TO THINK OR NOT TO THINK: ALLOCATING REASONING WHERE IT HELPS

Anonymous authors
Paper under double-blind review

ABSTRACT

Reinforcement learning (RL) has proven effective in enhancing the reasoning performance of large language models (LLMs), particularly in complex mathematical and programming tasks. However, this capability comes with systematic *length misallocation*, in which models devote excessive reasoning to simple questions while terminating prematurely on harder ones, degrading inference efficiency with negligible accuracy improvement. Many length-adaptive methods mitigate this issue by allocating token budgets according to question difficulty, under the implicit assumption that harder questions benefit monotonically from extended reasoning. In contrast, we find that the effect of reasoning length on accuracy is concentrated on *partially solvable* questions. Our further analysis reveals that explicit length rewards can produce unintended training dynamics. Motivated by these findings, we propose **CARE**—**C**ontrastive **A**ccuracy **R**eward **E**stimation—which compares the beneficial length adjustment per question from online sampled responses and applies adaptive length rewards within Group Relative Policy Optimization, with no extra hyperparameters or additional inference cost. Experiments across multiple reasoning benchmarks demonstrate that our method improves Pass@1 by up to 4% while simultaneously reducing reasoning length by 37%, achieving higher token efficiency. Code will be available upon the acceptance of this paper.

1 INTRODUCTION

Reinforcement learning (RL) has achieved remarkable success in enhancing the reasoning capabilities of large language models (LLMs), as exemplified by OpenAI-o1 (Jaech et al., 2024), DeepSeek-R1 (Guo et al., 2025), and Kimi-1.5 (Team et al., 2025). Building on advances in chain-of-thought prompting (Wei et al., 2022; Li et al., 2024; Merrill & Sabharwal, 2023) and self-improvement via verifiable rewards (Zelikman et al., 2022; Ouyang et al., 2022; Lightman et al., 2023), RL-trained LLMs exhibit sophisticated reasoning behaviors, including self-reflection, verification, and exploration of alternative paths. However, a persistent side-effect has emerged alongside these improvements: *length misallocation*. Specifically, such models systematically overthink simple questions by generating unnecessarily long reasoning traces (Chen et al., 2024), while underthinking harder ones through premature abandonment of promising solution paths (Wang et al., 2025) and reluctance to revise their initial answers (Kumar et al., 2024). These behaviors inflate inference costs without improving accuracy, motivating a growing line of work on reasoning length control.

To mitigate length misallocation, recent studies incorporate explicit length signals into RL training objectives. One line of work targets *static length compression*: Arora & Zanette (2025) adjust rewards as a function of reasoning length; Hou et al. (2025) imposes progressively tightened token limits; Aggarwal & Welleck (2025) optimizes policies under user-specified length budgets; Team et al. (2025) constrains length via a penalty term. A complementary direction pursues *adaptive length allocation*: Shen et al. (2025) and Dai et al. (2025) adapt length penalties using online difficulty signals; Xiang et al. (2025) adjusts length penalties inversely with question difficulty, and other methods (Fang et al., 2025; Zhang et al., 2025) select reasoning modes based on task complexity. Despite their differences, these approaches share an implicit assumption: *the optimal reasoning length scales monotonically with question difficulty, meaning that allocating more tokens to harder questions improves accuracy, whereas compressing easier ones may sacrifice little accuracy.* Notably, Wu et al. (2025) demonstrate that the relationship between reasoning length and accuracy can be non-monotonic, calling this assumption into question. Nevertheless, it remains unclear how explicit

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

length rewards affect questions of varying difficulty and whether the resulting training dynamics remain stable over the course of training. We investigate both questions in Section 4.

Our experiments on the Qwen3 family models (Yang et al., 2025) reveal that the effect of reasoning length on accuracy is *not* universally present across difficulty levels but is concentrated on partially solvable questions—those whose accuracy falls in $(0.25, 0.75]$, where the model generates correct answers for a proportion of sampled responses. For these questions, accuracy peaks near the mean length of correct responses and degrades when reasoning length is substantially shorter or longer. This selective pattern challenges the implicit assumption in several prior methods. Moreover, we find undesirable behaviors: *short-reward training can induce longer outputs, while long-reward training increases length through cyclic repetition rather than deeper reasoning in later training stages.*

Motivated by these findings, we propose **C**ontrastive **A**ccuracy **R**eward **E**stimation (**CARE**). For each question during training, we estimate whether longer or shorter reasoning is currently more beneficial and use this signal to determine the direction of the length term in the reward. Since this estimate is derived directly from the sampled response group, it incurs no additional inference cost and adapts online as the policy shifts. This design avoids counterproductive length pressure by encouraging extended reasoning where it improves accuracy and promoting conciseness where brevity suffices.

Our main contributions are as follows:

- We demonstrate that reasoning length primarily affects accuracy on partially solvable questions, and that directly applying length rewards can lead to unanticipated training effects, including length inflation and cyclic repetition.
- We propose **CARE**, a lightweight design, which estimates the beneficial length adjustment per question from online sampled responses at no additional inference cost and integrates directly into the GRPO (Shao et al., 2024) framework without modifying the underlying training pipeline.
- We evaluate on Qwen3-1.7B, 4B and 8B across multiple public reasoning benchmarks, and show that our method improves Pass@1 by up to 4 percentage points over GRPO on HMMT 2026, and up to 37% fewer tokens than the base model on average for Qwen3-4B, achieving better token efficiency without introducing additional hyperparameters or inference cost.

2 Related Work

Reinforcement Learning for LLMs Reasoning. RL has demonstrated considerable success in advancing LLM reasoning capability, with notable examples including OpenAI-o1 (Jaech et al., 2024) and DeepSeek-R1 (Guo et al., 2025), which show that RL can elicit self-reflection and verification without relying on human-annotated trajectories. Team et al. (2025) further scales reinforcement learning with verifiable rewards (RLVR) to 128K-token contexts and employs a length penalty to curb verbosity during training. Yu et al. (2025) addresses exploration and training-stability challenges that arise in long-chain RL through Clip-Higher, Dynamic Sampling, and Token-Level policy gradient loss. However, Yue et al. (2025) show that current RLVR primarily improves sampling efficiency over reasoning paths already present in the base model, rather than eliciting new reasoning patterns or capabilities.

Overthinking and Underthinking in RL-Trained Models. Although RL-based training has markedly improved LLMs' mathematical reasoning, it also introduces length misallocation. Chen et al. (2024) show that models frequently overthink simple questions, consuming extra tokens with negligible accuracy improvement. Wang et al. (2025) identify the opposite issue, underthinking, where models abandon promising paths prematurely. Su et al. (2025) demonstrate that both behaviors coexist, with overthinking on easy questions and underthinking on hard ones. Yeo et al. (2025) further find that, without reward shaping, reasoning length can grow uncontrollably during training and eventually degrade performance. Shojaee et al. (2025) report that standard models can surpass reasoning models on easy questions, while both fail on highly complex ones. An et al. (2025) improve efficiency by suppressing harmful reasoning patterns while reinforcing useful ones, and Sui et al. (2025) argue that existing approaches still do not fully resolve the overthinking and underthinking trade-off.

Efficient and Adaptive Reasoning via RL. To address length misallocation, RL-based approaches have developed along two main directions. The first focuses on efficient reasoning: Arora & Zanette (2025) train models to reduce reasoning length under a single scalar control; Hou et al. (2025) imposes progressively stricter token budgets under GRPO (Shao et al., 2024) with zero reward for over-budget outputs, and Aggarwal & Welleck (2025) introduces Length-Controlled Policy Optimization to support smooth accuracy–compute trade-offs under user-specified budgets. The second focuses on question-adaptive reasoning: Shen et al. (2025) estimates question difficulty via a token-length budget metric and adjusts penalties accordingly; Xiang et al. (2025) reallocates compute by scaling length penalties with question solve rates, and Fang et al. (2025) adaptively selects short-form or long-form reasoning based on question complexity. These methods share the assumption that harder questions benefit monotonically from longer reasoning. In contrast, our method makes no prior assumption about the preferred length adjustment based on question difficulty; instead, it estimates the beneficial length adjustment per question from online rollouts and applies length control only when the sampled responses indicate a clear accuracy gap.

3 Preliminary: Group Relative Policy Optimization

GRPO (Shao et al., 2024) is a PPO-based (Schulman et al., 2017) RL algorithm tailored for LLM reasoning that replaces the critic network in standard PPO with group-relative reward normalization. For each question q, GRPO samples a group of G responses $\{o_1, o_2, \ldots, o_G\}$ from the old policy $\pi_{\theta_{\text{old}}}$, and computes a group-normalized advantage for each response:

$$\hat{A}_i = \frac{r(q, o_i) - \mu_r}{\sigma_r}, \quad \mu_r = \frac{1}{G}\sum_{j=1}^{G} r(q, o_j), \quad \sigma_r = \sqrt{\frac{1}{G}\sum_{j=1}^{G} \left(r(q, o_j) - \mu_r\right)^2} \tag{1}$$

where μ_r and σ_r are the group-level mean and standard deviation of rewards, serving as a self-contained baseline in place of the value network required by PPO. The policy is then updated by maximizing the clipped surrogate objective:

$$\mathcal{L}_{\text{GRPO}}(\theta) = \mathbb{E}_{q\sim P(\mathcal{Q}),\, \{o_i\}_{i=1}^{G}\sim\pi_{\theta_{\text{old}}}(\cdot|q)} \left[\frac{1}{G}\sum_{i=1}^{G} \frac{1}{|o_i|}\sum_{t=1}^{|o_i|} \left\{ \min\left[\frac{\pi_\theta(o_{i,t} \mid q, o_{i,<t})}{\pi_{\theta_{\text{old}}}(o_{i,t} \mid q, o_{i,<t})} \hat{A}_i, \right.\right.\right. \\ \left.\left.\left. \text{clip}\left(\frac{\pi_\theta(o_{i,t} \mid q, o_{i,<t})}{\pi_{\theta_{\text{old}}}(o_{i,t} \mid q, o_{i,<t})}, 1-\varepsilon, 1+\varepsilon \right) \hat{A}_i \right] \right\} - \beta\, \mathbb{D}_{\text{KL}}[\pi_\theta \,\|\, \pi_{\text{ref}}] \right] \tag{2}$$

where ε is the clipping threshold, β controls the strength of the KL penalty, $\mathbb{D}_{\text{KL}}[\pi_\theta \| \pi_{\text{ref}}]$ denotes the KL divergence between the current policy and the frozen reference policy π_{ref}, and $P(\mathcal{Q})$ denotes the distribution over the question set $\mathcal{Q}$.

4 The Relationship between Reasoning Length and Accuracy

Before presenting our method, we conduct *motivating experiments* to investigate how reasoning length interacts with accuracy. We first show that reasoning length selectively influences accuracy depending on question difficulty (Section 4.1). We then demonstrate that training with length rewards produces effects misaligned with the conventional difficulty–token allocation assumption (Section 4.2). Finally, we show that length rewards may induce unexpected behaviors in later training stages (Section 4.3).

Setup. All experiments in this section are conducted on the Qwen3 family models (Yang et al., 2025) using mathematical reasoning benchmarks, including GSM8K (Cobbe et al., 2021), MATH500 (Lightman et al., 2023)[1], and MATH (Hendrycks et al., 2021). For difficulty-dependent analyses, we estimate the difficulty of each question as its average accuracy over sampled responses and assign it to a difficulty subset accordingly, partitioning into *hard* $[0, 0.25]$, *partially solvable* $(0.25, 0.75]$, and *easy* $(0.75, 1.0]$. For the training-based studies, we isolate the effect of length control from any particular reward design by adopting two controlled proxies: a *short reward* $r = y - L/L_{\max}$ that penalizes reasoning length to encourage concise outputs, and a *long reward* $r = y + L/L_{\max}$ that incentivizes longer outputs, where $y \in \{0, 1\}$ indicates response correctness and L denotes the

[1]MATH500 consists of 500 problems from the test split of MATH (Hendrycks et al., 2021).

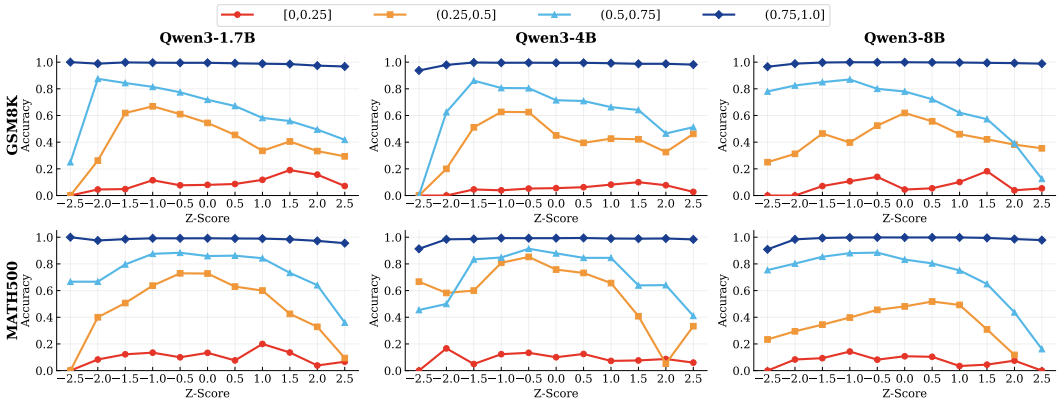


Figure 1: Relationship between reasoning length and accuracy across difficulty groups on GSM8K (top) and MATH500 (bottom) for Qwen3-1.7B, 4B, and 8B. Reasoning lengths are z-score normalized per question using correct-response statistics. Partially solvable questions (Blue/Orange curves) show a clear peak near zero z-score, whereas easy and hard questions show little sensitivity to length.

reasoning length. Subsection-specific experimental protocols and full implementation details are provided in Appendix A; additional results are in Appendices B and C.

