Prover Agent: An Agent-Based Framework for Formal Mathematical Proofs

Kaito Baba[1], Chaoran Liu[2], Shuhei Kurita[3 2], Akiyoshi Sannai[4 5 6 2 7]

[1]The University of Tokyo, Tokyo, Japan [2]Research and Development Center for Large Language Models, National Institute of Informatics, Tokyo, Japan
[3]National Institute of Informatics, Tokyo, Japan [4]Kyoto University, Kyoto, Japan [5]Shiga University, Shiga, Japan
[6]RIKEN Center for Advanced General Intelligence for Science Program, Kobe, Japan [7]National Institute of Science Technology Policy (NISTEP), Tokyo, Japan

Our code

Our paper

Motivation

- Large language models (LLMs):
  - ✔ Capable of powerful reasoning and generation
  - ✘ Prone to errors and hallucinations
- Formal proof assistants (e.g., Lean):
  - ✔ Verify mathematical correctness
  - ✘ Not generative; requires painstaking meticulous detail

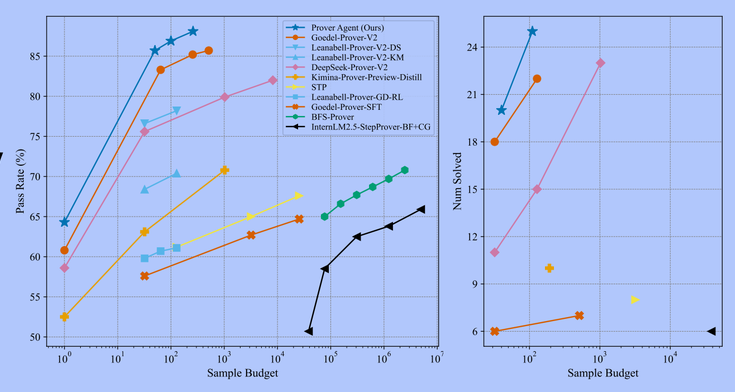
➡ LLM-based formal proving is gaining attention

✘ Yet, a large gap remains between informal reasoning and formal proving

Our Goal: Bridge this gap

Our Contributions

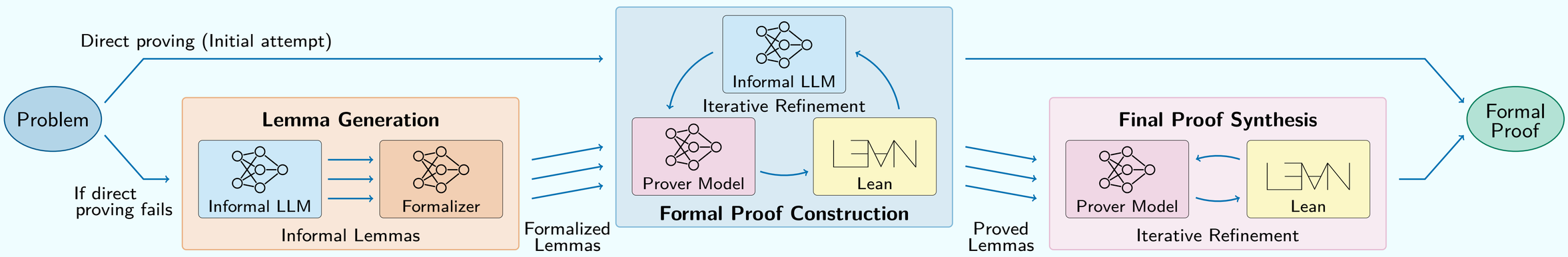
- Coordination of informal and formal reasoning with Lean feedback
- Auxiliary lemma generation for strategy discovery
  - Helps discover strategies even when the solution path is not apparent at first
- State-of-the-art theorem-proving performance among methods using small language models
- Efficiency in inference-time cost
  - Much smaller sample budget than prior work



MiniF2F PutnamBench

Comparison of theorem-proving performance

Prover Agent



◇ Three Key Components of Prover Agent ◇

1 Lemma Generation via Informal Reasoning

- Generate auxiliary lemmas
  - Specific cases
  - Potentially useful intermediate facts
- Not limited to subgoals of predefined proof sketch
  - Key difference from prior approachs
- e.g. Problem: Show that $n^2 + an$ is even ($n \in \mathbb{N}$, a: even)
  - Consider $n^2 + n$ or $n^2 + 3n$

