firmament of waters heaved the colossal orbs. He saw God's foot upon the treadle of the loom, and spoke it; and therefore his shipmates called him mad. So man's insanity is heaven's sense; and wandering from all mortal reason, man comes at last to that celestial thought, which, to reason, is absurd and frantic; and weal or woe, feels then uncompromised, indifferent as his God.

Outline

- Understanding
  - The problem of multiple proofs
  - The problem of conceptual possibility
  - Some vague intuitions
- Formal verification
  - Understanding mathematical assertions
  - Understanding mathematical proofs
  - Understanding mathematical types
  - Understanding mathematical inference
  - Understanding mathematical diagrams
- The philosophy of mathematics
  - Mathematical methods and abilities
  - Mathematical concepts
  - Mathematical ease and difficulty

Formal verification

Formal verification involves the use of formal methods to verify correctness, for example:

- verifying that a circuit description, an algorithm, or a network or security protocol meets its specification; or
- verifying that a proof of a mathematical theorem is correct.

"Interactive theorem proving" is one important approach.

Working with a proof assistant involves conveying enough information to the system to confirm that there is a formal axiomatic proof.

In fact, most proof systems actually construct a formal proof object, a complex piece of data that can be verified independently.

Interactive theorem proving

Some theorems formalized to date: the prime number theorem, the four-color theorem, the Jordan curve theorem, Gödel's first incompleteness theorem, Dirichlet's theorem, Cartan fixed-point theorems

There are good libraries for elementary number theory, real and complex analysis, measure-theoretic probability, linear algebra, Galois theory, ...

See the *Journal of Automated Reasoning*, *Journal of Formalised Reasoning*, *Journal of Formalized Mathematics*, the *Interactive Theorem Proving* conference, and Freek Wiedijk's list of 100 theorems.

Interactive theorem proving

Georges Gonthier headed a project to verify the Feit-Thompson theorem in Coq, with a group of researchers.

- The original 1963 journal publication ran 255 pages.
- The formalization is constructive.
- The development includes libraries for finite group theory, linear algebra, and representation theory.

The project was completed on September 20, with roughly

- 170,000 lines of code,
- 4,200 definitions, and
- 15,000 theorems.

Interactive theorem proving

Thomas Hales' *Flyspeck* project, to verify a proof of the Kepler conjecture, is nearing completion (HOL light, Isabelle).

- Three essential uses of computation: enumerating tame hypermaps, proving nonlinear inequalities, showing infeasibility of linear programs.
- The formalization led to even stronger results.

Vladimir Voevodsky has launched a project to develop "univalent foundations" for algebraic topology (Coq).

- Constructive dependent type theory has natural homotopy-theoretic interpretations.
- Rules for identity types characterize homotopy theories abstractly.
- One can consistently add an axiom to the effect that "isomorphic structures are identical."

Understanding mathematical language

Contemporary proof systems rely on a variety of frameworks:

- set theory (Mizar)
- higher-order logic (HOL, HOL light, Isabelle, . . .)
- (constructive) dependent type theory (Coq)

Assertions are made in the corresponding assertion language.

For example, here is Hales' statement of the Jordan curve theorem in HOL light:

```
!C. simple_closed_curve top2 C ==>
  (?A B. top2 A /\ top2 B /\
    connected top2 A /\ connected top2 B /\
  ~(A = EMPTY) /\ ~(B = EMPTY) /\
   (A INTER B = EMPTY) /\ (A INTER C = EMPTY) /\
       (B INTER C = EMPTY) /\
      (A UNION B UNION C = euclid 2)
```

Understanding mathematical language

Steve Kieffer implemented a parser for an extension of set theory designed by Harvey Friedman, and entered hundreds of definitions from Suppes' *Set theory* and Munkres' *Topology*.

