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 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

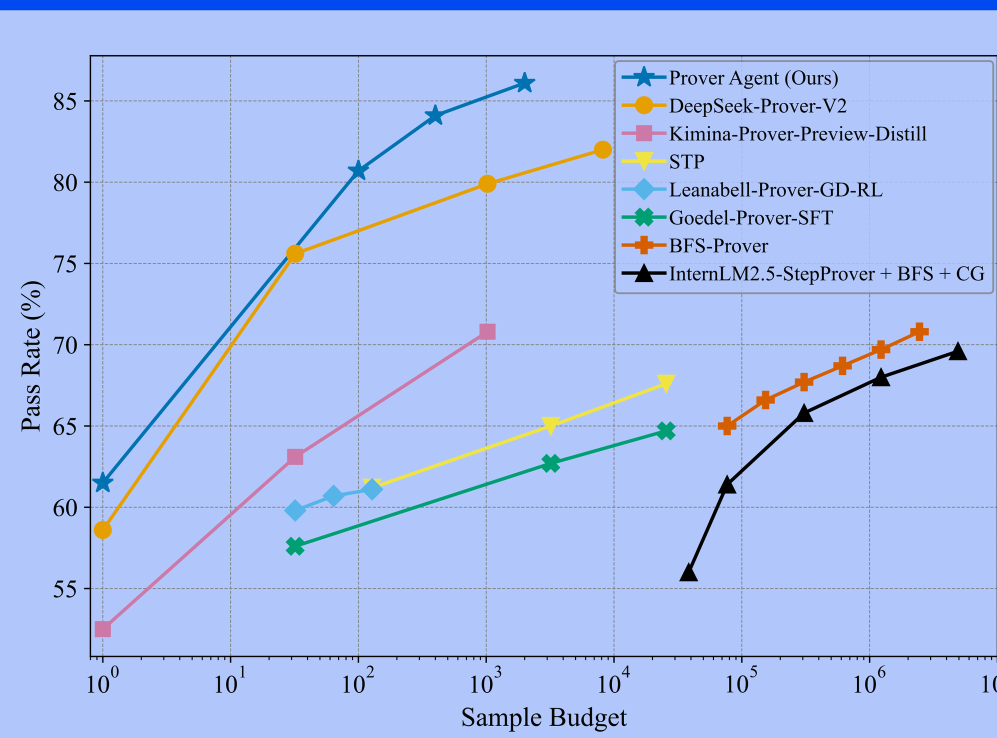
➡ LLM-based formal proving is gaining attention

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

Our goal: Bridge this gap

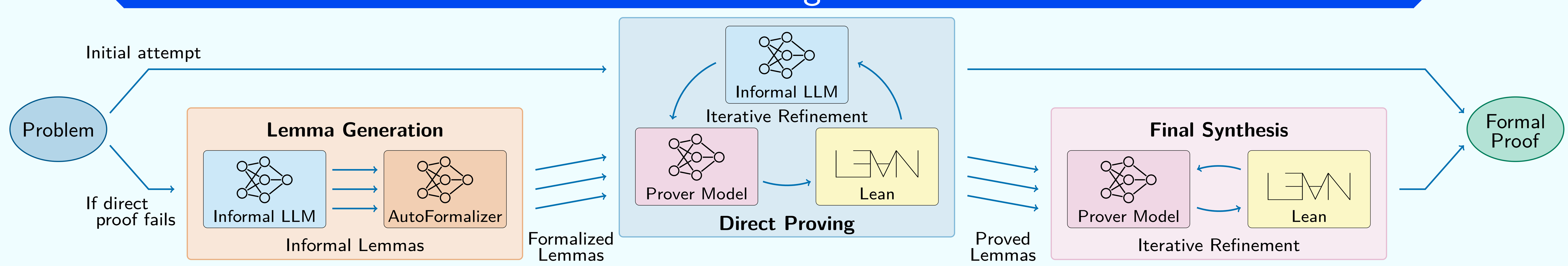
Our Contributions

- 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



Comparison of theorem-proving performance on the MiniF2F benchmark

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 approaches
- 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: `DeepSeek-Prover-V2-7B`
- AutoFormalizer: `Kimina-Autoformalizer-7B`