4.1 Length Sensitivity Across Difficulty Subsets

We first examine how reasoning length influences accuracy on the base models without any training intervention. For each question, reasoning lengths are z-score normalized using the mean and standard deviation of all the correct samples within that question, and the average accuracy is then computed for each length bin across all questions in each difficulty subset.

As shown in Figure 1, the effect of reasoning length on accuracy is concentrated on partially solvable questions. For easy questions, accuracy remains high across nearly all length z-score intervals, with only a slight drop at large positive z-scores, indicating that performance is largely insensitive to length in this subset. In contrast, for partially solvable questions, accuracy peaks when reasoning lengths are near or slightly below the mean length of correct samples, and declines when responses are substantially shorter or longer. For the hardest questions, accuracy remains uniformly low and shows little sensitivity to reasoning length, implying that allocating additional tokens yields little benefit when the model is already unable to solve them. This pattern suggests that length control strategies should primarily target partially solvable questions, as they represent the subset where reasoning length has the greatest impact on outcomes.

Finding 1

Partially solvable questions are *length-sensitive*: Accuracy peaks around a moderate normalized reasoning length, and both insufficient and excessive length degrade accuracy.

4.2 Accuracy Misalignment of Conventional Token Allocation

Prior work (Shen et al., 2025; Xiang et al., 2025; Fang et al., 2025) implicitly assumes that harder questions generally require longer reasoning length to be solved. While Section 4.1 reveals that reasoning length selectively influences accuracy most on partially solvable questions, it remains unclear whether training with length rewards produces analogous effects. To probe this, we train short-reward and long-reward Qwen3-1.7B variants and evaluate the base, short-reward, and long-reward models on the MATH benchmarks. Questions are grouped into four difficulty subsets by base model accuracy, and each subset is visualized as a density heatmap over the ΔLength–ΔAccuracy grid, where ΔLength and ΔAccuracy denote the per-question change in reasoning length and accuracy relative to the base model, respectively.

Figure 2 shows that short-reward training systematically reduces reasoning length across all four difficulty subsets, yet its effect on accuracy is inconsistent. For the hardest subset, most questions

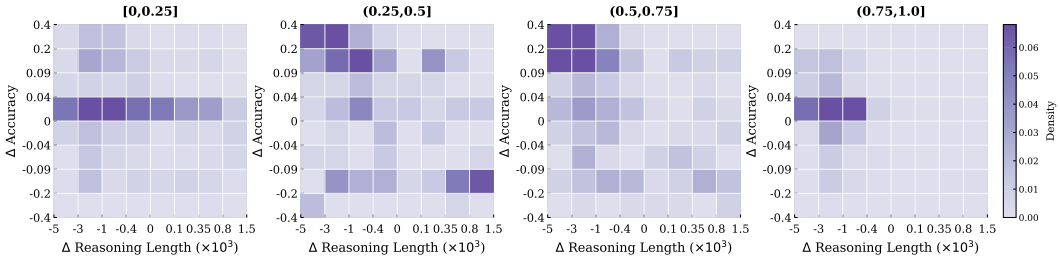


Figure 2: Density heatmaps of Δlength vs. Δaccuracy for short-reward Qwen3-1.7B across four difficulty subsets. A notable fraction of questions in the $(0.25, 0.5]$ and $(0.5, 0.75]$ subsets show improved accuracy, contradicting the assumption that harder questions benefit from additional tokens.

tend to cluster around near-zero or slightly positive ΔAccuracy, suggesting that reducing reasoning length has limited impact on questions that the model already fails to solve. For the easiest subset, the ΔAccuracy remains concentrated near zero with only a slight drop. However, in the (0.25, 0.5] subset questions that the model partially solves, accuracy improves under length reductions. For these borderline-hard questions, the model has sufficient capability to reach the correct answer, whereas longer responses often introduce unnecessary reasoning steps that override the correct trajectory and thereby lower observed accuracy. A similar pattern holds across different model–reward combinations.

Finding 2

The conventional difficulty-token allocation strategy, guided purely by empirical accuracy, may not fully align with where adjusting reasoning length in practice improves accuracy.

4.3 Unexpected Training Effects under Length Rewards

Section 4.2 reveals that the conventional difficulty–token allocation strategy does not adequately match the actual accuracy effects of reasoning length across questions. Beyond this misalignment, we also observe that both long-reward and short-reward training strategies exhibit distinct and unexpected behaviors across different models during the training process.

We train Qwen3-1.7B, Qwen3-4B, and Qwen3-8B, tracking five training dynamics throughout the training process. Taking Qwen3-4B under short-reward training as a representative example, we find that training initially succeeds in reducing reasoning length while maintaining correctness. However, beyond this initial point the training dynamics drift noticeably. Mean reasoning length begins to increase counterintuitively, driven primarily by the growing length of incorrect responses. Specifically, under short-reward training, the model is encouraged to produce shorter outputs for both correct and incorrect answers. This compression signal provides no explicit incentive to preserve progress-making action toward valid solutions. Consequently, we conjecture that continued optimization impairs the model's ability to complete valid solution trajectories, leading to invalid, incoherent, or degenerate responses on an increasing number of questions. This contradiction between the intended objective and the actual observed behavior on questions is what drives the observed degradation. By contrast, long-reward training successfully increases reasoning length. However, comparing model outputs for the same questions across different training stages, we observe that the later increase in reasoning length tends to coincide with a

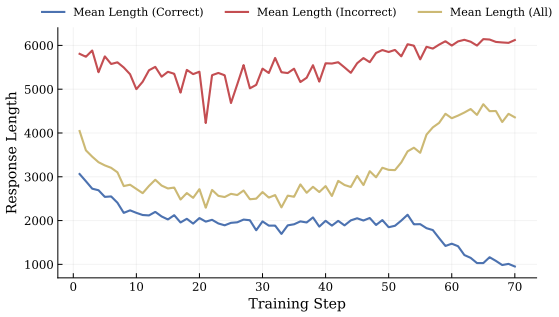


Figure 3: Training trajectory of Qwen3-4B under the short reward.

higher frequency of cyclic actions, suggesting that these additional tokens may reflect repetitive and redundant reasoning patterns rather than deeper or more productive exploration.

5 METHODOLOGY

Section 4 exposes limitations of existing length-reward approaches in mitigating length misallocation. To this end, we estimate whether longer or shorter reasoning is currently more beneficial for each question at each training step. This estimate is then converted into a signed length term within the reward. The resulting reward integrates directly into GRPO without modifying the training pipeline.

5.1 CONTRASTIVE ACCURACY–LENGTH ESTIMATION

Given a question q, we compare whether longer or shorter responses are currently more beneficial by comparing accuracy between the shorter and longer halves of the sampled responses. We sample a group of G responses $\{o_i\}_{i=1}^{G}$ from the old policy $\pi_{\theta_{\text{old}}}$. Let $y_i \in \{0, 1\}$ denote the correctness of response o_i, and let L_i denote its length. We sort the group by length in ascending order:

$$L_{(1)} \leq L_{(2)} \leq \cdots \leq L_{(G)} \tag{3}$$

where (i) denotes the index after sorting. Assuming G is even, we split the sorted responses into a shorter half and a longer half:

$$\mathcal{S}(q) = \{(1), \ldots, (G/2)\}, \qquad \mathcal{L}(q) = \{(G/2+1), \ldots, (G)\} \tag{4}$$

We then compute the average accuracy of each half:

$$\bar{y}_{\mathcal{S}}(q) = \frac{1}{|\mathcal{S}(q)|} \sum_{i \in \mathcal{S}(q)} y_i, \qquad \bar{y}_{\mathcal{L}}(q) = \frac{1}{|\mathcal{L}(q)|} \sum_{i \in \mathcal{L}(q)} y_i \tag{5}$$

Based on this comparison, we define a question-level length adjustment direction:

$$d(q) = \begin{cases} -1, & \bar{y}_{\mathcal{S}}(q) > \bar{y}_{\mathcal{L}}(q) \\ 0, & \bar{y}_{\mathcal{S}}(q) = \bar{y}_{\mathcal{L}}(q) \\ +1, & \bar{y}_{\mathcal{S}}(q) < \bar{y}_{\mathcal{L}}(q) \end{cases} \tag{6}$$

The adjustment direction signal $d(q)$ encodes whether extended reasoning is currently beneficial $(+1)$, redundant (-1), or uninformative (0) for a given question q. A value of $+1$ indicates that longer responses are empirically more accurate; -1 indicates that concise responses outperform and that length compression is appropriate; 0 indicates no clear contrastive signal, in which case no length adjustment is applied. Since $d(q)$ is estimated directly from the sampled response group, it incurs no additional inference cost and naturally adapts as the policy updates during training.

5.2 DIRECTION-CONDITIONED REWARD

We convert the question-level directional signal into a reward by combining answer correctness with a signed length term. For each response o_i, we define

$$r(q, o_i) = y_i + d(q) \cdot \frac{L_i}{L_{\max}} \tag{7}$$

Equivalently,

$$r(q, o_i) = \begin{cases} y_i - \dfrac{L_i}{L_{\max}}, & \bar{y}_{\mathcal{S}}(q) > \bar{y}_{\mathcal{L}}(q) \\ y_i, & \bar{y}_{\mathcal{S}}(q) = \bar{y}_{\mathcal{L}}(q) \\ y_i + \dfrac{L_i}{L_{\max}}, & \bar{y}_{\mathcal{S}}(q) < \bar{y}_{\mathcal{L}}(q) \end{cases} \tag{8}$$

where $y_i \in \{0, 1\}$ denotes the correctness of the generated response o_i, L_i denotes its corresponding length, and $L_{\max}$ is the maximum reasoning length used during training.

The signed length term $d(q) \cdot L_i / L_{\max}$ converts the directional signal into a reward component: it encourages longer outputs when extended reasoning improves accuracy, penalizes length when shorter reasoning suffices, and vanishes when no directional preference is detected. Crucially, the correctness reward y_i remains the principal term, while the length component serves only as a secondary shaping signal, ensuring that the model does not over-optimize length and thereby impair correctness.

5.3 Integration with GRPO

We integrate $r(q, o_i)$ directly into GRPO (Section 3). The group-normalized advantage $\hat{A}_i$ is computed according to Eq. (1), and the policy is updated by maximizing $\mathcal{L}_{\mathrm{GRPO}}(\theta)$ in Eq. (2). No other component of the training pipeline is modified.

In summary, our method assumes no fixed relationship between question difficulty and appropriate reasoning length. Instead, the beneficial length adjustment of a given question q is estimated online from sampled responses at no additional inference cost and without introducing extra hyperparameters. The reward adaptively encourages longer reasoning when it improves accuracy, penalizes it when concise responses suffice, and remains neutral otherwise, thereby keeping length control aligned with what is beneficial for each question throughout training in a lightweight manner.

6 Experiments

To validate the effectiveness of CARE in allocating reasoning length, we conduct experiments on Qwen3-1.7B, 4B and 8B (Yang et al., 2025) across four mathematical reasoning benchmarks under the same training setup, comparing against the base model, GRPO, and two length-control baselines.

6.1 Experimental Setup

Training. All experiments are conducted using the VERL (Sheng et al., 2025) framework.[2] We train Qwen3-1.7B, 4B and 8B on MATH (Hendrycks et al., 2021) with the AdamW (Loshchilov & Hutter, 2017) optimizer at a learning rate of 1×10^{-6} and a batch size of 256 questions per step. For each question, 16 responses are sampled at a temperature of 1.0 and top-p of 1.0, with the maximum reasoning length $6{,}144$ tokens. The model parameters are then updated every 256 sampled responses. The KL penalty coefficient β and clipping threshold ε are set to 0.001 and 0.2, respectively.

Baselines. We compare CARE against four methods: (1) the Qwen3 base model without RL training; (2) GRPO (Shao et al., 2024) with a pure correctness reward $r = y$; and (3) two adaptive length-control baselines, ALP (Xiang et al., 2025) and DAST (Shen et al., 2025).

Evaluation. We evaluate on four mathematical reasoning benchmarks: AIME 2025 (Balunović et al., 2025), AIME 2026 (Balunović et al., 2025), HMMT 2026 (Balunović et al., 2025),[3] and MATH500 (Lightman et al., 2023). For each question, we sample 32 responses with temperature 0.6, top-p 0.95, and a maximum reasoning length of $16{,}384$ tokens. Pass@1 is then adopted to evaluate the model's reasoning capability, computed using the unbiased estimator for Pass@k proposed in Chen et al. (2021), where $k = 1$. We also report reasoning length in tokens. The two metrics reflect whether length misallocation has been effectively mitigated: a method that allocates reasoning length well should achieve higher accuracy without inflating token consumption.

