BULLETIN (New Series) OF THE
AMERICAN MATHEMATICAL SOCIETY
https://doi.org/10.1090/bull/1826
Article electronically published on February 15, 2024

PROOF IN THE TIME OF MACHINES

ANDREW GRANVILLE

Abstract. We are concerned here with the nature of proof and what proof will become in this age of machines. We do so by comparing the values associated with (traditional) community based proof verification to those associated with computer proof verification. We finish by proposing ways that computer proofs might incorporate successful strategies from human experiences.

1. Introduction

Professional pure mathematicians are complicit in the undergraduate fantasy that mathematics is solidly built up from a bedrock of axioms into an immutable structure of objectively proven true results. This conceit persuades generations of undergraduates that mathematics is beautiful and timeless. However, any experienced working researcher proceeds rather differently, building on a library of knowledge (as found in papers, preprints, and books, as well as tacit knowledge) which they assume and hope is correct (or at least correctable). In practice, a *formal proof*, chasing a proof back to the axioms, can seem both tiresome and more-or-less pointless since progress is mostly measured in what is new and added to our knowledge with reference to the distant horizon, not by a tedious and hard-to-follow level of rigour, safety checks on what is far behind us. On the other hand, *intuitive proofs* not only provide acceptable justification of claims to a community of researchers (at least for now), but also stimulate, motivate, and inspire the next generation of developments.

However, change is afoot. With the advent of usable proof verification systems that have a large library of already given definitions, together with a wide panorama of fundamental theorems inexorably deduced from axioms, a researcher might hope to have a formal proof verification available by inputting their standard *intuitive* proof (albeit with modifications to fit what the proof system knows and needs) and feel more confident that they have not missed some simple issue.

Moreover, once machines can standardly convert an intuitive proof into a formal proof and vice versa, surely machines can help in the process of proof, at first confirming proposed lemmas (by recognizing proofs that are already in their library or finding straightforward modifications of what is already there). At some point we might expect such machine proof-verifiers to be used as "black boxes", in which we won't need to understand what they're doing to believe they have found an objective proof. Much of pure mathematics research is focused on proofs of proposed statements, and it is hard to guess whether there will be any limits to

Received by the editors June 21, 2023.
2020 *Mathematics Subject Classification.* Primary 00A30, 68T05.

machines' participation in this process. This raises many questions about the future of research mathematics, and the role of humans in that future, as highlighted by Venkatesh's essay [18]. Here I will focus on one important aspect: Will we have more confidence that a machine-assisted formal proof is objectively proven than the current more intuitive proofs? And if so, on what basis? To properly consider such questions we should first appreciate what the current system of verifying proofs is and what advantages it has (for more details on many of these points, see [8]). We should also better understand what machine proof-verifiers actually do, including the role of the human team inputing the proof into the machine. Amidst this, I will deliberately express some controversial opinions in the interest of provoking further discussion.

A brief history of proof. Mathematicians *prove* new statements by reducing them to those that have already been accepted as true:

> If... understanding is as we posited, it is necessary for demonstrative understanding... to depend on things which are true and primitive and immediate and more familiar than and prior to and explanatory of the conclusion.
>
> — Aristotle

This recursive process must have some starting point(s), some set of *axioms*—ideally elegantly formulated, as few as possible and consistent (we shouldn't be able to justify a statement and its negation from our axioms!)—to start the proving. Moreover, it would be best if the axioms allow us to prove all the theorems we are interested in, and if not, we should always be able to formulate a new axiom to help.

As far as we know, Euclid made the first serious attempt to formulate axioms and these were refined and developed well into the twentieth century. Cantor in particular argued that mathematical progress depends on conceptual innovation so we should always be ready to adapt our axioms, which should be *evidently true* and consistent. However, Frege's widely touted system was inconsistent (Russell created paradoxes within the interpretation of its language), and then Russell and Whitehead's replacement was not entirely self-evident and was unwieldy to implement.

These issues led Hilbert to suggest that we should be able to *start from any set of consistent axioms* (with a well-defined simple language, and appropriate inference rules for proofs) and see where that leads; in other words, one does not have to start from immortal truths. It is the proofs that conjure the mathematics into existence. Frege disagreed, arguing that a line is a line, the physical entity in the common vernacular, and if you produce a theory yielding something different with an obscure though consistent set of axioms then you should discard that theory.

Hilbert's rules inevitably lead to a plurality of axiomatic systems: for example, whether there is an infinity in between the infinity given by the integers and the infinity given by the reals in size, is independent of the ten standard axioms (ZFC), as proved by Gödel and Cohen. Allowing such alternative foundations seems to reduce mathematics to a game with certain rules, rather than developments based on profound insights, but this is where mathematicians today apply Cantor's rules, that we only accept possibilities that lead to interesting innovation.

