

DRPO: EFFICIENT REASONING VIA DECOUPLED REWARD POLICY OPTIMIZATION

Anonymous authors

Paper under double-blind review

ABSTRACT

Recent large reasoning models (LRMs) driven by reinforcement learning algorithms (e.g., GRPO) have achieved remarkable performance on challenging reasoning tasks. However, these models suffer from overthinking, generating unnecessarily long and redundant reasoning even for simple questions, which substantially increases computational cost and response latency. While existing methods incorporate length rewards to GRPO to promote concise reasoning, they incur significant performance degradation. We identify the root cause: when rewards for correct but long rollouts are penalized, GRPO’s group-relative advantage function can assign them negative advantages, actively discouraging valid reasoning. To overcome this, we propose Decoupled Reward Policy Optimization (DRPO), a novel framework that decouples the length-based learning signal of correct rollouts from incorrect ones. DRPO ensures that reward signals for correct rollouts are normalized solely within the positive group, shielding them from interference by negative samples. The DRPO’s objective is grounded in integrating an optimized positive data distribution, which maximizes length-based rewards under a KL regularization, into a discriminative objective. We derive a closed-form solution for this distribution, enabling efficient computation of the objective and its gradients using only on-policy data and importance weighting. Of independent interest, this formulation is general and can incorporate other preference rewards of positive data beyond length. Experiments on mathematical reasoning tasks demonstrate DRPO’s significant superiority over six efficient reasoning baselines. Notably, with a 1.5B model, our method achieves 77% length reduction with only 1.1% performance loss on simple questions like GSM8k dataset, while the follow-up baseline sacrifices 4.3% for 68% length reduction.

1 INTRODUCTION

Recently, large reasoning models (LRMs) driven by Reinforcement Learning (RL) (Guo et al., 2025; Team et al., 2025b) have demonstrated remarkable performance on complex reasoning tasks like mathematics, coding, and scientific problem-solving. Unlike conventional language models that focus on direct thoughts and solutions, LRMs improve performance by generating extended chain-of-thought paths (Wei et al., 2022), allowing them to revisit intermediate steps, correct errors, and even explore alternative reasoning paths. This approach equips LRMs with stronger reasoning abilities and has become a standard paradigm to develop models capable of solving complex tasks.

However, existing LRM s suffer from overthinking with lengthy and redundant reasoning paths. As demonstrated by Chen et al. (2024), reasoning models like DeepSeek-R1 (Guo et al., 2025) need to generate about 1,000 tokens to answer “what is the answer of 2 plus 3”, while only around 10 tokens are needed for non-reasoning models. Such overly generated reasoning paths raise significant issues, leading to substantially increased computational cost and longer inference time. Numerous studies have been conducted to explore ways to eliminate redundant reasoning and improve the reasoning efficiency. A popular strategy is to introduce explicit reward shaping with length penalties in RL to guide the model toward concise reasoning (Arora & Zanette; Huang et al.; Xiang et al.; Aggarwal & Welleck), e.g., penalizing rewards of correct answers based on reasoning length to encourage shorter reasoning. Nevertheless, almost all existing methods fall short in preserving performance while shortening reasoning, causing substantial performance loss. This raises a key question: *how to guide RL for efficient reasoning with minimal performance drop?*

We identify the root cause underlying the insufficiency of existing RL-based efficient reasoning methods. The recent advancement of training efficient reasoning models has been largely built upon the Group Relative Policy Optimization (GRPO) framework, due to its groundbreaking performance (Guo et al., 2025). GRPO’s effectiveness hinges on its group-relative advantage function, which normalizes a rollout’s reward against the group average to create a learning signal that distinguishes positive from negative examples. Yet, this very strength becomes its greatest weakness when moving beyond simple correctness. We demonstrate that the framework is fundamentally ill-suited for composite rewards. Incorporating a length penalty reduces the reward for correct but long answers, often pushing their group-relative advantage below zero. Consequently, GRPO is misled into interpreting verbose correct answers as negative examples, discouraging valid reasoning and creating a significant optimization barrier (Figure 1).

An effective mechanism must not only distinguish right from wrong but also efficient right from inefficient right—assigning a strong positive signal to concise answers and a weaker positive signal to verbose ones, all while suppressing incorrect reasoning. To this end, we introduce Decoupled Reward Policy Optimization (DRPO), a novel RL framework that fundamentally rethinks how learning signals are constructed. DRPO’s core innovation is the decoupling of the learning signal calculation: it normalizes rewards for a correct rollout only against other correct rollouts, completely insulating them from the negative examples that corrupt GRPO’s signal. This ensures that length penalty proportionally reduces the positive signal of a long correct answer but never pushes it into negative territory, thereby achieving a more favorable trade-off between efficiency and accuracy. We formalize this intuition by deriving a generalized objective in a discriminative RL framework. This objective integrates a perturbed version of the on-policy positive data distribution, where the perturbation is designed explicitly to maximize a length-based reward. We derive a closed-form solution of the perturbed distribution, which allows us to efficiently optimize the objective without any additional data collection, using only on-policy samples via importance weighting.

Our contributions are four-fold:

- We diagnose a critical, previously overlooked deficiency in the widely-adopted GRPO framework: its group-relative advantage function is ill-suited for correctness–length composite rewards and actively harms learning when incentivizing efficiency.
- We propose Decoupled Reward Policy Optimization (DRPO), a new paradigm that decouples learning signals for positive and negative data. DRPO provides consistent, uncorrupted policy gradients for multi-reward optimization (e.g., correctness and length).
- We derive a rigorous formulation for DRPO by integrating a reward-maximizing, perturbed positive data distribution directly into a discriminative objective. We obtain a tractable closed-form solution to the perturbed distribution, yielding a practical algorithm requiring only on-policy data with no overhead.
- We conduct experiments to demonstrate the superiority of DRPO in training efficient reasoning models, substantially outperforming strong baselines across different model sizes and various mathematical reasoning benchmarks.

2 RELATED WORK

Large Reasoning Models. Earlier structured prompting approaches such as Chain-of-Thought (CoT) (Wei et al., 2022), Tree-of-Thought (ToT) (Yao et al., 2023), and Graph-of-Thought (GoT) (Besta et al., 2024) demonstrated the importance of decomposing complex problems into intermediate steps. However, these methods rely heavily on prompting and search heuristics, lacking a unified learning framework to optimize reasoning efficiency and robustness.

The breakthrough came with DeepSeek-R1, which revealed that large reasoning models (LRMs) trained via large-scale reinforcement learning, particularly GRPO (Shao et al., 2024), can autonomously acquire advanced reasoning behaviors such as branching, verification, and backtracking. This success inspired numerous follow-up studies: some aimed at reproducing the effectiveness of GRPO (Wen et al., 2025; Luo et al., 2025b; He et al., 2025), while others investigated its limitations and proposed refinements to further enhance reasoning performance (Yu et al., 2025; Li et al., 2025a) (Chen et al., 2025; Zheng et al., 2025). In parallel, many open-weight reasoning models have adopted GRPO, including Qwen-3 (Yang et al., 2025a), GLM-4.5 (Team et al., 2025a),

108 K2-Think (Cheng et al., 2025), and Goedel-solver (Lin et al., 2025), among others. Despite these
 109 advances, existing LRM often suffer from *overthinking*—producing unnecessarily long and redundant
 110 reasoning even for simple problems. This work aims to directly address this issue by training
 111 LRM to reason both *efficiently* and *effectively*.

112 **Efficient Reasoning in LRM.** To address the issue of overthinking in LRM, a variety of techniques
 113 have been proposed (Sui et al., 2025; Yue et al., 2025), including (1) training-free methods,
 114 which shorten the reasoning paths via prompt (Aytes et al., 2025; Han et al., 2024) or manipulating
 115 the decoding process (Yang et al., 2025b; Yong et al., 2025; Wang et al., 2025a; Liu et al., 2025a;
 116 Wang et al., 2025c); (2) Supervised Fine-tuning (SFT) methods, which rely on compressed reasoning
 117 datasets for finetuning. These datasets are curated via token-level selection (Yuan et al., 2025;
 118 Xia et al., 2025; Zhuang et al., 2025), step-level selection (Xiao et al., 2025; Cui et al., 2025; Wang
 119 et al., 2025b), path-level selection (Munkhbat et al., 2025; Ghosal et al., 2025); (3) RL-based methods,
 120 which carefully design reward mechanisms to guide the model to reason efficiently (Hou et al.;
 121 Aggarwal & Welleck; Arora & Zanette; Xiang et al.; Luo et al.; Huang et al.; Yi et al., 2025; Liu
 122 et al., 2025b; Xu et al., 2025; Fang et al., 2025; Li et al., 2025b).

123 Among these techniques, RL-based methods have been demonstrated to be one of the most effective
 124 approaches due to their scalability and flexibility. Specifically, L1 (Aggarwal & Welleck) integrates
 125 a length constraint into the reward function to encourage higher performance while meeting the
 126 length goal specified in the prompt. Arora & Zanette employs online RL with a length penalty
 127 based on the distribution of correct answers, penalizing correct responses longer than the average
 128 while encouraging those shorter than the average. ALP (Xiang et al.) adaptively adjusts length
 129 penalties according to problem difficulty, measured by pass rate, assigning stronger penalties to
 130 high pass-rate problems to discourage overlong reasoning. HAPO (Huang et al.) keeps track of
 131 the minimum length of correct responses for each question, penalizing outputs that exceed this
 132 length while rewarding those that are shorter. Nevertheless, these methods all suffer from misleading
 133 learning signals and fall short in preserving performance while shortening reasoning, due to the
 134 limitation of their adopted relative advantage function.

135 **Discriminative Learning for LRM.** Discriminative learning is a classical paradigm applied widely
 136 to traditional tasks like classifications (Bishop & Nasrabadi, 2006; Yang & Ying, 2022) and rank-
 137 ings (Burges et al., 2005; Cao et al., 2007). These methods follow the principle of raising scores
 138 for positive (correct) samples while lowering scores for negative (incorrect) ones. Recently, several
 139 works have explored applying the principle of discriminative learning to LRM training. For exam-
 140 ple, (Li et al., 2025a) proposed a discriminative constrained RL framework with verifiable binary
 141 rewards to finetune LRM. Lyu et al. (2025); Bai et al. (2025) utilize discriminative loss for behav-
 142 ior cloning on positive samples and policy gradient on negative samples. Su et al. (2025) leverages
 143 discriminative learning with positive-negative pairs for reasoning tasks. However, these methods are
 144 limited to binary accuracy rewards and don't address the challenge of overthinking in LRM.

145 3 LIMITATION OF INCORPORATING LENGTH PENALTY INTO GRPO

147 **Notations.** We study the fine-tuning of a generative reasoning model π_θ parameterized by θ . At each
 148 learning step, the previous model is denoted by π_{old} , which is responsible for generating answers to
 149 a given set of questions. For a question $q \in \Sigma^*$ (including its prompt), the output $o \in \Sigma^*$ is sampled
 150 from $\pi_{\text{old}}(\cdot|q)$, consisting of both reasoning traces and the final answer, where Σ^* denotes the space
 151 of all sequences of tokens with arbitrary length. More concretely, o is generated sequentially at
 152 the token level: $o_t \sim \pi_{\text{old}}(\cdot|q, o_{<t})$, for $t = 1, \dots, |o|$. The correctness reward $r_c(o|q) \in \{1, 0\}$
 153 for a given question q and its corresponding answer in the output o is verified by either matching
 154 the extracted answer against the ground-truth answer or a formal verification tool (Guo et al., 2025;
 155 Zhang et al., 2025). Let $\pi_{\text{old}}^+(\cdot|q)$ denote the conditional distribution of outputs when the reward
 156 is one (i.e., correct answers) and $\pi_{\text{old}}^-(\cdot|q)$ denote the conditional distribution of outputs when the
 157 reward is zero (i.e., incorrect answers). Let $[\cdot]_+ = \max(\cdot, 0)$ denote the hinge function.