6.2 Result

Table 1 presents the main results across multiple reasoning benchmarks under different reward designs. Experiments show that CARE achieves the best balance between accuracy and reasoning length across multiple benchmarks, attaining higher Pass@1 while avoiding unnecessary token expenditure.

On Qwen3-1.7B, CARE achieves 43.82% average Pass@1 with $8{,}154$ tokens, outperforming GRPO by 2.64% while reducing token usage by 9.9%. Compared with DAST, the improvement reaches

[2] We use VERL version 0.7.0.dev0.

[3] HMMT Feb 2026 is sourced from `https://huggingface.co/datasets/MathArena/hmmt_feb_2026`.

5.23% with 31.7% fewer tokens. The improvement is particularly evident on HMMT 2026, where CARE reaches 21.78% Pass@1, compared with 15.81% for DAST and 17.61% for GRPO.

On Qwen3-4B, CARE outperforms GRPO by 2.62% in Pass@1 with 11.1% fewer tokens, while achieving a 4.68% improvement and 37.7% token reduction over the base model. On AIME 2026, CARE achieves 55.62% Pass@1, exceeding ALP and DAST by 5.20% and 18.85%, respectively.

On Qwen3-8B, CARE maintains a strong overall accuracy–efficiency trade-off, achieving 59.62% average Pass@1 with 7,707 tokens. Compared with ALP, it improves average Pass@1 by 0.83% while reducing token usage by 7.0%. Although competing methods occasionally perform better on individual benchmarks, CARE consistently achieves the highest average Pass@1 with the lowest average token usage across all three model sizes.

Table 1: Results on reasoning benchmarks under different methods. For each benchmark, we report Pass@1 (%) and length (Token).

Method	AIME 2025		AIME 2026		HMMT 2026		MATH500		Average	
	Pass@1	Token	Pass@1	Token	Pass@1	Token	Pass@1	Token	Pass@1	Token
Qwen3-1.7B										
Qwen3-1.7B	26.88	13320.41	30.21	13716.90	16.67	14062.21	86.47	5246.36	40.06	11586.47
GRPO	30.42	10826.55	28.96	10245.45	17.61	11296.01	**87.74**	3829.58	41.18	9049.40
ALP	31.87	11623.52	32.08	11789.75	19.41	12170.17	85.76	3863.09	42.28	9861.63
DAST	26.04	14052.29	29.48	14312.01	15.81	14639.91	83.03	4746.27	38.59	11937.62
CARE	**33.96**	**9352.41**	**32.29**	**10073.94**	**21.78**	**10176.61**	87.25	**3012.70**	**43.82**	**8153.92**
Qwen3-4B										
Qwen3-4B	46.46	12967.93	50.83	12692.93	23.30	13816.37	90.21	4833.99	52.70	11077.81
GRPO	48.85	9334.90	52.60	8657.44	26.04	10112.92	91.54	2962.31	54.76	7766.89
ALP	50.83	9012.51	50.42	8231.33	27.84	9765.89	90.61	**2547.05**	54.93	7389.20
DAST	32.50	13973.33	36.77	14032.09	20.36	14566.25	86.74	4670.34	44.09	11810.50
CARE	**52.92**	**8455.09**	**55.62**	**7817.14**	**28.60**	**8735.88**	**92.39**	2618.05	**57.38**	**6906.54**
Qwen3-8B										
Qwen3-8B	45.73	13176.58	52.19	12623.87	25.66	13925.07	88.07	5073.17	52.91	11199.67
GRPO	**55.42**	10029.52	56.46	9246.88	32.10	11185.89	90.51	3022.00	58.62	8371.07
ALP	54.27	9854.36	57.92	9240.24	32.48	11088.53	90.50	2983.70	58.79	8291.71
DAST	51.67	11288.34	56.56	10643.80	29.45	12502.82	90.30	2965.53	57.00	9350.12
CARE	53.85	**9166.40**	**58.85**	**8548.70**	**32.67**	**10310.60**	**93.10**	**2803.10**	**59.62**	**7707.20**

6.3 Further Analyses

Stability of the Length Preference Signal. To evaluate the stability of CARE's length preference signal under different rollout group sizes, we use $d_{16}(q)$ computed from the full group $g = 16$ as the reference and randomly subsample $g \in \{4, 8, 12\}$ responses without replacement. For each subset, we recompute the short–long split and $d_g(q)$, repeat the procedure 20 times at every training step, and report exact agreement over the first, middle, and final thirds of the training steps, together with non-zero agreement when $d_{16}(q) \neq 0$. As shown in Table 2, both exact agreement and non-zero agreement increase consistently with group size, while the exact agreement remains stable across training stages, indicating that CARE is not sensitive to the rollout group size.

Effect of Accuracy-Gap Magnitude Weighting. We examine whether incorporating the magnitude of the short–long accuracy difference improves CARE by weighting the length term with $|\bar{y}_{\mathcal{L}} - \bar{y}_{\mathcal{S}}|$. We compare this CARE-Magnitude variant with the original CARE across Qwen3-1.7B, 4B and 8B on all four benchmarks, reporting Pass@1 and average reasoning length. Table 3 shows that the magnitude variant achieves slightly higher Pass@1 on some individual benchmarks, but the original CARE consistently achieves better average Pass@1 with fewer tokens across all three model sizes.

Table 2: Stability of CARE's length preference signal under different rollout group sizes. Exact agreement measures consistency between the subsampled signal $d_g(q)$ and the full-group reference $d_{16}(q)$. Non-zero agreement is computed only when $d_{16}(q) \neq 0$.

Model	g	Early Exact	Middle Exact	Late Exact	Overall Exact	Overall Non-Zero
Qwen3-1.7B	4	85.39%	87.19%	87.52%	**86.69%**	50.90%
	8	91.95%	92.81%	92.81%	**92.52%**	73.73%
	12	96.27%	96.23%	96.40%	**96.30%**	**87.91%**
Qwen3-4B	4	89.85%	90.67%	91.62%	**90.69%**	50.94%
	8	94.63%	94.73%	94.99%	**94.78%**	73.82%
	12	97.39%	97.44%	97.40%	**97.41%**	**87.92%**
Qwen3-8B	4	87.73%	91.27%	90.89%	**89.96%**	51.55%
	8	93.50%	95.52%	95.18%	**94.73%**	75.32%
	12	96.94%	98.05%	97.51%	**97.50%**	**88.80%**

Table 3: Ablation on accuracy-gap magnitude weighting. Each entry reports Pass@1 (%) / average response length (Token). CARE-Magnitude weights the length term by $|\bar{y}_{\mathcal{L}} - \bar{y}_{\mathcal{S}}|$.

Model	Method	AIME25	AIME26	HMMT26	MATH500	Average
Qwen3-1.7B	CARE-Magnitude	31.35/10464.91	30.73/10496.70	20.83/11380.66	85.08/3314.91	42.00/8914.30
	CARE	**33.96/9352.41**	**32.29/10073.94**	**21.78/10176.61**	**87.25/3012.70**	**43.82/8153.92**
Qwen3-4B	CARE-Magnitude	**53.02/8843.30**	51.98/8117.22	26.99/9450.59	90.59/2718.12	55.65/7282.31
	CARE	52.92/8455.09	**55.62/7817.14**	**28.60/8735.88**	**92.39/2618.05**	**57.38/6906.54**
Qwen3-8B	CARE-Magnitude	**54.17/9583.34**	**59.06/8874.49**	32.01/10569.64	90.65/2892.74	58.97/7980.05
	CARE	53.85/9166.40	58.85/8548.70	**32.67/10310.60**	**93.10/2803.10**	**59.62/7707.20**

Sensitivity to the Length-Term Coefficient. Although CARE uses a fixed unit coefficient for the length term by default, we introduce λ here solely for sensitivity analysis, where $\lambda = 1$ corresponds to the original CARE. We evaluate a smaller coefficient, $\lambda = 0.5$, across Qwen3-1.7B, Qwen3-4B and Qwen3-8B using the same four benchmarks and evaluation metrics. The results in Table 4 indicate that CARE remains effective with $\lambda = 0.5$, while the default $\lambda = 1$ consistently provides a better average accuracy–efficiency trade-off across all model sizes.

Table 4: Ablation on the length-term coefficient. Each entry reports Pass@1 (%) / average reasoning length (Token). We compare the default CARE setting $\lambda = 1$ with $\lambda = 0.5$.

Model	Method	AIME25	AIME26	HMMT26	MATH500	Average
Qwen3-1.7B	CARE ($\lambda = 0.5$)	32.81/10627.63	30.31/10700.56	20.55/11500.73	85.25/3397.02	42.23/9056.49
	CARE	**33.96/9352.41**	**32.29/10073.94**	**21.78/10176.61**	**87.25/3012.70**	**43.82/8153.92**
Qwen3-4B	CARE ($\lambda = 0.5$)	**53.02/8555.19**	51.25/8218.75	27.27/9179.88	90.84/2742.92	55.60/7174.18
	CARE	52.92/8455.09	**55.62/7817.14**	**28.60/8735.88**	**92.39/2618.05**	**57.38/6906.54**
Qwen3-8B	CARE ($\lambda = 0.5$)	**54.06/9674.28**	56.67/8993.24	**32.95/10813.47**	90.63/3004.62	58.58/8121.40
	CARE	53.85/9166.40	**58.85/8548.70**	32.67/10310.60	**93.10/2803.10**	**59.62/7707.20**

7 Conclusion

In this work, we propose CARE, a lightweight method for reasoning length control in LLMs. Empirical analysis reveals that reasoning length predominantly influences accuracy on partially solvable questions, and that naive length rewards induce detrimental training effects. CARE addresses these by comparing the beneficial length adjustment per question from online sampled responses and integrating it into GRPO at no additional overhead. Experiments demonstrate that CARE improves Pass@1 while reducing reasoning length, exhibits stable length-preference signals, and maintains a strong accuracy–efficiency trade-off under different weighting and coefficient settings.

AI USE STATEMENT

Generative AI tools were used solely for language polishing and to assist with experimental code.

ETHICS STATEMENT

This work studies reasoning efficiency using publicly available mathematical benchmarks and does not involve human subjects or private data.

REPRODUCIBILITY STATEMENT

We provide detailed training and evaluation configurations, model and dataset information, and compute resources in the main paper and appendix. Code and implementation details will be released upon acceptance.

REFERENCES

Pranjal Aggarwal and Sean Welleck. L1: Controlling how long a reasoning model thinks with reinforcement learning. *arXiv preprint arXiv:2503.04697*, 2025.

Sohyun An, Ruochen Wang, Tianyi Zhou, and Cho-Jui Hsieh. Don't think longer, think wisely: Optimizing thinking dynamics for large reasoning models. *arXiv preprint arXiv:2505.21765*, 2025.

Daman Arora and Andrea Zanette. Training language models to reason efficiently. *arXiv preprint arXiv:2502.04463*, 2025.

Mislav Balunović, Jasper Dekoninck, Ivo Petrov, Nikola Jovanović, and Martin Vechev. Matharena: Evaluating llms on uncontaminated math competitions, February 2025. URL `https://matharena.ai/`.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde De Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021.

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

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

Muzhi Dai, Shixuan Liu, and Qingyi Si. Stable reinforcement learning for efficient reasoning. *arXiv preprint arXiv:2505.18086*, 2025.

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

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638, 2025.

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

Bairu Hou, Yang Zhang, Jiabao Ji, Yujian Liu, Kaizhi Qian, Jacob Andreas, and Shiyu Chang. Thinkprune: Pruning long chain-of-thought of llms via reinforcement learning. *arXiv preprint arXiv:2504.01296*, 2025.

Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, et al. Openai o1 system card. *arXiv preprint arXiv:2412.16720*, 2024.

Aviral Kumar, Vincent Zhuang, Rishabh Agarwal, Yi Su, John D Co-Reyes, Avi Singh, Kate Baumli, Shariq Iqbal, Colton Bishop, Rebecca Roelofs, et al. Training language models to self-correct via reinforcement learning. *arXiv preprint arXiv:2409.12917*, 2024.

Zhiyuan Li, Hong Liu, Denny Zhou, and Tengyu Ma. Chain of thought empowers transformers to solve inherently serial problems. In *The Twelfth International Conference on Learning Representations*, 2024.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *The twelfth international conference on learning representations*, 2023.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

William Merrill and Ashish Sabharwal. The expressive power of transformers with chain of thought. *arXiv preprint arXiv:2310.07923*, 2023.

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

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

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

Yi Shen, Jian Zhang, Jieyun Huang, Shuming Shi, Wenjing Zhang, Jiangze Yan, Ning Wang, Kai Wang, Zhaoxiang Liu, and Shiguo Lian. Dast: Difficulty-adaptive slow-thinking for large reasoning models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pp. 2322–2331, 2025.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. Hybridflow: A flexible and efficient rlhf framework. In *Proceedings of the Twentieth European Conference on Computer Systems*, pp. 1279–1297, 2025.

Parshin Shojaee, Iman Mirzadeh, Keivan Alizadeh, Maxwell Horton, Samy Bengio, and Mehrdad Farajtabar. The illusion of thinking: Understanding the strengths and limitations of reasoning models via the lens of problem complexity. *arXiv preprint arXiv:2506.06941*, 2025.