Hilbert's hope was to find a set of consistent axioms that are complete (which allow us to to prove or disprove all mathematical statements), and provably so. However, in 1931, Kurt Gödel shook Hilbert's proposed mathematics to its foundations with his *incompleteness theorems*, which essentially say that no consistent system of axioms and rules that includes elementary arithmetic on the integers can be used to prove every true statement about the integers, nor can it prove the system itself to be consistent. Disaster. Gödel's theorems directly imply that

> There can be no rigorous justification for classical mathematics.
>
> — John von Neumann

Crisis response. Most pure mathematicians deal with this irresolvable foundational crisis by ignoring it.[1] For many mathematicians, Gödel's objections seem to be irrelevant to what they are working on. They need a formal reasoning system that is reliable in all reasonable circumstances (as in the recent rebirths of category theory). For the last century the basic axiomatic system (ZFC) has remained accepted as the essentially unchanged foundation for most of modern pure mathematics, despite Gödel's results, and yet it works. But there is no end-run around Gödel's theorems: various authors mistakenly believe that computers will somehow help, but that is without understanding the *Church–Turing thesis*. This suggests the working hypothesis that all sensible computational systems are *equivalent* and *universal*. This implies that they can each calculate anything that is calculable.[2] Therefore, they can imitate one another, and computers can perfectly imitate humans and human interaction.[3] Moreover, humans can perfectly imitate computers, so that our fundamental limitations are the same.[4]

> Digital computers . . . are intended to carry out any operations which could be done by a human.
>
> — Alan Turing

There is another, less appreciated, crisis. Simple counting arguments establish that NP-families of provable correct statements (those for which there is a proof of polynomial size in the length of the problem) form a rather tiny subset of the set of all families of provable correct statements in ZFC. Thus, in practice, no machine,

[1] We occupy the penthouse of a very high, seemingly solidly built tower, constructed over many generations. The engineers have informed us that the foundations are crumbling and cannot be fixed, and recommend demolition. However, they have no good plans for rebuilding to anything like the heights already reached. So instead we are hoping that we can continue on as if nothing is amiss, adding new floors and not worrying about what is going on below.

[2] There have been several distinct attempts to describe "effectively calculable" functions: Gödel and Herbrand's general recursive functions, Church's λ-calculus, and the functions calculable on Turing's theoretical model for a machine. Church, Kleene, and Turing showed that these definitions are equivalent, and so it is widely believed that since these were defined from such different perspectives and yet are equivalent, they must provide an appropriate notion of "effectively calculable".

[3] By definition, a human can perform the tasks of a Turing machine—at least a patient human with a large pile of scratch paper—and thus the theoretically equivalent capabilities of humans and machines. In the imitation game [17] an interrogator tries to distinguish between a human who tries to prove she is a human and a computer who tries to fool the interrogator into believing it is a woman. (In the original, pre-Turing game, the part of the computer is played by a mischievous male.) A computer that can imitate humans so perfectly will have passed the *Turing test*.

[4] Computers are faster than humans by some multiplicative factor, call it μ, so what a computer achieves in s seconds, a human can achieve in μs seconds. Nonetheless, humans can do all that computers can do with enough time and space.

no algorithm, no human or computer can hope to prove all—or even a tiny fraction of—the correct and provable mathematical statements deducible from this or any other plausible axiomatic system.[5]

2. Accepting a result by being convinced by a proof

We would like to believe that pure mathematics has objective standards of proof, that any theorem has been rigorously justified back to the axioms. But who verifies this, and how? One would like a highly skilled objective verifier, a trusted authority who can verify that the language and deduction rules have been validly used all the way back to the axioms. In a formal proof every inference is laid out so that the proof can be mechanically verified, requiring no intuition.[6] But would such a proof be convincing to mathematicians?

> Proof... supposedly establishes the undeniable truth of a piece of mathematics, [but] proof doesn't actually convince mathematicians of that truth... Something else does.
>
> — Eugenia Cheng [2]

Cheng claims that mathematical communication turns the author's beliefs into a *believed truth* of her reader via *plausible reasoning*. This reasoning takes the structure of a carefully worded intuitive proof, with enough deductive rigour to allay any fear of ambiguity or misdirection. It therefore fits into the mathematical community's perception of what is known and what should be known, suggesting a verifier should fit any new claims into a larger context to help the reader.