➡ Help discover overall proof strategy

✔ Mirrors how human mathematicians typically work

2 Formal Proof Construction Guided by Informal Reasoning and Iterative Feedback

- Leverage the stronger mathematical ability of the informal LLM
- Construct a formal proof using an informal proof as a guide
- Iteratively refine the proof based on Lean feedback
  - Can be seen as self-correction through in-context learning
  - Akin to how humans improve their understanding based on feedback

3 Final Proof Synthesis Guided by Verified Lemmas and Iterative Feedback

- Consider overall proof using the lemmas
  - Use only the verified lemmas
- Allows bottom-up strategy construction even when the full plan isn't initially clear
  - Prior work: top-down approach requiring the full plan upfront
- Iteratively refine the proof based on Lean feedback

Experiments

◇ Experimental Setup ◇

- Informal LLM: `DeepSeek-R1-0528-Qwen3-8B`
- Formal prover model: `Goedel-Prover-V2-8B`/`DeepSeek-Prover-V2-7B`
- Formalizer: `Goedel-Formalizer-V2-8B`/`Kimina-Autoformalizer-7B`

◇ Comparison of Formal Theorem-Proving Performance ◇

Prover System		Method	Model Size	Sample Budget	miniF2F-test
Large Language Models					
DSP+ (Cao et al., 2025)	w/ QwQ, DeepSeek-V3, and BFS-Prover	Informal + Tree search	671B	1 128 1024	52.5% 74.2% 79.5%
	w/ DeepSeek-R1, DeepSeek-V3, and BFS-Prover			1024	80.7%
DeepSeek-Prover-V2 (Ren et al., 2025)		Whole-proof	671B	1 1024 8192	61.9% 86.6% 88.9%
Delta-Prover (Zhou et al., 2025)	w/ Gemini 2.5 Pro	Agent	unknown	16384	95.9%
Seed-Prover (Chen et al., 2025)		Whole-proof	unknown	unknown	**99.6%**
Medium Language Models					
Kimina-Prover-Preview (Wang et al., 2025)		Whole-proof	72B	1 1024 8192	52.9% 77.9% 80.7%
Goedel-Prover-V2 (Lin et al., 2025b)		Whole-proof	32B	32 1024 8192	88.1% 91.8% **92.2%**
Small Language Models					
DeepSeek-Prover-V1.5-RL + RMaxTS (Xin et al., 2025a)		Tree search	7B	$32 \times 16 \times 400$	63.5%
InternLM2.5-StepProver-BF + CG (Wu et al., 2024)		Tree search	7B	$256 \times 32 \times 600$	65.9%
HunyuanProver v16 + BFS + DC (Li et al., 2025)		Tree search	7B	$600 \times 8 \times 400$	68.4%
BFS-Prover (Xin et al., 2025b)		Tree search	7B	$2048 \times 2 \times 600$	70.8%
Leanabell-Prover-GD-RL (Zhang et al., 2025)		Whole-proof	7B	128	61.1%
Goedel-Prover-SFT (Lin et al., 2025a)		Whole-proof	7B	25600	64.7%
STP (Dong & Ma, 2025)		Whole-proof	7B	25600	67.6%
Kimina-Prover-Preview-Distill (Wang et al., 2025)		Whole-proof	7B	1 32 1024	52.5% 63.1% 70.8%
DeepSeek-Prover-V2 (Ren et al., 2025)		Whole-proof	7B	1 32 1024 8192	58.6% 75.6% 79.9% 82.0%
Leanabell-Prover-V2-KM (Ji et al., 2025)		Whole-proof	7B	32 128	68.4% 70.4%
Leanabell-Prover-V2-DS (Ji et al., 2025)		Whole-proof	7B	32 128	76.6% 78.2%
Goedel-Prover-V2 (Lin et al., 2025b)		Whole-proof	7B	1 64 256 512	60.8% 83.3% 85.2% 85.7%
Prover Agent (Ours)	w/ DeepSeek-Prover-V2	(Direct proving w/o iterative refinement) (Direct proving w/o iterative refinement) (Direct proving w/ iterative refinement) (Final proof synthesis w/ lemma)	Agent	8B	1 50 100 260	61.5% 79.9% 82.0% 82.8%
	w/ Goedel-Prover-V2	(Direct proving w/o iterative refinement) (Direct proving w/o iterative refinement) (Direct proving w/ iterative refinement) (Final proof synthesis w/ lemma)			1 50 100 260	64.3% 84.4% 85.7% 86.5%
	w/ Ensemble of Goedel-Prover-V2 and DeepSeek-Prover-V2	(Direct proving w/o iterative refinement) (Direct proving w/o iterative refinement) (Direct proving w/ iterative refinement) (Final proof synthesis w/ lemma)			1 50 100 260	64.3% 85.7% 86.9% **88.1%**