Comparison of Formal Theorem-Proving Performance

Prover System		Method	Model Size	Sample Budget	miniF2F test
Large language models					
Kimina-Prover-Preview (Wang et al., 2025)		Whole-proof	72B	1 32 1024 8192	52.9% 68.9% 77.9% 80.7%
DeepSeek-Prover-V2 (non-CoT) (Ren et al., 2025)		Whole-proof	671B	1 32 1024 8192	59.5% 73.8% 76.7% 78.3%
DeepSeek-Prover-V2 (CoT) (Ren et al., 2025)		Whole-proof	671B	1 32 1024 8192	61.9% 82.4% 86.6% **88.9%**
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 + BFS + 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., 2025)		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 (non-CoT) (Ren et al., 2025)		Whole-proof	7B	1 32 1024 8192	55.5% 68.0% 73.2% 75.0%
DeepSeek-Prover-V2 (CoT) (Ren et al., 2025)		Whole-proof	7B	1 32 1024 8192	58.6% 75.6% 79.9% 82.0%
Prover Agent (Ours)	(Direct proving w/o iterative refinement) (Direct proving w/o iterative refinement) (Direct proving w/ iterative refinement) (Final Proof Synthesis w/ Lemma)	Agent	8B	1 100 400 2000	61.5% 80.7% 84.0% **86.1%**

- ✓ 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 IMO	Olympiad AIME	Olympiad AMC	Olympiad Sum	MATH Algebra	MATH Number Theory	MATH Sum	Custom Algebra	Custom Number Theory	Custom Induction	Custom 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)		100	70.0	80.0	82.2	78.8	82.9	88.3	85.4	66.7	75.0	62.5	67.6
	(Direct proving w/ iterative refinement)		400	80.0	80.0	88.9	85.0	84.3	91.7	87.7	66.7	75.0	62.5	67.6
	(Final Proof Synthesis w/ Lemma)		2000	**80.0**	80.0	**91.1**	**86.3**	85.7	91.7	88.5	72.2	87.5	75.0	76.5
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
```

参考文献

Dong, K. and Ma, T. STP: Self-play llm theorem provers with iterative conjecturing and proving. *arXiv preprint arXiv:2502.00212*, 2025.

Li, Y., Du, D., Song, L., Li, C., Wang, W., Yang, T., and Mi, H. HunyuanProver: A scalable data synthesis framework and guided tree search for automated theorem proving. *arXiv preprint arXiv:2412.20735*, 2025.

Lin, Y., Tang, S., Lyu, B., Wu, J., Lin, H., Yang, K., Li, J., Xia, M., Chen, D., Arora, S., and Jin, C. Goedel-Prover: A frontier model for open-source automated theorem proving. *arXiv preprint arXiv:2502.07640*, 2025.

Ren, Z. Z., Shao, Z., Song, J., Xin, H., Wang, H., Zhao, W., Zhang, L., Fu, Z., Zhu, Q., Yang, D., Wu, Z. F., Gou, Z., Ma, S., Tang, H., Liu, Y., Gao, W., Guo, D., and Ruan, C. DeepSeek-Prover-V2: Advancing formal mathematical reasoning via reinforcement learning for subgoal decomposition. 2025.

Wang, H., Unsal, M., Lin, X., Baksys, M., Liu, J., Santos, M. D., Sung, F., Vinyes, M., Ying, Z., Zhu, Z., Lu, J., de Saxcé, H., Bailey, B., Song, C., Xiao, C., Zhang, D., Zhang, E., Pu, F., Zhu, H., Liu, J., Bayer, J., Michel, J., Yu, L., Dreyfus-Schmidt, L., Tunstall, L., Pagani, L., Machado, M., Bourigault, P., Wang, R., Polu, S., Barroyer, T., Li, W.-D., Niu, Y., Fleureau, Y., Hu, Y., Yu, Z., Wang, Z., Yang, Z., Liu, Z., and Li, J. Kimina-prover preview: Towards large formal reasoning models with reinforcement learning. *arXiv preprint arXiv:2504.11354*, 2025. doi: 10.48550/arXiv.2504.11354.

Wu, Z., Huang, S., Zhou, Z., Ying, H., Wang, J., Lin, D., and Chen, K. InternLM2.5-StepProver: Advancing automated theorem proving via expert iteration on large-scale lean problems. 2024.

Xin, H., Ren, Z. Z., Song, J., Shao, Z., Zhao, W., Wang, H., Liu, B., Zhang, L., Lu, X., Du, Q., Gao, W., Zhu, Q., Yang, D., Gou, Z., Wu, Z. F., Luo, F., and Ruan, C. DeepSeek-Prover-V1.5: Harnessing proof assistant feedback for reinforcement learning and monte-carlo tree search, 2025a.

Xin, R., Xi, C., Yang, J., Chen, F., Wu, H., Xiao, X., Sun, Y., Zheng, S., and Shen, K. BFS-Prover: Scalable best-first tree search for llm-based automatic theorem proving. 2025b.

Zhang, J., Wang, Q., Ji, X., Liu, Y., Yue, Y., Zhang, F., Zhang, D., Zhou, G., and Gai, K. Leanabell-Prover: Posttraining scaling in formal reasoning. *arXiv preprint arXiv:2504.06122*, 2025.