November 25, 2025

Does Reinforcement Learning Really Incentivize Reasoning Capacity in LLMs Beyond the Base Model?

Yang Yue[1*†], **Zhiqi Chen**[1*], **Rui Lu**[1], **Andrew Zhao**[1], **Zhaokai Wang**[2], **Yang Yue**[1], **Shiji Song**[1], **and Gao Huang**[1✉]

[1] LeapLab, Tsinghua University [2] Shanghai Jiao Tong University

[*] Equal Contribution [†] Project Lead [✉] Corresponding Author

Reinforcement Learning with Verifiable Rewards (RLVR) has recently demonstrated notable success in enhancing the reasoning performance of large language models (LLMs), particularly in mathematics and programming tasks. It is widely believed that, similar to how traditional RL helps agents to explore and learn new strategies, RLVR enables LLMs to continuously self-improve, thus acquiring novel reasoning abilities that exceed the capacity of the corresponding base models. In this study, we take a critical look at *the current state of RLVR* by systematically probing the reasoning capability boundaries of RLVR-trained LLMs across various model families, RL algorithms, and math/coding/visual reasoning benchmarks, using pass@k at large k values as the evaluation metric. While RLVR improves sampling efficiency towards correct paths, we surprisingly find that current training *rarely* elicit fundamentally new reasoning patterns. We observe that while RLVR-trained models outperform their base models at smaller values of k (*e.g.*, k=1), base models achieve higher pass@k score when k is large. Moreover, we observe that the reasoning capability boundary of LLMs often narrows as RLVR training progresses. Further coverage and perplexity analysis shows that the reasoning paths generated by RLVR models are already included in the base models' sampling distribution, suggesting that their reasoning abilities originate from and are *bounded* by the base model. From this perspective, treating the base model as an upper bound, our quantitative analysis shows that six popular RLVR algorithms perform similarly and remain far from optimal in fully leveraging the potential of the base model. In contrast, we find that distillation can introduce new reasoning patterns from the teacher and genuinely expand the model's reasoning capabilities. Taken together, our findings suggest that current RLVR methods have not fully realized the potential of RL to elicit genuinely novel reasoning abilities in LLMs. This underscores the need for improved RL paradigms, such as effective exploration mechanism, more deliberate and large-scale data curation, fine-grained process signal, and multi-turn agent interaction, to unlock this potential.

Project Page: `https://limit-of-RLVR.github.io`

1. Introduction

The development of reasoning-centric large language models (LLMs), such as OpenAI-o1 (Jaech et al., 2024), DeepSeek-R1 (Guo et al., 2025), and Kimi-1.5 (Team et al., 2025), has significantly advanced the frontier of LLM capabilities, particularly in solving complex logical tasks involving mathematics and programming. In contrast to traditional instruction-tuned approaches that rely on human-curated

The first author Yang Yue (乐洋) and the sixth author Yang Yue (乐阳) share the same English name but different Chinese names. *Correspond to: {le-y22, zq-chen23}@mails.tsinghua.edu.cn, gaohuang@tsinghua.edu.cn.*