✔ State-of-the-art performance among methods using SLMs

✔ High success rate under low sample budget

✔ Better performance than prior work through coordination

◇ Performance on Olympiad-Level Problems ◇

		Model Size	Sample Budget	Olympiad				MATH			Custom			
				IMO	AIME	AMC	Sum	Algebra	Number Theory	Sum	Algebra	Number Theory	Induction	Sum
Number of Problems				20	15	45	80	70	60	130	18	8	8	34
Prover Agent (Ours)	(Direct proving w/o iterative refinement)	8B	1	40.0	53.3	62.2	55.0	71.4	60.0	66.2	55.6	75.0	50.0	58.8
	(Direct proving w/o iterative refinement)		50	70.0	80.0	82.2	78.8	80.0	88.3	83.8	66.7	75.0	62.5	67.6
	(Direct proving w/ iterative refinement)		100	70.0	80.0	86.7	81.3	84.3	88.3	86.2	66.7	75.0	62.5	67.6
	(Final proof synthesis w/ lemma)		260	**70.0**	80.0	**88.9**	**82.5**	84.3	88.3	86.2	66.7	75.0	75.0	70.6
DeepSeek-Prover-V2 (Ren et al., 2025)		671B	8192	50.0	**93.3**	77.8	73.8	**100.0**	**96.7**	**98.5**	**83.3**	**87.5**	**100.0**	**88.2**

- Show strong performance on Olympiad-level problems
  - ➡ Suggest that coordination with informal reasoning may be the key
    - Olympiad-level problems require a high degree of mathematical reasoning
- Consistent gap in MATH and Custom
  - ➡ Suggests that model size and sample budget may play a more significant role here
    - Prover model also possesses a certain level of mathematical reasoning ability

◇ Case Study: Success with Lemma-Guided Proofs ◇

- Problem:

```
theorem induction_nfactltnexpnm1ngt3
  (n : ℕ)
  (h₀ : 3 ≤ n) :
  (n)! < n^(n - 1) := by sorry
```

- Reasoning trace w/ lemmas:
  - ✔ Consider the specific cases for $n = 3, 4, 5$
    - ➡ Clearly identify the use of mathematical induction
  - ✔ Employ proof techniques used in the lemmas
- Reasoning trace w/o lemmas:
  - ✘ Proof strategy is unclear
  - ✘ The details cannot be worked out sufficiently

- Generated lemmas:

```
theorem base_case_3 : (3 : ℕ)! < 3^(3 - 1) := by
  have h_main : (3 : ℕ)! < 3^(3 - 1) := by
   -- Calculate the factorial and the power step-by-step
   norm_num [Nat.factorial, Nat.pow_succ, Nat.mul_assoc]
   -- Use 'decide' to confirm the inequality
   <;> decide
  exact h_main
```

```
theorem exponent_inequality {n : ℕ} (hn : 2 ≤ n) :
    n^(n - 1) < (n + 1)^(n - 1) := by
  have h_main : n^(n - 1) < (n + 1)^(n - 1) := by
    have h₀ : n - 1 + 1 = n := by
      omega
    have h₁ : n ≥ 2 := by omega
    have h₂ : n - 1 ≥ 1 := by omega
    have h₃ : n < n + 1 := by omega
    -- Use the fact that if a < b and n > 0, then a^n < b^n
    exact calc
      n^(n - 1) < (n + 1)^(n - 1) := by
        -- Apply the lemma that if a < b and n > 0, then a^n < b^n
        exact Nat.pow_lt_pow_of_lt_left h₃ (by omega)
      _ = (n + 1)^(n - 1) := by rfl
  exact h_main
```

The full reasoning trace can be found in the appendix of our paper.

AI, Proof and Verification 2026 (AIPV 2026), collocated with the 27th International Symposium on Formal Methods (FM 2026).