DEFINITION MunkTop.13.2: 2-ary function `Basisgentop`. If $\texttt{TOPBASIS}[\mathscr{B}, X]$ then $\texttt{Basisgentop}(\mathscr{B}, X) \simeq$
$(!\mathscr{T} \subseteq \wp(X))((\forall U \subseteq X)(U \in \mathscr{T} \leftrightarrow (\forall x \in U)(\exists B \in \mathscr{B})$
$(x \in B \wedge B \subseteq U)))$.

DEFINITION MunkTop.13.3.c: 0-ary function `Krealtop`.
$\texttt{Krealtop} \simeq \texttt{Basisgentop}(\texttt{Stdrealtopbasis} \cup \{V \subseteq \mathbb{R} :$
$(\exists W \in \texttt{Stdrealtopbasis})$
$(V = W \backslash \{\texttt{Incl}_{\texttt{FrR}}(1_{\texttt{N}}/n) : n \in \mathbb{N}\})\}, \mathbb{R})$.

Understanding mathematical language

Natural language output:

Definition: If $\mathscr{B}$ is a basis for a topology on X then *the topology on X generated by $\mathscr{B}$* is the unique $\mathscr{T} \subseteq \wp(X)$ such that for every $U \subseteq X$, $U \in \mathscr{T}$ if and only if for every $x \in U$, there exists $B \in \mathscr{B}$ such that $x \in B$ and $B \subseteq U$.

Definition: *The K-topology on* $\mathbb{R}$ is the topology on $\mathbb{R}$ generated by the standard basis for a topology on $\mathbb{R}$ union the set of $V \subseteq \mathbb{R}$ such that there exists W in the standard basis for a topology on $\mathbb{R}$ such that $V = W \setminus \{1/n : n \in \mathbb{N}\}$.

Understanding mathematical language

Understanding mathematical language, involves, in part, being able to identify the fundamental logical and mathematical structure of an assertion: recognizing connectives and quantifiers, function application, predication, and so on.

Understanding mathematical proof

Think of an ordinary proof as a high-level description of, or recipe for constructing, a fully detailed axiomatic proof.

In formal verification, it is common to refer to proofs as "code."

```
lemma prime_factor_nat: "n ~= (1::nat) ==>
    EX p. prime p & p dvd n"
  apply (induct n rule: nat_less_induct)
  apply (case_tac "n = 0")
  using two_is_prime_nat apply blast
  apply (case_tac "prime n")
  apply blast
  apply (subgoal_tac "n > 1")
  apply (frule (1) not_prime_eq_prod_nat)
  apply (auto intro: dvd_mult dvd_mult2)
done
```

Understanding mathematical proof

```
proof (induct n rule: less_induct_nat)
  fix n :: nat
  assume "n ~= 1" and
    ih: "ALL m < n. m ~= 1 --> (EX p. prime p & p dvd m)"
  then show "EX p. prime p & p dvd n"
  proof -
  { assume "n = 0"
    moreover note two_is_prime_nat
    ultimately have ?thesis by auto }
  moreover
  { assume "prime n" then have ?thesis by auto }
  moreover
  { assume "n ~= 0" and "~prime n"
    with `n ~= 1` have "n > 1" by auto
    with `~prime n` and not_prime_eq_prod_nat obtain m k where
      "n = m * k" and "1 < m" and "m < n" by blast
    with ih obtain p where "prime p" and "p dvd m" by blast
    with `n = m * k` have ?thesis by auto }
  ultimately show ?thesis by blast
```

