A New Style of Mathematical Proof*

William M. Farmer[1]

McMaster University, Canada
wmfarmer@mcmaster.ca
http://imps.mcmaster.ca/wmfarmer/

7 September 2018

Abstract. Mathematical proofs will play a crucial role in building a *universal digital mathematics library (UDML)*. Traditional and formal style proofs do not adequately fulfill all the purposes that mathematical proofs have. We propose a new style of proof that fulfills seven purposes of mathematical proofs. We believe this style of proof is needed to build a highly interconnected UDML.

Keywords: Mathematical proof, traditional proof style, formal proof style, universal digital mathematics library, little theories method, theory graphs, flexiformalization, cross checks.

1 Introduction

Over the course of the next few decades, mathematical software systems will revolutionize how mathematical knowledge is expressed, organized, and applied. The end product of this revolution will be a *universal digital mathematics library (UDML)* containing vast amounts of highly interconnected mathematical knowledge.

We believe that the mathematical knowledge in a UDML should be represented in accordance with the *little theories method* [2] as a *theory graph* [4] consisting of axiomatic theories as nodes and theory morphisms as directed edges. The theories — which may have different underlying logics — serve as abstract mathematical models. The morphisms — which are meaning-preserving mappings from the formulas of one theory to the formulas of another — serve as information conduits that enable theory components such as definitions and theorems to be transported across the graph [1]. A theory graph enables mathematical knowledge to be formalized in the most convenient underlying logic at the most convenient level of abstraction using the most convenient vocabulary and then applied in many different contexts. In addition, the morphisms and other connections in a theory graph provide an infrastructure for finding relevant concepts and facts in the theory graph, e.g., all the definitions that are equivalent to a given definition.

* This paper is published in: J. H. Davenport, M. Kauers, G. Labahn, and J. Urban, eds, *Mathematical Software — ICMS 2018*, *Lecture Notes in Computer Science*, Vol. 10931, pp. 175–181, Springer, 2018. This research was supported by NSERC.

As one would expect, mathematical proofs will have a crucial role to play in the building of a UDML. They will serve as threads that tie the knowledge in a UDML together. We will argue that both the traditional proofs that appear in mathematical books and articles and the formal proofs developed using proof assistants are not adequate for the job and that a new style of proof is needed.

2 Styles of Mathematical Proof

A *proof* is a deductive argument intended to show that a mathematical statement is a logical consequence of a set of premises. There are many styles of proof. Some proofs *describe* a deduction of the statement from the premises, while other proofs *prescribe* the steps needed to produce the deduction. Many proofs are presented in a *two-column format* where each line in the left column is an intermediate result in a deduction and the corresponding line in the right column explains why the result is justified. Some proofs contain *computations* (e.g., numeric or algebraic simplifications) or *constructions* (e.g., via straightedge and compass). *Geometry proofs* are deductions guided by a geometric drawing. *Visual proofs* are presented by a series of diagrams or an animation.

The proofs presented in mathematical books and articles usually exhibit a particular style that we call the *traditional proof style*. Proofs of this style are arguments written in a stylized form of natural language with a heavy use of special symbols. In traditional proofs the terminology and notation may be ambiguous, assumptions may be unstated, and the argument may contain logical gaps. However, the reader is expected to be able to resolve the ambiguities, identify the unstated assumptions, and fill in the gaps in the argument. The writer — whose purpose is to serve some particular community of readers — has the freedom to express the argument in whatever manner is deemed most effective. This includes exhibiting other styles of proof within the traditional style.

The *formal proof style* is to present a proof as a derivation in a proof system for a formal logic. Using software systems, formal proofs can be interactively developed and mechanically checked. This style of proof is highly constrained by the logic, proof system, and the fact that every detail must be verified. On the other hand, there is a very high level of assurance that the statement proved is indeed a theorem of the proof system. Although the traditional proof style dominates mathematics, the formal proof style is beginning to make some modest inroads in mathematical practice.

3 Purposes of Mathematical Proof

Mathematical proofs serve (at least) seven purposes. For each of the seven, we describe what the purpose is and compare how well traditional and formal proofs fulfill the purpose.

Purpose 1: Communication

The main purpose of a proof given in a textbook or scientific article is to *communicate* to the reader why a mathematical statement follows from a set of premises. Proofs constructed for communication are used to convey insight and to build intuition. The highly flexible style of traditional proofs is usually a much better vehicle for communication than the highly constrained style of formal proofs. This is especially true when the writer is more concerned about high-level ideas than low-level details (that often can be mechanically checked by computation). However, formal proofs can be much more effective at presenting intricate syntactic manipulations than traditional proofs.