158 Following the success of DeepSeek-R1, Group Relative Policy Optimization (GRPO) is widely
 159 adopted in existing RL-based efficient reasoning methods to estimate the relative advantage from
 160 group rewards instead of the critic model. In the following, we illustrate the limitations of incorpo-
 161 rating length penalty into GRPO for promoting efficient reasoning. We note that similar limitations
 occur to other RL methods, such as RLOO (Ahmadian et al., 2024), and other REINFORCE-based

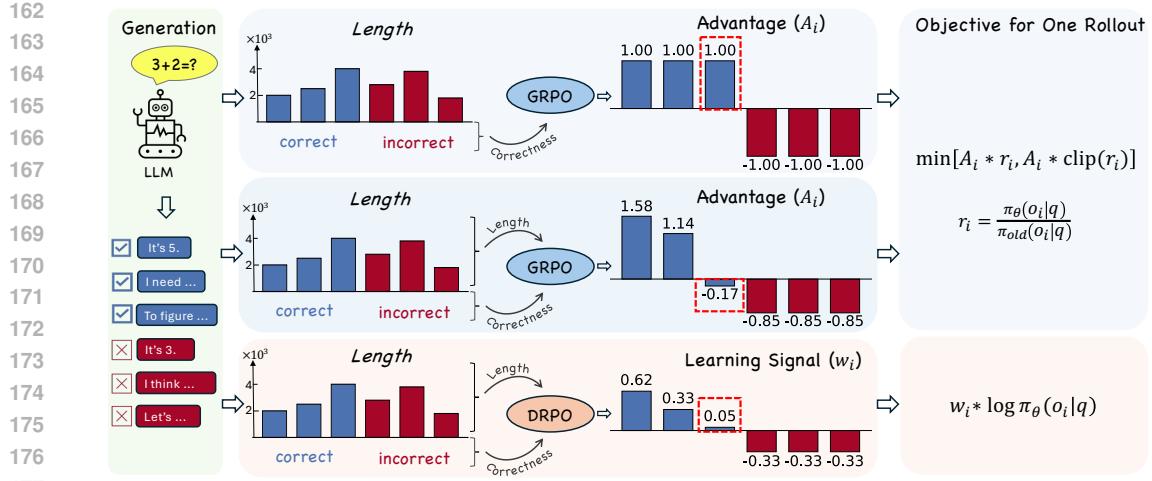


Figure 1: Illustration of the limitation of GRPO with length penalty and the benefit of our approach. Suppose $[1, 1, 1, 0, 0, 0]$ are the accuracy rewards for 6 responses, and $[0.73, 0.6, 0.2, 0, 0, 0]$ are the rewards after applying the length penalty to correct answers. Using the group-relative advantage calculation of GRPO, the advantages for the third response shift from **1** (without length penalty) to **-0.17** (with length penalty added), inadvertently penalizing the third correct response, which may substantially harm performance. In contrast, our proposed DRPO reduces the learning signal for lengthy and correct responses but never pushes them to the negative territory.

methods (Hu et al., 2025; Chu et al., 2025), which couple rewards for correct and incorrect answers to compute advantages for learning.

The GRPO objective for maximization is given by:

$$\mathcal{J}_{\text{GRPO}}(\theta) = \mathbb{E}_q \mathbb{E}_{\{o_i\}_{i=1}^G \sim \pi_{\text{old}}(\cdot|q)} \left[\frac{1}{G} \sum_{i=1}^G \frac{1}{|o_i|} \sum_{t=1}^{|o_i|} \min \left(r_{i,t} A(o_i|q), \text{clip}(r_{i,t}, 1 - \epsilon, 1 + \epsilon) A(o_i|q) \right) \right] - \beta \mathbb{D}_{\text{KL}}(\pi_\theta || \pi_{\text{ref}}) \quad (1)$$

where $r_{i,t} = \frac{\pi_\theta(o_{i,t}|q, o_{i,<t})}{\pi_{\text{old}}(o_{i,t}|q, o_{i,<t})}$, π_{ref} is a frozen reference model, $\mathbb{D}_{\text{KL}}(\cdot, \cdot)$ denotes the KL divergence between two distributions, and $A(o_i|q)$ denotes the relative advantage of output o_i , which quantifies how much better the reward of o_i denoted by $r(o_i|q)$ compared to the group average reward and guides the learning direction. Specifically, $A(o_i|q)$ is computed by

$$A(o_i|q) = \frac{r(o_i|q) - \text{mean}(r(o_1|q), r(o_2|q), \dots, r(o_G|q))}{\text{std}(r(o_1|q), r(o_2|q), \dots, r(o_G|q))}. \quad (2)$$

Let us consider incorporating a length penalty into GRPO. Existing RL methods with length control reveal a shared principle: penalizing rewards of correct answers based on reasoning length to encourage shorter reasoning, e.g., $r(o|q) = r_c(o|q) - r_l(o|q)$, where $r_l(\cdot)$ is a length-based cost or reward function (refer to Table 2 for detailed formulations). The consequence is that the reward of a correct with long output will be shifted down relatively. Incorporating this reward into GRPO's group relative advantage calculation may bias the intended effect, misleading the learning process. Let us consider an illustrative example in Figure 1. Suppose there are six generated outputs with different lengths, whose correctness rewards are $[1, 1, 1, 0, 0, 0]$ and corresponding lengths are $[2000, 2500, 4000, 2800, 3800, 3200]$. After combining the length reward with correctness reward, the combined reward of each answer becomes $[0.73, 0.6, 0.2, 0, 0, 0]$ ¹. So far, it looks like that all the designs work well since short correct answers have larger positive rewards and longer correct answers have smaller positive rewards while incorrect answers have zero rewards. However, when computing group relative advantage $A(o|q)$ in GRPO (i.e., Eqn. (2)), the advantage for the third correct response shifts from **1** (without length penalty) to **-0.17** (with length penalty added). This

¹These values are calculated with the formula proposed in (Arora & Zanette). We note that various reward combination designs in the literature lead to the same issue. Refer to Appendix A.3 for a detailed discussion.

216 negative signal will discourage valid reasoning and create a significant optimization barrier, which
 217 may substantially harm performance.
 218

219 4 DRPO: DECOUPLED REWARD POLICY OPTIMIZATION 220

221 The main reason of getting a negative learning signal in GRPO for a verbose correct answer is that
 222 its reshaped reward could become less than the mean reward of all samples including positive and
 223 negative ones. Our approach to avoid this issue is to decouple the rewarding of positive and negative
 224 samples so that the length rewards are only normalized within the positive group. To this end,
 225 we develop our approach based on a recent work (Li et al., 2025a), which proposes a discriminative
 226 optimization framework (DisCO) that directly increases the generative likelihood of positive answers
 227 and decrease that of negative answers. Below, we first introduce DisCO and then present our novel
 228 approach of integrating length rewards into DisCO’s objective.
 229

230 4.1 DISCRIMINATIVE CONSTRAINED POLICY OPTIMIZATION (DISCO) 231

232 DisCO was proposed to address several inherent limitations of GRPO, including difficulty bias and
 233 clipping operations. Let $s_\theta(o, q)$ be a scoring function, which measures the generative likelihood of
 234 answer o given the input q . In this paper, we will consider $s_\theta(o, q) = \frac{1}{|o|} \sum_{t=1}^{|o|} \log \pi_\theta(o_t | q, o_{<t})$,
 235 which is effective as demonstrated in (Li et al., 2025a). The objective of DisCO is formulated as:
 236

$$\begin{aligned} \max_{\theta} \quad & \mathbb{E}_q \left[\mathbb{E}_{o \sim \pi_{\text{old}}^+(\cdot | q)} s_\theta(o, q) - \tau \log \left(\mathbb{E}_{o' \sim \pi_{\text{old}}^-(\cdot | q)} \exp \left(\frac{s_\theta(o', q)}{\tau} \right) \right) \right], \\ \text{s.t.} \quad & \mathbb{D}_{\text{KL}}(\pi_{\text{old}} || \pi_\theta) \leq \delta, \end{aligned} \quad (3)$$

237 where $\delta > 0$ is a hyper-parameter. The intuition behind this formulation is straightforward: it
 238 increases the scores of positive responses $o \sim \pi_{\text{old}}^+(\cdot | q)$ while decreasing the scores of negative re-
 239 sponses, aggregated through a log-sum-exp function. The log-sum-exp has its roots in discriminative
 240 learning, appearing in losses such as cross-entropy and contrastive loss, and naturally emphasizes
 241 hard negatives by assigning them larger learning signal. The constraint $\mathbb{D}_{\text{KL}}(\pi_{\text{old}} || \pi_\theta) \leq \delta$, inspired
 242 by TRPO (Schulman et al., 2015), is added to ensure the stability of training.
 243

244 While DisCO demonstrates impressive gains in reasoning performance over GRPO, the length of
 245 its reasoning is uncontrolled, leaving the challenge of enhancing reasoning efficiency unresolved.
 246 Moreover, the above objective is derived under a binary reward setting, which does not accept
 247 flexible reward design. In the following, we discuss how to incorporate length rewards into the
 248 framework to encourage efficient reasoning.
 249

250 4.2 DECOUPLED REWARD POLICY OPTIMIZATION 251

252 We consider a simple length reward $r_l(o) = 1 - \frac{|o|}{C}$ for any correct response o , where C is a
 253 constant denoting maximum response length. An intuitive idea is to assign a weight to positive
 254 answers before their scores $s_\theta(o, q)$ in (3) such that a shorter answer is assigned with a larger weight
 255 than a longer answer. Below, we formalize this idea by proposing a principled objective.
 256

257 Our goal is to maximize the score of correct outputs with high length rewards while penalizing those
 258 of wrong outputs regardless of their lengths. Suppose we have a distribution P_q^* , which specifies a
 259 distribution of correct outputs with high length rewards given a question q . Then, we can modify the
 260 objective in (3) as
 261

$$\max \mathbb{E}_q \left[\mathbb{E}_{o \sim P_q^*} s_\theta(o, q) - \tau \log \left(\mathbb{E}_{o' \sim \pi_{\text{old}}^-(\cdot | q)} \exp \left(\frac{s_\theta(o', q)}{\tau} \right) \right) \right]. \quad (4)$$

262 This can be explained that if we have an off-policy data distribution P_q^* of correct outputs with
 263 high length rewards, we can use its sampled data to steer the model training such that it generates
 264 correct outputs with high length rewards more likely. However, the issue is that P_q^* is not readily
 265 available. Although a naive solution is to curate such data manually, as in SFT-based efficient
 266 reasoning methods, it requires substantial human effort and lacks scalability. To address this, we
 267 draw insights from reinforcement learning from human feedback (RLHF) (Bai et al., 2022; Ouyang
 268 et al., 2022; Ziegler et al., 2019) by finding an optimal policy P_q^* that maximizes the length reward
 269

270 with a KL regularization:

$$271 \quad P_q^* = \arg \max_{P \in \mathcal{P}} \mathbb{E}_{o \sim P} r_l(o) - \lambda \mathbb{D}_{\text{KL}}(P, \pi_{\text{old}}^+(\cdot|q)), \quad (5)$$

273 where $\lambda > 0$ is a regularization parameter, \mathcal{P} denotes the set of all probability measures P on correct
274 data given q , which are absolutely continuous with respect to $\pi_{\text{old}}^+(\cdot|q)$, i.e., $\pi_{\text{old}}^+(o|q) = 0$ indicates
275 $P(o) = 0$.

276 However, unlike RLHF that uses a LLM to learn P^* , we derive its closed analytical solution similar
277 to (Rafailov et al., 2023) (See Appendix A.1 for a complete derivation):
278

$$279 \quad P_q^*(o) = \frac{\pi_{\text{old}}^+(o|q) \exp(r_l(o)/\lambda)}{\mathbb{E}_{o \sim \pi_{\text{old}}^+(\cdot|q)} \exp(r_l(o)/\lambda)}. \quad (6)$$

281 As a result, we have

$$283 \quad \mathbb{E}_{o \sim P_q^*} s_\theta(o, q) = \sum_{o \in \Sigma^*} \frac{\pi_{\text{old}}^+(o|q) \exp(r_l(o)/\lambda)}{\mathbb{E}_{o \sim \pi_{\text{old}}^+(\cdot|q)} \exp(r_l(o)/\lambda)} s_\theta(o, q) \\ 284 \\ 285 \quad = \mathbb{E}_{o \sim \pi_{\text{old}}^+(\cdot|q)} \frac{\exp(r_l(o)/\lambda)}{\mathbb{E}_{o \sim \pi_{\text{old}}^+(\cdot|q)} \exp(r_l(o)/\lambda)} s_\theta(o, q).$$

288 Plugging this formulation back into (4), we obtain the final objective function:

$$289 \quad \max \quad \mathbb{E}_q \left[\mathbb{E}_{o \sim \pi_{\text{old}}^+(\cdot|q)} \frac{\exp(r_l(o)/\lambda)}{\mathbb{E}_{o \sim \pi_{\text{old}}^+(\cdot|q)} \exp(r_l(o)/\lambda)} s_\theta(o, q) - \tau \log \left(\mathbb{E}_{o' \sim \pi_{\text{old}}^-(\cdot|q)} \exp \left(\frac{s_\theta(o', q)}{\tau} \right) \right) \right] \\ 290 \\ 291 \quad s.t. \quad \mathbb{D}_{\text{KL}}(\pi_{\text{old}} || \pi_\theta) \leq \delta. \quad (7)$$

294 It is notable that the final objective only relies on the on-policy data, and it has an explanation that
295 each positive data is assigned with a weight $\omega(o|q) = \frac{\exp(r_l(o)/\lambda)}{\mathbb{E}_{o \sim \pi_{\text{old}}^+(\cdot|q)} \exp(r_l(o)/\lambda)}$ informed by its length
296 but normalized only within the positive data. It is notable that when $\lambda = +\infty$, then $\omega(o|q) = 1$ and
297 the above objective reduces to that of DisCO in (3).