Jinyan Su, Jennifer Healey, Preslav Nakov, and Claire Cardie. Between underthinking and overthinking: An empirical study of reasoning length and correctness in llms. *arXiv preprint arXiv:2505.00127*, 2025.

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

K Team, A Du, B Gao, B Xing, C Jiang, C Chen, C Li, C Xiao, C Du, C Liao, et al. k1. 5: Scaling reinforcement learning with llms, 2025. *URL https://arxiv. org/abs/2501.12599*, 2025.

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

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.

Yuyang Wu, Yifei Wang, Ziyu Ye, Tianqi Du, Stefanie Jegelka, and Yisen Wang. When more is less: Understanding chain-of-thought length in llms. *arXiv preprint arXiv:2502.07266*, 2025.

Violet Xiang, Chase Blagden, Rafael Rafailov, Nathan Lile, Sang Truong, Chelsea Finn, and Nick Haber. Just enough thinking: Efficient reasoning with adaptive length penalties reinforcement learning. *arXiv preprint arXiv:2506.05256*, 2025.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Edward Yeo, Yuxuan Tong, Morry Niu, Graham Neubig, and Xiang Yue. Demystifying long chain-of-thought reasoning in llms. *arXiv preprint arXiv:2502.03373*, 2025.

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

Yang Yue, Zhiqi Chen, Rui Lu, Andrew Zhao, Zhaokai Wang, Shiji Song, and Gao Huang. Does reinforcement learning really incentivize reasoning capacity in llms beyond the base model? *arXiv preprint arXiv:2504.13837*, 2025.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah Goodman. Star: Bootstrapping reasoning with reasoning. *Advances in Neural Information Processing Systems*, 35:15476–15488, 2022.

Jiajie Zhang, Nianyi Lin, Lei Hou, Ling Feng, and Juanzi Li. Adaptthink: Reasoning models can learn when to think. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 3716–3730, 2025.

A Experimental Details for Reasoning Length and Accuracy

The following provides the shared experimental configurations for Section 4.

A.1 Sampling Configuration

The sampling configuration is summarized in Table 5. We use 64 samples per question to obtain a stable per-question difficulty estimate and to ensure sufficient coverage across different reasoning lengths. The bin width of 0.5 is chosen to balance granularity with statistical reliability within each bin. Temperature and top-p are both set to 1.0 to ensure diversity in reasoning lengths.

Table 5: Sampling configuration for difficulty estimation on GSM8K and MATH500.

Inference Engine	Samples per Question	Temperature	Top-p	Max Reasoning Length	Bin Width
vLLM	64	1.0	1.0	16,384	0.5

A.2 Filtering Protocol and Difficulty Composition

For z-score normalization, we exclude questions with fewer than three correct samples. Table 6 reports the number and proportion of retained questions for each model and dataset.

Table 6: Number of questions retained after z-score filtering on GSM8K and MATH500.

	Qwen3-1.7B	Qwen3-4B	Qwen3-8B
GSM8K	1286/1319 (97.50%)	1277/1319 (96.82%)	1292/1319 (97.95%)
MATH500	475/500 (95.00%)	474/500 (94.80%)	479/500 (95.80%)

For completeness, before applying the correct-response filtering criterion, we additionally report the fine-grained composition of the hard and easy difficulty subsets used in Section 4.1. Let p denote the per-question accuracy under the same sampling configuration. The hard difficulty subset $[0, 0.25]$ is divided into $p = 0$ and $0 < p \leq 0.25$, while the easy difficulty subset $(0.75, 1]$ is divided into $0.75 < p < 1$ and $p = 1$.

Table 7: Fine-grained composition of the hard and easy difficulty subsets. Each entry reports the number of questions (fraction within the corresponding interval).

Model	Dataset	Hard $[0, 0.25]$		Easy $(0.75, 1]$	
		$p = 0$	$0 < p \leq 0.25$	$0.75 < p < 1$	$p = 1$
Qwen3-1.7B	GSM8K	33 (30.56%)	75 (**69.44%**)	179 (**15.85%**)	950 (84.15%)
	MATH500	25 (36.76%)	43 (**63.24%**)	112 (**28.28%**)	284 (71.72%)
Qwen3-4B	GSM8K	42 (41.18%)	60 (**58.82%**)	184 (**15.62%**)	994 (84.38%)
	MATH500	26 (40.62%)	38 (**59.38%**)	107 (**25.72%**)	309 (74.28%)
Qwen3-8B	GSM8K	23 (30.26%)	53 (**69.74%**)	70 (**5.75%**)	1148 (94.25%)
	MATH500	21 (40.38%)	31 (**59.62%**)	68 (**16.00%**)	357 (84.00%)

A.3 Length–Accuracy Relationships Across Model Families

We additionally evaluate Gemma-4-E2B-it, Gemma-4-E4B-it, and DeepSeek-R1-Distill-Llama-8B on GSM8K and MATH500, following the same experimental procedure as in Section 4.1. For each model, we sample 64 responses per question and normalize response lengths using the mean and standard deviation of its correct responses. We then report the average accuracy within each normalized length interval for every difficulty subset.

Figure 4 shows that the length–accuracy relationships for the Gemma-4 and DeepSeek-R1-Distill-Llama models are broadly consistent with those observed in Section 4.1, although these patterns are not equally evident across models and difficulty subsets.

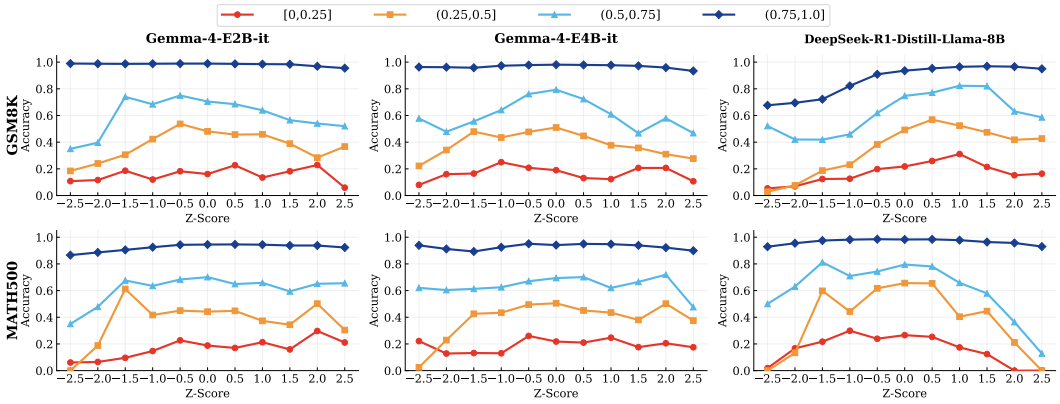


Figure 4: Relationship between reasoning length and accuracy across difficulty groups on GSM8K (top) and MATH500 (bottom) for Gemma-4-E2B-it, Gemma-4-E4B-it, and DeepSeek-R1-Distill-Llama-8B.

A.4 Training Configuration

The training setup is listed in Table 8. We adopt GRPO as the advantage estimator with a sampling group size of 16 per question. The global batch size is set to 256, meaning that 256 distinct questions are sampled at each training step, generating a total of $256 \times 16 = 4{,}096$ responses. The mini batch size is also set to 256, meaning that the model updates its parameters once every 256 collected responses, mitigating the risk of converging to local optima.

Table 8: Hyperparameters for training.

Advantage Estimator	Training Temperature	Global Batch Size	Mini Batch Size	Rollout Number	Regularization	Actor Clip Ratio	Max Response Length	Learning Rate
GRPO	1.0	256	256	16	KL/Entropy	0.2	6144	1e-6

A.5 Evaluation Protocol

The evaluation settings are reported in Table 9. We generate 32 responses per question at temperature 0.6 and top-p 0.95, with the maximum reasoning length set to 16,384 tokens.

Table 9: Evaluation configuration.

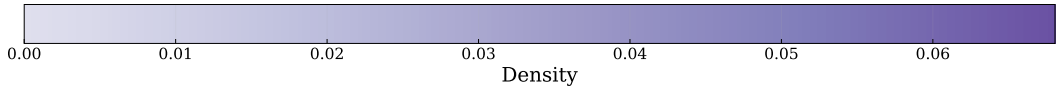
Samples per Question	Temperature	Top-p	Max Reasoning Length	Aggregation
32	0.6	0.95	16,384	Mean over samples

B Accuracy Misalignment Analysis

Here we describe the visualization protocol and present full results for all model–reward combinations.

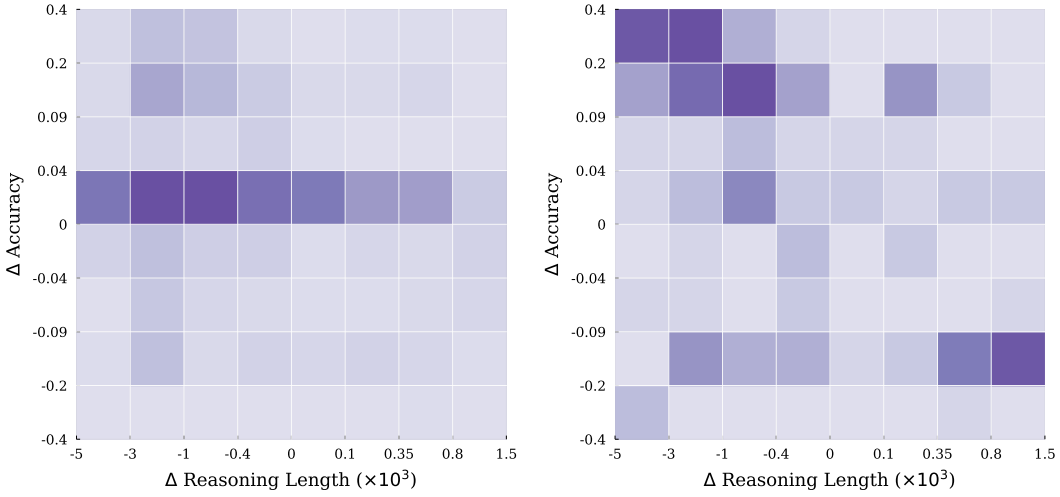
B.1 Heatmap Visualization Protocol

Each difficulty subset, split by base-model accuracy, is visualized as a density heatmap on a shared 8×8 grid. The horizontal axis (ΔReasoning length) measures the change in token relative to the base model for the same question, and the vertical axis (ΔAccuracy) measures the corresponding change in per-question accuracy. Each cell reports the within-bin fraction of questions. Darker cells indicate a higher concentration of questions in that region. All heatmaps share the following density scale:



B.2 QWEN3-1.7B

Short-Reward. As shown in Figure 5, under short-reward training, Qwen3-1.7B exhibits a strong leftward shift across all difficulty subsets, confirming that the short reward successfully compresses reasoning length. Notably, the $(0.25, 0.5]$ shows a concentration of density in the upper-left region, indicating that removing redundant reasoning steps improves accuracy for partially solvable questions. The hardest subset $[0, 0.25]$ remains clustered near zero Δ Accuracy, while the easiest subset $(0.75, 1.0]$ exhibits only a slight accuracy decline under length reduction.



(a) Difficulty Subset [0, 0.25]

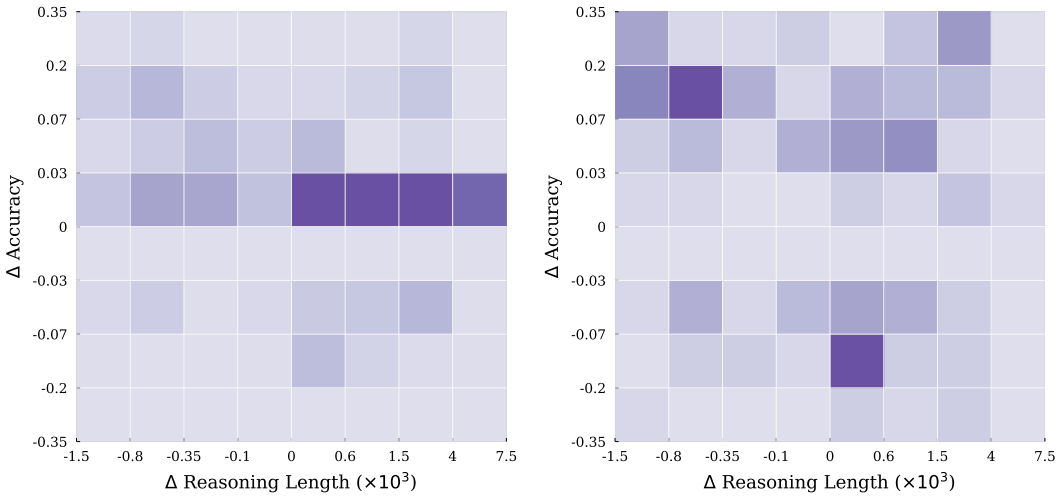
(b) Difficulty Subset (0.25, 0.5]

(c) Difficulty Subset (0.5, 0.75]

(d) Difficulty Subset (0.75, 1.0]

Figure 5: Density heatmaps of Δlength vs. Δaccuracy for **Qwen3-1.7B** under Short-Reward.