Indeed, for highly sophisticated advances, the mathematical community accepts proofs if experts can fully appreciate the overall strategy, and then if each part of the proof can be verified by an expert on that technique. Moreover, the process of verification often inspires new perspectives in the research community: Verifiers are rarely looking just to agree, rather, they are looking to incorporate new ideas and techniques into their own repertoire. They do not do this by simply reading the words of the author, but rather by re-interpreting the text (or lecture) into their own terms and matching up the new ideas with what they know or believe.

Believing a proof: A community perspective. We believe a proof, although only exposed to part of the argument, if it fits into our idea of the subject area, and if the details left to the reader seem standard, and are *robust*, in that the odd error should be easily fixable. Moreover, an author's belief in their own work can be enhanced by examination and good questions from colleagues:

> There is no... mathematician so expert in his science as to place entire confidence in his proof immediately on his discovery of it... Every time he runs over his proofs, his confidence increases; but still more by the approbation of his friends; and is rais'd to its ut-

[5]There is a secondary issue—is there an algorithm that in polynomial time can find proofs for the theorems in a given NP-family? If P=NP,

> then everything we are trying to do can be done... However, I think most people believe that P≠NP.
>
> — Avi Wigderson.

[6]Though this ideal presupposes consensus on the meaning and interpretation of each step and the inviolability of logic and language on the part of the verifier.

> most perfection by the universal assent and applauses of the learned world.
>
> — David Hume (1739)

Proofs are accepted as correct by peers interested in similar questions and are aware of the techniques used in the field.[7] To verify proofs mathematicians place the proof into a context they understand, their deep knowledge allowing them to skip and accept much tacit knowledge while verifying the (relatively few) details that are different from what has gone before. The peer wishes to add to their own intuition and scope, not simply agree that the proof's argument is correct. The reader is not passive. She wants to understand, to synthesize, and to use the ideas in her own research.

Different people get different things out of a reading and therefore a new research article can inspire new ideas in hitherto unforeseen directions. Even the same person can, at different times, get different things from reading an article; our understandings do change over time, sometimes even how we approach the whole area. Proofs accepted by these community standards might be wrong since the details are not carefully checked by the verifier (and indeed, most details are usually of less interest, as an experienced reader can reconstruct them) but the verifier is not looking for strict proof but rather understanding that allows the new work to be contextualized.[8] Proof and understanding are not synonymous, and we have to appreciate and accept how they match and how they differ.

A piece of mathematics feels right if it is about what "ought to be" [2], rather than "useful, fun, intriguing, beautiful, proved in detail", a perspective which motivates the approach that many take to proving theorems. Indeed, Paul Erdös claimed that an objective supreme being has a book which contains the perfect proof for every true theorem, each of which is short and elegant; Short, so it is easy to verify; and elegant, so one knows that the statement fits so well that it must be true.

Part of refereeing is deciding on the interest of the submission. Venkatesh [18] argues that "the value we assign to a work of mathematics is purely subjective, in the sense that it depends solely on the perception of that work, and not on any objective quality." Mathematics has long been directed by this kind of subjectivity and famously, Littlewood would ask referees, "Is it new? Is it correct? Is it surprising?" These questions are important, since jobs, status, and grants depend on where an author's work is published. Can there be an objective standard to judge new work or must we maintain our subjective choices? Can proof verification systems imitate (or improve on) human judgement?

The robust nature of proof. We would like a review system in which the proof guarantees the theorem, so a competent mathematician does not need to verify each proof she uses herself. Familiarity simplifies scrutiny (at some risk of favoring

[7] Ideally, a reviewer should be truly independent, but in practice—for high-level research based on the literature—no one but an expert has the interest and skills to review the work, especially if it is complicated enough to require a substantial time commitment. Moreover, an expert knows what is more-or-less standard in a new work and what is novel, and so can better focus their efforts on where identifiable mistakes are more likely to appear.

[8] And slip-ups can occur: In [8] we discuss some famous mistakes in world-leading mathematical research that were accepted by the community for a while, including works of Voevodsky, Biss, Mochizuki, and, to a lesser extent, Wiles; and how to understand this acceptance in terms of what the accepted standards of proof are.

conformity) so papers are more quickly and accurately refereed, focusing on those details that might be most likely, in the experience of the referee, to cause concern. This supports a belief in the *robust* nature of proof; we believe that not much can go wrong with well-used technical tools and so we make assumptions about what needs verifying. And even if there is a mistake, experience shows that a simple modification should be enough to make the argument work. Any experienced researcher does this regularly in developing their own work, so when they encounter minor technical flaws in the work of others, they tend to believe that they are fixable.