299 We solve the optimization problem (7) similarly as (Li et al., 2025a). In particular, the expectations
300 are replaced by empirical averages and the KL divergence is estimated by using sampled data, and
301 the constraint is handled by adding a penalty function $\beta_0 [\mathbb{D}_{\text{KL}}(\pi_{\text{old}} || \pi_\theta) - \delta]_+^2$ to the objective, where
302 β_0 is a penalty constant. For completeness, we present a full algorithm for solving (7) in Algorithm 1
303 in the Appendix. We refer to this method as Decoupled Reward Policy Optimization (DRPO).

305 5 EXPERIMENTS

307 **Datasets.** We validate our method on mathematical reasoning tasks. Specifically, we train mod-
308 els on the DeepScaleR-Preview-Dataset (Luo et al., 2025c), which consists of approximately 40.3k
309 question-answer pairs sourced from AIME problems from 1984 to 2023, AMC problems before
310 2023, Omni-MATH (Gao et al., 2024) and Still (Min et al., 2024) datasets. We evaluate all the mod-
311 els on math problems with different levels of difficulty, including (a) easy level: GSM8K (Cobbe
312 et al., 2021), (b) medium level: MATH-500 (Hendrycks et al., 2021), (c) hard level: Olympiad-
313 Bench (He et al., 2024), and (d) very hard level: AIME (aggregating 2024 and 2025). **To verify the**
314 **generalizability of our method to non-mathematical reasoning tasks, we also conducted experiments**
315 **on K&K logic puzzle dataset (Xie et al., 2024), which is included in the Appendix A.4.**

316 **Models.** We adopt three reasoning models as our base models: DeepSeek-R1-Distill-Qwen-1.5B
317 model, DeepSeek-R1-Distill-Qwen-7B, and DeepSeek-R1-Distill-Llama-8B, and conduct RL fine-
318 tuning from them.

319 **Baselines.** We compare our methods with six of the most recent state-of-the-art efficient reason-
320 ing methods, including (1) the method in (Arora & Zanette), which integrates length reshaped re-
321 wards into the RLOO advantage function, referred to as RLOO-LP; (2) ALP (Xiang et al.), which
322 uses a length penalty in GRPO that is the length scaled by the solving rate of each question; (3)
323 HAPO (Huang et al.), which penalizes the responses longer than the shortest correct answer in the
324 history while rewarding those that are shorter; (4) L1-max (Aggarwal & Welleck), which is a rea-

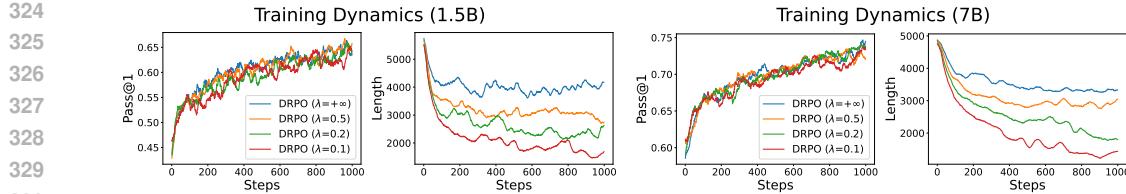


Figure 2: Training dynamics of DRPO with different regularization weights λ . The left two plots are for fine-tuning the 1.5B model, and the right two are for fine-tuning the 7B model. $\lambda = +\infty$ denotes the reference method DisCO, which does not incorporate length rewards in training.

soning language model that produces outputs satisfying a maximum length constraint given in the prompt; (5) ShorterBetter (SB) (Yi et al., 2025), which aims to match Sample Optimal Length defined as the shortest correct response among multiple generations; (6) LASER-D (Liu et al., 2025b), which employs a step length reward function based on difficulty-aware dynamic target length. We train models for methods (1)-(3) using the experimental settings described below. For methods (4)-(6), we evaluate the models provided in their original works. All the compared models were finetuned from the same base models on the DeepScaleR-Preview dataset, except L1-max, which was trained on DeepScaleR-1.5B-Preview. We summarize different reward designs of the above baselines in Table 2 in Appendix.

Training Details. For all the training, we employ the AdamW optimizer with a weight decay of 0.01 and set the learning rate to a constant $2e^{-6}$ for 1.5B model, $1e^{-6}$ for 7B model, and $5e^{-7}$ for 8B model, following Li et al. (2025a). We set the batch size to 128 for each step of RL, the mini-batch size to 32 for each iteration of model update, and sample 8 responses per question for training. For RLOO-LP, we tune their weight parameter $\alpha \in \{0.05, 0.1, 0.2\}$. For ALP, we tune their penalty weight $\beta \in \{1e^{-9}, 1e^{-8}, 1e^{-7}\}$. For HAPO, we tune their weight parameter $w \in \{0.01, 0.1, 1\}$. For the proposed method, we tune $\lambda \in \{0.5, 0.2, 0.1\}$. These parameters serve the same role that controls the tradeoff between efficiency and accuracy. For all other hyperparameters, we follow the default values from their official papers. Details are provided in Appendix A.2. The generation budget is limited to 8k tokens for both training and evaluation.

Evaluation. We use Pass@1 averaged over the 16 generated answers per prompt as the performance metric and use the averaged number of tokens as the length metric. For all methods, we train the model for 1000 RL steps to enable convergence and conduct evaluation every 200 steps. The models with the best pass@1 are reported, as we aim to enhance reasoning efficiency with minimal performance reduction. For models that are trained by us, we set temperature = 0.6 and top-p = 0.95, consistent with the training setup. For L1-MAX and LASER-D, we also use temperature = 0.6 and top-p = 0.95, while we use temperature = 0.9 and top-p = 0.9 for SB, following the original paper.

In addition to pass@1 and reasoning length, following (Luo et al., 2025a; Yi et al., 2025), we also adopt Accuracy Efficiency Score (AES) as a supplementary metric. AES integrates performance and reasoning length into a single measure, directly quantifying the trade-off between accuracy and computational cost. The AES is computed by:

$$\text{AES} = \begin{cases} \alpha * \Delta_{\text{Length}} + \beta * |\Delta_{\text{Acc}}|, & \text{if } \Delta_{\text{Acc}} \geq 0, \\ \alpha * \Delta_{\text{Length}} - \gamma * |\Delta_{\text{Acc}}|, & \text{if } \Delta_{\text{Acc}} < 0. \end{cases}$$

where $\alpha, \beta, \gamma > 0$, $\Delta_{\text{Length}} = \frac{\text{Length}_{\text{ref}} - \text{Length}_{\text{model}}}{\text{Length}_{\text{ref}}}$ and $\Delta_{\text{Acc}} = \frac{\text{Acc}_{\text{model}} - \text{Acc}_{\text{ref}}}{\text{Acc}_{\text{ref}}}$, where the quantities with subscript ‘‘ref’’ means each method’s baseline that does not consider length reward, and that with subscript ‘‘model’’ means the model for evaluation. In our experiments, we use the default values $\alpha = 1, \beta = 3$ same as (Luo et al., 2025a), but set $\gamma = 10$ to emphasize the importance of minimizing performance degradation.

5.1 VISUALIZATION OF LEARNING PROCESS

To directly verify the effectiveness of the proposed method, we present training dynamics of DRPO with different regularization weights λ in Figure 2, where $\lambda=+\infty$ corresponds to the reference method DisCO, which does not employ length reward during training. In terms of performance (first and third figures in Figure 2), we can see that DRPO with smaller λ values exhibits marginally

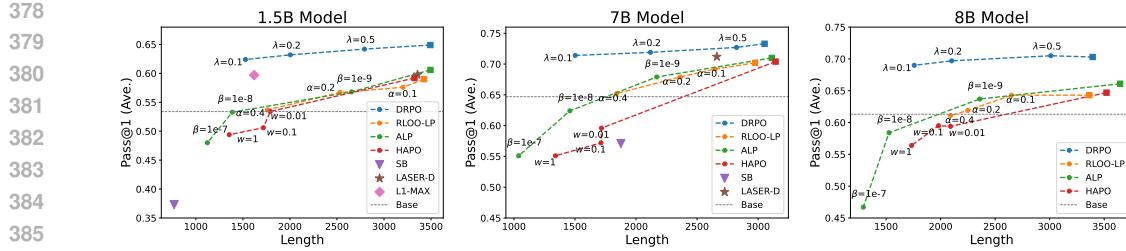


Figure 3: Comparison of performance-efficiency trade-off. Left is for fine-tuning 1.5B model, middle is for fine-tuning 7B model and right is for fine-tuning 8B model. Grey lines represent the base model performance before finetuning, with generation length of 4698 for 1.5B model, 4119 for 7B model, and 4325 for 8B model. Squares denote models trained with reference methods without length penalties, i.e., $\lambda=+\infty$ (corresponding to DisCO) for DRPO, $\alpha=0$ for RLOO-LP, $\beta=0$ (corresponding to GRPO) for ALP, $w=0$ for HAPO. Triangles denote the models trained by other works.

worse or comparable performance compared with DisCO, while the second and fourth figures show that smaller λ values lead to substantial reductions in response length, with $\lambda=0.1$ reducing length by over 50% relative to $\lambda=+\infty$ (DisCO). These observations demonstrate the effectiveness of DRPO to achieve more concise reasoning while maintaining nearly unchanged training performance. In the following section, we will evaluate the generalization of DRPO to test datasets, compared with other strong efficient reasoning baselines.

5.2 COMPARISON WITH BASELINES

In this part, we evaluate the effectiveness of the proposed method on test datasets, compared with existing efficient reasoning baselines.

Trade-off between performance and efficiency. We present the trade-off between performance and efficiency for various methods in Figure 3, where the averaged pass@1 over four math datasets of different difficulty is reported. We observe that our proposed DRPO consistently achieves significantly better performance-efficiency trade-off than all baselines on finetuning 1.5B, 7B and 8B models, including the models trained by other work. Notably, relative to the reference learning method (square marker), the proposed DRPO on finetuning 7B model in Figure 3 (Middle) efficiently reduces reasoning length from 3053 to 1502 (51% length reduction) with only 2.6% loss of performance via varying λ to control the tradeoff, demonstrating the effectiveness of DRPO to reduce reasoning length while preserving the reasoning capability. In contrast, all the efficient reasoning baselines suffer from severe performance degradation when the reasoning length is reduced. For example, RLOO-LP reduces reasoning length from 2975 to 1841 (38% length reduction) but incurs a 7.1% loss in performance on finetuning 7B model, with ALP showing a similar trend. Compared with DRPO, these methods trade off more performance for less reduction in length, highlighting the superiority of our method.

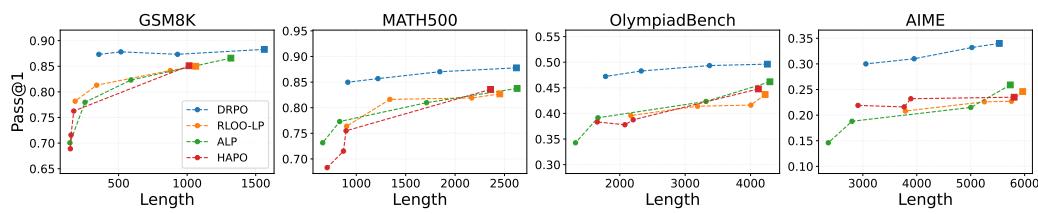
Evaluating Trade-off via AES. Additionally, we directly quantify the effectiveness of our proposed method in enhancing reasoning efficiency with minimal performance drop, using the Accuracy-Efficiency Score (AES). AES is positive when the model reduces output length while maintaining or enhancing accuracy, and negative when accuracy deteriorates. For fair comparison, reference model in AES is each method’s counterpart without length reward, i.e., $\lambda=+\infty$ for DRPO, $\alpha=0$ for RLOO-LP, $\beta=0$ for ALP, $w=0$ for HAPO. We present the best AES score for each method in Table 1 and defer detailed results in Appendix A.6. From Table 1, we observe that almost all the baseline methods exhibit negative AES scores across 1.5B, 7B, and 8B models, indicating the inefficiency of existing methods in reducing reasoning length while preserving performance. In contrast, DRPO consistently achieves a positive AES score for finetuning 1.5B, 7B, and 8B models, highlighting its capability of improving reasoning efficiency while maintaining performance.

5.3 LENGTH REDUCTION FOR DIFFERENT PROBLEM DIFFICULTY

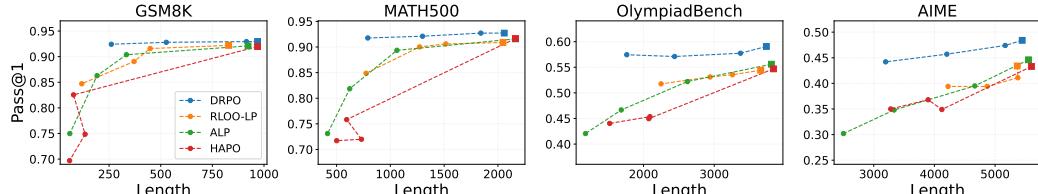
In this part, we study the impact of length reduction on questions with varying difficulties. In Figure 4, we present the performance of different methods across four math datasets of increasing

Table 1: Accuracy Efficiency Score (AES) Comparison with Baselines. The best AES score for each method is presented.

