Reinforcement Learning for Reasoning in Small LLMs: What Works and What Doesn't

Quy-Anh Dang[1,2], **Chris Ngo**[2]
[1]VNU University of Science, Vietnam
[2]Knovel Engineering Lab, Singapore
{quyanh.dang, chris.ngo}@knoveleng.com

Abstract

Enhancing the reasoning capabilities of large language models (LLMs) typically relies on massive computational resources and extensive datasets, limiting accessibility for resource-constrained settings. Our study investigates the potential of reinforcement learning (RL) to improve reasoning in small LLMs, focusing on a 1.5-billion-parameter model, `DeepSeek-R1-Distill-Qwen-1.5B`, under strict constraints: training on 4 NVIDIA A40 GPUs (48 GB VRAM each) within 24 hours. Adapting the Group Relative Policy Optimization (GRPO) algorithm and curating a compact, high-quality mathematical reasoning dataset, we conducted three experiments to explore model behavior and performance. Our results demonstrate rapid reasoning gains - e.g., AMC23 accuracy rising from 63% to 80% and AIME24 reaching 46.7%, surpassing `o1-preview` - using only 7,000 samples and a $42 training cost, compared to thousands of dollars for baseline models. However, challenges such as optimization instability and length constraints emerged with prolonged training. These findings highlight the efficacy of RL-based fine-tuning for small LLMs, offering a cost-effective alternative to large-scale approaches. We release our code and datasets as open-source resources, providing insights into trade-offs and laying a foundation for scalable, reasoning-capable LLMs in resource-limited environments. All are available at https://github.com/knoveleng/open-rs.

1 Introduction

Recent advancements in large language models (LLMs) have significantly advanced the pursuit of artificial general intelligence (AGI), with models such as GPT-4o (OpenAI 2024a), Claude 3.5 Sonnet (Anthropic 2024), and Gemini 1.5 (Google 2024) demonstrating unprecedented capabilities. A pivotal aspect of this progress is the integration of post-training techniques into the training pipeline. These methods - including supervised fine-tuning (SFT) and reinforcement learning (RL) - enhance reasoning accuracy, align models with societal values, and adapt them to user preferences, all while demanding fewer computational resources than pre-training (OpenAI 2024b). A notable innovation in this domain is OpenAI's o1 series, which leverages inference-time scaling through extended Chain-of-Thought (CoT) reasoning to achieve remarkable performance in mathematics, coding, and scientific reasoning tasks (OpenAI 2024b). However, despite these breakthroughs, scaling reasoning capabilities at test time remains a persistent challenge for the broader research community, largely due to limited access to proprietary methodologies and resources.

Efforts to bolster LLM reasoning have explored diverse strategies. Process-based reward models (Uesato et al. 2022; Lightman et al. 2023a; Wang et al. 2023) guide models toward structured problem-solving, while RL approaches (Kumar et al. 2024) optimize performance through feedback-driven learning. Search algorithms, such as Monte Carlo Tree Search (MCTS) and Beam Search, have also been employed to enhance reasoning depth (Feng et al. 2024; Xin et al. 2024; Trinh et al. 2024). Although these methods have driven incremental gains, they fall short of the general reasoning prowess exhibited by the o1 series. Recently, the DeepSeek-R1 model (DeepSeek-AI 2025) has emerged as a competitive alternative, utilizing RL with the Group Relative Policy Optimization (GRPO) algorithm. Built on the 671-billion-parameter DeepSeek-V3, DeepSeek-R1 matches o1's reasoning performance (DeepSeek-AI 2025). Yet, the sheer scale and computational demands of such models - often exceeding hundreds of billions of parameters - render them impractical for self-hosting by most organizations outside major technology firms, limiting their broader adoption.

In contrast, small LLMs, typically ranging from 1 to 10 billion parameters, present a resource-efficient alternative with potential for widespread deployment. Previous studies have demonstrated the feasibility of enhancing small LLMs through RL-based fine-tuning inspired by DeepSeek-R1 (Luo et al. 2025; Team 2025b). However, these efforts often rely on expansive datasets (hundreds of thousands to millions of samples) or incur significant computational costs, undermining their accessibility for resource-constrained settings. This tension motivates two central research questions:

1. *How do small LLMs behave when fine-tuned under strict resource constraints, such as limited computational power and training time?*
2. *Can their reasoning performance be elevated using an RL-based approach akin to DeepSeek-R1's methodology, and if so, how?*

These questions naturally extend to a practical inquiry: If viable, how should such an approach be imple-

Copyright © 2026, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

mented for small LLMs, and if not, what are the fundamental limitations? Addressing these, we investigate the reasoning capacity of a 1.5-billion-parameter model, `DeepSeek-R1-Distill-Qwen-1.5B`, under stringent constraints: training on a cluster of 4 NVIDIA A40 GPUs (48 GB VRAM each) within a 24-hour window. Our methodology adapts the GRPO-based RL framework from DeepSeek-R1, tailoring it to the resource-limited context of small LLMs. We assess performance on a suite of mathematical reasoning benchmarks, a domain requiring structured, logical problem-solving that serves as a robust testbed for reasoning ability.

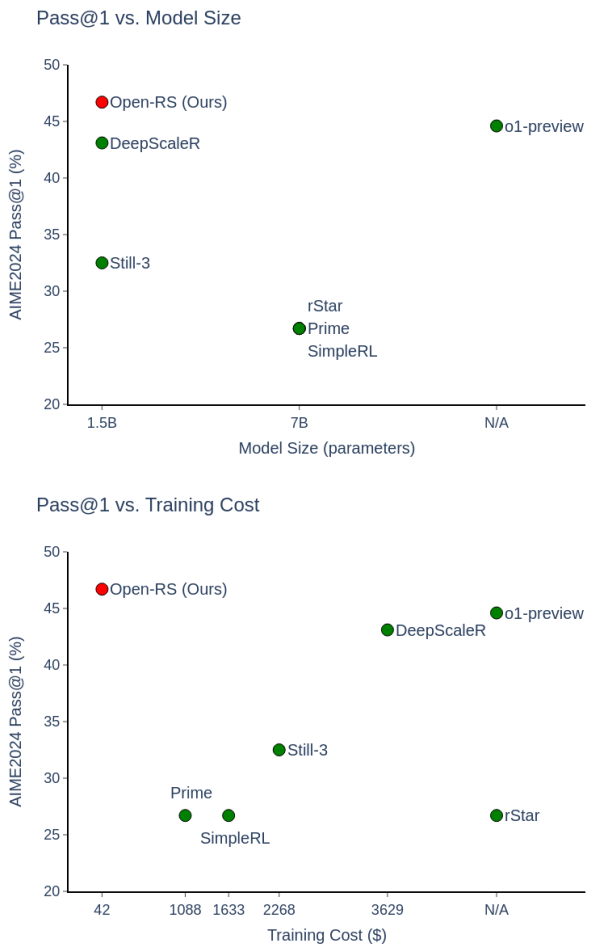

Figure 1: Comparison of zero-shot pass@1 performance versus model size (left) and computational cost (right). Our Open-RS (red point) achieves the highest AIME24 score (46.7%), outperforming `o1-preview` (44.6%) and other models (green points). Additionally, Open-RS models exhibit the lowest computational cost at approximately $42.