Commonalities between formal and intuitive proofs. All proofs are constructed in a similar way: Starting from agreed upon axioms we construct proofs of given statements. To advance far we need to avoid going back to the axioms all the time, so we need to build a library of statements that we know to be true and are unambiguously stated. Traditionally, this library is stored in research articles, and synthesized in books. Machines can also store this in their own language(s).

A reader hopes to trace a thread back to the point she believes a claim with no further explanation. This can be achieved through interactive links that allow a reader to dig progressively deeper, whether in a formal or intuitive proof.[9]

Naively, this discussion suggests that a researcher can make advances through logical deductions from what they quote from the library of known results. However, researchers typically build awareness of their subject through speculation and proof, and seeing new questions in context, the exact statement of already known results plays a subsidiary role.

Very different processes. Formal proofs chase the details of a proof back to the axioms, like a child tirelessly asking "Why?" (until one gets back to immutable truths). But, at the end of that process, does the child remember what they asked at the start and how they got to the end? And when the formal proof-verifier reports that a proof is correct, then do we really understand why any more than the child who is told, "Because I told you so"?

Community proof-verifiers typically learn from their work, and cannot only reproduce the proof (in principle), but can adapt the learned ideas to other questions. Participants in a community expect proofs they can understand, interpret, appreciate, and even use, if possible. They can be excited to find an alternative or clearer proof, though that seems to play no role in a formal system.

Eternal truths? Venkatesh [18] states that "a proof is ... an argument compelling consensus" and de Toffoli [5], that "Shar[ing] mathematical arguments... [is] a necessary condition of mathematical justification." But why do we need consensus, or indeed anyone else's input, if proofs build solidly from the axioms to the latest theorem? De Toffoli [4] argues that "criteria of acceptability for rigorous proofs are not carved in stone... but are indexed to a mathematical community in a particular time," citing proofs in basic calculus. Her argument implies that proofs inevitably belong to the latest paradigm. But surely there are simple immutable truths that are eternal, and not paradigm dependent?

> A triangle. This seems to be extremely simple, and you'd think we... know all about it... Even if we prove that it possesses all the

[9]Patrick Massot [11] recently announced that software tools are being developed to automatically convert formal proofs into such human-readable interactive proofs.

> attributes we can conceive of, some other mathematician, perhaps 1000 years into the future, may detect further properties in it; so we'll never know for sure that we have grasped everything that there is to grasp about the triangle.
>
> — René Descartes (1648)

In other words, no matter how much we feel we understand on a given question, we might have only scraped the surface and there may be a lot more to know. Moreover, we have no surefire way to know whether we have a complete understanding. And if our understanding cannot be considered complete or immutable even on simple questions, then can formal proofs be believed to work in every context?

But, surely in mathematics, objective proof follows from well-formulated axioms, language, and deductive rules?[10] If so, a formal proof can be believed to work in every potential context, and therefore mathematics deals in objective, eternal truth.

To decide which of these two perspectives to believe, we need to ask what scientific objectivity is, and whether it is attainable.

3. Myths of scientific objectivity

> Scientific objectivity... expresses the idea that scientific claims, methods, results, and scientists themselves... should not be influenced by particular perspectives, value judgments, community bias, or personal interests.[11]

Copernican theory was long objectively refuted by passages from the Bible. Racism and sexism were long supported and sustained by supposedly objective societal evidence. Yet today these supposedly "objective stances" are, objectively subjective and self-serving (see [9] for much more on this theme). Typically, calling an assertion "objectively true" gives it authority; it is rhetoric designed to quell doubt,[12] so it is important to know how such claims are justified.

If we agree to the above definition of objectivity (in that it dispells subjectivity), then how attainable is such objectivity in any science?

Karl Popper defined "objectivity" as faithfulness to facts, interpreted without the bias of a premeditated perspective, with the ability to adjust to changing information. However, this does not mean that objectivity necessarily correctly identifies eternal truths. He claimed that theories gradually come closer to truth, depending on our better collection and understandings of data, and becoming more objective over time. Thus, to Popper, objective truth is always something in front of us, to be strived for, but not objectively attainable.

Scientific realism claims that objective truths are independent of perception,[13] though Hilbert's plurality of possible axiomatic bases (and alternative definitions given in mathematical works) suggest that many different perceptions are equally

[10] Indeed, in 1895, Peano observed that "Imprecise ideas cannot be represented by symbols," inferring that if we appropriately use symbols then everything will be precise, and resulting truths will be objectively deduced. However, even if we agree that this is a way to be precise, one can be precise and wrong!

[11] For this quote and a more thorough and less opinionated version of the ideas in this section, see the Stanford Encyclopaedia of Philosophy [14].