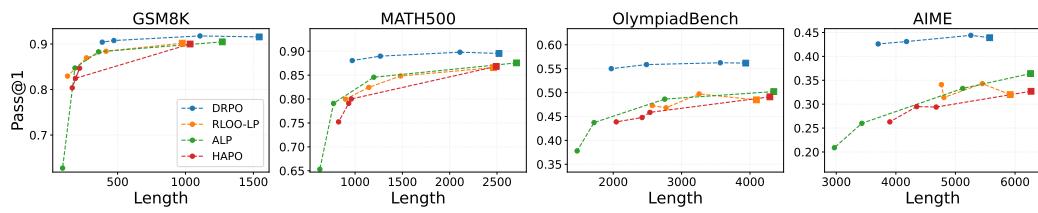
	Method	Pass@1	Length	AES
1.5B Model	RLOO-LP	0.567	2531	-0.129
	ALP	0.606	3494	-0.387
	HAPO	0.534	1791	-0.519
	DRPO	0.624	1527	0.178
7B Model	RLOO-LP	0.692	2649	-0.033
	ALP	0.679	2170	-0.134
	HAPO	0.596	1717	-1.080
	DRPO	0.714	1502	0.249
8B Model	RLOO-LP	0.619	2249	0.251
	ALP	0.637	2358	-0.01
	HAPO	0.595	1981	-0.366
	DRPO	0.690	1758	0.297



(a) 1.5B Model



(b) 7B Model



(c) 8B Model

Figure 4: Performance-efficiency tradeoff on individual datasets with increasing difficulty levels from left to right. (a) is for finetuning 1.5B model, (b) is for finetuning 7B model, (c) is for finetuning 8B model. Squares denote models trained with reference methods without length penalties (i.e., $\lambda=+\infty$ (corresponding to DisCO) for DRPO, $\alpha=0$ for RLOO-LP, $\beta=0$ (corresponding to GRPO) for ALP, $w=0$ for HAPO).

difficulty, from GSM8K (easy) to AIME (very hard). As shown in Figure 4, we observe that: (1) on relatively easy questions like GSM8K, DRPO significantly reduces generation length from 1563 to 356 (77.2% length reduction) for 1.5B model and from 969 to 261 (73.1% length reduction) for 7B model with negligible performance drop (-1.1% for 1.5B model and -0.6% for 7B model). In contrast, all the baselines still bring dramatic performance degradation, indicated by a steep slope. (2) With question difficulty increasing from left to right in Figure 4, all the models exhibit longer reasoning length and introduce increasing performance drop, which is reasonable since difficult

486 problems inherently require longer reasoning paths to solve. **(3)** When comparing Figure 4(a) and
 487 two models from the same LLM family, we can see that 7B models generally demand
 488 shorter reasoning length than 1.5B models. **Besides, models with larger sizes, e.g. 7B and 8B**
 489 **models, are more resilient to length reduction introduced by DRPO, as evidenced by flatter slopes**
 490 **than 1.5B models.** **(4)** Across various difficulty levels, DRPO consistently achieves a better trade-off
 491 between performance and length than all the baselines, demonstrating its effectiveness in guiding
 492 efficient reasoning.

493 We conduct further case studies in Appendix A.8 to analyze the efficient reasoning behavior after
 494 DRPO training, compared with DisCO. It is shown that models trained by DRPO preserve existing
 495 reflection capability while effectively reducing redundant back-and-forth reasoning.

496

497 6 CONCLUSION AND DISCUSSIONS

498

499 In this work, we revealed a key limitation of existing RL-based efficient reasoning methods, that their
 500 reliance on group-relative advantages can mislead learning when length rewards are included. To
 501 address this, we proposed DRPO, which decouples learning signals of correct and incorrect answers
 502 so that length penalties reduce the positive signals for correct reasoning but never reverse them. Our
 503 method integrates an optimized positive data distribution under a KL regularization into a discrimi-
 504 native objective, for which we derived a closed-form solution, enabling efficient computation using
 505 only on-policy data with importance weighting. Experiments on math reasoning tasks demonstrate
 506 significant superiority of our approach, compared with existing efficient reasoning methods.

507

508 A limitation of this study is that experiments were confined to 1.5B and 7B models due to compu-
 509 tational constraints. Extending the proposed approach to larger models or broader reasoning tasks
 510 remains an interesting direction. Besides, while DRPO is designed for efficient reasoning, its for-
 511 mulation is general and can be extended to incorporate other rewards on positive data beyond length,
 512 such as process rewards. Another interesting direction is to incorporate mechanisms to adapt λ to the
 513 difficulty level of questions, larger λ for more difficult questions and smaller λ for easier questions
 514 as implied by our results in Figure 4, which we leave for future investigation.

515

516 REPRODUCIBILITY STATEMENT

517

518 We provide detailed settings of our experiments in the experimental part and the appendix to help
 519 reproduce the results. Furthermore, we include the source code in the supplementary materials to
 520 help with reproducibility.

521

522

523 LLM USAGE

524

525 We use LLM only to help correct grammar errors and polish writing.

526

527 REFERENCES

528

529 Pranjal Aggarwal and Sean Welleck. L1: Controlling How Long A Reasoning Model Thinks With
 530 Reinforcement Learning. URL <http://arxiv.org/abs/2503.04697>.

531

532 Arash Ahmadian, Chris Cremer, Matthias Gallé, Marzieh Fadaee, Julia Kreutzer, Olivier Pietquin,
 533 Ahmet Üstün, and Sara Hooker. Back to basics: Revisiting reinforce style optimization for learn-
 534 ing from human feedback in llms. *arXiv preprint arXiv:2402.14740*, 2024.

535

536 Daman Arora and Andrea Zanette. Training Language Models to Reason Efficiently. URL <http://arxiv.org/abs/2502.04463>.

537

538 Simon A Aytes, Jinheon Baek, and Sung Ju Hwang. Sketch-of-thought: Efficient llm reasoning with
 539 adaptive cognitive-inspired sketching. *arXiv preprint arXiv:2503.05179*, 2025.

540

541 Lei Bai, Zhongrui Cai, Yuhang Cao, Maosong Cao, Weihan Cao, Chiyu Chen, Haojong Chen, Kai
 542 Chen, Pengcheng Chen, Ying Chen, et al. Intern-s1: A scientific multimodal foundation model.
 543 *arXiv preprint arXiv:2508.15763*, 2025.

540 Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn
 541 Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless
 542 assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*,
 543 2022.

544 Maciej Besta, Nils Blach, Ales Kubicek, Robert Gerstenberger, Michał Podstawski, Lukas Giani-
 545 nazzi, Joanna Gajda, Tomasz Lehmann, Hubert Niewiadomski, Piotr Nyczek, and Torsten Hoe-
 546 fler. Graph of thoughts: solving elaborate problems with large language models. In *Proceedings*
 547 *of the Thirty-Eighth AAAI Conference on Artificial Intelligence and Thirty-Sixth Conference on*
 548 *Innovative Applications of Artificial Intelligence and Fourteenth Symposium on Educational Ad-*
 549 *vances in Artificial Intelligence, AAAI'24/IAAI'24/EAAI'24*. AAAI Press, 2024. ISBN 978-1-
 550 57735-887-9. doi: 10.1609/aaai.v38i16.29720. URL <https://doi.org/10.1609/aaai.v38i16.29720>.

552 Christopher M Bishop and Nasser M Nasrabadi. *Pattern recognition and machine learning*, vol-
 553 ume 4. Springer, 2006.

555 Chris Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Greg Hul-
 556 lender. Learning to rank using gradient descent. In *Proceedings of the 22nd international confer-
 557 ence on Machine learning*, pp. 89–96, 2005.

559 Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. Learning to rank: from pairwise
 560 approach to listwise approach. In *Proceedings of the 24th international conference on Machine
 561 learning*, pp. 129–136, 2007.

562 Aili Chen, Aonian Li, Bangwei Gong, Binyang Jiang, Bo Fei, Bo Yang, Boji Shan, Changqing Yu,
 563 Chao Wang, Cheng Zhu, et al. Minimax-m1: Scaling test-time compute efficiently with lightning
 564 attention. *arXiv preprint arXiv:2506.13585*, 2025.

565 Xingyu Chen, Jiahao Xu, Tian Liang, Zhiwei He, Jianhui Pang, Dian Yu, Linfeng Song, Qiuzhi Liu,
 566 Mengfei Zhou, Zhuosheng Zhang, et al. Do not think that much for 2+ 3=? on the overthinking
 567 of o1-like llms. *arXiv preprint arXiv:2412.21187*, 2024.

569 Zhoujun Cheng, Richard Fan, Shibo Hao, Taylor W. Killian, Haonan Li, Suqi Sun, Hector Ren,
 570 Alexander Moreno, Daqian Zhang, Tianjun Zhong, Yuxin Xiong, Yuanzhe Hu, Yutao Xie,
 571 Xudong Han, Yuqi Wang, Varad Pimpalkhute, Yonghao Zhuang, Aaryamanvikram Singh, Xuezhi
 572 Liang, Anze Xie, Jianshu She, Desai Fan, Chengqian Gao, Liqun Ma, Mikhail Yurochkin, John
 573 Maggs, Xuezhe Ma, Guowei He, Zhiting Hu, Zhengzhong Liu, and Eric P. Xing. K2-think:
 574 A parameter-efficient reasoning system, 2025. URL <https://arxiv.org/abs/2509.07604>.

576 Xiangxiang Chu, Hailang Huang, Xiao Zhang, Fei Wei, and Yong Wang. Gpg: A simple and strong
 577 reinforcement learning baseline for model reasoning. *arXiv preprint arXiv:2504.02546*, 2025.

578 Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser,
 579 Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to
 580 solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

582 Yingqian Cui, Pengfei He, Jingying Zeng, Hui Liu, Xianfeng Tang, Zhenwei Dai, Yan Han, Chen
 583 Luo, Jing Huang, Zhen Li, et al. Stepwise perplexity-guided refinement for efficient chain-of-
 584 thought reasoning in large language models. *arXiv preprint arXiv:2502.13260*, 2025.

585 DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu,
 586 Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu,
 587 Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao
 588 Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan,
 589 Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao,
 590 Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Honghui Ding,
 591 Huajian Xin, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jiawei Wang, Jingchang
 592 Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai
 593 Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang,
 Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang,

594 Minghui Tang, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang,
 595 Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang,
 596 R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhua Chen, Shengfeng
 597 Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing
 598 Wu, Shengfeng Ye, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wanjia Zhao, Wen
 599 Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong
 600 Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu,
 601 Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xi-
 602 aoshua Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia
 603 Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng
 604 Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong
 605 Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong,
 606 Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou,
 607 Y. X. Zhu, Yanhong Xu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying
 608 Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda
 609 Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu,
 610 Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu
 611 Zhang, and Zhen Zhang. DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Rein-
 612 forcement Learning. URL <http://arxiv.org/abs/2501.12948>.

613 Gongfan Fang, Xinyin Ma, and Xinchao Wang. Thinkless: Llm learns when to think. *arXiv preprint*
 614 *arXiv:2505.13379*, 2025.

615 Bofei Gao, Feifan Song, Zhe Yang, Zefan Cai, Yibo Miao, Qingxiu Dong, Lei Li, Chenghao Ma,
 616 Liang Chen, Runxin Xu, et al. Omni-math: A universal olympiad level mathematic benchmark
 617 for large language models. *arXiv preprint arXiv:2410.07985*, 2024.

618 Soumya Suvra Ghosal, Souradip Chakraborty, Avinash Reddy, Yifu Lu, Mengdi Wang, Dinesh
 619 Manocha, Furong Huang, Mohammad Ghavamzadeh, and Amrit Singh Bedi. Does thinking
 620 more always help? understanding test-time scaling in reasoning models. *arXiv preprint*
 621 *arXiv:2506.04210*, 2025.

622 Dongyoung Go, Tomasz Korbak, Germán Kruszewski, Jos Rozen, Nahyeon Ryu, and Marc Dymet-
 623 man. Aligning language models with preferences through f-divergence minimization. *arXiv*
 624 *preprint arXiv:2302.08215*, 2023.

625 Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu,
 626 Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms
 627 via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

628 Tingxu Han, Zhenting Wang, Chunrong Fang, Shiyu Zhao, Shiqing Ma, and Zhenyu Chen. Token-
 629 budget-aware llm reasoning. *arXiv preprint arXiv:2412.18547*, 2024.

630 Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Leng Thai, Junhao Shen, Jinyi Hu,
 631 Xu Han, Yujie Huang, Yuxiang Zhang, et al. Olympiadbench: A challenging benchmark for
 632 promoting agi with olympiad-level bilingual multimodal scientific problems. *arXiv preprint*
 633 *arXiv:2402.14008*, 2024.