Our study yields three primary contributions:

1. We systematically analyze the reasoning potential of small LLMs under specific computational constraints, providing a practical lens on their scalability and deployment feasibility.
2. We offer actionable insights into the efficacy and challenges of RL-based fine-tuning for small LLMs, bridging the gap between theoretical advancements and real-world applicability.
3. We release our source code and curated datasets as open-source resources, fostering reproducibility and encouraging further exploration by the research community.

Our findings illuminate the promise of RL-based methods to enhance small LLMs' reasoning capabilities, achieving competitive performance with minimal resources (Figure 1). Simultaneously, they reveal critical challenges - such as data efficiency, optimization stability, and length constraints - that must be addressed to fully realize this potential. These insights lay the groundwork for developing lightweight, reasoning-capable LLMs suitable for resource-constrained environments, advancing the democratization of advanced AI technologies.

The remainder of this paper is structured as follows: Section 2 details our methodology, including data curation, RL algorithm, and reward design; Section 3 presents three experiments, their results, and comparative analyses; and Section 4 summarizes key findings. Additional details, including related work, discussion, hyperparameter setups, and supplementary results, are provided in the Appendix.

2 Methodology

In this section, we outline our approach to optimizing the reasoning capabilities of small large language models (LLMs) under computational constraints. Our methodology comprises two primary components: (1) the curation of a high-quality, mathematics-focused dataset, and (2) the application of a resource-efficient reinforcement learning (RL) algorithm. These components are designed to balance performance gains with practical limitations, such as reduced computational overhead and privacy considerations.

High-Quality Dataset Curation

To minimize training costs while maximizing reasoning performance, we curate a compact, high-quality dataset tailored to mathematical reasoning. This dataset is derived from two existing sources: the s1 dataset (Muennighoff et al. 2025) and the DeepScaleR dataset (DeepSeek-AI 2025). By filtering and refining these datasets, we ensure that our training data is both relevant and challenging, enabling efficient learning for small LLMs.

s1 Dataset The s1 dataset (Muennighoff et al. 2025) is a general-purpose reasoning corpus comprising 59,029 questions sourced from diverse domains, including NuminaMATH (LI et al. 2024), AIME problems (1983–2021), OlympicArena (Huang et al. 2024), OmniMath (Gao et al. 2024), AGIEval (Zhong et al. 2023), probability questions from Stanford University's Statistics Department PhD Qualifying Exams (https://statistics.stanford.edu), and brainteasers from PuzzledQuant (https://www.puzzledquant.com). Although the dataset spans multiple disciplines - such as Astronomy, Biology, Chemistry, Computer Science, Geography, Mathematics, and Physics - our focus is exclusively on mathematical reasoning.

To isolate mathematics-specific examples, we adopt a filtering workflow inspired by (Muennighoff et al. 2025). First, we retain only questions with solutions containing the LaTeX command `\boxed{}`, a common indicator of mathematical answers, reducing the dataset to 31,323 examples. Next, we employ the distilled model `DeepSeek-R1-Distill-Qwen-1.5B` to eliminate trivial questions, yielding 21,533 examples. Finally, to ensure data quality, we use `Qwen2.5-7B-Instruct` to remove noisy or multi-part questions, resulting in a final set of 18,615 high-quality mathematical reasoning examples – **open-s1** dataset.

DeepScaleR Dataset The DeepScaleR dataset (Luo et al. 2025) contains 40,315 mathematics-specific questions drawn from AIME (1984–2023), AMC (prior to 2023), Omni-MATH, and the Still dataset. Unlike the s1 dataset, DeepScaleR is pre-filtered to focus solely on mathematics, with redundant questions removed and solutions extracted from raw text using retrieval-augmented generation (RAG) and advanced LLMs like `Gemini-1.5-Pro-002`. To further refine this dataset, we apply `Qwen2.5-Math-7B-Instruct` to exclude easy questions, reducing the set to 21,044 examples – **open-deepscaler** dataset. We opt for `Qwen2.5-Math-7B-Instruct` over `DeepSeek-R1-Distill-Qwen-1.5B` - used for the s1 dataset - to introduce diversity in filtering criteria and avoid excessive overlap between the two datasets.

Final Dataset Combining the refined **open-s1** dataset (18,615 examples) and **open-deepscaler** (21,044 examples), we obtain a final high-quality dataset of 39,659 mathematical reasoning questions. This curated corpus strikes a balance between scale and specificity, enabling effective training of small LLMs under resource constraints.

Reinforcement Learning Algorithm

To train small LLMs efficiently, we adopt the Group Relative Policy Optimization (GRPO) algorithm (Shao et al. 2024), as utilized in (DeepSeek-AI 2025). GRPO eliminates the need for a separate critic model - typically as large as the policy model - by estimating baselines from group scores, thereby reducing computational overhead. For each question q, GRPO samples a group of G outputs $\{o_1, o_2, \ldots, o_G\}$ from the old policy $\pi_{\theta_{\text{old}}}$ and optimizes the policy π_θ by maximizing the following objective:

$$\begin{aligned}\mathcal{J}_{\text{GRPO}}(\theta) = \mathbb{E}_{[q\sim P(Q),\{o_i\}_{i=1}^{G}\sim\pi_{\theta_{\text{old}}}(O|q)]} \\ \frac{1}{G}\sum_{i=1}^{G}\left(\min\left(\frac{\pi_\theta(o_i|q)}{\pi_{\theta_{\text{old}}}(o_i|q)}A_i,\right.\right. \\ \left.\left.\text{clip}\left(\frac{\pi_\theta(o_i|q)}{\pi_{\theta_{\text{old}}}(o_i|q)}, 1-\epsilon, 1+\epsilon\right)A_i\right)\right) \\ -\beta\mathbb{D}_{\text{KL}}(\pi_\theta||\pi_{\text{ref}}) \qquad (1)\end{aligned}$$

where the KL-divergence term is defined as:

$$\mathbb{D}_{\text{KL}}(\pi_\theta||\pi_{\text{ref}}) = \frac{\pi_{\text{ref}}(o_i|q)}{\pi_\theta(o_i|q)} - \log\frac{\pi_{\text{ref}}(o_i|q)}{\pi_\theta(o_i|q)} - 1, \quad (2)$$

and the advantage A_i is computed from a group of rewards $\{r_1, r_2, \ldots, r_G\}$:

$$A_i = \frac{r_i - \text{mean}(\{r_1, r_2, \ldots, r_G\})}{\text{std}(\{r_1, r_2, \ldots, r_G\})}. \quad (3)$$

Here, ϵ and β are hyperparameters controlling the clipping range and KL penalty, respectively.

Reward Models The reward function is critical to guiding RL optimization. We employ a rule-based reward system comprising three components, designed to balance correctness, efficiency, and structure without relying on resource-intensive neural reward models:

- **Accuracy Reward**: This evaluates whether the model's response is correct, requiring the final answer to be presented in a `\boxed{}` format for reliable verification. A binary score (1 for correct, 0 for incorrect) ensures simplicity and objectivity.
- **Cosine Reward**: This augments the accuracy reward by scaling it based on response length using a cosine schedule. Shorter correct solutions receive higher rewards, while longer incorrect solutions are penalized less severely, incentivizing concise yet accurate reasoning.
- **Format Reward**: This enforces structural clarity by requiring the model to encapsulate its reasoning process within `<think>` and `</think>` tags, awarding a positive score for compliance.