Long-Reward. Figure 6 presents the long-reward results for Qwen3-1.7B. Unlike the short-reward setting, all subsets exhibit a rightward shift, reflecting longer reasoning traces. However, this length increase does not consistently lead to higher accuracy. For the partially solvable subsets $(0.25, 0.5]$ and $(0.5, 0.75]$, density spreads across both positive and negative ΔAccuracy regions despite reasoning length increases, suggesting that additional tokens contribute unreliably to improving reasoning quality. Both the hardest subset $[0, 0.25]$ and easiest subset $(0.75, 1.0]$ show rightward concentration with ΔAccuracy fluctuating around zero, confirming that extended reasoning on questions the model either reliably solves or consistently fails to solve provides negligible benefit.



(a) Difficulty Subset [0, 0.25]

(b) Difficulty Subset (0.25, 0.5]

(c) Difficulty Subset (0.5, 0.75]

(d) Difficulty Subset (0.75, 1.0]

Figure 6: Density heatmaps of Δlength vs. Δaccuracy for **Qwen3-1.7B** under Long-Reward.

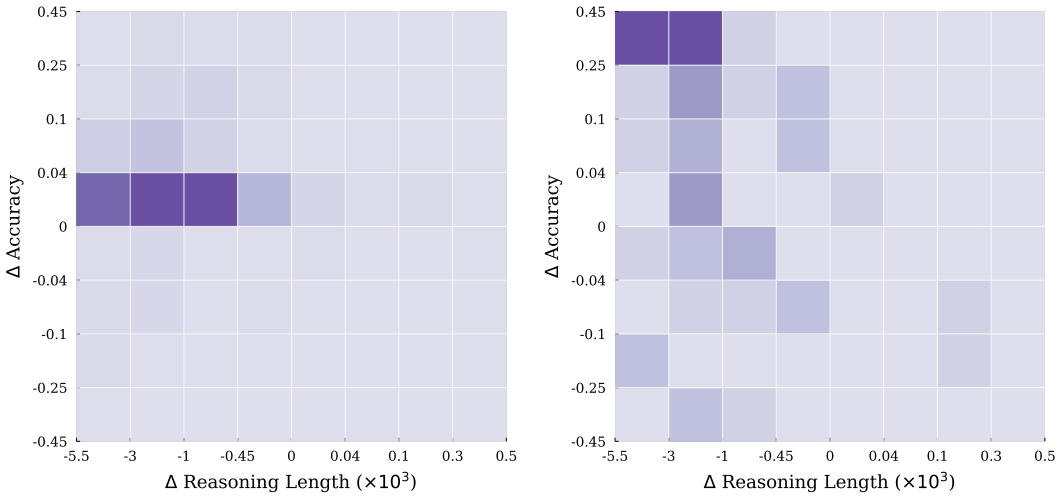
Key Observations for Qwen3-1.7B.

- Both the hardest subset $[0, 0.25]$ and the easiest subset $(0.75, 1.0]$ show ΔAccuracy concentrated near zero under short-reward and long-reward training, indicating that questions at these two subsets are largely insensitive to reasoning length changes.
- The partially solvable subsets $(0.25, 0.5]$ and $(0.5, 0.75]$ are sensitive to reasoning length changes. Short-reward training produces relatively more concentrated accuracy improvement than long-reward training, which distributes ΔAccuracy across both positive and negative regions.

B.3 QWEN3-4B

Short-Reward. The short-reward heatmaps for Qwen3-4B are shown in Figure 7. Consistent with the 1.7B results, length compression is evident across all difficulty subsets, as indicated by the leftward-concentrated density. Most of the density is concentrated on negative ΔLength regions, with only limited mass extending into the positive region. The most striking observation appears in the $(0.25, 0.5]$ subset, where a large fraction of questions concentrate in the upper-left region, with the strongest positive ΔAccuracy occurring at relatively large negative ΔLength regions, indicating that substantial length reductions could coincide with notable accuracy improvements for partially

solvable questions. The $(0.5, 0.75]$ subset displays a similar but relatively weaker trend. These two partially solvable subsets also exhibit a broader spread along the ΔAccuracy axis than the hardest and easiest subsets. For both the hardest $[0, 0.25]$ and easiest $(0.75, 1.0]$ subsets, ΔAccuracy remains clustered around zero despite substantial length reductions, indicating that reasoning length has limited influence on questions at these two difficulty subsets.



(a) Difficulty Subset [0.0, 0.25]

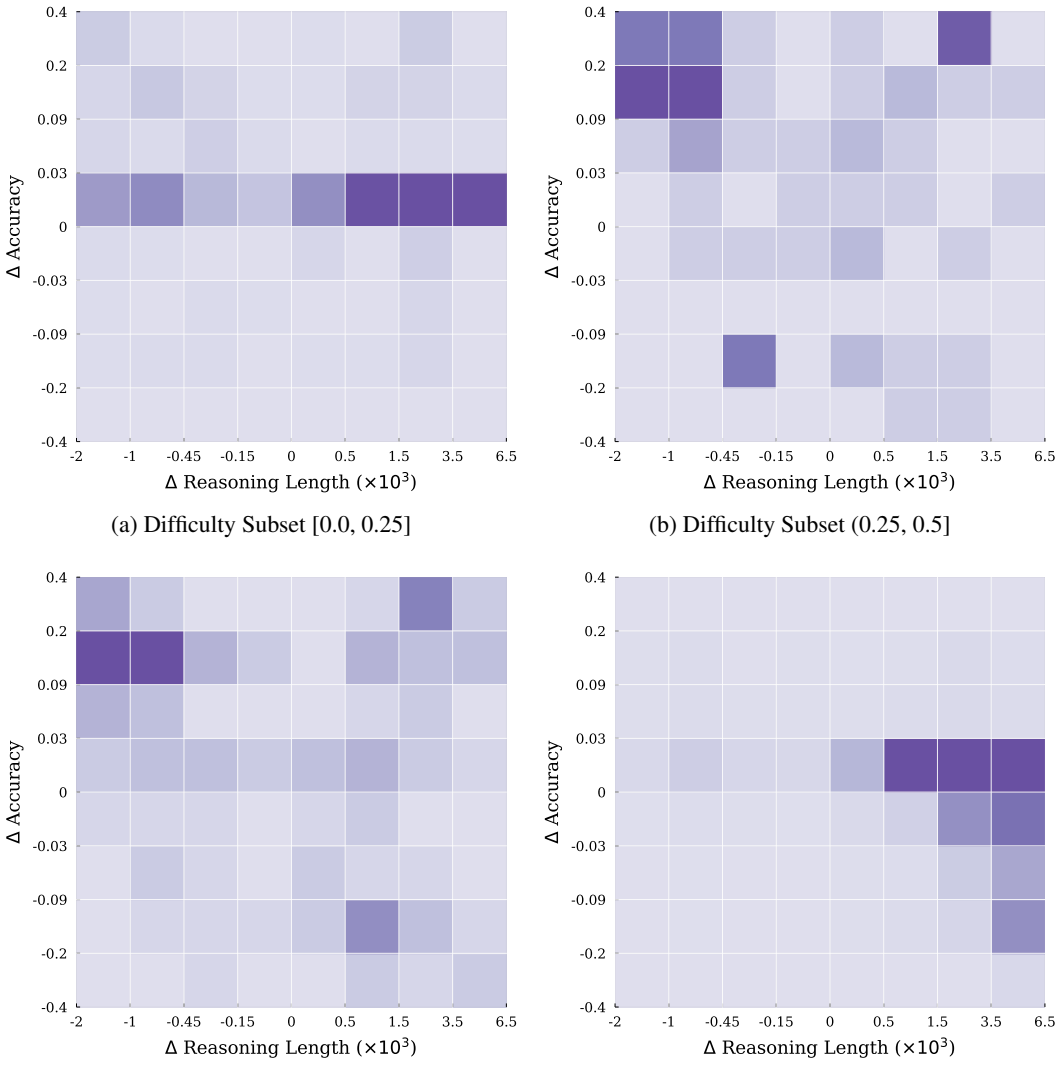
(b) Difficulty Subset (0.25, 0.5]

(c) Difficulty Subset (0.5, 0.75]

(d) Difficulty Subset (0.75, 1.0]

Figure 7: Density heatmaps of Δlength vs. Δaccuracy for **Qwen3-4B** under Short-Reward.

Long-Reward. The long-reward heatmaps for Qwen3-4B are shown in Figure 8. All four subsets exhibit a rightward shift extending up to $6{,}500$ tokens. The density also spans a substantially broader range of positive ΔLength regions across the four subsets. This rightward spread is visible across the entire difficulty subset. In particular, the $(0.25, 0.5]$ subset exhibits no consistent accuracy trend under length extension. One cluster achieves accuracy improvement in the upper-right region at large length increases, while another shows accuracy decline, indicating that the long reward indiscriminately extends reasoning length yet produces divergent accuracy effects across different questions. Moreover, the $(0.5, 0.75]$ subset displays a similar trend to that of the $(0.25, 0.5]$ subset, with density distributed across multiple ΔAccuracy regions. For both the hardest $[0, 0.25]$ and easiest $(0.75, 1.0]$ subsets, ΔAccuracy remains clustered near zero despite reasoning length increases, though the easiest subset $(0.75, 1.0]$ is more prone to accuracy decline as reasoning length grows.



(a) Difficulty Subset [0.0, 0.25]

(b) Difficulty Subset (0.25, 0.5]

(c) Difficulty Subset (0.5, 0.75]

(d) Difficulty Subset (0.75, 1.0]

Figure 8: Density heatmaps of Δlength vs. Δaccuracy for **Qwen3-4B** under Long-Reward.

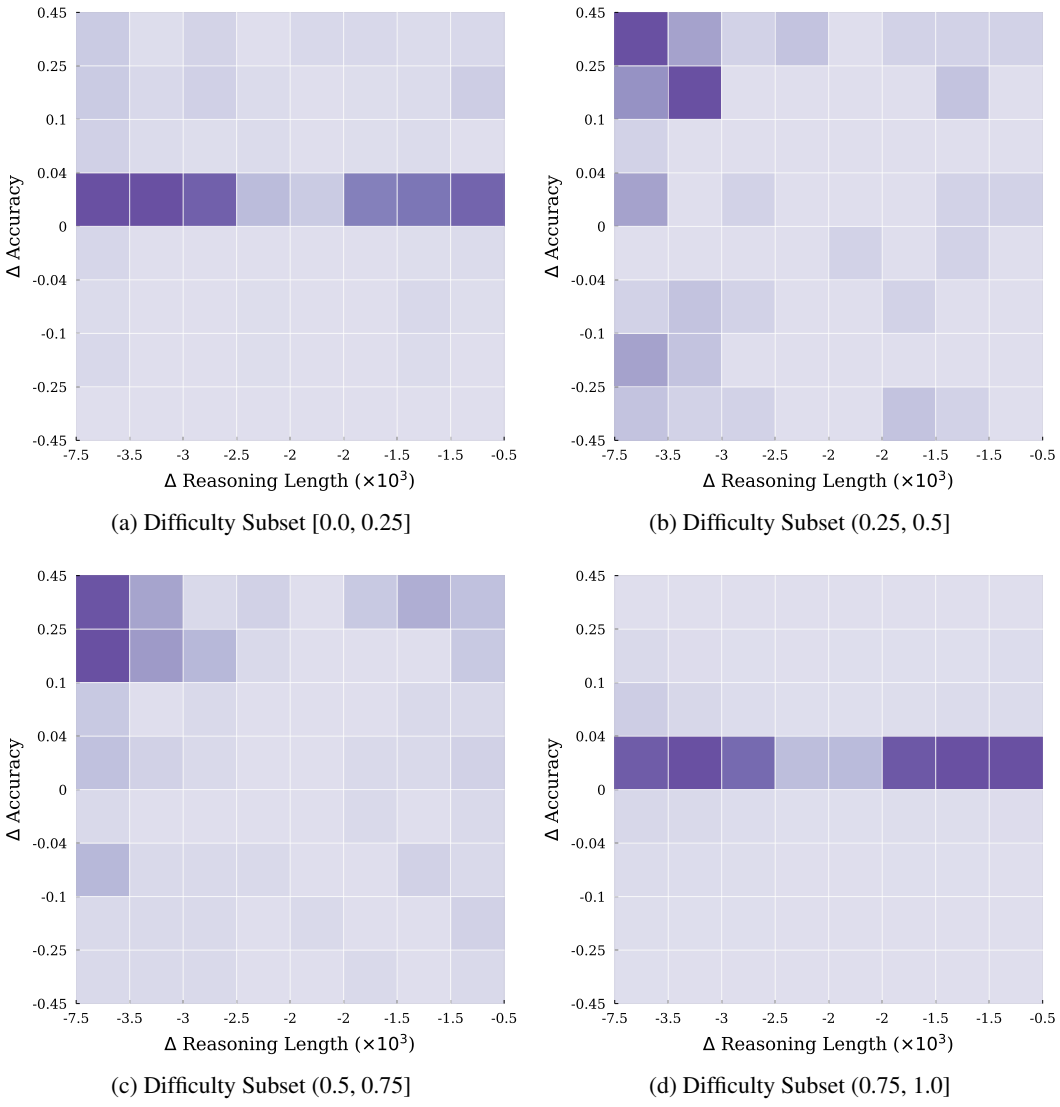
Key Observations for Qwen3-4B.