634 Jujie He, Jiacai Liu, Chris Yuhao Liu, Rui Yan, Chaojie Wang, Peng Cheng, Xi-
 635 aoyu Zhang, Fuxiang Zhang, Jiacheng Xu, Wei Shen, Siyuan Li, Liang Zeng, Tian-
 636 wen Wei, Cheng Cheng, Bo An, Yang Liu, and Yahui Zhou. Skywork open
 637 reasoner series. <https://capricious-hydrogen-41c.notion.site/Skywork-Open-Reasoner-Series-1d0bc9ae823a80459b46c149e4f51680>,
 638 2025. Notion Blog.

639 Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song,
 640 and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. *arXiv*
 641 *preprint arXiv:2103.03874*, 2021.

642 Bairu Hou, Yang Zhang, Jiabao Ji, Yujian Liu, Kaizhi Qian, Jacob Andreas, and Shiyu Chang.
 643 ThinkPrune: Pruning Long Chain-of-Thought of LLMs via Reinforcement Learning. URL
 644 <http://arxiv.org/abs/2504.01296>.

648 Jian Hu, Jason Klein Liu, Haotian Xu, and Wei Shen. Reinforce++: An efficient rlhf algorithm with
 649 robustness to both prompt and reward models. *arXiv preprint arXiv:2501.03262*, 2025.

650

651 Chengyu Huang, Zhengxin Zhang, and Claire Cardie. HAPO: Training Language Models to Reason
 652 Concisely via History-Aware Policy Optimization. URL <http://arxiv.org/abs/2505.11225>.

653

654 Gang Li, Ming Lin, Tomer Galanti, Zhengzhong Tu, and Tianbao Yang. Disco: Reinforcing large
 655 reasoning models with discriminative constrained optimization. *arXiv preprint arXiv:2505.12366*,
 656 2025a.

657

658 Zheng Li, Qingxiu Dong, Jingyuan Ma, Di Zhang, and Zhifang Sui. Selfbudgeter: Adaptive token
 659 allocation for efficient llm reasoning. *arXiv preprint arXiv:2505.11274*, 2025b.

660

661 Yong Lin, Shange Tang, Bohan Lyu, Ziran Yang, Jui-Hui Chung, Haoyu Zhao, Lai Jiang, Yihan
 662 Geng, Jiawei Ge, Jingruo Sun, Jiayun Wu, Jiri Gesi, Ximing Lu, David Acuna, Kaiyu Yang,
 663 Hongzhou Lin, Yejin Choi, Danqi Chen, Sanjeev Arora, and Chi Jin. Goedel-prover-v2: Scaling
 664 formal theorem proving with scaffolded data synthesis and self-correction, 2025. URL <https://arxiv.org/abs/2508.03613>.

665

666 Kaiyuan Liu, Chen Shen, Zhanwei Zhang, Junjie Liu, Xiaosong Yuan, et al. Efficient reasoning
 667 through suppression of self-affirmation reflections in large reasoning models. *arXiv preprint
 arXiv:2506.12353*, 2025a.

668

669 Wei Liu, Ruochen Zhou, Yiyun Deng, Yuzhen Huang, Junteng Liu, Yuntian Deng, Yizhe Zhang,
 670 and Junxian He. Learn to reason efficiently with adaptive length-based reward shaping. *arXiv
 preprint arXiv:2505.15612*, 2025b.

671

672 Haotian Luo, Li Shen, Haiying He, Yibo Wang, Shiwei Liu, Wei Li, Naiqiang Tan, Xiaochun Cao,
 673 and Dacheng Tao. O1-Pruner: Length-Harmonizing Fine-Tuning for O1-Like Reasoning Pruning.
 674 URL <http://arxiv.org/abs/2501.12570>.

675

676 Haotian Luo, Li Shen, Haiying He, Yibo Wang, Shiwei Liu, Wei Li, Naiqiang Tan, Xiaochun Cao,
 677 and Dacheng Tao. O1-pruner: Length-harmonizing fine-tuning for o1-like reasoning pruning.
 678 *arXiv preprint arXiv:2501.12570*, 2025a.

679

680 Michael Luo, Sijun Tan, Justin Wong, Xiaoxiang Shi, William Y. Tang, Manan Roongta, Colin Cai,
 681 Jeffrey Luo, Li Erran Li, Raluca Ada Popa, and Ion Stoica. Deepscaler: Surpassing o1-preview
 682 with a 1.5b model by scaling rl, 2025b. Notion Blog.

683

684 Michael Luo, Sijun Tan, Justin Wong, Xiaoxiang Shi, William Y Tang, Manan Roongta, Colin Cai,
 685 Jeffrey Luo, Tianjun Zhang, Li Erran Li, et al. Deepscaler: Surpassing o1-preview with a 1.5 b
 686 model by scaling rl. *Notion Blog*, 2025c.

687

688 Chengqi Lyu, Songyang Gao, Yuzhe Gu, Wenwei Zhang, Jianfei Gao, Kuikun Liu, Ziyi Wang,
 689 Shuaibin Li, Qian Zhao, Haian Huang, et al. Exploring the limit of outcome reward for learning
 690 mathematical reasoning. *arXiv preprint arXiv:2502.06781*, 2025.

691

692 Yingqian Min, Zhipeng Chen, Jinhao Jiang, Jie Chen, Jia Deng, Yiwen Hu, Yiru Tang, Jiapeng
 693 Wang, Xiaoxue Cheng, Huatong Song, et al. Imitate, explore, and self-improve: A reproduction
 694 report on slow-thinking reasoning systems. *arXiv preprint arXiv:2412.09413*, 2024.

695

696 Tergel Munkhbat, Namgyu Ho, Seo Hyun Kim, Yongjin Yang, Yujin Kim, and Se-Young Yun. Self-
 697 training elicits concise reasoning in large language models. *arXiv preprint arXiv:2502.20122*,
 698 2025.

699

700 Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong
 701 Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to fol-
 702 low instructions with human feedback. *Advances in neural information processing systems*, 35:
 703 27730–27744, 2022.

704

705 Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea
 706 Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances
 707 in neural information processing systems*, 36:53728–53741, 2023.

702 John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region
 703 policy optimization. In *International conference on machine learning*, pp. 1889–1897. PMLR,
 704 2015.

705 Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang,
 706 Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathemati-
 707 cal reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

708 Xuerui Su, Shufang Xie, Guoqing Liu, Yingce Xia, Renqian Luo, Peiran Jin, Zhiming Ma, Yue
 709 Wang, Zun Wang, and Yuting Liu. Trust region preference approximation: A simple and stable
 710 reinforcement learning algorithm for llm reasoning. *arXiv preprint arXiv:2504.04524*, 2025.

711 Yang Sui, Yu-Neng Chuang, Guanchu Wang, Jiamu Zhang, Tianyi Zhang, Jiayi Yuan, Hongyi Liu,
 712 Andrew Wen, Shaochen Zhong, Na Zou, et al. Stop overthinking: A survey on efficient reasoning
 713 for large language models. *arXiv preprint arXiv:2503.16419*, 2025.

714 5 Team, Aohan Zeng, Xin Lv, Qinkai Zheng, Zhenyu Hou, Bin Chen, Chengxing Xie, Cunxiang
 715 Wang, Da Yin, Hao Zeng, Jiajie Zhang, Kedong Wang, Lucen Zhong, Mingdao Liu, Rui Lu,
 716 Shulin Cao, Xiaohan Zhang, Xuancheng Huang, Yao Wei, Yean Cheng, Yifan An, Yilin Niu,
 717 Yuanhao Wen, Yushi Bai, Zhengxiao Du, Zihan Wang, Zilin Zhu, Bohan Zhang, Bosi Wen,
 718 Bowen Wu, Bowen Xu, Can Huang, Casey Zhao, Changpeng Cai, Chao Yu, Chen Li, Chendi
 719 Ge, Chenghua Huang, Chenhui Zhang, Chenxi Xu, Chenzheng Zhu, Chuang Li, Congfeng Yin,
 720 Daoyan Lin, Dayong Yang, Dazhi Jiang, Ding Ai, Erle Zhu, Fei Wang, Gengzheng Pan, Guo
 721 Wang, Hailong Sun, Haitao Li, Haiyang Li, Haiyi Hu, Hanyu Zhang, Hao Peng, Hao Tai, Haoke
 722 Zhang, Haoran Wang, Haoyu Yang, He Liu, He Zhao, Hongwei Liu, Hongxi Yan, Huan Liu, Hui-
 723 long Chen, Ji Li, Jiajing Zhao, Jiamin Ren, Jian Jiao, Jiani Zhao, Jianyang Yan, Jiaqi Wang, Jiayi
 724 Gui, Jiayue Zhao, Jie Liu, Jijie Li, Jing Li, Jing Lu, Jingsen Wang, Jingwei Yuan, Jingxuan Li,
 725 Jingzhao Du, Jinhua Du, Jinxin Liu, Junkai Zhi, Junli Gao, Ke Wang, Lekang Yang, Liang Xu, Lin
 726 Fan, Lindong Wu, Lintao Ding, Lu Wang, Man Zhang, Minghao Li, Minghuan Xu, Mingming
 727 Zhao, Mingshu Zhai, Pengfan Du, Qian Dong, Shangde Lei, Shangqing Tu, Shangtong Yang,
 728 Shaoyou Lu, Shijie Li, Shuang Li, Shuang-Li, Shuxun Yang, Sibo Yi, Tianshu Yu, Wei Tian,
 729 Weihan Wang, Wenbo Yu, Weng Lam Tam, Wenjie Liang, Wentao Liu, Xiao Wang, Xiaohan Jia,
 730 Xiaotao Gu, Xiaoying Ling, Xin Wang, Xing Fan, Xingru Pan, Xinyuan Zhang, Xinze Zhang,
 731 Xiuqing Fu, Xunkai Zhang, Yabo Xu, Yandong Wu, Yida Lu, Yidong Wang, Yilin Zhou, Yiming
 732 Pan, Ying Zhang, Yingli Wang, Yingru Li, Yinpei Su, Yipeng Geng, Yitong Zhu, Yongkun Yang,
 733 Yuhang Li, Yuhao Wu, Yujiang Li, Yunan Liu, Yunqing Wang, Yuntao Li, Yuxuan Zhang, Zezhen
 734 Liu, Zhen Yang, Zhengda Zhou, Zhongpei Qiao, Zhuoer Feng, Zhuorui Liu, Zichen Zhang, Zi-
 735 han Wang, Zijun Yao, Zikang Wang, Ziqiang Liu, Ziwei Chai, Zixuan Li, Zuodong Zhao, Wen-
 736 guang Chen, Jidong Zhai, Bin Xu, Minlie Huang, Hongning Wang, Juanzi Li, Yuxiao Dong,
 737 and Jie Tang. Glm-4.5: Agentic, reasoning, and coding (arc) foundation models, 2025a. URL
 738 <https://arxiv.org/abs/2508.06471>.

739
 740 Kimi Team, Angang Du, Bofei Gao, Bowei Xing, Changjiu Jiang, Cheng Chen, Cheng Li, Chenjun
 741 Xiao, Chenzhuang Du, Chonghua Liao, et al. Kimi k1.5: Scaling reinforcement learning with
 742 llms. *arXiv preprint arXiv:2501.12599*, 2025b.

743 Chenlong Wang, Yuanning Feng, Dongping Chen, Zhaoyang Chu, Ranjay Krishna, and Tianyi
 744 Zhou. Wait, we don't need to "wait"! removing thinking tokens improves reasoning efficiency.
 745 *arXiv preprint arXiv:2506.08343*, 2025a.

746
 747 Yibo Wang, Li Shen, Huanjin Yao, Tiansheng Huang, Rui Liu, Naiqiang Tan, Jiaxing Huang, Kai
 748 Zhang, and Dacheng Tao. R1-compress: Long chain-of-thought compression via chunk compres-
 749 sion and search. *arXiv preprint arXiv:2505.16838*, 2025b.

750
 751 Yue Wang, Qiuwei Liu, Jiahao Xu, Tian Liang, Xingyu Chen, Zhiwei He, Linfeng Song, Dian Yu,
 752 Juntao Li, Zhuosheng Zhang, et al. Thoughts are all over the place: On the underthinking of
 753 o1-like llms. *arXiv preprint arXiv:2501.18585*, 2025c.

754
 755 Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny
 Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in
 neural information processing systems*, 35:24824–24837, 2022.

756 Liang Wen, Yunke Cai, Fenrui Xiao, Xin He, Qi An, Zhenyu Duan, Yimin Du, Junchen Liu, Lifu
 757 Tang, Xiaowei Lv, et al. Light-r1: Curriculum sft, dpo and rl for long cot from scratch and beyond.
 758 *arXiv preprint arXiv:2503.10460*, 2025.

759

760 Heming Xia, Chak Tou Leong, Wenjie Wang, Yongqi Li, and Wenjie Li. Tokenskip: Controllable
 761 chain-of-thought compression in llms. *arXiv preprint arXiv:2502.12067*, 2025.