3 Experiments

To address the research questions outlined in Section 1 - namely, how reinforcement learning (RL) can enhance the reasoning abilities of small large language models (LLMs) and what practical insights emerge under computational constraints - we design three experiments to analyze the training behavior of small LLMs. These experiments aim to provide empirical evidence of performance improvements and offer actionable guidance for future research and industrial applications.

Experimental Setup

We select `DeepSeek-R1-Distill-Qwen-1.5B` (DeepSeek-AI 2025) as our base model for training. This 1.5-billion-parameter model, distilled from larger architectures, is chosen for its balance of efficiency and reasoning potential. Notably, we bypass the supervised fine-tuning (SFT) phase - typically a precursor to RL for performance enhancement (Chu et al. 2025) - hypothesizing that the model's pretraining is sufficient to leverage RL directly. For the RL phase, we employ the Group Relative Policy Optimization (GRPO) algorithm, as detailed in Section 2, due to its computational efficiency.

Training is conducted on a cluster of 4 NVIDIA A40 GPUs (48GB VRAM each), imposing constraints that limit us to sampling 6 outputs per step with a maximum completion length of 4096 tokens. To facilitate this, we adapt `open-r1` (Face 2025), an open-source reproduction of DeepSeek-R1 by the Hugging Face team, customizing it to align with our objectives. The training phase is restricted to 1

epoch, completed within a 24-hour window, reflecting real-world resource limitations. Hyperparameters and additional configurations are detailed in Appendix E.

Benchmark Datasets

To evaluate the reasoning capabilities of our small LLM, we choose five mathematics-focused benchmark datasets: **AIME24** [1], **MATH-500** (Lightman et al. 2023b; Hendrycks et al. 2021), **AMC23** [2], **Minerva** (Lewkowycz et al. 2022a) and **OlympiadBench** (He et al. 2024). Details of the datasets are provided in Appendix C.

Baseline Models

To contextualize our results, we compare our trained model against a range of baselines: `Llama-3.1-70B-Instruct` (AI 2024a), `o1-preview` (AI 2024b), `Qwen-2.5-Math-7B-Instruct` (Yang et al. 2024), `rStar-Math-7B` (Guan et al. 2025), `Eurus-2-7B-PRIME`, `Qwen2.5-7B-SimpleRL` (Zeng et al. 2025) (Cui et al. 2025), `DeepSeek-R1-Distill-Qwen-1.5B` (DeepSeek-AI 2025), `DeepScaleR-1.5B-Preview` (Luo et al. 2025), `Still-3-1.5B-Preview` (Team 2025b).

This selection enables a robust comparison across model sizes, training methodologies, and reasoning strategies, highlighting the efficacy of our approach for small LLMs. Details of the baselines are provided in Appendix D.

Evaluation Metric

We adopt the zero-shot pass@1 metric to measure performance, defined as the proportion of problems correctly solved on the first attempt without prior examples. This metric emphasizes the model's ability to reason independently, aligning with our goal of enhancing intrinsic reasoning capabilities in small LLMs. Final answers are required in `\boxed{}` format for consistent automated evaluation.

Process and Results

In this subsection, we present three experiments designed to enhance the reasoning abilities of small LLMs using reinforcement learning (RL), follow the methodology in Section 2. We analyze training progress, evaluate performance across benchmarks, and compare our models against baselines, highlighting key insights and their implications for future work.

Experiment 1: Impact of High-Quality Data. In Experiment 1, we train the `DeepSeek-R1-Distill-Qwen-1.5B` model using the **open-s1** dataset (18,615 samples) from Section 2, with a maximum completion length of 4096 tokens. We employ accuracy and format rewards, as described in Section 2. Although the full dataset corresponds to approximately 1500 global steps for one epoch, computational constraints

[1]https://huggingface.co/datasets/AI-MO/aimo-validation-aime
[2]https://huggingface.co/datasets/AI-MO/aimo-validation-amc

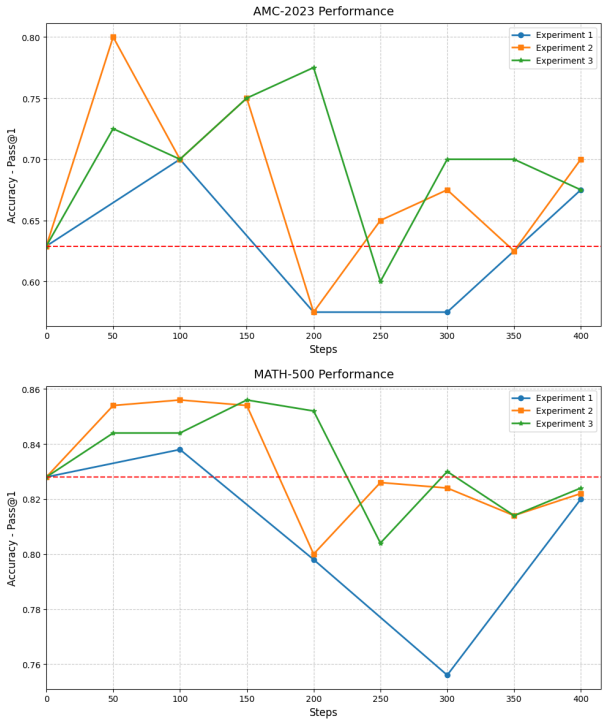

Figure 2: Performance of the model on **AMC23** (left) and **MATH-500** (right) across global training steps. The red dashed line indicates the baseline score at the start of training.

(24-hour limit on 4x A40 GPUs) restrict training to 500 global steps.

Performance on AMC23 improves from 63% to 70% and on MATH-500 from 83% to 84% within the first 50–100 steps (see Figure 2). However, after 200 steps, accuracy degrades significantly, dropping below 60% on AMC23 and to 80% on MATH-500. Figure 3 illustrates this trend, showing unstable accuracy rewards and completion lengths fluctuating near 4000 tokens initially, then decreasing to around 3000 tokens by 100 global steps (approximately 3000 local steps on a single GPU). Post-200 steps, lengths increase again, accompanied by unreadable content and non-English outputs.

This degradation suggests that the model struggles with the complexity of **open-s1**, often exceeding the 4096-token limit before producing a final answer. The initial length reduction reflects adaptation to the format reward, but the subsequent increase and language drift indicate reward misalignment. We derive the following insight:

Insight 1

Small LLMs can achieve rapid reasoning improvements with limited high-quality data within 50–100 steps, but performance degrades with prolonged training under strict length constraints.