[12] "Best practice" is rhetoric currently used by bureaucrats to quell dissent and to passive-aggressively shame doubters. Is a claim of "objective truth" any better?

[13] A room has a certain temperature, whether or not you find it hot or I find it cold.

valid! Even supposing we begin with the same axioms and definitions, each researcher's language use and viewpoint may lead to subtle differences with equally valid interpretations (which may be complementary, but it can be difficult to be certain). But who decides whose perception is objectively correct?

Kuhn [10] was particularly skeptical of claims of objectivity, claiming that all observations are made through the lens of a reigning paradigm, perceived and conceptualized by the latest (but not last) theoretical assumptions.[14] The paradigm guides scientists' approach to their work, setting the community standards.[15] This viewpoint makes it hard to believe in a theory-independent language, observations, or even objectivity; however, the concept of objectivity is arguably meaningful within a particuliar paradigm.

One might argue that precision, simplicity, coherence, and scope of an experiment or theory are value-free, and so *should* motivate preferences. However, if we had written accuracy, elegance, fit to theory, and connections to other parts of the theory, then these concepts are all inescapably embedded in some paradigm.

There have been massive paradigm shifts in mathematics in the last fifty years, in the value placed on different areas, stemming from the powerful influence of Grothendieck. His rethinking of algebraic geometry, functional analysis, and much else led the mathematical establishment to pay little attention to extraordinary work in any field not closely aligned with Grothendieck's work. For example, combinatorics was trivialized as "mere calculation" (as I heard Atiyah proclaim in the late 1990s)[16] and it took a long time—and substantial controversy—before a Fields medal was once again presented to research outside a narrow band of topics.

Despite all these good reasons to be skeptical about the possibility of objectivity in science, mathematicians try to discuss their ideas in unambiguous language with clear rules of inference. One can therefore argue that within this paradigm, pure mathematicians' claims are less influenced by preconceived notions, value judgements, and community bias than other sciences, and so are, arguably, relatively objective. For what it's worth, mathematical objectivity has suffered less from contextual influences than other sciences from pressures like climate denial, general politics, and needs of large businesses such as drug and food companies, though there are notable recent examples like cryptography in industry and government, interpretation of statistical data during the pandemic, and now the sometimes exaggerated claims of the machine learning industry.

Difficulties of definitions. No single definition is useful for many of the concepts studied in modern mathematics, and so all definitions should be reconciled.[17] This can be an issue for formal verifiers, especially as in cutting edge areas, many competing approaches (including definitions) can be tried, and only experience will

[14]But surely experimental evidence is objective? However, only if the experimental apparatus is reliable, and typically we only believe it's reliable if it confirms results obtained in line with our earlier beliefs. Thus experimental evidence is only understood from a perspective within its own paradigm.

[15]This can be consistent with Popper, for example, in considering the popular example of special relativity enhancing and supplanting Newtonian mechanics.

[16]Indeed, shamefully, Szemerédi was not recognized in the general mathematics community for his great 1975 work on structure in nonsparse sets [16] until the 2012 Abel prize.

[17]We are used to this in, for example, defining real numbers, or synonymous concepts in algebra and geometry.

determine which works best most often (and those judgement calls are inevitably subjective).

All formal languages must make choices as to how to explain ideas, meaning some thoughts are emphasized, others de-emphasized, or even excluded.[18] Therefore, some new developments are easily expressed and worked on, while others are not. That is a lot to reconcile with the notion of objectivity.

The uncertainty principle of objective proof verification. The history of mathematical practice suggests that *the less one questions a proof, the more susceptible it is to error.* This important principle strongly suggests one must find a wide variety of ways to explain and to verify any given proof, even a computer proof, and to look at it from as many different perspectives as possible.[19]

4. Computers and proofs

There are currently three main uses of computers in proofs:

• *Calculations in establishing a proof*: Authors might reduce their question to a large finite problem with too many cases to resolve by hand, and then eliminate the cases by computer (like the proofs of the four color theorem (4CT), Kepler's conjecture for 3-d sphere-packing, God's number for Rubik's cube, and the classification of finite simple groups). Authors might need to construct a tool with delicate and special properties that might only exist in high dimension, and this can only feasibly be found by computer (like Maynard's 400-dimensional sieve in his work on short gaps between primes). Authors can use computers to calculate large examples which may inspire understanding and then proofs; this is the traditional role of papers in *Mathematics of Computation*, and we have seen some spectacular recent examples using machine learning (see, e.g., [3]).[20]

• *Assistance in verifying the logic of an author's arguments, perhaps interactively, "computer-assisted proofs"*: This has been most useful in verifying long, detailed proofs, where angels fear to tread. For example, a team worked with Gonthier to verify the most recent proof of 4CT (which has 32 steps involving 633 subgraphs that need to be reduced) using the Coq v7.3.1 proof assistant; a team worked with Hales to verify his proof of Kepler's conjecture[21] using the HOL Light and Isabelle proof assistants.