762

763 Violet Xiang, Chase Blagden, Rafael Rafailov, Nathan Lile, Sang Truong, Chelsea Finn, and Nick
 764 Haber. Just Enough Thinking: Efficient Reasoning with Adaptive Length Penalties Reinforcement
 765 Learning. URL <http://arxiv.org/abs/2506.05256>.

766

767 Yang Xiao, Jiashuo Wang, Ruiyuan Yuan, Chunpu Xu, Kaishuai Xu, Wenjie Li, and Pengfei Liu.
 768 Limopro: Reasoning refinement for efficient and effective test-time scaling. *arXiv preprint
 769 arXiv:2505.19187*, 2025.

770

771 Chulin Xie, Yangsibo Huang, Chiyuan Zhang, Da Yu, Xinyun Chen, Bill Yuchen Lin, Bo Li, Badih
 772 Ghazi, and Ravi Kumar. On memorization of large language models in logical reasoning. *arXiv
 773 preprint arXiv:2410.23123*, 2024.

774

775 Tian Xie, Zitian Gao, Qingnan Ren, Haoming Luo, Yuqian Hong, Bryan Dai, Joey Zhou, Kai Qiu,
 776 Zhirong Wu, and Chong Luo. Logic-rl: Unleashing llm reasoning with rule-based reinforcement
 777 learning. *arXiv preprint arXiv:2502.14768*, 2025.

778

779 Yuhui Xu, Hanze Dong, Lei Wang, Doyen Sahoo, Junnan Li, and Caiming Xiong. Scalable chain of
 780 thoughts via elastic reasoning. *arXiv preprint arXiv:2505.05315*, 2025.

781

782 An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang
 783 Gao, Chengan Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu,
 784 Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin
 785 Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang,
 786 Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui
 787 Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang
 788 Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yingqi
 789 Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan
 790 Qiu. Qwen3 technical report, 2025a. URL <https://arxiv.org/abs/2505.09388>.

791

792 Chenxu Yang, Qingyi Si, Yongjie Duan, Zheliang Zhu, Chenyu Zhu, Qiaowei Li, Zheng Lin, Li Cao,
 793 and Weiping Wang. Dynamic early exit in reasoning models. *arXiv preprint arXiv:2504.15895*,
 794 2025b.

795

796 Tianbao Yang and Yiming Ying. Auc maximization in the era of big data and ai: A survey. *ACM
 797 computing surveys*, 55(8):1–37, 2022.

798

799 Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik R
 800 Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. In
 801 *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL <https://openreview.net/forum?id=5Xc1ecx01h>.

802

803 Jingyang Yi, Jiazheng Wang, and Sida Li. Shorterbetter: Guiding reasoning models to find optimal
 804 inference length for efficient reasoning. *arXiv preprint arXiv:2504.21370*, 2025.

805

806 Xixian Yong, Xiao Zhou, Yingying Zhang, Jinlin Li, Yefeng Zheng, and Xian Wu. Think or not?
 807 exploring thinking efficiency in large reasoning models via an information-theoretic lens. *arXiv
 808 preprint arXiv:2505.18237*, 2025.

809

810 Qiyi Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Tiantian Fan, Gaohong
 811 Liu, Lingjun Liu, Xin Liu, et al. Dapo: An open-source llm reinforcement learning system at
 812 scale. *arXiv preprint arXiv:2503.14476*, 2025.

813

814 Hang Yuan, Bin Yu, Haotian Li, Shijun Yang, Christina Dan Wang, Zhou Yu, Xueyin Xu, Weizhen
 815 Qi, and Kai Chen. Not all tokens are what you need in thinking. *arXiv preprint arXiv:2505.17827*,
 816 2025.

810 Linan Yue, Yichao Du, Yizhi Wang, Weibo Gao, Fangzhou Yao, Li Wang, Ye Liu, Ziyu Xu, Qi Liu,
811 Shimin Di, et al. Don't overthink it: A survey of efficient r1-style large reasoning models. *arXiv*
812 *preprint arXiv:2508.02120*, 2025.

813

814 Chong Zhang, Yue Deng, Xiang Lin, Bin Wang, Dianwen Ng, Hai Ye, Xingxuan Li, Yao Xiao,
815 Zhanfeng Mo, Qi Zhang, et al. 100 days after deepseek-r1: A survey on replication studies and
816 more directions for reasoning language models. *arXiv preprint arXiv:2505.00551*, 2025.

817

818 Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang,
819 Yuqiong Liu, Rui Men, An Yang, et al. Group sequence policy optimization. *arXiv preprint*
820 *arXiv:2507.18071*, 2025.

821

822 Ren Zhuang, Ben Wang, and Shuifa Sun. Accelerating chain-of-thought reasoning: When goal-
823 gradient importance meets dynamic skipping. *arXiv preprint arXiv:2505.08392*, 2025.

824

825 Daniel M Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B Brown, Alec Radford, Dario Amodei, Paul
826 Christiano, and Geoffrey Irving. Fine-tuning language models from human preferences. *arXiv*
827 *preprint arXiv:1909.08593*, 2019.

828

829

830

831

832

833

834

835

836

837

838

839

840

841

842

843

844

845

846

847

848

849

850

851

852

853

854

855

856

857

858

859

860

861

862

863

864 **A APPENDIX**865
866 **A.1 DERIVING THE OPTIMUM OF THE KL-CONSTRAINED REWARD MAXIMIZATION**
867 **OBJECTIVE**
868869 In this part, we derive the optimal policy P_q^* that maximizes the length reward with a KL regularization:
870
871

872
$$\max_{P \in \mathcal{P}} \mathbb{E}_{o \sim P} r_l(o) - \lambda \mathbb{D}_{\text{KL}}(P, \pi_{\text{old}}^+(\cdot|q)), \quad (8)$$

873 where $\lambda > 0$ is a regularization parameter, $r_l(o)$ denotes length reward of response o , \mathcal{P} denotes
874 the set of all probability measures P on correct data given q , which are absolutely continuous with
875 respect to $\pi_{\text{old}}^+(\cdot|q)$, i.e., $\pi_{\text{old}}^+(o|q) = 0$ indicates $P(o) = 0$.
876877 Following prior work Rafailov et al. (2023); Go et al. (2023), we have
878

879
$$\begin{aligned} & \max_{P \in \mathcal{P}} \mathbb{E}_{o \sim P} r_l(o) - \lambda \mathbb{D}_{\text{KL}}(P, \pi_{\text{old}}^+(\cdot|q)) \\ &= \max_{P \in \mathcal{P}} \mathbb{E}_{o \sim P} r_l(o) - \lambda \mathbb{E}_{o \sim P} \log \frac{P(o|q)}{\pi_{\text{old}}^+(o|q)} \\ &= \max_{P \in \mathcal{P}} \mathbb{E}_{o \sim P} \left(r_l(o) - \lambda \log \frac{P(o|q)}{\pi_{\text{old}}^+(o|q)} \right) \\ &= \min_{P \in \mathcal{P}} \mathbb{E}_{o \sim P} \left(\log \frac{P(o|q)}{\pi_{\text{old}}^+(o|q)} - r_l(o)/\lambda \right) \\ &= \min_{P \in \mathcal{P}} \mathbb{E}_{o \sim P} \left(\log \frac{P(o|q)}{\frac{1}{Z(q)} \pi_{\text{old}}^+(o|q) \exp(r_l(o)/\lambda)} - \log Z(q) \right) \end{aligned} \quad (9)$$

889 where $Z(q) = \sum_o \pi_{\text{old}}^+(o|q) \exp(r_l(o)/\lambda)$ is a partition function, which doesn't depend on P .
890891 Let's first define $\bar{P}(o|q) = \frac{1}{Z(q)} \pi_{\text{old}}^+(o|q) \exp(r_l(o)/\lambda)$. Since $\bar{P}(o|q) \geq 0$ for all o and
892 $\sum_o \bar{P}(o|q) = 1$, $\bar{P}(o|q)$ is a valid probability distribution. Thus, we can reformulate (9) as:
893

894
$$\begin{aligned} & \min_{P \in \mathcal{P}} \mathbb{E}_{o \sim P} \left(\log \frac{P(o|q)}{\frac{1}{Z(q)} \pi_{\text{old}}^+(o|q) \exp(r_l(o)/\lambda)} - \log Z(q) \right) \\ &= \min_{P \in \mathcal{P}} \left(\mathbb{D}_{\text{KL}}(P, \bar{P}) - \log Z(q) \right) \end{aligned} \quad (10)$$

898 Since $Z(q)$ doesn't depend on P , the minimum of (10) is achieved by minimizing the first KL term.
899 With Gibbs' inequality that KL-divergence is minimized at 0 if and only if the two distributions are
900 identical. Therefore, we have the optimal solution:
901

902
$$P_q^*(o) = \bar{P}(o|q) = \frac{1}{Z(q)} \pi_{\text{old}}^+(o|q) \exp(r_l(o)/\lambda) \quad (11)$$

903
$$= \frac{\pi_{\text{old}}^+(o|q) \exp(r_l(o)/\lambda)}{\mathbb{E}_{o \sim \pi_{\text{old}}^+(\cdot|q)} \exp(r_l(o)/\lambda)}. \quad (12)$$

904 **A.2 DETAILED HYPERPARAMETER SETTING**
905906 In this section, we provide detailed hyperparameter settings used in our experiments.
907908 For all the methods, we employ the AdamW optimizer with a weight decay of 0.01 and set constant
909 learning rate to $2e^{-6}$ for 1.5B model and $1e^{-6}$ for 7B model, following Li et al. (2025a). We set the
910 batch size to 128 for each step of RL, the mini-batch size to 32 for each iteration of model update,
911 and sample 8 responses per question for training. The generation budget is limited to 8k tokens for
912 both training and evaluation. The temperature is set to 0.6 for training.
913914 For RLOO-LP, we use RLOO advantage estimator and clip ratio $\epsilon = 0.2$. We tune their weight
915 parameter $\alpha \in \{0.05, 0.1, 0.2\}$. Following their paper, we normalize its loss by the length of the
916 response.
917

918 Table 2: Various reward designs for efficient reasoning, where $r_c(o|q) = \mathbb{I}(o \text{ is correct}) \in \{1, 0\}$
919 denotes correctness reward.

Method	$r(o, q)$
RLOO-LP (Arora & Zanette)	$r_c(o, q) - \alpha * r_c(o, q) * \sigma(\frac{ o - \text{mean}\{ o_i , r_c(o_i, q) = 1\}}{\text{std}\{ o_i , r_c(o_i, q) = 1\}})$
ALP (Xiang et al.)	$r_c(o, q) - \beta * o * \max(\text{mean}\{r_c(o_i, q)\}, K^{-1})$
HAPO (Huang et al.)	$r_c(o, q) + w * \max(\cos(\min(\frac{\pi}{2} \frac{ o }{h(q)}, \pi)), c) r_c(o, q) + w * \min(\cos(\min(\frac{\pi}{2} \frac{ o }{h(q)}, \pi)), 0)(1 - r_c(o, q))$
L1-MAX (Aggarwal & Welleck)	$r_c(o, q) * \text{clip}(\alpha(L_T - o) + \delta, 0, 1)$
SB (Yi et al., 2025)	$\alpha * r_c(o, q) - \beta * \text{abs}(o - L_{SOL}(q)),$ where $L_{SOL}(q) = \min\{ o_i , r_c(o_i, q) = 1\}$ if at least one response is correct, $\text{mean}\{ o_i \}$ otherwise
LASER-D (Liu et al., 2025b)	$r_c(o, q) + r_c(o, q) * \alpha \mathbb{I}(o \leq L_A)$