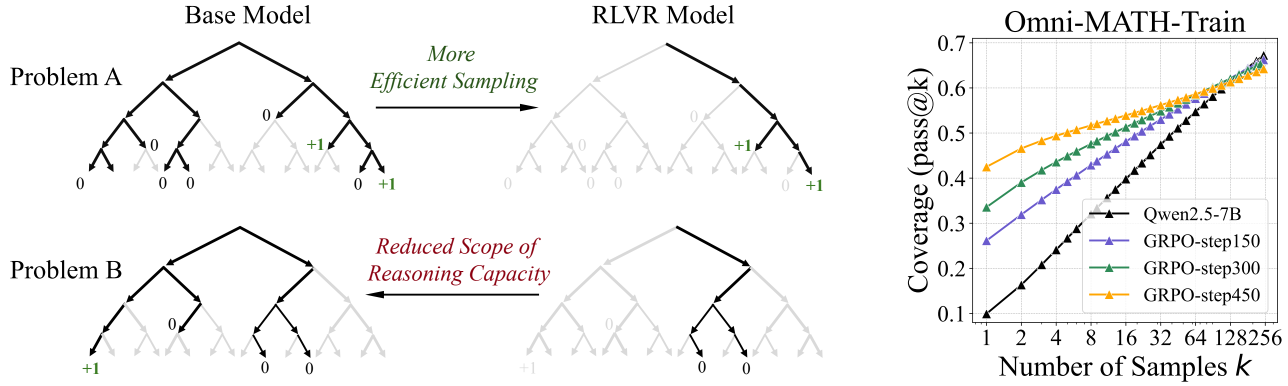

Figure 1: **(Left)** The effect of current RLVR on LLM's reasoning ability. Search trees are generated by repeated sampling from the base and RLVR-trained models for a given problem. Grey indicates paths that are unlikely to be sampled by the model, while **black** indicates paths that are likely to be sampled. Green indicates correct paths, which has positive rewards. Our key finding is that all reasoning paths in the RLVR model are already present in the base model. For certain problems like Problem A, RLVR training biases the distribution toward rewarded paths, improving sampling efficiency. However, this comes at the cost of reduced scope of reasoning capacity: For other problems like Problem B, the base model contains the correct path, whereas that of the RLVR model does not. **(Right)** As RLVR training progresses, the average performance (*i.e.*, pass@1) improves, but the coverage of solvable problems (*i.e.*, pass@256) decreases, indicating a reduction in LLM's reasoning boundary.

annotations (Achiam et al., 2023; Grattafiori et al., 2024), the key driver behind this leap forward is large-scale *Reinforcement Learning with Verifiable Rewards* (RLVR) (Lambert et al., 2024; Guo et al., 2025). RLVR starts with a pretrained base model or one fine-tuned on long chains of thought (CoT) data, optimizing it via reinforcement learning based on simple, automatically computable rewards. These rewards are determined by whether the model's output matches a ground-truth solution in mathematics or passes unit tests in code, thus enabling scalability without human labeling. This framework has gained significant attention due to its simplicity and practical effectiveness. In traditional RL settings such as game playing (*e.g.*, Atari, Go), agents often autonomously discover new strategies and surpass even human-level performance through self-improvement (Mnih et al., 2015; Silver et al., 2017). Inspired by this success, it is widely believed that RLVR similarly enables LLMs to autonomously develop novel reasoning patterns, including enumeration, self-reflection, and iterative refinement, surpassing the capabilities of their base models (Guo et al., 2025). Consequently, RLVR has been considered a promising path toward continuously self-evolving LLMs, potentially bringing us closer to more powerful intelligence (Guo et al., 2025).

However, despite its empirical success, the underlying effectiveness of current RLVR remains underexamined. This raises a fundamental question: ***Does current RLVR genuinely enable LLMs to acquire novel reasoning abilities–similar to how traditional RL discovers new strategies through exploration–or does it simply utilize reasoning patterns already in the base model?***

To rigorously answer this question, we must first assess the reasoning capability boundaries of both base and RLVR-trained models. Traditional evaluation metrics rely on average score from greedy decoding or nucleus sampling (Holtzman et al., 2020), which reflects average-case behavior. However, these metrics risk underestimating the true potential of a model, especially if it fails on difficult problems after limited attempts, despite being capable of solving them with more sampling. To overcome this limitation, we adopt the pass@k metric (Brown et al., 2024), where a problem is considered solved if any of the k sampled outputs is correct. By allowing multiple attempts, pass@k reveals whether a model has the potential to solve a problem. The average pass@k across a dataset thus reflects the proportion of problems a model can potentially solve within k trials, offering a more robust view of its reasoning boundary. This provides a rigorous test on whether the RLVR training yields fundamentally transcending capacity, enabling the model to solve problems that the base model cannot.