Experiment 2: Balancing Easy and Hard Problems. Building on Experiment 1, we hypothesize that mixing easier

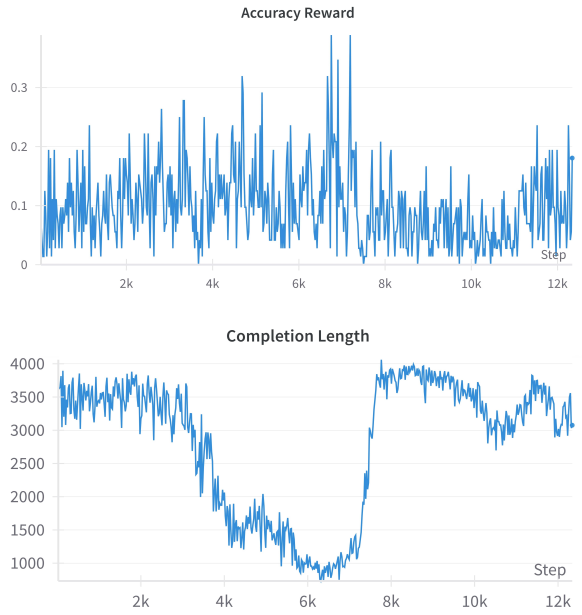

Figure 3: Accuracy reward (left) and completion length (right) of outputs in Experiment 1 across local steps. Note that global steps are distributed across 4 GPUs, with 100 global steps approximating 3000 local steps.

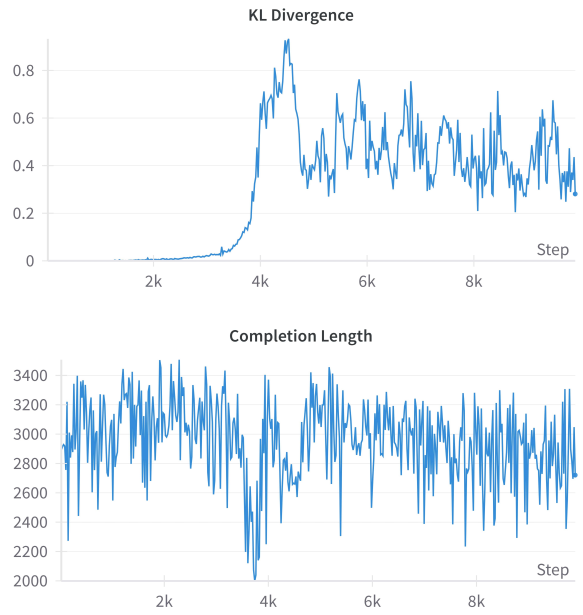

Figure 4: KL divergence (left) and completion length (right) of outputs in Experiment 2 across local steps.

problems with challenging ones could stabilize training and reduce completion lengths. We construct a dataset of 7000 samples: 3000 from **open-s1**, 3000 from **open-deepscaler**, and 1000 easier problems from the raw **DeepScaleR** dataset (Section 2). The maximum completion length is reduced to 3584 tokens, retaining accuracy and format rewards.

Initial completion lengths drop to approximately 2800 tokens, and performance improves significantly: AMC23 rises from 63% to 80%, and MATH-500 from 83% to 85% within 50–100 steps (Figure 2). However, after 150–200 steps (approximately 4000 local steps), performance declines, and KL divergence becomes unstable (Figure 4), with mixed-language outputs reemerging.

The improved initial performance validates our hypothesis, suggesting that easier problems encourage concise reasoning, while harder ones maintain complexity. However, the late-stage instability highlights persistent challenges with length constraints and multilingual tendencies. We note:

Insight 2

Incorporating a mix of easy and hard problems under reduced length constraints enhances early performance and stabilizes reasoning behavior, though long-term stability remains elusive.

Experiment 3: Controlling Length with Cosine Reward. Experiment 3 uses the same 7000-sample dataset as Experiment 2, but replaces the accuracy reward with a cosine reward to better control output length, as outlined in Section 2. We also append an instruction to the system prompt: *"Reply in English only, do not use other languages"*, avoiding a computationally expensive language reward function. The maximum completion length remains 3584 tokens.

Completion lengths stabilize between 1000 and 3500 tokens (Figure 5), a marked improvement over Experiment 2's 2000–3500 range. Performance on AMC23 and MATH-500 increases modestly compared to the baseline (63% to 72.5% and 83% to 84.4%, respectively) within 50 steps, though it lags behind Experiment 2's peak (Figure 2). After 200 steps, mixed-language content persists, reflecting the multilingual nature of `DeepSeek-R1-Distill-Qwen-1.5B`.

The cosine reward effectively regulates length, but the language issue suggests a need for explicit language constraints or extended completion lengths for complex tasks. We conclude:

Insight 3

Cosine rewards stabilize completion lengths, improving training consistency, but extending length limits is necessary for extremely hard tasks, particularly with multilingual base models.

Overall Comparison. We select checkpoints at 100, 50, and 50 global steps from Experiments 1, 2, and 3, naming them **Open-RS1**, **Open-RS2**, and **Open-RS3** (R for **Reasoning**, S for **Small**), respectively. These are evaluated against baselines from Section 3 across benchmarks from Section 3, using zero-shot pass@1 (Table 1).

Our models outperform most baselines, with average scores of 53.0% (Open-RS1), 55.7% (Open-

Model	AIME24	MATH-500	AMC23	Minerva	OlympiadBench	Avg.
General Models						
Llama-3.1-70B-Instruct	16.7	64.6	30.1	35.3	31.9	35.7
o1-preview	44.6	85.5	–	–	–	–
7B Models						
Qwen-2.5-Math-7B-Instruct	13.3	79.8	50.6	34.6	40.7	43.8
rStar-Math-7B	26.7	78.4	47.5	–	47.1	–
Eurus-2-7B-PRIME	26.7	79.2	57.8	38.6	42.1	48.9
Qwen2.5-7B-SimpleRL	26.7	82.4	62.5	**39.7**	43.3	50.9
1.5B Models						
DeepSeek-R1-Distill-Qwen-1.5B	28.8	82.8	62.9	26.5	43.3	48.9
Still-3-1.5B-Preview	32.5	84.4	66.7	29.0	45.4	51.6
DeepScaleR-1.5B-Preview	43.1	**87.8**	73.6	30.2	50.0	**57.0**
Our Models						
Open-RS1 (100 steps)	30.0	83.8	70.0	29.0	**52.4**	53.0
Open-RS2 (50 steps)	30.0	85.4	**80.0**	30.5	**52.4**	55.7
Open-RS3 (50 steps)	**46.7**	84.4	72.5	26.8	51.3	56.3

Table 1: Zero-shot pass@1 performance across benchmarks. **Bold** indicates the highest score per benchmark. Dashes (–) denote unavailable official scores. Scores for `o1-preview` are sourced from (AI 2024b); others from (Zeng et al. 2025; Luo et al. 2025). Our models are evaluated using the `lighteval` package (Fourrier et al. 2023).