Understanding mathematical proof

```
Theorem Burnside_normal_complement :
  'N_G(S) \subset 'C(S) -> 'O_p^'(G) ><| S = G.
Proof.
move=> cSN; set K := 'O_p^'(G); have [sSG pS _] := and3P sylS.
have [p'K]:  p^'.-group K /\ K <| G by rewrite pcore_pgroup pcore_normal.
case/andP=> sKG nKG; have{nKG} nKS := subset_trans sSG nKG.
have{pS p'K} tiKS: K :&: S = 1 by rewrite setIC coprime_TIg ?(pnat_coprime pS).
suffices{tiKS nKS} hallK: p^'.-Hall(G) K.
  rewrite sdprodE //= -/K; apply/eqP; rewrite eqEcard ?mul_subG //=.
  by rewrite TI_cardMg //= (card_Hall sylS) (card_Hall hallK) mulnC partnC.
pose G' := G^`(1); have nsG'G : G' <| G by rewrite der_normal.
suffices{K sKG} p'G': p^'.-group G'.
  have nsG'K: G' <| K by rewrite (normalS _ sKG) ?pcore_max.
  rewrite -(pquotient_pHall p'G') -?pquotient_pcore //= -/G'.
  by rewrite nilpotent_pcore_Hall ?abelian_nil ?der_abelian.
suffices{nsG'G} tiSG': S :&: G' = 1.
  have sylG'S : p.-Sylow(G') (G' :&: S) by rewrite (pSylow_normalI _ sylS).
  rewrite /pgroup -[#|_|](partnC p) ?cardG_gt0 // -{sylG'S}(card_Hall sylG'S).
  by rewrite /= setIC tiSG' cards1 mul1n pnat_part.
apply/trivgP; rewrite /= focal_subgroup_gen ?(p_Sylow sylS) // gen_subG.
apply/subsetP=> z; case/imset2P=> x u Sx; case/setIdP=> Gu Sxu ->{z}.
have cSS: forall y, y \in S -> S \subset 'C_G[y].
  move=> y; rewrite subsetI sSG -cent_set1 centsC sub1set; apply: subsetP.
  by apply: subset_trans cSN; rewrite subsetI sSG normG.
have{cSS} [y]: exists2 y, y \in 'C_G[x ^ u | 'J] & S :=: S :^ u :^ y.
```

Understanding mathematical proofs

Understanding mathematical proof involves, in part, being able to recognize contextual cues: explicit or implicit reliances on local assumptions, background knowledge, recently established facts, and so on; and to determine whether inferences are a matter of calculation, unwrapping definitions, applying a lemma, etc.

Understanding mathematical domains and structures

Let z be a complex number. Then

$$|e^z| = \left|\sum_{i=0}^{\infty} \frac{z^i}{i!}\right| \leq 1 + |z| + \left|\sum_{i=2}^{\infty} \frac{z^i}{i!}\right| \leq \ldots$$

What types of objects are these?

- i
- z^i
- The division symbol
- The less-than relation
- The summation symbol

Theorem provers use various forms of *type inference*.

Understanding mathematical types

Glossary:

- *Type inference*: methods of determining, in a given context, the type of a given term.
- *Overloading*: using the same symbol for more than one purpose (e.g. $+$ for the natural numbers and the reals)
- *Polymorphism*, *type classes*: using general operations and facts (like $x + y = y + x$) that have multiple instantiations.
- *Coercions*: casting a value of one type to another.
- *Implicit arguments*: systematically leaving out information when it can be inferred from context.
- *Unification and matching*: instantiating variables to get two terms to agree.

Understanding mathematical domains and structures

The following make sense in any commutative monoid:

$$\sum_{i<n+1} a_i = \left(\sum_{i<n} a_i\right) + a_n$$

$$\sum_{i \in S \cup T} a_i = \sum_{i \in S} a_i + \sum_{i \in T} a_i \quad \text{if } S \cap T = \emptyset$$

$$\sum_{i \in S} (a_i + b_i) = \sum_{i \in S} a_i + \sum_{i \in S} b_i$$

Also,

$$c \cdot \sum_{i \in S} a_i = \sum_{i \in S} c \cdot a_i$$

makes sense if $\cdot$ distributes over $+$.

Understanding mathematical domains and structures

Instances include not only specific sums (natural numbers, reals, rings, . . .) but also

- $\prod_{i \in S} a_i$
- $\bigvee_{i \in S} a_i, \quad \bigwedge_{i \in S} a_i$
- $\min_{i \in S} a_i, \quad \max_{i \in S} a_i,$
- $\bigcup_{i \in S} a_i, \quad \bigcap_{i \in S} a_i$
- $\mathrm{lcm}_{i \in S} a_i, \quad \gcd_{i \in S} a_i$

and many others.