929 Table 3: Illustration of negative learning signal of correct outputs of existing reward designs.

Method	Hyperparameter	Length	Correctness	Reward	Advantage
RLOO-LP	$\alpha=0.4$	[1500, 1200, 1900, 2200, 2800, 2000, 3600, 6400, 1300, 1200]	[1, 1, 1, 1, 1, 1, 1, 0, 0]	[0.87, 0.89, 0.85, 0.83, 0.79, 0.84, 0.74, 0.63, 0, 0]	[0.25, 0.27, 0.23, 0.21, 0.16, 0.22, 0.11, -0.02 , -0.72, -0.72]
ALP	$\beta=0.0001$	[1500, 1200, 1900, 2200, 2800, 2000, 3600, 6400, 1300, 1200]	[1, 1, 1, 1, 1, 1, 1, 0, 0]	[0.88, 0.9, 0.85, 0.82, 0.78, 0.84, 0.71, 0.49, -0.1, -0.1]	[0.74, 0.8, 0.65, 0.58, 0.46, 0.63, 0.28, -0.32 , -1.92, -1.9]
HAPO	$w=1, c=-0.7, h=1200$	[1500, 1200, 1900, 2200, 2800, 2000, 3600, 6400, 1300, 1200]	[1, 1, 1, 1, 1, 1, 1, 1, 0, 0]	[0.62, 1, 0.3, 0.3, 0.3, 0.3, 0.3, 0.3, -0.13, -0.1]	[0.99, 2.29, -0.1 , -0.1 , -0.1 , -0.1 , -0.1, -1.57, -1.12]
L1-MAX	$\alpha=0.0003, L_T=4000, \delta=0.5$	[1500, 1200, 1900, 2200, 2800, 2000, 3600, 6400, 1300, 1200]	[1, 1, 1, 1, 1, 1, 1, 1, 0, 0]	[1, 1, 1, 1, 1, 1, 1, 1, 0.86, 0.62, 0, 0, 0]	[0.8, 0.8, 0.8, 0.8, 0.48, 0.8, -0.06 , -1.48 , -1.48]
SB	$\alpha=2, \beta=0.001$	[1500, 1200, 1900, 2200, 2800, 2000, 3600, 6400, 1300, 1200]	[1, 1, 1, 1, 1, 1, 1, 1, 0, 0]	[1.7, 2, 1.3, 1, 0.4, 1.2, -0.4, -3.2, -0.1, 0]	[0.92, 1.14, 0.64, 0.43, 0.01, 0.57, -0.56 , -2.53 , -0.35, -0.28]
LASER-D	$\alpha=0.5, L_A=4000$	[1500, 1200, 1900, 2200, 2800, 2000, 3600, 6400, 1300, 1200]	[1, 1, 1, 1, 1, 1, 1, 1, 0, 0]	[1.5, 1.5, 1.5, 1.5, 1.5, 1.5, 1.5, 1.5, 1.0, 0]	[0.59, 0.59, 0.59, 0.59, 0.59, 0.59, 0.59, 0.59, -0.25 , -1.94, -1.94]

941 For ALP, we follow their paper to use GRPO advantage estimator and normalize its loss by the
942 total number of tokens. The KL Coefficient is set to 0.001. We tune their penalty weight $\beta \in$
943 $\{1e^{-9}, 1e^{-8}, 1e^{-7}\}$ and and clip ratio $\epsilon = 0.2$

944 For HAPO, we follow their paper to use GRPO advantage estimator, set the cutoff $c = -0.7$, KL
945 Coefficient to 0, and clip ratio $\epsilon = 0.2$. The loss is normalized by the length of the response. We
946 tune their weight parameter $w \in \{0.01, 0.1, 1\}$.

947 For the proposed DRPO, we follow Li et al. (2025a) to set constraint value $\delta = 1e^{-4}$, penalty
948 constant $\beta_0 = 1e^3$, $\tau = 10$. we tune regularization parameter $\lambda \in \{0.5, 0.2, 0.1\}$.

951 A.3 LIMITATION OF INCORPORATING LENGTH REWARD WITH GROUP ADVANTAGE

952 In this part, we summarize different reward designs of existing baselines, which incorporate a
953 length reward to encourage efficient reasoning, in Table 2. To illustrate the inherent limitation
954 of incorporating length reward with group advantage, in Table 3, we provide detailed examples
955 of how these reward designs fail to work with group advantage, resulting in misleading learning
956 signals. Specifically, we follow the hyperparameters used in their paper to calculate the ad-
957 vantage with RLOO advantage estimator for RLOO-LP method and GRPO advantage estima-
958 tor (i.e., Eqn. (2)) for other methods. RLOO advantage estimator is calculated as $A(o_i|q) =$
959 $r(o_i|q) - \text{mean}(r(o_1|q), \dots, r(o_{i-1}|q), r(o_{i+1}|q), \dots, r(o_G|q))$. As indicated by red values in Ta-
960 ble 3, all reward designs produce varying amounts of misleading learning signals. We see that HAPO
961 suffers most, yielding incorrect learning directions in 6 out of 10 cases. This could help explain why
962 HAPO exhibits larger performance degradation than other baselines in our experiments.

963 A.4 EXPERIMENTS ON NON-MATHEMATICAL REASONING TASK

964 To evaluate DRPO’s generalization on non-mathematical reasoning tasks, we conducted additional
965 experiments on the logic puzzle reasoning task. Following Xie et al. (2025); Su et al. (2025), the
966 training dataset is limited to 3 to 7-person logic puzzles with K&K logic puzzle dataset (Xie et al.,
967 2024) and the test dataset contains 2 to 8-person puzzles. We train 1.5B models for all methods
968 for 400 steps and conduct evaluation every 80 steps. As shown in Figure 5, DRPO method still
969 exhibits a substantially better trade-off than all other baselines, demonstrating the generalizability
970 of the proposed method to other tasks. Notably, DRPO significantly reduces the generation length
971

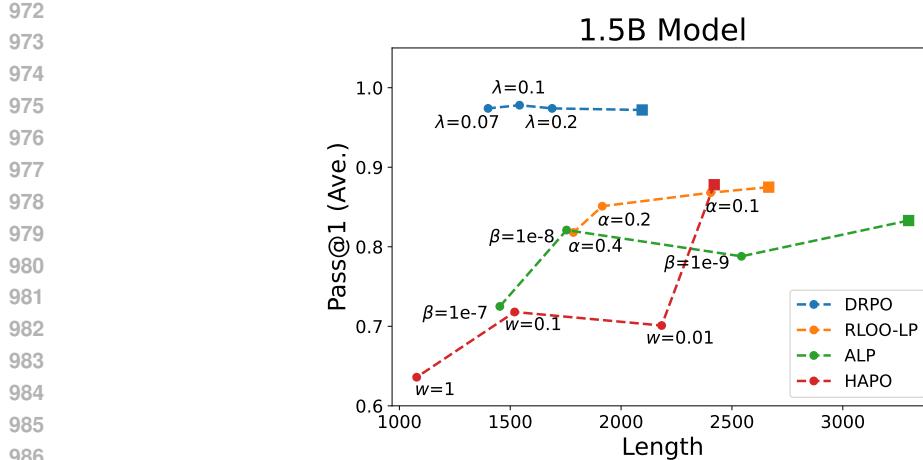


Figure 5: Comparison of performance-efficiency trade-off on logical puzzle reasoning task. Squares denote models trained with reference methods without length penalties, i.e., $\lambda = +\infty$ (corresponding to DisCO) for DRPO, $\alpha = 0$ for RLOO-LP, $\beta = 0$ (corresponding to GRPO) for ALP, $w = 0$ for HAPO. Triangles denote the models trained by other works.

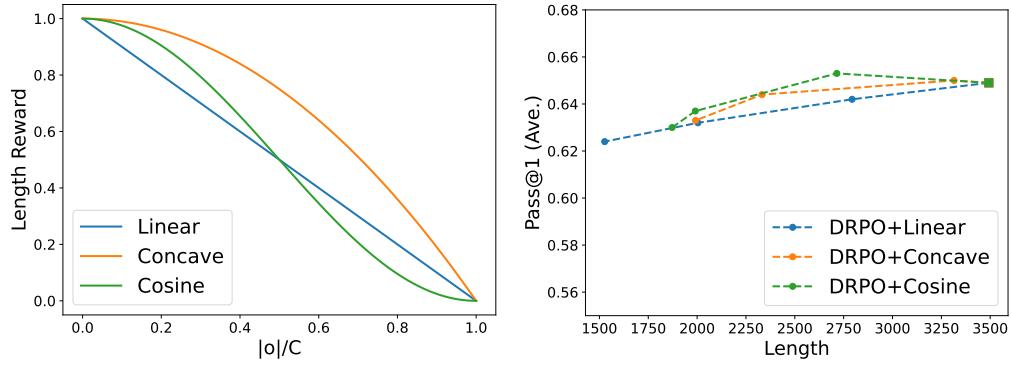


Figure 6: Ablation studies on different reward designs. Left is the length reward values with respect to $|o|/C$; Right is the performance-efficiency trade-offs for different reward designs on mathematical reasoning task. Square denotes the model trained without length penalties, i.e., $\lambda = +\infty$ (DisCO).

from 2095 to 1400 (33.2% length reduction) without any performance drop (from an accuracy of 0.972 to 0.974).

A.5 ABLATION STUDY ON LENGTH REWARD DESIGN

In the main experiments, we adopted a simple length reward $r_l(o) = 1 - \frac{|o|}{C}$, where C is a constant denoting maximum response length. To study the effect of different length reward designs in DRPO, we conduct experiments on mathematical reasoning task with (1) a concave length reward (i.e., $r_l(o) = 1 - (\frac{|o|}{C})^2$) and (2) cosine length reward (i.e., $0.5 + 0.5 \cos(\pi * \frac{|o|}{C})$), using a 1.5B model. For all reward design choices, we tune λ in $\{0.5, 0.2, 0.1\}$. The results are summarized in Figure 6. We observe that all reward designs achieve competitive performance. The linear reward provides the broadest trade-off spectrum, while the concave and cosine rewards tend to yield higher accuracy at the cost of longer reasoning lengths. These findings suggest that developing more sophisticated length rewards is a promising direction for further improving DRPO.

A.6 DETAILED AES PERFORMANCE

In this part, we present detailed AES performance for each method in Table 4, 5, and 6, where bold values denote the best AES performance for each method. We observe that all baseline methods

Table 4: Detailed AES performance for 1.5B models.

Method	GSM8K	MATH500	OlympiadBench	AIME	Ave.
RLOO-LP ($\alpha=0.1$)	0.078	0.021	-0.423	-0.738	-0.172
RLOO-LP ($\alpha=0.2$)	0.246	0.325	-0.279	-0.694	-0.129
RLOO-LP ($\alpha=0.4$)	0.026	-0.13	-0.454	-1.18	-0.412
ALP ($\beta=1e-9$)	0.06	0.016	-0.603	-1.571	-0.387
ALP ($\beta=1e-8$)	-0.189	-0.088	-0.91	-2.229	-0.602
ALP ($\beta=1e-7$)	-1.019	-0.518	-1.887	-3.774	-1.4
HAPO ($w=0.01$)	-0.209	-0.348	-0.879	0.202	-0.519
HAPO ($w=0.1$)	-0.74	-0.811	-1.066	-0.457	-0.969
HAPO ($w=1$)	-1.05	-1.127	-0.841	-0.182	-1.063
DRPO ($\lambda=0.5$)	0.296	0.21	0.152	-0.143	0.093
DRPO ($\lambda=0.2$)	0.614	0.296	0.184	-0.597	0.164
DRPO ($\lambda=0.1$)	0.662	0.332	0.098	-0.729	0.178

Table 5: Detailed AES performance for 7B models.

Method	GSM8K	MATH500	OlympiadBench	AIME	Ave.
RLOO-LP ($\alpha=0.1$)	0.391	0.23	-0.051	-0.532	-0.033
RLOO-LP ($\alpha=0.2$)	0.211	0.284	-0.046	-0.829	-0.122
RLOO-LP ($\alpha=0.4$)	0.045	-0.04	-0.103	-0.709	-0.331
ALP ($\beta=1e-9$)	0.451	0.265	-0.297	-0.984	-0.134
ALP ($\beta=1e-8$)	0.161	-0.351	-1.048	-1.798	-0.68
ALP ($\beta=1e-7$)	-0.923	-1.211	-1.745	-2.679	-1.573
HAPO ($w=0.01$)	-0.105	-1.001	-1.318	-1.676	-1.08
HAPO ($w=0.1$)	-0.999	-1.483	-1.252	-1.197	-1.42
HAPO ($w=1$)	-1.483	-1.407	-1.344	-1.503	-1.6
DRPO ($\lambda=0.5$)	0.053	0.106	-0.126	-0.155	-0.007
DRPO ($\lambda=0.2$)	0.439	0.303	0.015	-0.33	0.115
DRPO ($\lambda=0.1$)	0.672	0.514	0.254	-0.455	0.249

yield negative AES scores for almost all settings, underscoring their inefficiency in preserving performance while reducing reasoning length. In contrast, DRPO consistently achieves positive AES scores for most cases, demonstrating its effectiveness in improving reasoning efficiency without sacrificing performance.

A.7 PERFORMANCE IN TERMS OF PASS@K

In our main experiments, we reported pass@1 averaged over 16 sampled responses for each question to ensure reliability. This evaluation protocol is widely used in recent reasoning works, including those of our baseline methods and DeepSeek-R1 (DeepSeek-AI et al.). Nevertheless, we also include pass@16 as a complementary metric. As shown in Figure 7 with pass@16 metric, DRPO method consistently outperforms all other baselines by a large margin, exhibiting a substantially better trade-off and highlighting the robustness of our approach across metrics.

A.8 CASE STUDY

We analyze the reasoning path of DRPO, compared with DisCO, which corresponds to DRPO with $\lambda = +\infty$. Figure 8 shows the reasoning paths on an easy prompt, where DRPO reaches the correct answer with clear reasoning in only 89 tokens, achieving a 6 \times reduction compared to the 526 tokens required by DisCO. Although DisCO also produces the correct answer, its reasoning is highly re-

Table 6: Detailed AES performance for 8B models.