- Compared to 1.7B, the 4B model exhibits a narrower positive ΔLength range under short-reward, indicating more length compression with almost no questions showing length increases. Despite a wider ΔAccuracy improvement, the hardest $[0, 0.25]$ and easiest subsets $(0.75, 1.0]$ still concentrate near zero, further suggesting that these questions are insensitive to length changes.
- For the partially solvable subsets $(0.25, 0.5]$ and $(0.5, 0.75]$, the narrower positive length range leads to a sharper upper-left concentration under short-reward, indicating that Qwen3-4B achieves effective length compression on a broader range of questions within these subsets. Under long-reward, accuracy improvements remain inconsistent despite length extension.

B.4 QWEN3-8B

Short-Reward. Figure 9 illustrates the short-reward results for Qwen3-8B. Under short-reward training, density remains entirely in the negative ΔLength region in four difficulty subsets, showing strong compression relative to the base model. Notably, these large reductions in reasoning length do not always lead to accuracy degradation. Accuracy changes are concentrated in the two partially solvable subsets. In $(0.25, 0.5]$, density spans both positive and negative ΔAccuracy, but several

high-density regions lie above zero under large length reductions. The $(0.5, 0.75]$ subset shows a similar but clearer tendency, with positive accuracy changes concentrated toward the strongest compression, indicating that shortening reasoning can still coincide with improved performance. In contrast, the hardest and easiest subsets remain centered near zero ΔAccuracy throughout a broad range of negative ΔLength, suggesting limited sensitivity to such reductions.



(a) Difficulty Subset [0.0, 0.25]

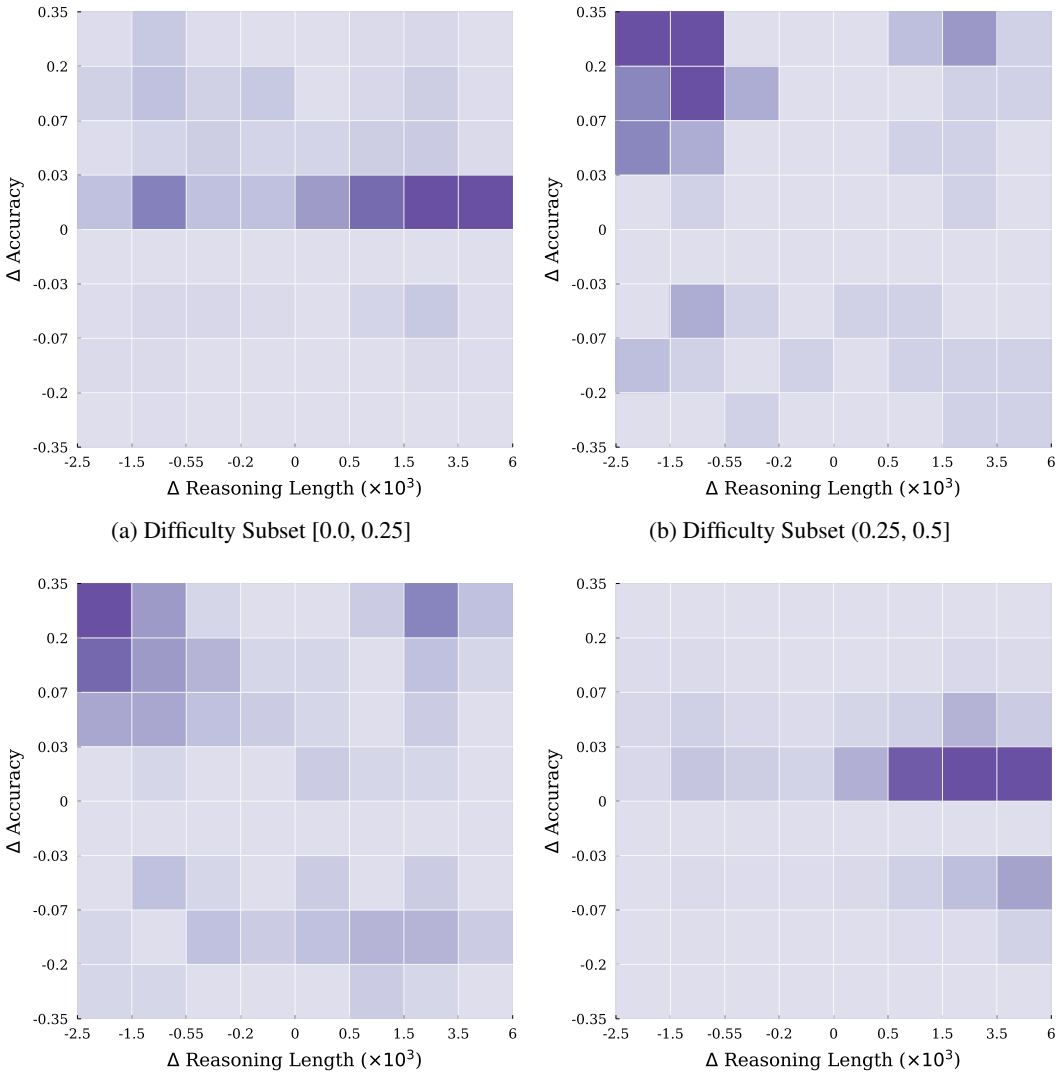
(b) Difficulty Subset (0.25, 0.5]

(c) Difficulty Subset (0.5, 0.75]

(d) Difficulty Subset (0.75, 1.0]

Figure 9: Density heatmaps of Δlength vs. Δaccuracy for **Qwen3-8B** under Short-Reward.

Long-Reward. Figure 10 presents the long-reward training results for Qwen3-8B. Similar to the smaller models, reasoning length varies substantially across all four difficulty subsets, while the corresponding accuracy changes remain highly inconsistent, particularly within the partially solvable subsets. In the $(0.25, 0.5]$ subset, notable accuracy improvements appear under both substantial length reductions and large length increases, indicating that longer or shorter reasoning does not consistently lead to better performance. The $(0.5, 0.75]$ subset exhibits a similar pattern, with positive and negative ΔAccuracy distributed across a broad range of positive and negative ΔLength regions. For the hardest $[0, 0.25]$ subset, ΔAccuracy remains concentrated near zero despite considerable variation in reasoning length. The easiest $(0.75, 1.0]$ subset shows a stronger shift toward increased reasoning length, yet most questions still remain close to zero ΔAccuracy, with a small fraction exhibiting accuracy degradation at larger length increases.



(a) Difficulty Subset [0.0, 0.25]

(b) Difficulty Subset (0.25, 0.5]

(c) Difficulty Subset (0.5, 0.75]

(d) Difficulty Subset (0.75, 1.0]

Figure 10: Density heatmaps of Δlength vs. Δaccuracy for **Qwen3-8B** under Long-Reward.

Key Observations for Qwen3-8B.

- Under short-reward training, four difficulty subsets exhibit reductions in reasoning length, yet the hardest and easiest subsets remain tightly concentrated around zero ΔAccuracy. This further indicates that large changes in reasoning length have limited influence on these questions.
- The partially solvable subsets respond more strongly to length changes. Short-reward produces positive ΔAccuracy primarily under length reductions, whereas long-reward distributes both accuracy improvements and declines over a much broader range of ΔLength.

B.5 Summary

Across model sizes and reward types, the heatmaps reveal a consistent pattern. The hardest questions $[0, 0.25]$ cluster near zero ΔAccuracy regardless of length changes, confirming that these questions are largely insensitive to reasoning length. The partially solvable subsets $(0.25, 0.5]$ and $(0.5, 0.75]$ exhibit notable accuracy shifts. Under short-reward training, length reduction can coincide with positive ΔAccuracy, whereas under long-reward training, additional token overhead does not effectively convert into consistent accuracy improvement. The easiest questions $(0.75, 1.0]$ show comparatively small accuracy variation, as the model already solves them reliably.

C UNEXPECTED BEHAVIORS

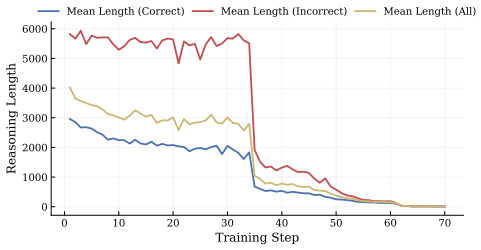
In this part, we report tracked training dynamics and trajectories alongside the cyclic reasoning definition and its comparison between stable and unstable intervals across different training configurations.

C.1 TRACKED TRAINING DYNAMICS

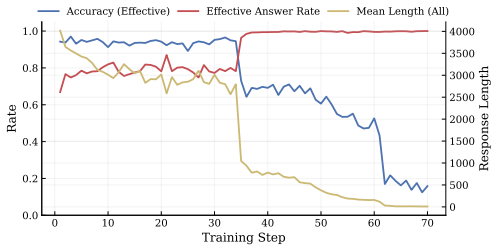
To characterize model's behavioral dynamics under both short-reward and long-reward length incentives, we mainly track the following five metrics across training:

- **Mean Reasoning Length**: average number of reasoning tokens across all $4{,}096$ sampled responses at each training step.
- **Mean Correct / Incorrect Reasoning Length**: average number of reasoning tokens conditioned on whether the response is correct or incorrect, computed over the $4{,}096$ samples at each step.
- **Effective Answer Rate**: the fraction of $4{,}096$ responses that contain a parsable final answer.
- **Correct Rate Among Effective Answers**: accuracy computed only over responses with a parsable final answer, excluding responses that fail to produce a valid output.

C.2 TRAINING TRAJECTORIES

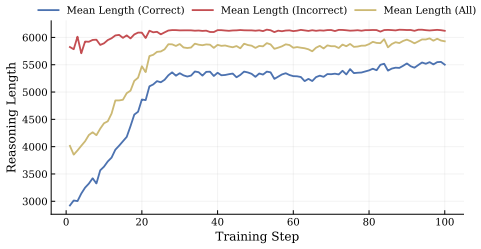


(a) Reasoning length by correctness

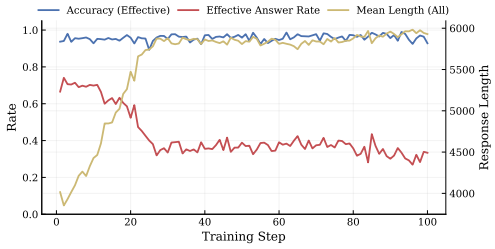


(b) Effective answer rate and accuracy

Figure 11: Training trajectories under Short-Reward for **Qwen3-1.7B**.

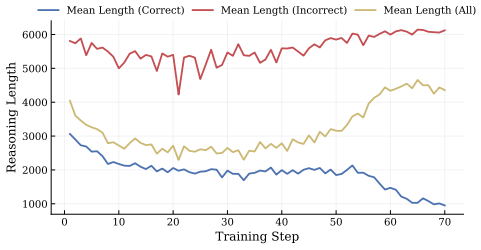


(a) Reasoning length by correctness

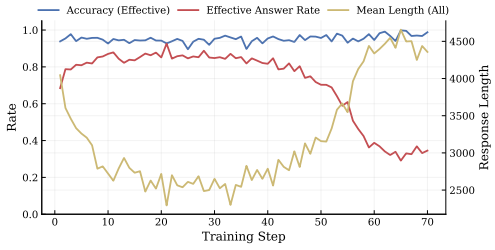


(b) Effective answer rate and accuracy

Figure 12: Training trajectories under Long-Reward for **Qwen3-1.7B**.

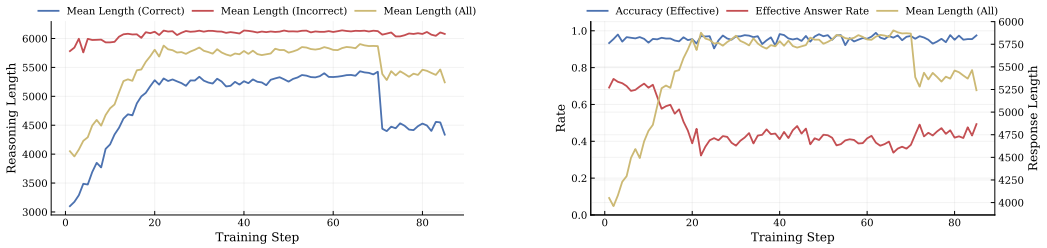


(a) Reasoning length by correctness



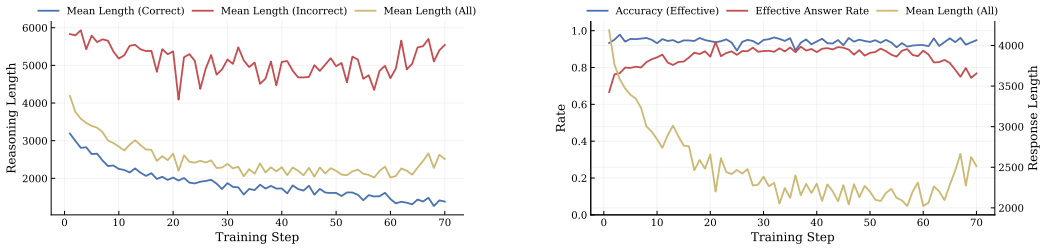
(b) Effective answer rate and accuracy

Figure 13: Training trajectories under Short-Reward for **Qwen3-4B**.