There have been striking recent advances with Lean, most notably, the verification, correction, and improvement of a key result of Clausen and Scholze, discussed elsewhere in this volume. There are now a lot of people working with Lean, and the quality and quantity of ideas that have been codified is extraordinary and many areas of mathematics are seeing nontrivial results being verified. Moreover, as more researchers contribute to the system, interaction should move towards something

[18]This is also true of spoken languages. People who speak two languages find that some concepts are more naturally explained in one, some in the other.

[19]Uncertainty principles express that two related quantities, usually a function f and Lf for some given operator L, cannot both have small support. See, e.g., [19] for some examples. Here we claim that the amount we question a proof and its susceptibility to subtle error are analogously related.

[20]Though beware of the hype; for example, the title, "Guiding human intuition with AI" which is a self-congratulatory way to describe—and purloin—the main role of computers in mathematics for the last fifty years.

[21]The original proof was so complicated that a team of a dozen referees worked for four years on verifying the proof and only reported they were "99% confident" that the proof was correct.

resembling the high-level practice of mathematicians. With further work on the input and output languages, a system like this could be user-friendly and become an integral part of the mathematician's arsenal.

These proof-verifiers are interactive, where the user interactively breaks the proof down into simpler objects that the machine already knows about, and gives Lean hints. The proof assistant will determine whether the statement is "obviously" true or false based on its current library. If not, the user enters more details. The proof assistant therefore forces the user to explain their arguments in a rigorous way, and to fill in simpler steps than human mathematicians might feel they need.[22] For example, for a proof with ten lemmas, some the theorem prover will see and resolve quickly, others it might need more details until it can see its way to a proof. In doing so, the program learns more, maintains a library, and is perhaps more efficient when it next encounters similar issues.

Proof assistants can't judge whether a mathematical statement is interesting or important, only whether it is consistent with what it has been shown. They should eventually (maybe even soon) require less help, perhaps much less help. The eventual proofs are not human-readable, but thanks to the groundbreaking work of Massot and his team [11] that should change, converting machine proofs into high-level arguments that humans understand and appreciate, which allows us to have more faith in the output. Nonetheless, can we really trust proof assistants to be correct if they are only really checked by their own internal logic (which might propagate a subtle error)?[23] What would be best is if these proofs can be independently verified, perhaps by different programs run on different machines. In effect, we propose refereeing computer proof-verifiers output within their own community![24] We can design the future based on what already works.

• *Proving claimed theorems, "computer-generated proofs"*: Computers have a huge memory and are fast, whereas humans are much more limited and so have to avoid severe combinatorial explosion by bringing in *tactics* early on in pruning search trees. For example, a human might

— recognise which hypothesis in a statement is likely to be most important;
— identify a subsidiary goal to aim for to break the proof into smaller steps;
— identify and adapt a known proof technique to this new situation;
— use a few examples to guess at properties, rapidly pruning the search tree.

All of these are difficult to implement on a computer, so what tactics best serve computer-generated proofs? Ganesalingam and Gowers [6] aimed to design a computer verifier to learn and think like a human, asking which order to try different tactics in, whether computers can learn from their past experience, and whether we can devise a theory that better mimics human choices or work with a mix of

[22]Scholze and Gonthier both reported that they learned a lot during this laborious input process!

[23]Proof assistants are programmed in a *kernel*, which plays the same role as machine code in regular software, so "must be trusted implicitly" and are "generally as simple as possible" [15]. Humans work in the *vernacular* which is then translated by an *elaborator* so that the human's instructions can be implemented by the kernel, suitably translated. This is analogous to all computer programming and so has similar potential for error.

[24]This will require them to share a common language.

the two.[25] One goal is to adapt and win Turing's "imitation game" [17], but now a machine-created proof should be indistinguishable from a great human proof.[26]

Can machine learning algorithms construct proofs of previously unproved results, taking leadership away from humans? For now, relatively little is understood about creativity and intuition, in particular, how humans move from one understanding to a different one. To simulate this on a machine seems, for now, to be far away.

Surely we have seen examples of computers that think? Before computers, librarians were often credited with knowing a lot more than they really did (as the gatekeepers of so much knowledge). Computers are much bigger repositories for knowledge, more accessible and less proscribed by others, and can achieve some surprising feats; it is not surprising that they get credited with powers that they do not yet possess.

