- **CMO 2024** (6 problems): The China Mathematical Olympiad, China's national championship
- **Putnam 2024** (12 problems): The William Lowell Putnam Competition, the preeminent mathematics competition for undergraduate students in North America
- **ISL 2024** (31 problems): The IMO Shortlist, a collection of problems proposed by participating countries and considered by the Problem Selection Committee for potential inclusion in IMO 2024
- **IMO-ProofBench** (60 problems): Developed by the DeepMind team behind DeepThink IMO-Gold (Luong and Lockhart, 2025), this benchmark (Luong et al., 2025) is divided into a basic set (30 problems, pre-IMO to IMO-Medium difficulty) and an advanced set (30 challenging problems simulating complete IMO examinations, up to IMO-Hard level)

3.3. Evaluation Results

3.3.1. One-Shot Generation

We first evaluate the model's ability to generate correct proofs without iterative refinement. On the in-house problems, we generated 8 proof samples per problem for each evaluated model. Proof correctness was measured by majority voting across 8 verification analyses produced by our final verifier. As shown in Figure 1, across all categories of CNML-level problems – algebra, geometry, number theory, combinatorics, and inequality – DeepSeekMath-V2 consistently outperforms GPT-5-Thinking-High (OpenAI, 2025) and Gemini 2.5-Pro (DeepMind, 2025), demonstrating superior theorem-proving ability across domains.

3.3.2. Sequential Refinement with Self-Verification

For challenging problems from competitions like IMO and CMO, models often cannot generate comprehensive and rigorous proofs in a single attempt within the 128K token limit. When this occurs, our proof generator recognizes its proof is invalid through self-verification but lacks the context length to resolve all identified issues in a single attempt.

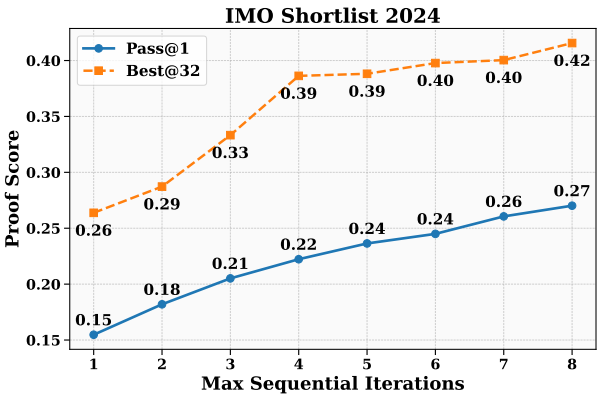

Figure 2 | Proof quality improvements as the maximum sequential iterations varies from 1 (no refinement) to 8 (initial generation plus up to 7 refinements based on self verification).

To explore how extended context and self-verification can improve proof quality, we evaluate sequential refinement with self-verification. This approach first generates a proof with self-analysis, then iteratively re-prompts the generator with its previous output (see Appendix A.4 for the refinement prompt), allowing it to address identified issues. The process continues until the generator assigns itself a perfect score or reaches the maximum number of sequential attempts.

Figure 2 demonstrates proof quality improvement through sequential refinement on IMO Shortlist 2024 problems. For each problem, we launched 32 independent refinement threads. Proof correctness was measured by majority voting across 32 verification analyses from our final verifier. We report two metrics in Figure 2: (1) Pass@1 – the average score of the final