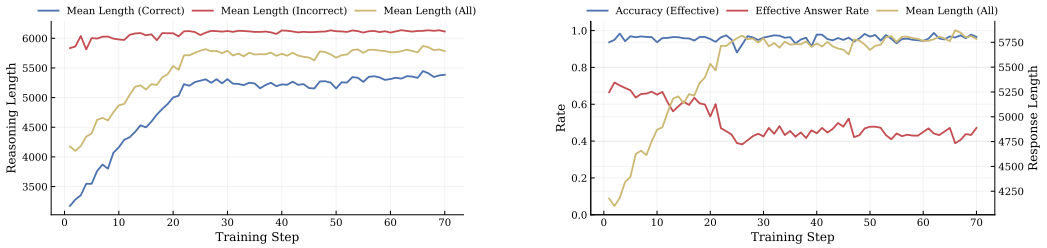
(a) Reasoning length by correctness (b) Effective answer rate and accuracy

Figure 14: Training trajectories under Long-Reward for **Qwen3-4B**.



(a) Reasoning length by correctness (b) Effective answer rate and accuracy

Figure 15: Training trajectories under Short-Reward for **Qwen3-8B**.



(a) Reasoning length by correctness (b) Effective answer rate and accuracy

Figure 16: Training trajectories under Long-Reward for **Qwen3-8B**.

Figures 11–16 present the training trajectories for Qwen3-1.7B, Qwen3-4B, and Qwen3-8B under short-reward and long-reward training. For Qwen3-1.7B with short-reward training (Figure 11), mean reasoning length gradually decreases before collapsing abruptly after approximately step 35. Meanwhile, accuracy among effective answers drops substantially in the later stage, indicating that aggressive length compression eventually degrades reasoning performance. Under long-reward training (Figure 12), mean reasoning length rapidly increases and then remains close to the maximum response length, while the effective answer rate declines to around 0.35 and accuracy among effective answers remains relatively stable. For Qwen3-4B under short-reward training (Figure 13), mean reasoning length initially decreases but later rebounds, primarily as incorrect responses remain substantially longer while correct responses continue to shorten. This rebound coincides with a marked decline in the effective answer rate, whereas accuracy among effective answers remains stable. Under long-reward training (Figure 14), reasoning length similarly increases rapidly and remains at a high level, accompanied by a lower effective answer rate but little change in effective-answer accuracy. Qwen3-8B exhibits a more stable trajectory under short-reward training (Figure 15). Mean reasoning length decreases steadily and remains low, with only a modest late-stage rebound compared with the smaller models, while accuracy among effective answers stays consistently high. In contrast, under long-reward training (Figure 16), reasoning length again increases rapidly and stabilizes near the upper range, while the effective answer rate decreases to around 0.4 despite nearly unchanged accuracy among effective answers. This suggests that the additional reasoning length does not contribute to significant improvement in reasoning capability but instead increases the proportion of responses that fail to reach an answer within the maximum length.

C.3 CYCLIC REASONING: DEFINITIONS

Definition of Cyclic Segment. To approximate the prevalence of unproductive repetitive behavior in model responses, we introduce the notion of a *Cyclic Segment*. Given a response, we first segment it into sentences using punctuation boundaries. A Cyclic Segment is defined as the substring spanning from the first occurrence of any cyclic keyword in a sentence to the end of that sentence, where the cyclic keywords are defined as follows (matched case-insensitively):

```
wait, but, if, check, again, alternatively
```

Two supplementary rules apply:

(1) If multiple cyclic keywords appear within the same sentence, they are counted as a single Cyclic Segment.

(2) Cyclic keywords appearing in different sentences are counted as separate Cyclic Segments.

Cyclic Reasoning Ratio (CRR). We define the *Cyclic Reasoning Ratio* (CRR) to measure the proportion of sentences attributed to cyclic behavior within a response. For a given question q, let $\mathcal{A}(q)$ denote the set of sampled responses. For each response $a \in \mathcal{A}(q)$, let $N_{\text{cyc}}(a)$ denote the number of sentences containing at least one Cyclic Segment and $N_{\text{tot}}(a)$ denote the total number of sentences. The CRR for question q is defined as:

$$\text{CRR}(q) = \frac{1}{|\mathcal{A}(q)|} \sum_{a \in \mathcal{A}(q)} \frac{N_{\text{cyc}}(a)}{N_{\text{tot}}(a)} \tag{9}$$

A higher CRR indicates that a greater proportion of sentences in the model output consist of cyclic reasoning patterns rather than substantive inference, and serves as a proxy for reasoning inefficiency.

Remark

These keywords were identified through manual inspection of model responses from unstable training intervals, where recurring patterns of unproductive reasoning loops were observed, such as re-checking a verified computation or reproducing near-identical paragraphs across successive reasoning steps. Individual occurrences of these keywords may correspond to productive reasoning, e.g., legitimate backtracking or self-verification. Our analysis therefore focuses on the *change* in their density ΔCRR between stable and unstable intervals, capturing the growth of repetitive patterns rather than isolated normal usage. We provide representative cases in Appendix I.3, illustrating how keyword-flagged sentences correspond to semantic repetition during unstable training intervals.

Training Interval Classification. To compare model behavior across different stages of training, we identify two distinct step intervals that reflect different patterns in the training dynamics.

Stable Interval: a training interval in which the training metrics remain in their normal range and no anomalous training dynamics are observed.

Unstable Interval: a training interval in which the training dynamics exhibit clear anomalies, indicating that the training process has deviated from its expected trajectory.

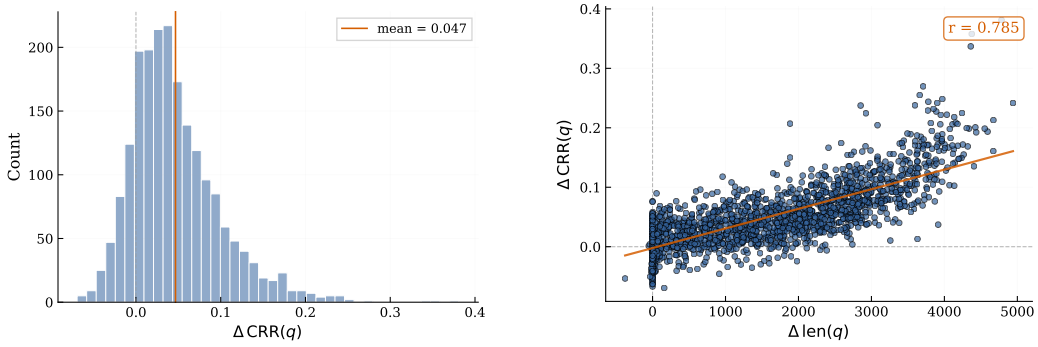
C.4 CYCLIC REASONING: RESULTS

We exclude Qwen3-1.7B under short-reward training from this analysis, as this setting exhibits length collapse rather than cyclic reasoning. A detailed case study is provided in Appendix I.1.

Table 10: Step intervals selected as the *Stable Interval* and *Unstable Interval* under short-reward and long-reward training for each model.

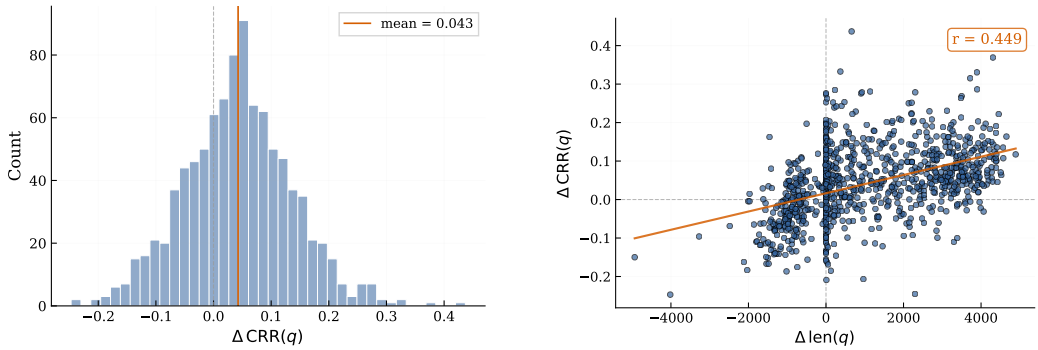
Model	Short-reward		Long-reward	
	Stable Interval	Unstable Interval	Stable Interval	Unstable Interval
Qwen3-1.7B	N/A	N/A	Steps 1–15	Steps 31–45
Qwen3-4B	Steps 1–10	Steps 60–70	Steps 1–15	Steps 31–45
Qwen3-8B	Steps 1–10	Steps 60–70	Steps 1–15	Steps 31–45

To examine whether the length increases observed during unstable training intervals are associated with cyclic reasoning, we compare the per-question CRR between the Stable and Unstable Intervals defined in Table 10. For each matched question, we compute $\Delta\,\mathrm{CRR} = \mathrm{CRR}_{\mathrm{unstable}} - \mathrm{CRR}_{\mathrm{stable}}$ and $\Delta\,\mathrm{len} = \mathrm{len}_{\mathrm{unstable}} - \mathrm{len}_{\mathrm{stable}}$. A positive $\Delta\,\mathrm{CRR}$ indicates that the proportion of cyclic segments increased during the unstable phase. Figures 17 – 21 present the distribution of $\Delta\,\mathrm{CRR}$ alongside its correlation with $\Delta\,\mathrm{len}$. The results suggest that longer reasoning traces do not always lead to deeper or more productive thinking, and may instead introduce redundant cyclic patterns that degrade reasoning efficiency. In particular, the distribution of $\Delta\,\mathrm{CRR}$ has a positive mean under long-reward settings, indicating that cyclic reasoning becomes more prevalent during unstable training phases. The positive correlation between $\Delta\,\mathrm{len}$ and $\Delta\,\mathrm{CRR}$ further confirms that the observed length increases are associated with a higher proportion of repetitive reasoning patterns rather than deeper exploration.



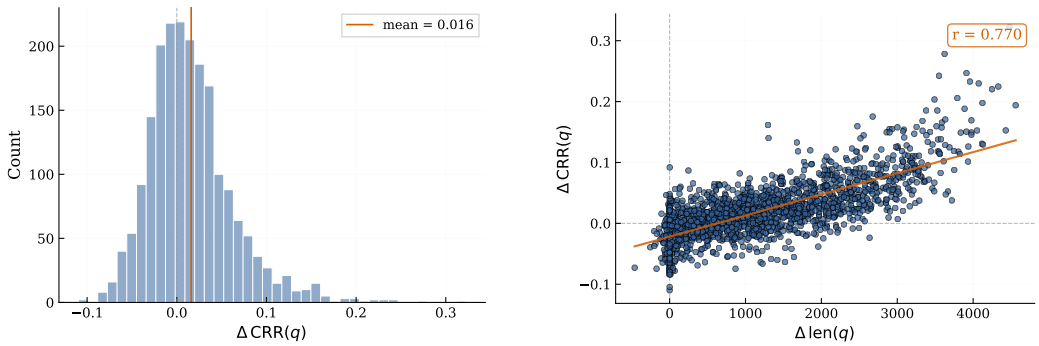
(a) Distribution of $\Delta\,\mathrm{CRR}$ (b) $\Delta\,\mathrm{CRR}$ vs $\Delta\,\mathrm{len}$

Figure 17: CRR comparison for **Qwen3-1.7B** under Long-Reward training.



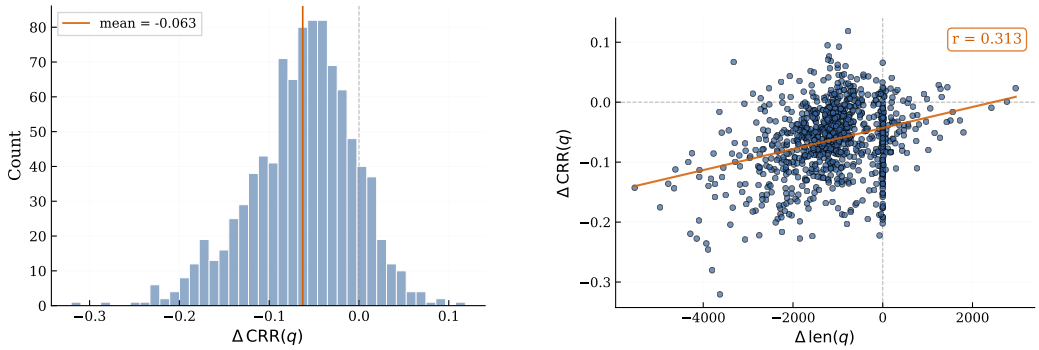
(a) Distribution of $\Delta\,\mathrm{CRR}$ (b) $\Delta\,\mathrm{CRR}$ vs $\Delta\,\mathrm{len}$

Figure 18: CRR comparison for **Qwen3-4B** under Short-Reward training.