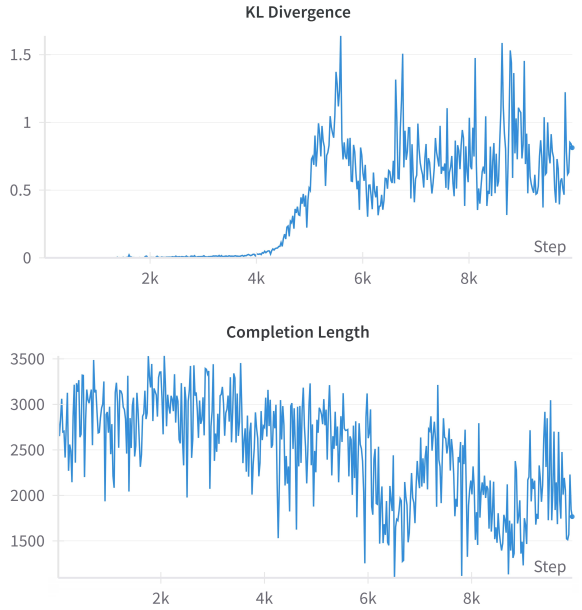

Figure 5: KL divergence (left) and completion length (right) of outputs in Experiment 3 across local steps.

RS2), and 56.3% (Open-RS3), compared to 57.0% for `DeepScaleR-1.5B-Preview`. Notably, Open-RS3 achieves the highest AIME24 score (46.7%), surpassing `o1-preview` (44.6%) and `DeepScaleR-1.5B-Preview` (43.1%). Open-RS2 excels on AMC23 (80.0%) and ties with Open-RS1 on OlympiadBench (52.4%), both outperforming `DeepScaleR-1.5B-Preview`. MATH-500 scores remain competitive, though Minerva performance lags behind 7B models, reflecting the complexity of cross-disciplinary reasoning.

We further compare training costs[3] and data efficiency (Tables 2 and 3, and Figure 1). Our approach, using 7000 samples with 6 outputs per step (42,000 total samples), costs approximately $42 on 4x A40 GPUs over 24 hours. In contrast, 7B models like `Qwen2.5-7B-SimpleRL` ($1633) and `Eurus-2-7B-PRIME` ($1088) and 1.5B models like `DeepScaleR-1.5B-Preview` ($3629) and `Still-3-1.5B-Preview` ($2268) require significantly more resources and data (e.g., 40k × 16 samples for `DeepScaleR`).

Our approach demonstrates that small LLMs can achieve competitive reasoning performance with minimal data and cost, offering a scalable alternative to resource-intensive baselines.

4 Conclusion

Our study investigated enhancing the reasoning abilities of small LLMs using RL, focusing on the 1.5-billion-parameter

[3]Cost estimates are calculated based on pricing from https://www.runpod.io/pricing.

	rStar-Math-7B	Eurus-2-7B-PRIME	Qwen2.5-7B-SimpleRL	Open-RS
Base Model	Qwen2.5-Math-7B	Qwen2.5-Math-7B	Qwen2.5-Math-7B	DeepSeek-R1-Distill-Qwen-1.5B
SFT Data	7.3M	230k	0	0
RM Data	7k	0	0	0
RM	None	Eurus-2-7B-SFT	None	None
RL Data	3.647M × 16	150k × 4	8k × 8	7k × 6
Hardware	10x 8 H100 80GB, 15x 4 A100 40GB	1x 8 A100 80GB	4x 6 A100 80GB	1x 4 A40 48GB
Time	–	72h	36h	24h
Cost Est.	–	$1088	$1633	$42

Table 2: Comparison of data usage and training costs for 7B models. Data are sourced from original papers or GitHub issues addressing author's resource constraints.

	DeepScaleR-1.5B-Preview	Still-3-1.5B-Preview	Open-RS
Base Model	DeepSeek-R1-Distill-Qwen-1.5B	DeepSeek-R1-Distill-Qwen-1.5B	DeepSeek-R1-Distill-Qwen-1.5B
SFT Data	0	0	0
RM Data	0	0	0
RM	None	None	None
RL Data	40k × 16	30k × 8	7k × 6
Hardware	8x A100 80GB	1x 8 A100 80GB	1x 4 A40 48GB
Time	240h	150h	24h
Cost Est.	$3629	$2268	$42

Table 3: Comparison of data usage and training costs for 1.5B models. Data are sourced from original papers or GitHub issues addressing author's resource constraints.

`DeepSeek-R1-Distill-Qwen-1.5B` under strict constraints. Adapting the GRPO algorithm and a compact mathematical reasoning dataset, we conducted three experiments to assess behavior and performance under resource limitations. Our findings show small LLMs can achieve significant reasoning gains with minimal resources - e.g., AMC23 accuracy rising from 63% to 80% and AIME24 reaching 46.7%, surpassing `o1-preview` - at a cost of $42 versus thousands for baselines. Open-RS variants averaged 53.0%–56.3% on benchmarks, demonstrating RL's viability for small LLMs. Releasing our code and datasets, we provide a framework for lightweight, reasoning-capable models, despite challenges like optimization stability, laying a foundation for future work.

References

AI, M. 2024a. Introducing Llama 3.1: Our most capable models to date. Published on July 23, 2024.

AI, O. 2024b. Introducing OpenAI o1-preview. Published on Dec 12, 2024.

Anthropic. 2024. Claude 3.5 Sonnet.

Chu, T.; Zhai, Y.; Yang, J.; Tong, S.; Xie, S.; Schuurmans, D.; Le, Q. V.; Levine, S.; and Ma, Y. 2025. SFT Memorizes, RL Generalizes: A Comparative Study of Foundation Model Post-training. arXiv:2501.17161.

Cobbe, K.; Kosaraju, V.; Bavarian, M.; Chen, M.; Jun, H.; Kaiser, L.; Plappert, M.; Tworek, J.; Hilton, J.; Nakano, R.; Hesse, C.; and Schulman, J. 2021. Training Verifiers to Solve Math Word Problems. arXiv:2110.14168.

Cui, G.; Yuan, L.; Wang, Z.; Wang, H.; Li, W.; He, B.; Fan, Y.; Yu, T.; Xu, Q.; Chen, W.; Yuan, J.; Chen, H.; Zhang, K.; Lv, X.; Wang, S.; Yao, Y.; Han, X.; Peng, H.; Cheng, Y.; Liu, Z.; Sun, M.; Zhou, B.; and Ding, N. 2025. Process Reinforcement through Implicit Rewards. arXiv:2502.01456.

DeepSeek-AI. 2025. DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning. arXiv:2501.12948.

Face, H. 2025. Open R1: A fully open reproduction of DeepSeek-R1.

Feng, X.; Wan, Z.; Wen, M.; McAleer, S. M.; Wen, Y.; Zhang, W.; and Wang, J. 2024. Alphazero-like Tree-Search can Guide Large Language Model Decoding and Training. arXiv:2309.17179.

Fourrier, C.; Habib, N.; Kydlíček, H.; Wolf, T.; and Tunstall, L. 2023. LightEval: A lightweight framework for LLM evaluation.

Gao, B.; Song, F.; Yang, Z.; Cai, Z.; Miao, Y.; Dong, Q.; Li, L.; Ma, C.; Chen, L.; Xu, R.; Tang, Z.; Wang, B.; Zan, D.; Quan, S.; Zhang, G.; Sha, L.; Zhang, Y.; Ren, X.; Liu, T.; and Chang, B. 2024. Omni-MATH: A Universal Olympiad Level Mathematic Benchmark For Large Language Models. arXiv:2410.07985.

Google. 2024. Our next-generation model: Gemini 1.5.