Another example: if H and K are subgroups of a group, G, then $H \cap K$ is both a set and a group. So, e.g., $1 \in H \cap K$, and $|H \cap K| \geq 1$.

Understanding mathematical domains and structures

Understanding mathematical conventions regarding domains and types involves being able to resolve ambiguities and infer type information from the context; being able to recognize concrete domains as implicitly embedded in other domains; being able to recognize concrete and abstract structures as instances of more general classes of structures; and so on.

Understanding mathematical inference

So far, we have just scratched the surface; this doesn't begin to get at nontrivial mathematical inferences.

Most systems employ automated techniques to fill in small gaps in reasoning.

One can distinguish between:

- Decision procedures and search procedures
- Domain-general methods and domain-specific methods
- "Principled" methods and heuristics

Understanding mathematical inference

Domain-general methods:

- Propositional theorem proving
- First-order theorem proving
- Higher-order theorem proving
- Equality reasoning
- Nelson-Oppen "combination" methods.

Domain-specific methods:

- Linear arithmetic (integer, real, or mixed)
- Nonlinear real arithmetic (real closed fields, transcendental functions)
- Algebraic methods (such as Gröbner bases)

Automated methods do especially well on large, homogeneous problems; but often fail to capture even the most straightforward mathematical inferences.

Understanding mathematical inference

Understanding mathematics involves being able to carry out straightforward mathematical inferences in specific mathematical domains, even when those inferences are difficult to spell out in formal axiomatic terms.

Understanding mathematical diagrams

Diagrammatic reasoning plays an important role in mathematics.

Since the end of the nineteenth century, they have been used only sparingly in "rigorous" proofs. Still:

- They are often used to accompany / illustrate a mathematical argument.
- Sometimes a diagram can be entirely convincing.
- Sometimes a diagrams can be viewed as shorthand for a longer text argument.
- In specific domains, diagram use is often governed by implicit conventions.

In fact, in some domains, diagrammatic arguments can be viewed as being as rigorous as text arguments.

Understanding mathematical diagrams

Ken Manders observed that, in a Euclidean proof, diagrams are used only in precise, restricted ways.

For example, the diagram can only be used to license certain types of topological (diagrammatic, co-exact) assertions. Other (metric, exact) assertions are licensed explicitly by the text.

Building on Mumma's dissertation, Ed Dean, John Mumma, and I:

- Gave a detailed analysis of Euclidean diagrammatic inference.
- Showed soundness and completeness with respect to the modern semantics.
- Used off-the-shelf automated reasoning tools to verify such inferences.

Understanding mathematical diagrams

Understanding mathematical diagram use involves being able to represent information in a diagram appropriately, and draw valid inferences from the information so represented.

Outline

- Understanding
  - The problem of multiple proofs
  - The problem of conceptual possibility
  - Some vague intuitions
- Formal verification
  - Understanding mathematical assertions
  - Understanding mathematical proofs
  - Understanding mathematical types
  - Understanding mathematical inference
  - Understanding mathematical diagrams
- The philosophy of mathematics
  - Mathematical methods and abilities
  - Mathematical concepts
  - Mathematical ease and difficulty

The philosophy of mathematics

What does this all have to do with the philosophy of mathematics?

Notes:

- This is not a turf war.
- Also not a matter of value judgement.

Rather, it is a question as to what role distinctly philosophical methods can play in relation to theorem proving, software engineering, and so on.

Talking about understanding

Understanding mathematical language, involves, in part, being able to identify the fundamental logical and mathematical structure of an assertion. . .

Understanding mathematical proof involves, in part, being able to recognize contextual cues. . .

Understanding mathematical conventions regarding domains and types involves being able to resolve ambiguities and infer type information from the context . . .

Understanding mathematics involves being able to carry out straightforward mathematical inferences in specific mathematical domains. . .

Understanding mathematical diagram use involves being able to represent information in a diagram appropriately, and draw valid inferences from the information so represented.