Machine learning typically develops an approach to very specific problems through simple algorithms cleverly constructed through use of experience (data). Creating a large database and analyzing it with specially formulated tools can be startlingly effective (like Google Translate or ChatGPT), but this is not the same as developing intuition (or even simulating intuition effectively).[27] There are as yet no "thinking machines".

Some of the recent machine learning tools (particularly those using large language models) can seem like they are pulling together ideas like a human, but their processes are so different that it is hard to believe that they can go beyond being an impressive, useful, but ultimately empty, facsimile, no more alive than Michelangelo's *David*.[28]

Computer errors. If formal proofs on a computer are to be objectively true then computers should become error-free. People often assume that if a computer program is reliable then it is "free from error" which flies in the face of their own experiences. Reliable systems, which have been heavily invested in, govern your credit card, phone, airlines, your university, and we have all had frustration with those! Besides hardware and software problems, there may be programming issues (e.g., not accounting for your particular situation) or implementation issues (e.g., ambiguous menus, or misleading instructions). Surely over time these can all be ironed out, and computers will be trouble free. However, upgrades might deal with a previously unmanaged case, but also provide new functions, which bring new issues! And if a system eventually becomes "perfect", how would we prove that it is indeed perfect?

Computer hardware glitches are usually recognized by outputs that are obviously inconsistent with other information. However, this does not help us when the

[25] And see Gowers's latest project at https://gowers.wordpress.com/2022/04/28/announcing-an-automatic-theorem-proving-project/#more-6531.

[26] In [6], Ganesalingam and Gowers selected problems to prove and got thousands of independent readers to try to distinguish which proofs were by their program and which by real people (see https://gowers.wordpress.com/2013/04/14). The results are encouraging, though of course this is not the Turing test, since it is not an independent arbiter that selected the problems.

[27] There is a lot of money and publicity surrounding the subject of machine learning and some other forms of artificial intelligence, so it is perhaps not surprising that many hyped advances are either exaggerated or are easily explained in terms of well-designed algorithms and extraordinary computing power.

[28] See Melanie Mitchell's wonderful book [12] for a forensic discussion of what underlies some of the recent developments in machine learning.

computer is calculating something that we know little about! Manufacturers rarely reveal concerns about their products (so as not to put off potential purchasers), so if there is a glitch, it is unlikely to be shared with users. Moreover, would a small mathematical error in a widely used computer chip be worth the cost of fixing for the manufacturers?[29]

Commercial software has about 1 bug per hundred lines of code; even the space program can only get it down to perhaps 1 per 10,000 lines. Corporations keep quiet about bugs to limit legal liability, and correcting bugs can create new bugs![30] There are recent program-verifiers designed to check that programs perform their claimed calculations, so there may soon be less errors, but that remains to be seen.

5. All the proofs yet to come...

Computer proof-verification and generation are fast-moving subjects and it is exciting to witness the beginning of these important developments. Brilliant people are getting involved and Kevin Buzzard takes the compelling view that, "the more people are familiar with the software, the sooner interesting things will happen"—Lean now has thousands of users.

My concern has been with the nature of proof and what proof will become. Now is surely the moment to identify how to best direct these developments to enhance the future of proofs in mathematical research. In this article I have largely focused on refuting the naive notion that formal proofs will improve objectivity. But more importantly, is that we should not lose the benefits of the community-based approach to proof that has long served us so well. Indeed, some of our traditions should guide our quest in developing computer proofs.

Most importantly, computers should serve our needs. Input should be in the standard lexicon, proofs should be presented to be human-readable, and verifiers developed to work alongside a research mathematician, giving her the ability to much more easily check out a proposed proof idea. When a program claims to have a proof then it should be verified by different computers run by different chips using different software.[31]

A formal computer proof is remarkably fragile, for if we find any errors at all then we have to wonder how perfect its logic is, and so it puts all of its claims into doubt. Therefore we will need to build in the robustness of traditional intuitive mathematical proofs so that minor flaws can be fixed.

Eventually, we would like informative proofs—not just verifications—*explaining* the mathematics; they need not give the shortest proof. This should allow us to modify what we learn and help us to develop new concepts. We should be able to interact with the program as one does with a (marvelously retentive) human. It can be desirable to give several proofs for the same theorem, for example, to highlight different themes—can a computer program be trained to identify such

[29]In 1993, Pentium released a chip which they subsequently found had a hardware bug affecting its floating point processor (even when dividing certain seven digit integers by each other). Rather than recall the flawed chips they kept quiet and corrected the problem in updates. However, Thomas Nicely made the error public in June, 1994 after he found it when he got a number theory calculation wrong. Pentium resisted a recall until IBM refused to ship their product.

