Math Reasoning in Times of AI: LEAN way of theorem proving

Jan Čepička, Josef Daněk, Petr Girg, Petr Nečesal & Jan Pospíšil

Department of Mathematics (KMA), Faculty of Applied Sciences (FAV)
University of West Bohemia in Pilsen (ZČU)

Powerful proof assistant

https://lean-lang.org

Mathematics can be approached through different computational lenses. **LEAN** acts like a programming language and proof assistant, where theorems and proofs are written as verifiable code. It enables mathematicians to formalize complex structures step by step, ensuring absolute rigor and supporting large-scale collaboration. **Automatic Theorem Provers (ATP)** push further toward full automation by using logic and search to establish results, excelling at routine or highly technical proofs but offering less conceptual guidance. In contrast, **Large Language Models (LLMs)** generate intuitive explanations, sketch proofs, and even draft LEAN code. Together, these approaches illustrate a spectrum: from LEAN's human-guided rigor, to ATP's automated precision, to LLMs' flexible but informal reasoning. The use of LEAN represents a new phase in a decades-long pursuit of automating research in mathematical analysis – **a tradition that has been evolving at KMA for over 30 years**.

Theorem (Limit Law for Sum of Convergent Sequences):

$$\lim_{n\to+\infty}(a_n+b_n)=\lim_{n\to+\infty}a_n+\lim_{n\to+\infty}b_n$$

Standard proof steps (when performed by hand)

- We have two convergent sequences (a_n) and (b_n).
- Let us denote their limits as c and d. Thus, we have that
$$a_n\to c,\qquad b_n\to d.$$
- Let us take an arbitrary $\varepsilon>0$.
- There exists an index $N_a\in\mathbb{N}$ such that $|a_n-c|<\frac{\varepsilon}{2}$ for all $n\geq N_a$ (due to $a_n\to c$).
- There exists an index $N_b\in\mathbb{N}$ such that $|b_n-d|<\frac{\varepsilon}{2}$ for all $n\geq N_b$ (due to $b_n\to d$).
- Let us take the index $N:=\max\{N_a,N_b\}$.
- Then for all $n\geq N$, we have
$$|a_n+b_n-(c+d)|=|(a_n-c)+(b_n-d)|\leq|a_n-c|+|b_n-d|<\tfrac{\varepsilon}{2}+\tfrac{\varepsilon}{2}=\varepsilon,$$
which means that
$$(a_n+b_n)\to(c+d),$$
and the proof is finished. ■

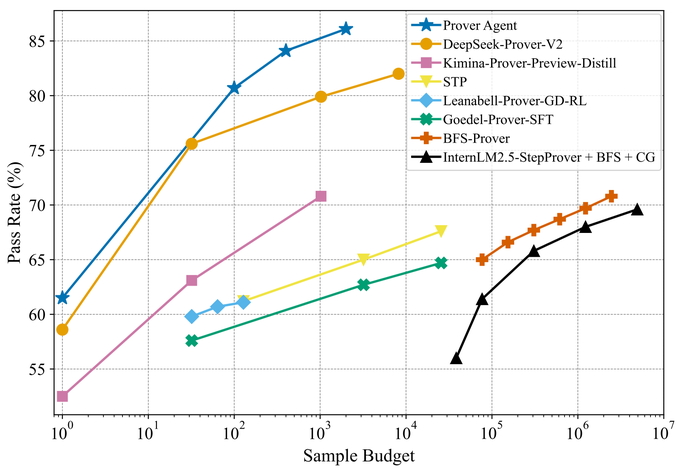
Proof in LEAN (machine-checkable down to the level of AXIOMs)

```
import Mathlib

def ConvergesTo (a : ℕ → ℝ) (c : ℝ) :=
  ∀ ε > 0, ∃ N, ∀ n ≥ N, |a n - c| < ε

theorem limit_add
  {a b : ℕ → ℝ} {c d : ℝ}
  (ha : ConvergesTo a c) (hb : ConvergesTo b d) :
  ConvergesTo (λ n => a n + b n) (c + d) :=
by
  intros ε ε_pos
  have ε2_pos : ε / 2 > 0 := by linarith
  obtain ⟨Na, haN⟩ := ha (ε / 2) ε2_pos
  obtain ⟨Nb, hbN⟩ := hb (ε / 2) ε2_pos
  let N := max Na Nb
  use N
  intros n hn
  have ha_appl := haN n (le_trans (le_max_left _ _) hn)
  have hb_appl := hbN n (le_trans (le_max_right _ _) hn)
  calc
    |a n + b n - (c + d)| = |(a n - c) + (b n - d)| := by ring_nf
    _ ≤ |a n - c| + |b n - d| := abs_add _ _
    _ < ε / 2 + ε / 2 := add_lt_add ha_appl hb_appl
    _ = ε := by ring
```