Method	GSM8K	MATH500	OlympiadBench	AIME	Ave.
RLOO-LP ($\alpha=0.1$)	0.378	0.198	0.282	0.295	0.212
RLOO-LP ($\alpha=0.2$)	0.515	0.203	0.313	0.315	0.251
RLOO-LP ($\alpha=0.4$)	0.064	-0.126	0.127	0.392	-0.119
ALP ($\beta=1e-9$)	0.476	0.215	0.053	-0.67	-0.01
ALP ($\beta=1e-8$)	0.22	-0.252	-0.686	-2.405	-0.584
ALP ($\beta=1e-7$)	-2.139	-1.773	-1.81	-3.733	-2.289
HAPO ($w=0.01$)	-0.021	-0.17	-0.251	-0.755	-0.413
HAPO ($w=0.1$)	0.197	-0.264	-0.445	-0.673	-0.366
HAPO ($w=1$)	-0.23	-0.667	-0.542	-1.579	-0.775
DRPO ($\lambda=0.5$)	0.29	0.172	0.098	0.091	0.122
DRPO ($\lambda=0.2$)	0.607	0.435	0.313	0.068	0.296
DRPO ($\lambda=0.1$)	0.624	0.449	0.292	0.04	0.297

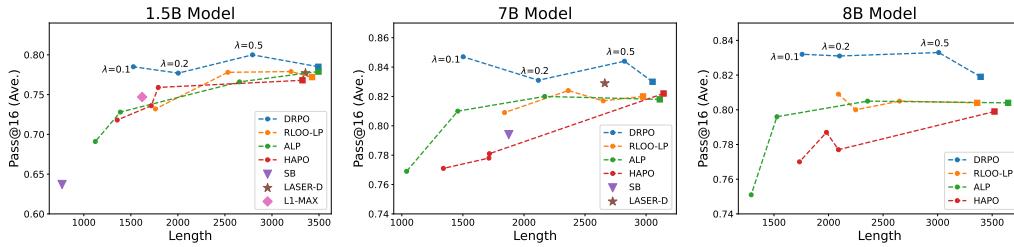


Figure 7: Comparison of performance-efficiency trade-off with pass@16 metric. Left is for fine-tuning 1.5B model, middle is for fine-tuning 7B model and right is for fine-tuning 8B model. Squares denote models trained with reference methods without length penalties, i.e., $\lambda=+\infty$ (corresponding to DisCO) for DRPO, $\alpha = 0$ for RLOO-LP, $\beta = 0$ (corresponding to GRPO) for ALP, $w = 0$ for HAPO. Triangles denote the models trained by other works.

Algorithm 1 Decoupled Reward Policy Optimization (DRPO)

```

1: Input: Initial policy model  $\pi_0$ , reward function  $r$ , question set  $\mathcal{D}$ , hyperparameter  $\delta, \beta, \tau, \lambda$ .
2: Policy model  $\pi_\theta = \pi_0$ 
3: for Step = 1,  $\dots$ ,  $T$  do
4:   Sample a batch of questions  $\mathcal{B}$  from  $\mathcal{D}$ 
5:   Update the old policy model  $\pi_{\text{old}} = \pi_\theta$ 
6:   For each question  $q \in \mathcal{B}$ , sample  $n$  responses  $\{o_i\}_{i=1}^n \sim \pi_{\text{old}}(\cdot|q)$  denoted by  $S_q$  and partition
    it into  $S_q^+$  and  $S_q^-$  based on correctness rewards  $r(o_i|q) \in \{0, 1\}$ 
7:   for minibatch  $\mathcal{B}_m \in \mathcal{B}$  do
8:     Compute KL divergence estimator by

$$\hat{\mathbb{D}}_{KL}(\theta) = \frac{1}{\sum_{q \in \mathcal{B}_m} \sum_{o \in S_q} |o|} \sum_{q \in \mathcal{B}_m} \sum_{o \in S_q} \sum_{t=1}^{|o|} \log \frac{\pi_{\theta, \text{old}}(o_t|q, o_{<t})}{\pi_\theta(o_t|q, o_{<t})}$$

9:     Compute gradient estimator of the objective in Eqn. 7 by

$$G_1 = \frac{1}{|\mathcal{B}_m|} \sum_{q \in \mathcal{B}_m} \left( \sum_{o \in S_q^+} \frac{\exp(r_l(o)/\lambda)}{\sum_{o^* \in S_q^+} \exp(r_l(o^*)/\lambda)} \nabla s_\theta(o, q) - \nabla \left( \tau \log \sum_{o' \in S_q^-} \exp(\frac{s_\theta(o', q)}{\tau}) \right) \right)$$

10:    Compute gradient estimator of a penalty function of the constraint by  $G_2 = 2\beta_0[\hat{\mathbb{D}}_{KL}(\theta) - \delta]_+ \nabla \hat{\mathbb{D}}_{KL}(\theta)$ 
11:    Update  $\pi_\theta$  with Adam-W using the gradient estimator  $G = G_1 + G_2$ 
12:  end for
13: end for

```

1134 dundant, including four occurrences of reflection words (e.g., wait) and three repeated answers. By
1135 contrast, DRPO’s reasoning involves only a single reflection and one correct answer.
1136

1137 We further evaluate the reasoning efficiency on a more challenging prompt in Figure 9 and Figure 10.
1138 As shown in Figure 9, DisCO requires 4497 tokens, about 10 times longer than the 455 tokens of
1139 DRPO, to solve the question. Moreover, the reasoning of DisCO contains 78 reflection words and 19
1140 answers, six of which are incorrect. In comparison, DRPO produces only two correct answers and
1141 five reflection words as seen in Figure 10. It is worth noticing that reasoning from DisCO explicitly
1142 says that ‘but I’m not 100% sure’ before concluding the final answer. It suggests that excessive
1143 back-and-forth in the reasoning, indicated by frequent use of reflection words, may result in less
1144 confident answers.
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187

<p>1188</p> <p>1189</p> <p>1190</p> <p>1191</p> <p>1192</p> <p>1193</p> <p>1194</p> <p>1195</p> <p>1196</p> <p>1197</p> <p>1198</p> <p>1199</p> <p>1200</p> <p>1201</p> <p>1202</p> <p>1203</p> <p>1204</p> <p>1205</p> <p>1206</p> <p>1207</p> <p>1208</p> <p>1209</p> <p>1210</p> <p>1211</p> <p>1212</p> <p>1213</p> <p>1214</p> <p>1215</p> <p>1216</p> <p>1217</p> <p>1218</p> <p>1219</p> <p>1220</p> <p>1221</p> <p>1222</p> <p>1223</p> <p>1224</p> <p>1225</p> <p>1226</p> <p>1227</p> <p>1228</p> <p>1229</p> <p>1230</p> <p>1231</p> <p>1232</p> <p>1233</p> <p>1234</p> <p>1235</p> <p>1236</p> <p>1237</p> <p>1238</p> <p>1239</p> <p>1240</p> <p>1241</p>	<p>Prompt 1</p>	<p>A robe takes 2 bolts of blue fiber and half that much white fiber. How many bolts in total does it take? Let's think step by step and output the final answer within <code>\boxed{}</code>.</p>	<p>Answer from DisCO (DRPO $\lambda = +\infty$) 526 tokens</p> <p>Okay, so I have this problem here: A robe takes 2 bolts of blue fiber and half that much white fiber. I need to figure out how many bolts in total it takes. Hmm, let me break this down step by step.</p> <p>First, the problem says the robe uses 2 bolts of blue fiber. That's straightforward. Now, it also says it takes half that much white fiber. So, I need to find out how much white fiber is that.</p> <p>Let me write this out. Blue fiber is 2 bolts. White fiber is half of that. So, I can calculate the white fiber as:</p> <p>White fiber = $(1/2) * \text{Blue fiber}$ White fiber = $(1/2) * 2$</p> <p>Hmm, let me do that math. Half of 2 is 1, so white fiber is 1 bolt.</p> <p>Now, the total number of bolts is the sum of blue fiber and white fiber. So, that would be:</p> <p>Total bolts = Blue fiber + White fiber Total bolts = $2 + 1$</p> <p>Which equals 3. So, the total number of bolts needed is 3.</p> <p>Wait, let me just make sure I didn't skip any steps or make any mistakes. So, blue fiber is 2, white fiber is half of that, so 2 divided by 2 is indeed 1. Adding them together, $2 + 1$ is 3. Yeah, that seems right.</p> <p>I don't think I need to consider any other factors here, like mixing them or something. The problem just says it takes 2 bolts of blue and half that much white. So, 2 blue and 1 white, totaling 3.</p> <p>Wait, another way to think about it: If half that much white is 1 bolt, then 2 blue plus 1 white is 3 total. Yeah, that's correct.</p> <p>I don't think I need to overcomplicate this. It's pretty straightforward. So, the total number of bolts is 3.</p> <p>**Final Answer** The total number of bolts needed is <code>\boxed{3}</code>.</p> <p></think></p> <p>The robe takes 2 bolts of blue fiber. The white fiber required is half of that, which is:</p> <pre> \[\text{White fiber} = \frac{1}{2} \times 2 = 1 \text{bolt} \] </pre> <p>Adding the blue and white fibers together:</p> <pre> \[\text{Total bolts} = 2 + 1 = 3 \] </pre> <p>The total number of bolts needed is <code>\boxed{3}</code>.</p>	<p>Answer from DRPO ($\lambda = 0.1$) 89 tokens</p>	<p>Okay, so I need to figure out how many total bolts a robe takes. It says the robe takes 2 bolts of blue fiber and half that much white fiber. Wait, so blue is 2 bolts, white is half of blue, which is 1 bolt. So total is $2+1=3$ bolts. That seems right.</p> <p></think></p>	<p>The total number of bolts needed is 3.</p>	<p><code>\boxed{3}</code></p>
---	------------------------	--	--	---	---	---	-------------------------------

Figure 8: Example reasoning for Prompt 1 from DisCo (DRPO $\lambda = +\infty$) and DRPO ($\lambda = 0.1$). Words with green color are answers, and with blue color are reflection words. DRPO reaches the correct answer with clear reasoning in only 89 tokens, achieving a 6x reduction compared to the 526 tokens required by DisCO.

1242

Prompt 2

1243

Josh decides to try flipping a house. He buys a house for \$80,000 and then puts in \$50,000 in repairs. This increased the value of the house by 150%. How much profit did he make? Let's think step by step and output the final answer within `\boxed{}`.

1244

Answer from DisCO (DRPO with $\lambda = +\infty$) 4497 tokens

1245

1246

1247

1248

1249

1250

1251

1252

1253

1254

1255

1256

1257

1258

1259

1260

1261

1262

1263

1264

1265

1266

1267

1268

1269

1270

1271

1272

1273

1274

1275

1276

1277

1278

1279

1280

1281

1282

1283

1284

1285

1286

1287

1288

1289

1290

1291

1292

1293

Figure 9: Example reasoning for Prompt 2 from DisCO (DRPO $\lambda = +\infty$). Words in green/red color are correct and incorrect answers, respectively. Blue words are reflection words. The reasoning of DisCO contains 78 reflection words and 19 answers, six of which are incorrect.

1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306

Prompt 2

1307
1308 Josh decides to try flipping a house. He buys a house for \$80,000 and then puts in \$50,000 in repairs. This
1309 increased the value of the house by 150%. How much profit did he make? Let's think step by step and
output the final answer within \boxed{ }.

1310 **Answer to prompt 2 from DRPO ($\lambda = 0.1$) 455 tokens**

1311 Okay, so Josh buys a house for \$80,000 and spends another \$50,000 on repairs. So total cost is 80k +50k=130k.
1312 Then the house's value increased by 150%. **Wait**, increased by 150% means it's now 100%+150%=250% of the
1313 original value.
1314 **Wait**, original cost was \$80,000, but after repairs, the total investment is \$130,000, right? **Wait**, maybe I should
1315 clarify: when they say the house's value increased by 150%, is it based on the original purchase price or the
1316 total cost?
1317 Hmm, usually, such problems consider the increase based on the cost basis. So if the house increased by
1318 150%, that's 150% of the original purchase price, which was \$80,000.
1319 So 150% of 80k is $1.5 \times 80k = 120k$. So the value went up by 120k, so new value is $80k + 120k = 200k$.
1320 **Wait**, but he spent 50k on repairs. So his total investment was 130k, and he sold it for 200k. **So profit is 200k -**
1321 **130k=70k.**
1322 **Wait**, let me double-check:
1323 Purchase price:80k
1324 Repairs:50k
1325 Total cost:130k
1326 Value increased by150%: so 150% of 80k=120k, so new value=80k+120k=200k.
1327 He sold it for 200k, so profit=200k-130k=70k.
1328 Yes, **so profit is \$70,000.**
1329 </think>
1330 Josh made a profit of \$70,000.
1331 The answer is \boxed{70000}.
1332
1333
1334
1335

1336
1337 Figure 10: Example reasoning for Prompt 2 from DRPO ($\lambda = 0.1$). Words in green/red color are
1338 correct and incorrect answers, respectively. Blue words are reflection words. DisCO uses 4497
1339 tokens, about 10 times longer than the 455 tokens of DRPO.

1340
1341
1342
1343
1344
1345
1346
1347
1348
1349