[30]Noncommercial, open source software like Lean might have less bugs, and identified bugs may be more easily fixed, but there are still no guarantees that none remain!

[31]Then it would be highly unlikely to get an error at the same point in the claimed proof given by each system, though this is still no guarantee of correctness.

variants?[32] It is hard to describe what a good proof is (as in Erdös's "Book") but we know it when we see it: Can our computer programs always find such proofs? And, more worryingly, if computers do so much of the heavy lifting, will humans continue to appreciate book proofs? It is hard to predict the future of proof as we begin this period of profound change in the way in which we work with cutting-edge mathematics.

References

[1] Kevin Buzzard, *The Xena Project*, a blog at https://xenaproject.wordpress.com/

[2] Eugenia Cheng, *Mathematics, morally*, 2004 (preprint).

[3] Alex Davies, Peter Veličković, Geordie Williamson, et al., *Advancing mathematics by guiding human intuition with AI*, Nature 600 (2021), 70–74.

[4] Silvia De Toffoli, *Reconciling Rigor and intuition*, Erkenntnis **86** (2021), no. 6, 1783–1802, DOI 10.1007/s10670-020-00280-x. MR4331052

[5] Silvia De Toffoli, *Groundwork for a fallibilist account of mathematics*, The Philosophical Quarterly **71** (2021), 823–844.

[6] M. Ganesalingam and W. T. Gowers, *A fully automatic theorem prover with human-style output*, J. Automat. Reason. **58** (2017), no. 2, 253–291, DOI 10.1007/s10817-016-9377-1. MR3600894

[7] Kurt Gödel, *The modern development of the foundations of mathematics in the light of philosophy*, draft of a 1961 presentation to the American Philosophical Society, Collected Works, Vol III, Oxford (1995), 375–388.

[8] Andrew Granville, *Accepted proofs: Objective truth, or culturally robust?*, Annals of Math. and Philosophy 2 (2023), 66 pgs.

[9] Donna Haraway, *Situated knowledges: The science question in feminism and the privilege of partial perspective*, Feminist studies, 14 (1988), 575–599.

[10] Thomas S. Kuhn, *The Structure of Scientific Revolutions* (2nd ed), Chicago, University of Chicago Press, 1962.

[11] Patrick Massot, *Formal mathematics for mathematicians and mathematics students* lecture on youtube.com (https://youtu.be/tp_h3vzkObo).

[12] Melanie Mitchell, *Artificial Intelligence: A Guide for Thinking Humans*, Farrar, Straus and Giroux, 2019.

[13] Rodrigo Ochigame, *Automated mathematics and the reconfiguration of proof and labor*, Bull. Amer. Math. Soc. (N.S.), **(61)** 2024, no. 3, ISSN: 0273-0979.

[14] Julian Reiss and Jan Sprenger, *Scientific Objectivity*, The Stanford Encyclopedia of Philosophy (Winter 2020 Edition), Edward N. Zalta (ed.), https://plato.stanford.edu/cgi-bin/encyclopedia/archinfo.cgi?entry=scientific-objectivity

[15] Michael Shulman, *Strange new universes: Proof assistants and synthetic foundations*, Bull. Amer. Math. Soc. (N.S.), **(61)** 2024, no. 2, ISSN: 0273-9079.

[16] E. Szemerédi, *On sets of integers containing no k elements in arithmetic progression*, Acta Arith. **27** (1975), 199–245, DOI 10.4064/aa-27-1-199-245. MR369312

[17] A. M. Turing, *Computing machinery and intelligence*, Mind **59** (1950), 433–460, DOI 10.1093/mind/LIX.236.433. MR37064

[18] Akshay Venkatesh, *How we place value in mathematics*, Bull. Amer. Math. Soc. (N.S.), **(61)** 2024, no. 2, ISSN: 0273-9079.

[19] Avi Wigderson and Yuval Wigderson, *The uncertainty principle: variations on a theme*, Bull. Amer. Math. Soc. (N.S.) **58** (2021), no. 2, 225–261, DOI 10.1090/bull/1715. MR4229152

Département de Mathématiques et Statistique, Université de Montréal, CP 6128 succ Centre-Ville, Montréal, QC H3C 3J7, Canada.

Email address: andrew.granville@umontreal.ca

[32] One can ask whether a formalization of a given intuitive proof is going to be the same proof? When an intuitive proof is dissected into what is required for, say, Lean to work with, it will look very different and rest on a rather different looking library of knowledge. And how different will the same proof look when modified for a different proof-verification language?