Comparison of AI agents for mathematical proofs

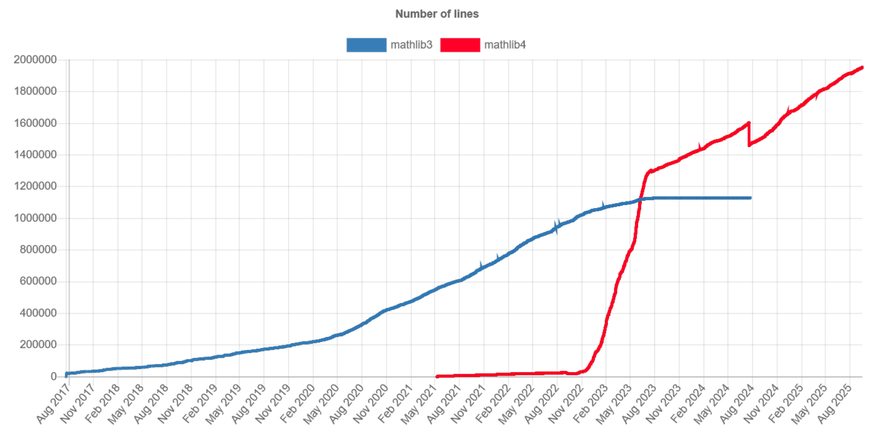


Mathlib — comprehensive mathematical library

Mathlib is a community-driven library of formalized mathematics written in Lean. It provides the building blocks needed for mathematical research across many domains of mathematics and computer science.

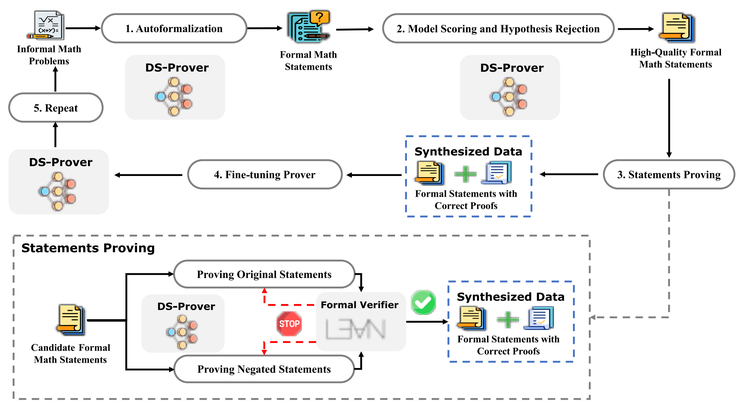
Mathlib statistics	Definitions	Theorems	Contributors
	115437	232180	653

Mathlib code growth



DeepSeek-Prover

In the following diagram, we can see how DS-Prover uses LEAN to prove theorems in LLMs.



References

- J. Avigad, L. de Moura, S. Kong, S. Ullrich, Theorem Proving in Lean 4, 2023, https://leanprover.github.io/theorem_proving_in_lean4/.
- K. Baba, C. Liu, S. Kurita, A. Sannai, Prover agent: An agent-based framework for formal mathematical proofs (2025). URL https://arxiv.org/abs/2506.19923
- D. T. Christiansen, Functional Programming in Lean, 2023, https://lean-lang.org/functional_programming_in_lean/.
- Lean Community, mathlib documentation, https://leanprover-community.github.io/mathlib_docs/ (2023).
- P. Song, K. Yang, A. Anandkumar, Lean copilot: Large language models as copilots for theorem proving in lean, in: Proceedings of the International Conference on Neuro-symbolic Systems, vol. 288 of Proceedings of Machine Learning Research, PMLR, 2025, pp. 144–169.
- T. Tao, Analysis I, vol. 37 of Texts Read. Math., 2nd ed., New Delhi: Hindustan Book Agency, 2009.
- T. Tao, Analysis I (online version), https://teorth.github.io/analysis/ (2025).
- H. Xin, D. Guo, Z. Shao, Z. Ren, Q. Zhu, B. Liu, C. Ruan, W. Li, X. Liang, Advancing theorem proving in LLMs through large-scale synthetic data, in: The 4th Workshop on Mathematical Reasoning and AI at NeurIPS'24, 2024.

http://www.kma.zcu.cz cepicka@kma.zcu.cz, danek@kma.zcu.cz, pgirg@kma.zcu.cz, pnecesal@kma.zcu.cz, honik@kma.zcu.cz