Guan, X.; Zhang, L. L.; Liu, Y.; Shang, N.; Sun, Y.; Zhu, Y.; Yang, F.; and Yang, M. 2025. rStar-Math: Small LLMs Can Master Math Reasoning with Self-Evolved Deep Thinking. arXiv:2501.04519.

He, C.; Luo, R.; Bai, Y.; Hu, S.; Thai, Z.; Shen, J.; Hu, J.; Han, X.; Huang, Y.; Zhang, Y.; Liu, J.; Qi, L.; Liu, Z.; and Sun, M. 2024. OlympiadBench: A Challenging Benchmark for Promoting AGI with Olympiad-Level Bilingual Multimodal Scientific Problems. In Ku, L.-W.; Martins, A.; and Srikumar, V., eds., *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 3828–3850. Bangkok, Thailand: Association for Computational Linguistics.

Hendrycks, D.; Burns, C.; Kadavath, S.; Arora, A.; Basart, S.; Tang, E.; Song, D.; and Steinhardt, J. 2021. Measuring Mathematical Problem Solving With the MATH Dataset. *NeurIPS*.

Huang, Z.; Wang, Z.; Xia, S.; Li, X.; Zou, H.; Xu, R.; Fan, R.-Z.; Ye, L.; Chern, E.; Ye, Y.; Zhang, Y.; Yang, Y.; Wu, T.; Wang, B.; Sun, S.; Xiao, Y.; Li, Y.; Zhou, F.; Chern, S.; Qin, Y.; Ma, Y.; Su, J.; Liu, Y.; Zheng, Y.; Zhang, S.; Lin, D.; Qiao, Y.; and Liu, P. 2024. OlympicArena: Benchmarking Multi-discipline Cognitive Reasoning for Superintelligent AI. arXiv:2406.12753.

Kojima, T.; Gu, S. S.; Reid, M.; Matsuo, Y.; and Iwasawa, Y. 2022. Large Language Models are Zero-Shot Reasoners. In Koyejo, S.; Mohamed, S.; Agarwal, A.; Belgrave, D.; Cho, K.; and Oh, A., eds., *Advances in Neural Information Processing Systems*, volume 35, 22199–22213. Curran Associates, Inc.

Kumar, A.; Zhuang, V.; Agarwal, R.; Su, Y.; Co-Reyes, J. D.; Singh, A.; Baumli, K.; Iqbal, S.; Bishop, C.; Roelofs, R.; et al. 2024. Training language models to self-correct via reinforcement learning. *arXiv preprint arXiv:2409.12917*.

Lewkowycz, A.; Andreassen, A. J.; Dohan, D.; Dyer, E.; Michalewski, H.; Ramasesh, V. V.; Slone, A.; Anil, C.; Schlag, I.; Gutman-Solo, T.; Wu, Y.; Neyshabur, B.; Gur-Ari, G.; and Misra, V. 2022a. Solving Quantitative Reasoning Problems with Language Models. In Oh, A. H.; Agarwal, A.; Belgrave, D.; and Cho, K., eds., *Advances in Neural Information Processing Systems*.

Lewkowycz, A.; Andreassen, A. J.; Dohan, D.; Dyer, E.; Michalewski, H.; Ramasesh, V. V.; Slone, A.; Anil, C.; Schlag, I.; Gutman-Solo, T.; Wu, Y.; Neyshabur, B.; Gur-Ari, G.; and Misra, V. 2022b. Solving Quantitative Reasoning Problems with Language Models. In Oh, A. H.; Agarwal, A.; Belgrave, D.; and Cho, K., eds., *Advances in Neural Information Processing Systems*.

LI, J.; Beeching, E.; Tunstall, L.; Lipkin, B.; Soletskyi, R.; Huang, S. C.; Rasul, K.; Yu, L.; Jiang, A.; Shen, Z.; Qin, Z.; Dong, B.; Zhou, L.; Fleureau, Y.; Lample, G.; and Polu, S. 2024. NuminaMath.

Lightman, H.; Kosaraju, V.; Burda, Y.; Edwards, H.; Baker, B.; Lee, T.; Leike, J.; Schulman, J.; Sutskever, I.; and Cobbe, K. 2023a. Let's verify step by step. *arXiv preprint arXiv:2305.20050*.

Lightman, H.; Kosaraju, V.; Burda, Y.; Edwards, H.; Baker, B.; Lee, T.; Leike, J.; Schulman, J.; Sutskever, I.; and Cobbe, K. 2023b. Let's Verify Step by Step. *arXiv preprint arXiv:2305.20050*.

Luo, M.; Tan, S.; Wong, J.; Shi, X.; Tang, W. Y.; Roongta, M.; Cai, C.; Luo, J.; Zhang, T.; Li, L. E.; Popa, R. A.; and Stoica, I. 2025. DeepScaleR: Surpassing O1-Preview with a 1.5B Model by Scaling RL. https://github.com/agentica-project/deepscaler. Github.

Madaan, A.; Tandon, N.; Gupta, P.; Hallinan, S.; Gao, L.; Wiegreffe, S.; Alon, U.; Dziri, N.; Prabhumoye, S.; Yang, Y.; Gupta, S.; Majumder, B. P.; Hermann, K.; Welleck, S.; Yazdanbakhsh, A.; and Clark, P. 2023. Self-Refine: Iterative Refinement with Self-Feedback. In Oh, A.; Naumann, T.; Globerson, A.; Saenko, K.; Hardt, M.; and Levine, S., eds., *Advances in Neural Information Processing Systems*, volume 36, 46534–46594. Curran Associates, Inc.

Muennighoff, N.; Yang, Z.; Shi, W.; Li, X. L.; Fei-Fei, L.; Hajishirzi, H.; Zettlemoyer, L.; Liang, P.; Candès, E.; and Hashimoto, T. 2025. s1: Simple test-time scaling. arXiv:2501.19393.

Nye, M.; Andreassen, A. J.; Gur-Ari, G.; Michalewski, H.; Austin, J.; Bieber, D.; Dohan, D.; Lewkowycz, A.; Bosma, M.; Luan, D.; Sutton, C.; and Odena, A. 2021. Show Your Work: Scratchpads for Intermediate Computation with Language Models. arXiv:2112.00114.

OpenAI. 2024a. Hello GPT-4o.

OpenAI. 2024b. Learning to reason with LLMs.

Rajani, N. F.; McCann, B.; Xiong, C.; and Socher, R. 2019. Explain Yourself! Leveraging Language Models for Commonsense Reasoning. In Korhonen, A.; Traum, D.; and Màrquez, L., eds., *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 4932–4942. Florence, Italy: Association for Computational Linguistics.

Reynolds, L.; and McDonell, K. 2021. Prompt Programming for Large Language Models: Beyond the Few-Shot Paradigm. In *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems*, CHI EA '21. New York, NY, USA: Association for Computing Machinery. ISBN 9781450380959.

Shao, Z.; Wang, P.; Zhu, Q.; Xu, R.; Song, J.; Bi, X.; Zhang, H.; Zhang, M.; Li, Y. K.; Wu, Y.; and Guo, D. 2024. DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models. arXiv:2402.03300.