Purpose 2: Certification

Another important purpose of a proof is to *certify* that a mathematical statement follows from a set of premises. Such a proof serves as a certificate that can be independently checked. Since a traditional proof is written for a particular audience, it may not be easily checked by someone outside of this audience. Moreover, a traditional proof may contain mistakes that are not easily noticed by a reader, even a reader in the intended audience. In contrast, a formal proof can be mechanically checked by software alone. A formal proof thus offers the highest level of certification.

Purpose 3: Discovery

A proof is often formulated to be a provisional argument that a mathematician can use to *discover* new theorems. This idea is brilliantly expressed in *Proofs and Refutations* by Imre Lakatos [7]. See also Yehuda Rav, "Why Do We Prove Theorems?" [8]. Traditional proofs are well suited for expressing provisional arguments that can be analyzed by humans. Formal proofs are too rigid to express provisional arguments and thus are poorly suited for this task. On the other hand, machines can be used to discover various kinds of structure embodied in a formal proof, but it is much more difficult to analyze traditional proofs in this way.

Purpose 4: Learning

The most effective way to *learn* mathematics is to read and write proofs. Traditional proofs are today generally much easier to read and write than formal proofs. However, a reader of a traditional proof may have to work harder on resolving ambiguities, identifying unstated assumptions, and filling in the gaps in the argument, and a writer may have to work harder on verifying that each step of the argument is valid. With effective software support, reading and writing formal proofs could become almost as easy as reading and writing traditional proofs.

Purpose 5: Universality

A proof is *universal* if it is expressed without any superfluous ideas and can thus be applied in every context in which the conditions of the proof hold. Traditional proofs can be expressed in a universal manner, but the underlying mathematical foundation is usually implicit. Traditional proofs are thus untethered; they do not have a precise mathematical home. Formal proofs have a precise mathematical home, but the home is usually not connected to many other contexts in which the proof can be applied. Hence both traditional and formal proofs fall short in achieving universality.

Purpose 6: Coherency

A theorem is *coherent* with a body of mathematical knowledge if it properly fits into the body without any contradictions or unexpected relationships. A proof by itself does not establish that the theorem it proves is coherent. Most mathematicians are reluctant to accept a theorem on only the basis of its proof. There is always the possibility of error, especially if the proof is not machine checked. Georg Kreisel has noted in several of his papers, e.g., in [5, p. 126] and [6, p. 145], that a better way to avoid error than carefully checking a proof is to use *cross checks* to compare the result with known facts. For example, the proof can be checked against similarly structured proofs and the theorem can be compared with consequences of the theorem or related versions of the theorem that have been independently proven. Although cross checks are very important, they are rarely written down and are not considered as part of either a traditional or a formal proof.

Purpose 7: Beauty

Mathematics is a utilitarian art form like architecture or industrial design. The desire to create *beauty* (what mathematicians call *elegance*) is one of the strongest driving forces in mathematics. Mathematicians seek to develop proofs that are beautiful as well as correct. Indeed some mathematicians will not accept a theorem until an elegant proof of the theorem has been found. It is safe to say that most mathematicians find it easier to write beautiful proofs in the highly flexible traditional proof style than in the highly constrained formal proof style.

Summary

Table 1 summarizes the differences between traditional and formal proofs. As can be seen, neither traditional proofs nor formal proofs fulfill all the purposes that mathematical proofs have. Furthermore, both styles lack the capacity to fully achieve universality and coherency.

	Traditional Proofs	Formal Proofs
Communication	●	◒
Certification	◒	●
Discovery (Human)	●	○
Discovery (Machine)	○	●
Learning (Reading)	◓	◒
Learning (Writing)	◓	◒
Universality	◒	◒
Coherency	○	○
Beauty	●	○

● : high; ◓ : medium high; ◒ : medium low; ○ : low.

Table 1. Traditional vs. Formal Proofs

4 A Proposed New Style of Proof

Since traditional and formal proofs do not adequately achieve universality and coherency, they are not adequate for building a highly interconnected UDML. We therefore propose a new style of proof that is better suited for threading together the concepts and facts in a UDML. This new proof style has four components:

1. A *home theory* HT consisting of a formal logic Log, a language Lang in Log, and a set Axms of formulas in Lang that serve as the axioms of the theory.
2. A *theorem* Thm that is a formula in Lang purported to be a logical consequence of Axms.
3. An *argument* Arg that shows Thm is a logical consequence of Axms.
4. A set CC of *cross checks* that compare Arg with similar arguments and the theorem with related theorems.

