

# Ax-Prover: Revolutionizing Automated Theorem Proving with Multi-Agent Systems

In the ever-evolving landscape of artificial intelligence, Ax-Prover emerges as a groundbreaking multi-agent system designed to automate theorem proving in mathematics and quantum physics. By seamlessly integrating the creative reasoning capabilities of large language models (LLMs) with the formal verification rigor of the Lean proof assistant, Ax-Prover addresses longstanding challenges in automated reasoning and sets a new standard for efficiency and adaptability.

## Architecture: A Symphony of Agents

At the heart of Ax-Prover lies a sophisticated architecture comprising three specialized agents: the Orchestrator, Prover, and Verifier. Coordinated through the Model Context Protocol (MCP), these agents engage in a closed-loop process of problem dispatch, iterative construction, and verification, ensuring the generation of formally validated Lean proofs.

- **Orchestrator:** The maestro of this ensemble, the Orchestrator schedules proof tasks, distributes subtasks, manages feedback, and maintains the refinement loop. It orchestrates the collaborative efforts of the Prover and Verifier, ensuring that proofs are either verified or resources are optimally utilized.
- **Prover:** Leveraging the linguistic prowess of general-purpose LLMs, such as Claude Sonnet 4, the Prover synthesizes natural language proof sketches and incrementally translates them into Lean code. By utilizing Lean tools via MCP, the Prover enforces correctness through regular verification, bridging the gap between creative intuition and formal precision.
- **Verifier:** With a meticulous eye for detail, the Verifier operates on diagnostics from Lean to ensure that proofs are error-free and devoid of unproven placeholders. By collaborating closely with the Prover, the Verifier guarantees the integrity and reliability of the final proof.

## Benchmark Performance: Setting New Standards

Ax-Prover's prowess is evident in its impressive benchmark performance across various mathematical and scientific domains. Evaluated on both existing and newly created Lean benchmarks, Ax-Prover consistently outperforms specialized provers and achieves competitive results on established ones.

- **NuminaMath-LEAN:** Ax-Prover achieves an overall accuracy of 51%, with a notable Pass@1 rate of 26% on unsolved problems, showcasing its ability to tackle complex mathematical challenges.
- **Abstract Algebra AA:** With an overall accuracy of 64%, Ax-Prover surpasses Mathlib LLMs, demonstrating its expertise in abstract algebraic structures.
- **QuantumTheorems QT:** Achieving a remarkable overall accuracy of 96%, Ax-Prover provides full coverage of easy problems and excels in quantum theory theorem proving.
- **PutnamBench:** Ranked third with an accuracy of 14%, Ax-Prover exhibits strong sample efficiency, outperforming other specialized provers using fewer compute resources.

## Generalization and Domain Adaptability

Unlike systems confined to narrow domains, Ax-Prover harnesses the broad-domain knowledge inherent in general-purpose LLMs. Through the MCP, it maintains up-to-date interaction with Lean libraries, enabling rapid adaptation to diverse disciplines such as algebra, quantum physics, and cryptography. This adaptability is further enhanced by its modular multi-agent framework, which supports component interchangeability and parallel development.

## Practical Use Case: Cryptography Theorem Formalization

Ax-Prover's collaborative capabilities were put to the test in the formalization of a cryptography theorem related to branch number computation for non-singular matrices over finite fields. By co-structuring the proof, verifying lemmas, and error-checking intermediate steps, Ax-Prover assisted a human expert in completing the formalization on modest hardware within two working days. This practical use case underscores Ax-Prover's usability and potential to accelerate scientific research.

## Future Directions: Towards a Learning Scientific Assistant

As Ax-Prover continues to evolve, ongoing development efforts focus on parallelization, long-term memory modules, and enhanced reasoning capabilities. These enhancements aim to transform Ax-Prover into a continually learning, memory-augmented scientific assistant capable of reliable reasoning across formalizable domains. By addressing known limitations of specialization and enabling rapid formalization in emerging fields, Ax-Prover paves the way for verifiable scientific artificial intelligence.

In conclusion, Ax-Prover represents a significant advancement in automated theorem proving, combining the strengths of LLMs and Lean to create a robust, adaptable, and collaborative framework. Its achievements in benchmark performance, domain adaptability, and practical applications position it as a cornerstone of modern scientific discovery, driving innovation and expanding the frontiers of knowledge.

**Contributor:** Alessandro Linzi