Using the pass@k metric, we conduct extensive experiments across various benchmarks, covering multiple LLM families, model sizes, and RLVR algorithms to compare base models with their RLVR-trained counterparts. We uncover several surprising findings that offer a more comprehensive assessment of the effectiveness of current RLVR training and reveal the gap between existing RLVR methods and the ideal goals of RL-discovering genuinely new reasoning strategies:

• **Current RLVR models often exhibit narrower reasoning coverage than their base models.** In pass@k curves, although RLVR models outperform their base models at small k, it is surprising that base models consistently surpass RLVR models across all benchmarks and LLM families as k increases. This suggests that current RLVR training does *not* expand, and even reduce the scope of reasoning over solvable problems. Manual inspection of model responses shows that, for most problems, the base model can produce *at least one* correct CoT, implying that it can already generate correct reasoning paths for problems that were previously considered only solvable for RLVR models.

• **Reasoning paths generated by current RLVR model already exist in its base model.** To further investigate this phenomenon, we analyze the accuracy distribution. The results show that although RLVR improves average performance (*i.e.*, pass@1) by sampling more efficiently on problems already solvable by the base model, it does not enable the model to solve new problems. Further perplexity analysis reveals that the reasoning paths produced by RLVR models already exist within the output distribution of the base model. These findings indicate that RLVR does not introduce fundamentally new reasoning capabilities and that the reasoning capacity of current RLVR models remains bounded by that of its base model. This effect of RLVR is illustrated in Figure 1 (left).

• **Current RLVR algorithms perform similarly and remain far from optimal.** Treating the base model as an upper bound, we define the *sampling efficiency gap* (Δ_{SE}), shown in Figure 8 (top), as the difference between an RL model's pass@1 and the base model's pass@k (with $k = 256$ as a proxy for upper-bound performance). This metric quantifies how closely an RL algorithm approaches the optimal bound. Across all algorithms (e.g., PPO, GRPO, Reinforce++), Δ_{SE} shows only minor variation yet remains consistently large, suggesting that current RLVR methods, while improving sampling efficiency, are still far from optimal.

• **RLVR and distillation are fundamentally different.** While RLVR improves reasoning scores by more efficiently sampling high-reward outputs, it does not elicit new reasoning capabilities and remains constrained within the base model's capacity. In contrast, distillation can transfer new reasoning patterns from a stronger teacher to the student. As a result, distilled models often demonstrate an expanded reasoning scope beyond that of the base model.

In conclusion, our findings show that current RLVR methods, while improving sampling efficiency, rarely elicit novel reasoning beyond the base model. This highlights a gap between existing RLVR methods and the goals of reinforcement learning, underscoring the need for improved RL paradigms such as better exploration, continual data scaling, fine-grained process signal, and multi-turn agent interaction.

2. Preliminaries

In this section, we first outline the fundamentals of RLVR, then introduce the pass@k metric to evaluate reasoning boundaries, and explain why it is preferred over alternatives like best-of-N.

2.1. Reinforcement Learning with Verifiable Rewards

Verifiable Rewards. Let π_θ be an LLM with parameters θ that generates a token sequence $\mathbf{y} = (y_1, \ldots, y_T)$ conditioned on a natural-language prompt x. A deterministic *verifier* $\mathcal{V}$ returns a binary reward: $r = \mathcal{V}(x, \mathbf{y}) \in \{0, 1\}$, where $r = 1$ if and only if the model's final answer is exactly correct. A format reward may also be added to encourage the model to explicitly separate the reasoning process from the final answer. The goal of RL is to learn a policy to maximize the expected reward: $J(\theta) = \mathbb{E}_{x\sim\mathcal{D}}\left[\mathbb{E}_{\mathbf{y}\sim\pi_\theta(\cdot|x)}[\,r\,]\right]$, where $\mathcal{D}$ is the distribution of prompts.

RLVR Algorithms. Proximal Policy Optimization (PPO) (Schulman et al., 2017) proposed using the following clipped surrogate to maximize the objective:

$$\mathcal{L}_{\text{CLIP}} = \mathbb{E}\left[\min(r_t(\theta)A_t,\ \text{clip}(r_t(\theta), 1-\epsilon, 1+\epsilon)A_t)\right], \tag{1}$$

where $r_t(\theta) = \pi_\theta(y_t|x, \mathbf{y}_{<t})/\pi_{\theta_{\text{old}}}(y_t|x, \mathbf{y}_{<t})$, and A_t is the advantage estimated by a value network V_ϕ.