Talking about understanding

Understanding mathematical language, involves, in part, being able to identify the fundamental logical and mathematical structure of an assertion. . .

Understanding mathematical proof involves, in part, being able to recognize contextual cues. . .

Understanding mathematical conventions regarding domains and types involves being able to resolve ambiguities and infer type information from the context . . .

Understanding mathematics involves being able to carry out straightforward mathematical inferences in specific mathematical domains. . .

Understanding mathematical diagram use involves being able to represent information in a diagram appropriately, and draw valid inferences from the information so represented.

Mathematical methods and abilities

Informally, we often explain our ascriptions of understanding by characterizing the associated abilities.

This provides a helpful way of thinking about mathematical knowledge:

- not just a list of definitions and theorems, knowing *that* certain statements are true;
- but a protocol, a manner of thinking, a form of life, knowing *how* to proceed.

Mathematical method and abilities

Straightforward model:

- We face various tasks (solving a problem, proving a theorem, verifying an inference, developing a theory, forming a conjecture).
- "Reasoning" involves passage though various epistemic states.
- "Understanding" (methods, techniques, procedures, protocols, tactics, strategies, . . .) makes this passage possible.

Understanding involves:

- Being able to recognize the nature of the objects and questions before us.
- Being able to marshall the relevant background knowledge and information.
- Being able to traverse space the of possibilities before us in a fruitful way.
- Being able to identify features of the context that help us cut down complexity.

Mathematical methods and abilities

We lack a clear methodological framework:

- Algorithms are overly specific; different "methods" may account for the same ability.
- Yet there is a compositional aspect to methods and abilities.
- Methods are fallible.
- Identity criteria are murky.

Machine models, cognitive models, programming languages, psychological data, etc. seem to provide the wrong level of description.

We need a level of abstraction that is appropriate for talking about the interesting features of the *mathematics*.

Mathematical concepts

Conventional psychological approaches (involving categories and exemplars) don't do well with mathematical concepts.

- Membership can be sharply defined.
- Mathematical concepts can evolve over time.
- Understanding a concept admits degrees.
- Various things can "improve our understanding" of a concept.
- One can speak of implicit uses of a concept.

Mathematical concepts

One solution: think of a mathematical concept as a bundle of abilities.

For example, the group concept includes:

- Knowing the definition of a group.
- Knowing common examples of groups, and being able to recognize implicit group structures when it is fruitful to do so.
- Knowing how to construct groups from other groups or other structures, in fruitful ways.
- Recognizing that there are different kinds of groups (abelian, nilponent, solvable, finite vs. infinite, continuous vs. discrete) and being able/prone to make these distinctions.
- Knowing various theorems about groups, and when and how to apply them.

Mathematical ease and difficulty

Foundational reduction washes out many of the nuances.

- There is only one type of mathematical object (set).
- There is only one binary relation (element-of).
- One only needs one "method" of proof: unwrap definitions and search.

This makes it hard to recognize

- differences between algebraic and geometric methods;
- differences between elementary proofs, conceptual proofs, and so on;
- the value of a good definition.

Mathematical ease and difficulty

It comes down to *complexity*: the differences in organization and expression matter because we have limited time, energy, memory, processing capacity.

But how can we measure complexity?

- Computer science: algorithmic complexity
- Logic: descriptive complexity, length of proof
- Cognitive science and psychology: timing tasks and cognitive models.

We need measures that are better tailored to the *mathematics*.

Concluding remarks

How to get started:

- Find much more precise, focused questions.
- Look to domains of application, such as
  - formal verification and automated reasoning
  - mathematical pedagogy and cognitive science
  - history (and historiography) of mathematics
  - mathematics itself

Over time, small but concrete advances will hopefully come together to give us a coherent theory of mathematical understanding.

Concluding remarks

And what if they don't?

Then we will have merely contributed to the conceptual foundations of automated reasoning, cognitive science, pedagogy, history of science, and so on — and learned some interesting things about mathematics as well.