Learning to prove with proof assistants?

Emmanuel Beffara

LIG, Université Grenoble Alpes

including joint work with Iro Bartzia, Antoine Meyer, Julien Narboux, Faustine Oliva, Martin Bodin, Rémi Molinier, Luigi Bernardi …

IRN-LI kickoff meeting
Roma, 12 maggio 2025

Introduction

Introducing the topic

We all know

- what it means to prove things (practically),
- what makes something a mathematical proof (intuitively),
- what is a formal proof (choose your preferred formalism).

Many of us

- teach math or computer science,
- want to make students understand logic.

Some of us

- prove some of their results in proof assistants,
- consider that formalised proofs have extra value,
- appreciate how proof assistants are like calculators for proofs.

Why not use proof assistants as a tool for teaching logic?

Proof assistants and teaching

Many attempts to use proof assistants for teaching, with the intent to address the difficulties of students:

- By experts in logic or programming languages, since 1975
- Used to teach logic, mathematics or computer science
- Development of interfaces and tools, on top of general purpose proof assistants or as standalone systems
- Observation of advantages in both proving and proof understanding

> We already have some sense that proof assistants greatly diminish the need for verification and justification, but we know almost nothing of their potential contribution to other roles of proof, such as explanation, communication, discovery, and systematization, or how they now may become more relevant as pedagogical motivation for the learning of proof in the classroom.

Proof Technology in Mathematics Research and Training
Hanna, Reid & de Villiers, 2019

Formal vs practical proof

What makes a proof

Characteristics of a proof

What makes an argument a mathematical proof:

- convincingness (for its target audience)
- surveyability (by the community, cf. peer reviewing)
- formalisability (in principle or in practice)

The four-color problem and its philosophical significance
Tymoczko, 1979

Ideas developed after the beginnings of computer-aided proving.

- What is the status of the computer program in Appel and Haken's proof of the four-color theorem published in 1977?
- What is the status of the software ecosystem in a proof formalised in a proof assistant today?

The act of proving

The roles of proving

- validating assertions (verification, explanation)
- communicating (systematisation, transmission to a community)
- part of the research process (exploration of theories, conjecturing)

Proof, explanation and exploration: an overview
Hanna, 2000

Types of reasoning involved in the act of proving:

1. deductive
2. abductive
3. inductive
4. conceptualization
5. counterexample finding
6. ...

Types of reasoning involved in a formal proof:

1. deductive

Proving in the classroom

Why prove in the classroom?

- the status of mathematical statements
- forms of validation for abstract notions
- understanding of mathematical notions

How is proof taught?

- by mimicry of model practices
- through situations which require systematic logical study
- by gradual transition from argumentation/explanation to proof

Processus de preuve et situations de validation
Balacheff, 1987

What goals when using proof assistants?

- teaching proof assistants – formal methods in CS curricula
- teaching mathematical proof – various experiments in first years

Utilisation des assistants de preuves pour l'enseignement en L1.
Kerjean, Le Roux, Massot, Mayero, Mesnil, Modeste, Narboux & Rousselin, 2022.

Interacting with a proof assistant

Interaction in proving

A proving task in a teaching situation ideally implies

- exploration, trial and error, conjectures;
- contradictory debate for
  - trying one's arguments and refining them,
  - reaching consensus within a group;
- "internal debate" of the same kind.

A kind of transposition of mathematicians' practice.

Proofs and refutations. The logic of mathematical discovery
Lakatos, 1976

Research questions

- What are the possible effects of using proof assistants on students' learning of proof?
- What characteristics of proof assistants are likely to strengthen or obstruct these effects?
- How do they shape the activity of proving?

Linguistic aspect of interaction

Different points of view on proof can lead to different choices:

Focus	Kind of proof assistant
Pure validation	Proof assistant using tactic languages (Coq, Lean...)
Validation and communication, transfer to written proof	Proof assistant with declarative or controlled natural language (Mizar, Isabelle/Isar, Lean Verbose, Coq Waterproof, Lurch...)
Conceptualization, conjectures, counterexamples	HRL (Pease and Colton), QED-Tutrix (Richard) (intelligent tutors for learning)

Kinds of languages to describe proofs:

Imperative We give orders (called tactics) to complete the proof tree

Declarative The proof is a sequence of mathematical assertions and their justification.

But everything is essentially **text**.

Categorising effects of proof assistants

Aspects affected by the use of proof assistants

- Memorizing and formulation
- Manipulation of formal statements
- Perception of a reasoning's structure
- Language level and precision

Features that can have an impact

- *Language and interaction mode*: type of user input, imperative or declarative style, object naming, possibility of writing ill-formed statements
- *Automation and user assistance*: mathematical libraries, rule selection and application, scope management, rule chaining and automated computation, type of feedback
- *Proof structure and proof state visualization*: global or local viewpoint on proof, status of statements, possibility to create new definitions and lemmas

Observations

- Students can solve more exercises with a point-and-click user interface.
- The tools have a tendency to induce trial-and-error strategies.
- Controlled natural language has an effect on student productions.
- Tunnel effect: limiting the view to local state and hiding global goals and directions

Proof assistants for undergraduate mathematics and computer science education: elements of a priori analysis.
Bartzia, EB, Meyer & Narboux, ongoing work.

Logical foundations and the classroom

Foundations of proof assistants

Proof assistants are built on different logical foundations:

- Type theory (Coq, Lean, Matita, Agda)
- Set theory (Mizar, Isabelle/ZFC)
- Higher-order logic (HOL-Light, Isabelle/HOL)

Foundations can be visible in the proof language (e.g. apply a lemma to an argument).

What to do with the foundations?

Two strategies:

1. Try to hide the foundations
   Example: Distinction between the introduction rule for ⇒ ("Assume that n is even") and introduction rule for ∀ ("Fix n / Take n") in Lean Verbose and CoqWaterProof.
2. Explicitly teach the proof assistant and its foundations

Foundations and the teaching of logic

One step away from proof assistants, foundations provide ways to understand the nature of mathematical reasoning:

- proofs as programs – computing ways to refute potential contradictions
- proofs as strategies
  - mathematical debate as a combinatorial game
  - proofs as winning strategies
- games as precursors to proofs – the reasoning structures of mathematical proof without the formalism

ongoing work with Bernardi