The home theory is a node in a UDML and a formal context for the proof. It is connected via meaning-preserving morphisms to other theories in the UDML. Ideally, the home theory is at the optimal level of abstraction for the proof and contains only the concepts and assumptions needed to express the proof's argument and theorem.

The theorem is a formal statement of what the proof's argument shows. It can be transported via appropriate morphisms to other theories in which the conditions of the proof hold. The home theory HT and the theorem Thm together thus serve as a specification of the set of theories T and formulas A in the UDML's theory graph such that T is an instance of HT, A is an instance of Thm, and A is a theorem of T. In this way, the proof fulfills the purpose of universality.

The argument Arg has both a traditional component for communication, human-oriented discovery, learning, and beauty and a formal component for certification, machine-oriented discovery, and learning. The two components are tightly integrated so that, for example, a reader of the traditional component

can switch, if desired, to the formal component when a gap in the argument is reached. It is not necessary that the formal component is a complete formal proof of the theorem. The formal component can even be totally absent. Thus the proof is *flexiformal* [3].

The set of cross checks should be carefully chosen to show that the theorem is coherent with the web of previously established facts in the UDML. There are various kinds of cross checks that can be in CC. One kind is a similar proof of a similar theorem. A second kind is a logical consequence of Thm in HT that has been proved independently of Thm. For example, the logical consequence could be a special case of Thm or a corollary of Thm. A third kind is an instance of Thm in an instance of HT that has been proved independently of Thm. For example, the instance of Thm could be an expression of Thm in a more concrete setting than HT or the dual of Thm in HT under some notion of duality. With the set CC the proof thus fulfills the purpose of coherency.

Of course, it is possible that a cross check fails. This could indicate that a mistake has been made or that something is not adequately understood. Thus failed cross checks are valuable because they can lead to finding hidden mistakes and making new discoveries.

In summary, the new style of proof we propose is a mixture of the traditional and formal proof styles in which the context of the proof and the statement proved are formal, the argument of the proof is expressed in a traditional style, and parts of the argument may be integrated with formal derivations. The home theory of the proof is a node in a theory graph of a UDML that is an optimal expression of the context of the proof. And the cross checks of the proof connect the proof and the theorem to similar proofs and related theorems in the theory graph.

5 Conclusion

We have proposed a new style of proof that contains elements of the traditional and formal styles of proof. It fulfills the seven purposes of mathematical proofs including universality and coherency. We believe this proof style is the thread that is needed to interconnect the concepts and facts in a UDML. We also believe its use will promote the formalization of mathematical knowledge while preserving the benefits of both traditional and formal proofs.

Acknowledgments

The author would like to thank the referees for their comments. This research was supported by NSERC.

References

1. Barwise, J., Seligman, J.: Information Flow: The Logic of Distributed Systems, Tracts in Computer Science, vol. 44. Cambridge University Press (1997)

2. Farmer, W.M., Guttman, J.D., Thayer, F.J.: Little theories. In: Kapur, D. (ed.) Automated Deduction—CADE-11. Lecture Notes in Computer Science, vol. 607, pp. 567–581. Springer (1992)
3. Kohlhase, M.: The flexiformalist manifesto. In: Voronkov, A., Negru, V., Ida, T., Jebelean, T., Petcu, D., Watt, S.M., Zaharie, D. (eds.) 14th International Workshop on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC 2012). pp. 30–36. IEEE Press (2013)
4. Kohlhase, M.: Mathematical knowledge management: Transcending the one-brain-barrier with theory graphs. European Mathematical Society (EMS) Newsletter **92**, 22–27 (June 2014)
5. Kreisel, G.: Some uses of proof theory for finding computer programs. In: Colloque international de logique: Clermont-Ferrand 18-25 juillet 1975, Colloques internationaux du Centre national de la recherche scientifique, vol. 249, pp. 123–133. Centre national de la recherche scientifique (1977)
6. Kreisel, G.: Mathematical logic: Tool and object lesson for science. Synthese **62**, 139–151 (1985)
7. Lakatos, I.: Proofs and Refutations. Cambridge University Press (1976)
8. Rav, Y.: Why do we prove theorems. Philosophia Mathematica **7**, 5–41 (1999)