Shinn, N.; Cassano, F.; Berman, E.; Gopinath, A.; Narasimhan, K.; and Yao, S. 2023. Reflexion: Language Agents with Verbal Reinforcement Learning. arXiv:2303.11366.

Team, K. 2025a. Kimi k1.5: Scaling Reinforcement Learning with LLMs. arXiv:2501.12599.

Team, R. S. 2025b. STILL-3-1.5B-preview: Enhancing Slow Thinking Abilities of Small Models through Reinforcement Learning.

Trinh, T.; Wu, Y.; Le, Q.; He, H.; and Luong, T. 2024. Solving Olympiad Geometry without Human Demonstrations. *Nature*.

Uesato, J.; Kushman, N.; Kumar, R.; Song, F.; Siegel, N.; Wang, L.; Creswell, A.; Irving, G.; and Higgins, I. 2022. Solving math word problems with process-and outcome-based feedback. *arXiv preprint arXiv:2211.14275*.

Wang, P.; Li, L.; Shao, Z.; Xu, R.; Dai, D.; Li, Y.; Chen, D.; Wu, Y.; and Sui, Z. 2023. Math-Shepherd: A Label-Free Step-by-Step Verifier for LLMs in Mathematical Reasoning. *arXiv preprint arXiv:2312.08935*.

Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; ichter, b.; Xia, F.; Chi, E.; Le, Q. V.; and Zhou, D. 2022. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. In Koyejo, S.; Mohamed, S.; Agarwal, A.; Belgrave, D.; Cho, K.; and Oh, A., eds., *Advances in Neural Information Processing Systems*, volume 35, 24824–24837. Curran Associates, Inc.

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. 2024. DeepSeek-Prover-V1.5: Harnessing Proof Assistant Feedback for Reinforcement Learning and Monte-Carlo Tree Search. arXiv:2408.08152.

Yang, A.; Zhang, B.; Hui, B.; Gao, B.; Yu, B.; Li, C.; Liu, D.; Tu, J.; Zhou, J.; Lin, J.; Lu, K.; Xue, M.; Lin, R.; Liu, T.; Ren, X.; and Zhang, Z. 2024. Qwen2.5-Math Technical Report: Toward Mathematical Expert Model via Self-Improvement. arXiv:2409.12122.

Ye, Y.; Huang, Z.; Xiao, Y.; Chern, E.; Xia, S.; and Liu, P. 2025. LIMO: Less is More for Reasoning. arXiv:2502.03387.

Yeo, E.; Tong, Y.; Niu, X.; Neubig, G.; and Yue, X. 2025. Demystifying Long Chain-of-Thought Reasoning in LLMs. In *Scaling Self-Improving Foundation Models without Human Supervision*.

Zelikman, E.; Wu, Y.; Mu, J.; and Goodman, N. 2022. STaR: Bootstrapping Reasoning With Reasoning. In Oh, A. H.; Agarwal, A.; Belgrave, D.; and Cho, K., eds., *Advances in Neural Information Processing Systems*.

Zeng, W.; Huang, Y.; Liu, W.; He, K.; Liu, Q.; Ma, Z.; and He, J. 2025. 7B Model and 8K Examples: Emerging Reasoning with Reinforcement Learning is Both Effective and Efficient. https://hkust-nlp.notion.site/simplerl-reason. Notion Blog.

Zhong, W.; Cui, R.; Guo, Y.; Liang, Y.; Lu, S.; Wang, Y.; Saied, A.; Chen, W.; and Duan, N. 2023. AGIEval: A Human-Centric Benchmark for Evaluating Foundation Models. arXiv:2304.06364.

Contents

A Related Work

Reasoning in Large Language Models

A substantial body of research has investigated methods to enhance the reasoning capabilities and factual accuracy of large language models (LLMs). Early approaches predominantly relied on prompting techniques to elicit structured reasoning. For instance, scratchpad-style prompting encourages models to break down problems into intermediate steps (Nye et al. 2021), while verification mechanisms assess the correctness of generated outputs (Cobbe et al. 2021). Chain-of-thought (CoT) prompting has emerged as a particularly effective strategy, leveraging demonstrations of step-by-step reasoning to improve performance on complex tasks (Wei et al. 2022; Kojima et al. 2022; Reynolds and McDonell 2021). More recently, techniques such as intermediate self-reflection have been proposed to enable models to iteratively refine their reasoning processes (Shinn et al. 2023; Madaan et al. 2023).

In parallel, supervised fine-tuning (SFT) has been employed to embed reasoning capabilities directly into LLMs. Studies such as (Lewkowycz et al. 2022b) and (Rajani et al. 2019) demonstrate that fine-tuning on high-quality datasets can enhance problem-solving abilities. Notably, integrating CoT reasoning into SFT has shown significant promise; works like (Zelikman et al. 2022; Muennighoff et al. 2025; Ye et al. 2025) illustrate that fine-tuning with small, carefully curated datasets of CoT examples can yield substantial performance gains. However, these efforts have predominantly focused on large-scale LLMs, typically ranging from 7 billion to over 100 billion parameters. This reliance on massive models limits accessibility and practicality for resource-constrained settings, motivating the exploration of alternative approaches for smaller LLMs.

Reasoning with Reinforcement Learning

Reinforcement learning (RL) has emerged as a powerful paradigm for improving reasoning in LLMs, particularly for tackling complex, multi-step problems. Unlike SFT, which often optimizes for imitation of training data, RL enables models to learn from feedback, enhancing generalization to both in-distribution and out-of-distribution tasks (Chu et al. 2025; Yeo et al. 2025). Recent advancements underscore the efficacy of RL in this domain. For example, (OpenAI 2024b) and (DeepSeek-AI 2025) demonstrate that RL-based training can significantly boost reasoning performance, while (Team 2025a) explores scaling laws for RL-driven LLMs. These studies highlight RL's ability to refine decision-making processes by optimizing for task-specific rewards, such as correctness or logical coherence.

Despite these advances, RL-based methods are not without limitations. They typically demand substantial computational resources, often exceeding those required for SFT, and are predominantly applied to large LLMs. This focus on scale renders RL impractical for smaller models and restricts its adoption outside well-resourced organizations, such as major technology firms. Furthermore, privacy concerns arise when deploying such models, as self-hosting becomes infeasible for most academic or industrial entities with limited infrastructure. Consequently, there remains a critical gap in the literature: the application of RL to enhance reasoning in small LLMs under resource and privacy constraints.

B Limitations & Discussion

While our study demonstrates the promise of RL-based fine-tuning for enhancing the reasoning abilities of small LLMs, several limitations and broader implications warrant discussion. These insights not only contextualize our findings but also highlight avenues for future research.

Limitations

First, our experiments were constrained by a 24-hour training window on a modest cluster of 4 NVIDIA A40 GPUs (48 GB VRAM each), limiting the number of global steps (e.g., 500 in Experiment 1 versus a potential 1500 for one epoch). This restriction curtailed our ability to fully explore the long-term behavior of the model, particularly beyond 200 steps, where performance degradation and multilingual outputs emerged. Second, the maximum completion length (4096 tokens in Experiment 1, reduced to 3584 in Experiments 2 and 3) proved insufficient for extremely hard problems in the **open-s1** dataset, forcing the model to truncate reasoning processes prematurely. This suggests that our methodology may underexploit the potential of small LLMs on complex tasks requiring extended reasoning chains.

Third, the multilingual nature of the base model, `DeepSeek-R1-Distill-Qwen-1.5B`, introduced unintended language drift after 150–200 steps, despite efforts to enforce English-only outputs via prompts in Experiment 3. This limitation reflects a trade-off in using a pre-trained, multilingual foundation, which, while efficient, complicates monolingual optimization. Finally, our evaluation focused exclusively on mathematical reasoning benchmarks, leaving the generalizability of our approach to other domains - such as scientific reasoning or coding - unexplored. These constraints highlight the need for cautious interpretation of our results within the specified scope.

Discussion

Our findings reveal a nuanced trade-off between efficiency and reasoning depth in small LLMs. The rapid performance gains observed in the first 50–100 steps across all experiments (Insight 1) suggest that small, high-quality datasets can effectively bootstrap reasoning capabilities, aligning with prior work on data efficiency in RL (Chu et al. 2025). However, the subsequent degradation underscores a sensitivity to over-optimization under fixed length constraints, a challenge also noted in larger models like DeepSeek-R1 (DeepSeek-AI 2025). Experiment 2's success with mixed difficulty levels (Insight 2) indicates that curriculum-like strategies could mitigate this. Meanwhile, the cosine reward's stabilizing effect in Experiment 3 (Insight 3) suggests a promising direction for controlling reasoning verbosity, though it sacrifices peak accuracy compared to Experiment 2.

Comparatively, our Open-RS variants achieved performance rivaling or exceeding state-of-the-art 1.5B models (e.g., `DeepScaleR-1.5B-Preview`) and even some 7B

models, at a fraction of the cost and data volume. This efficiency challenges the prevailing reliance on massive datasets and computational resources in reasoning enhancement (OpenAI 2024b; Luo et al. 2025), offering a scalable alternative for resource-constrained environments. However, the persistent multilingual drift and length limitations point to inherent challenges in adapting multilingual base models and optimizing for complex tasks within tight bounds.

Future Directions

These limitations suggest several research avenues. First, extending training duration or employing multi-stage length schedules could address truncation issues, allowing the model to handle harder problems without compromising stability. Second, incorporating a lightweight language reward or monolingual pre-filtering of the base model might mitigate language drift, enhancing output consistency. Third, expanding the benchmark suite to include non-mathematical domains would test the generalizability of our approach, aligning with broader AGI goals. Finally, exploring hybrid methods - such as combining GRPO with search algorithms like MCTS (Feng et al. 2024) - could further deepen reasoning capacity without significantly increasing resource demands.

In conclusion, our work demonstrates that RL-based fine-tuning can unlock substantial reasoning potential in small LLMs, even under stringent constraints. By identifying key trade-offs and offering practical insights, we pave the way for developing efficient, reasoning-capable models that balance performance and accessibility - a critical step toward democratizing advanced AI technologies.

C Datasets

Detail datasets are used in Section 3.

- **AIME24** [4]: 30 problems from the 2024 American Invitational Mathematics Examination, emphasizing advanced high-school-level reasoning.
- **AMC23** [5]: 40 problems from the 2023 American Mathematics Competition, testing foundational mathematical skills.
- **MATH-500** (Lightman et al. 2023b; Hendrycks et al. 2021): A subset of 500 problems from the MATH benchmark, sourced from various mathematics competitions and spanning algebra, calculus, and geometry.
- **Minerva** (Lewkowycz et al. 2022a): 272 undergraduate-level problems across physics, biology, chemistry, economics, and other sciences, requiring quantitative reasoning (mathematics subset used).
- **OlympiadBench** (He et al. 2024): 675 Olympiad-level problems in mathematics and physics, designed to challenge advanced reasoning abilities.

Table 4 summarizes the datasets and their sample sizes. This diverse collection ensures a comprehensive assessment of the model's reasoning generalization across problem types and difficulty levels.

[4]https://huggingface.co/datasets/AI-MO/aimo-validation-aime
[5]https://huggingface.co/datasets/AI-MO/aimo-validation-amc

Table 4: Benchmark Datasets and Sample Sizes for Evaluation

Dataset	# Samples
AIME24	30
MATH-500	500
AMC23	40
Minerva	272
OlympiadBench	675

D Baseline Models

The description of baseline models in Section 3.

- **General-Purpose Large Models**:
  - `Llama-3.1-70B-Instruct` (AI 2024a): A 70B-parameter model optimized for instruction-following.
  - `o1-preview` (AI 2024b): A high-performing reasoning model from OpenAI.
- **Mathematics-Focused 7B Models**:
  - `Qwen-2.5-Math-7B-Instruct` (Yang et al. 2024): An RL-trained model with a reward model for mathematical reasoning.
  - `rStar-Math-7B` (Guan et al. 2025): Uses Monte Carlo Tree Search (MCTS) for deep reasoning and self-evolution.
  - `Eurus-2-7B-PRIME` (Cui et al. 2025): Employs the PRIME method with online RL and process rewards.
  - `Qwen2.5-7B-SimpleRL` (Zeng et al. 2025): Applies Proximal Policy Optimization (PPO) with rewards based on final answers.
- **Mathematics-Focused 1.5B Models**:
  - `DeepSeek-R1-Distill-Qwen-1.5B` (DeepSeek-AI 2025): The original untrained baseline.
  - `DeepScaleR-1.5B-Preview` (Luo et al. 2025): Fine-tuned with GRPO on 40,000 math problem-answer pairs across multiple RL stages.
  - `Still-3-1.5B-Preview` (Team 2025b): RL-trained with a focus on slow thinking (e.g., tree search) using 30,000 curated math examples.

E Hyperparameter Setup

Table 5 show parameters that used in training phase.

Table 5: Hyperparameter Setups for GRPO Trainer

Parameter	**Value**
General Settings	
bf16	true
use_vllm	true
vllm_device	auto
vllm_enforce_eager	true
vllm_gpu_memory_utilization	0.7
vllm_max_model_len	4608
do_eval	false
output_dir	data/OpenRS-GRPO
overwrite_output_dir	true
Training Configuration	
gradient_accumulation_steps	4
gradient_checkpointing	true
gradient_checkpointing_kwargs	use_reentrant: false
learning_rate	1.0e-06
lr_scheduler_type	cosine_with_min_lr
lr_scheduler_kwargs	min_lr_rate: 0.1
warmup_ratio	0.1
max_steps	500
num_train_epochs	1
per_device_train_batch_size	6
per_device_eval_batch_size	6
Generation Settings	
max_prompt_length	512
max_completion_length	3584 or 4096
num_generations	6
temperature	0.7
seed	42
Logging and Saving	
log_completions	true
log_level	info
logging_first_step	true
logging_steps	1
logging_strategy	steps
save_strategy	steps
save_steps	50
report_to	wandb
Reward Configuration	
reward_funcs	format, accuracy (cosine)
reward_weights	1.0, 2.0
Hub Settings	
hub_model_id	OpenRS-GRPO
hub_strategy	every_save
push_to_hub	true