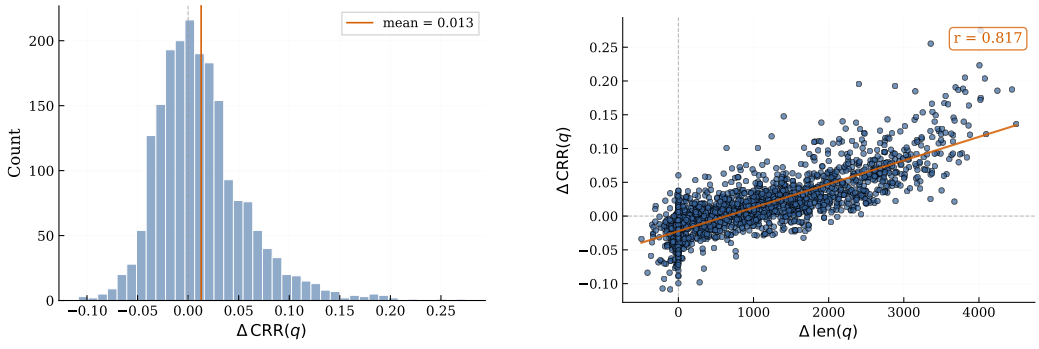
(a) Distribution of Δ CRR (b) Δ CRR vs Δ len

Figure 19: CRR comparison for **Qwen3-4B** under Long-Reward training.



(a) Distribution of Δ CRR (b) Δ CRR vs Δ len

Figure 20: CRR comparison for **Qwen3-8B** under Short-Reward training.



(a) Distribution of Δ CRR (b) Δ CRR vs Δ len

Figure 21: CRR comparison for **Qwen3-8B** under Long-Reward training.

D Accuracy Gap Analysis

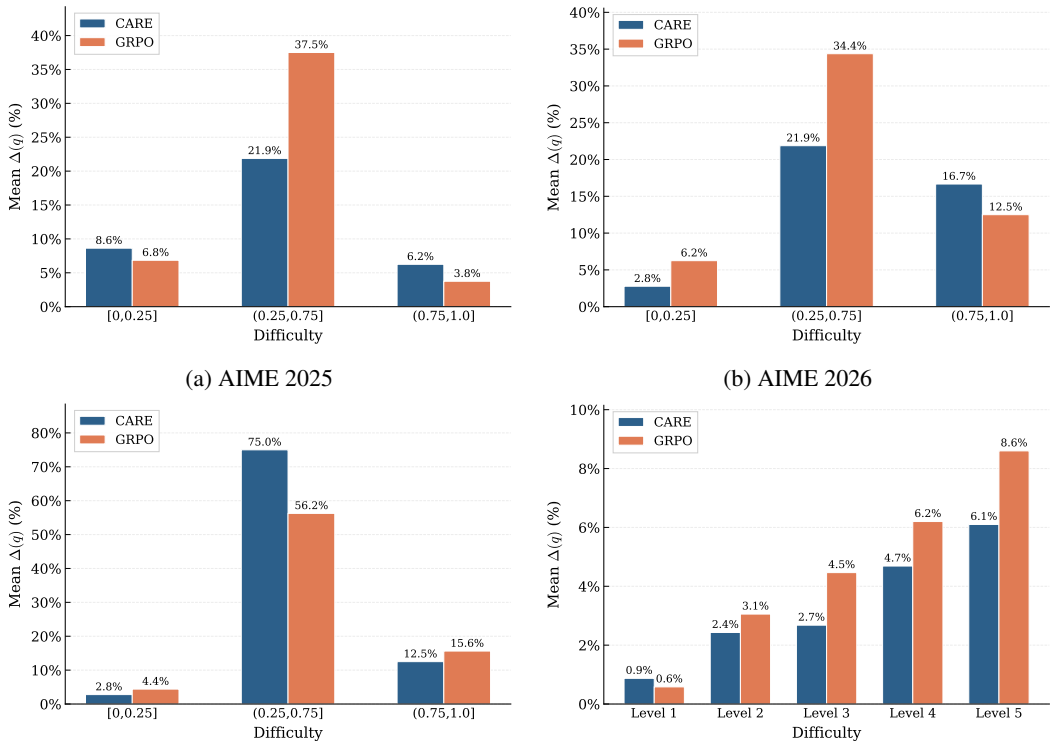
In this part, we provide the full directional signal analysis between CARE and GRPO, across AIME 2025, AIME 2026, HMMT 2026, and MATH500 for Qwen3-1.7B, 4B and 8B.

D.1 Metrics and Difficulty Partitioning

For each question q, the accuracy gap $\Delta(q) = |\bar{y}_{\mathcal{S}}(q) - \bar{y}_{\mathcal{L}}(q)|$ measures the absolute accuracy difference between the shorter and longer response halves, capturing the degree of the length sensitivity of each question. We compute $\Delta(q)$ for GRPO and CARE and aggregate it within each difficulty group across all benchmarks. For MATH500, questions are partitioned by the dataset's built-in difficulty labels (Levels 1–5). Since AIME 2025, AIME 2026, and HMMT 2026 lack predefined labels, difficulty is estimated from the base model's per-question Pass@1 and partitioned into three subsets: $[0,\ 0.25]$, $(0.25,\ 0.75]$, and $(0.75,\ 1.0]$. The two middle subsets $(0.25,\ 0.5]$ and $(0.5,\ 0.75]$ are merged into $(0.25,\ 0.75]$ to ensure sufficient sample sizes, as these benchmarks contain few questions and finer splitting would introduce high variance in $\Delta(q)$ estimates.

D.2 Experiment Results

Figures 22 – 24 report the mean accuracy gap $\Delta(q)$ at each difficulty group across different benchmarks. Over the majority of benchmarks, CARE achieves a lower mean $\Delta(q)$ than GRPO, indicating that the directional signal successfully reduces the accuracy gap between shorter and longer response halves. This reduction is most consistent on the partially solvable subset $(0.25,\ 0.75]$ and on harder MATH500 levels, with the clearest consistent reduction appearing at Level 5, where such length sensitivity is more prevalent. On the hardest subsets $[0, 0.25]$, both methods exhibit comparably lower $\Delta(q)$ than the partially solvable subsets, confirming that hard questions for models are insensitive to length variation.



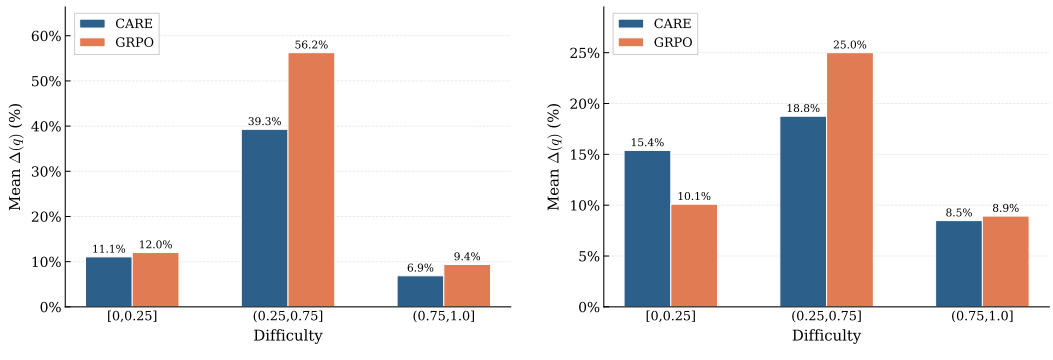
(a) AIME 2025

(b) AIME 2026

(c) HMMT 2026

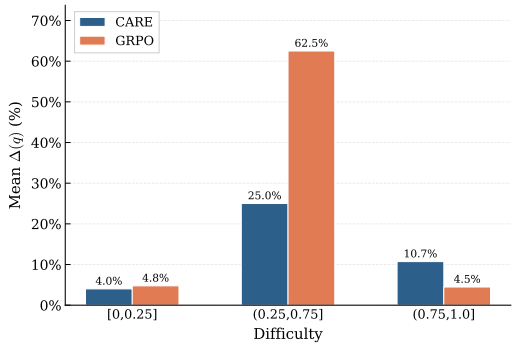
(d) MATH500

Figure 22: Accuracy gap $\Delta(q)$ at each difficulty group for **Qwen3-1.7B**.

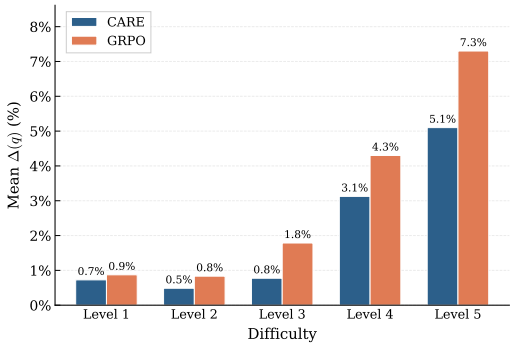


(a) AIME 2025

(b) AIME 2026

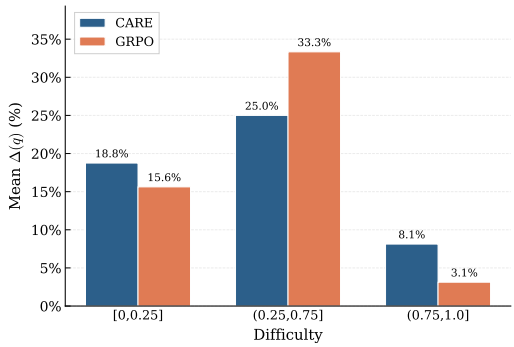


(c) HMMT 2026

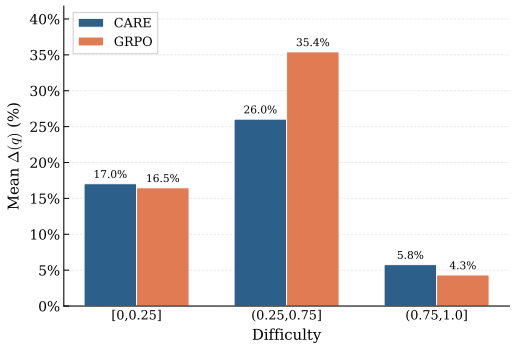


(d) MATH500

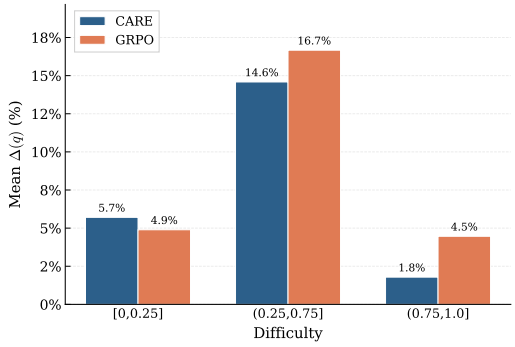
Figure 23: Accuracy gap $\Delta(q)$ at each difficulty group for **Qwen3-4B**.



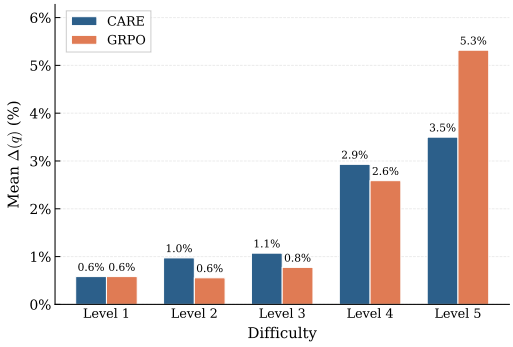
(a) AIME 2025



(b) AIME 2026



(c) HMMT 2026



(d) MATH500

Figure 24: Accuracy gap $\Delta(q)$ at each difficulty group for **Qwen3-8B**.

D.2.1 Benchmark-Specific Analysis

On AIME 2025 and AIME 2026, the reduction in $\Delta(q)$ generally concentrates on the partially solvable subset, with this pattern particularly clear for Qwen3-1.7B and Qwen3-4B. For Qwen3-4B, CARE reduces the partially solvable gap from 56.2% to 39.3% on AIME 2025 and from 25.0% to 18.8% on AIME 2026, while the hardest and easiest subsets show smaller or reversed differences. On MATH500, the reduction is more evident at harder difficulty levels for Qwen3-1.7B and Qwen3-4B, whereas for Qwen3-8B the clearest decrease appears at Level 5, from 5.3% to 3.5%.

A different pattern emerges on HMMT 2026. CARE achieves lower $\Delta(q)$ on the hardest and partially solvable subsets for Qwen3-4B, on the hardest and easiest subsets for Qwen3-1.7B, and on the partially solvable and easiest subsets for Qwen3-8B. Since all three base models attain relatively low accuracy on HMMT 2026, questions are concentrated disproportionately in the hardest subset, leaving the remaining subsets with limited sample sizes. These smaller subsets make $\Delta(q)$ estimates more susceptible to per-question variation, although CARE still achieves higher overall Pass@1 than GRPO on HMMT 2026, as shown in Table 1.

E Length Consistency and Accuracy

In this section, we compute the dispersion $D(q)$ for CARE and GRPO on Qwen3-1.7B, Qwen3-4B and Qwen3-8B across AIME 2025, AIME 2026, HMMT 2026, and MATH500, measuring how consistently the model converges to a stable reasoning length when correctly solving a question q.

E.1 Metrics and Difficulty Partitioning

For each question q, the dispersion $D(q)$ is defined as the mean absolute deviation of correct reasoning lengths normalized by their mean, quantifying how tightly correct responses cluster around a common reasoning length. Formally, let $\{L_i\}_{i\in C(q)}$ denote the lengths of correct responses for question q and $\bar{L} = \frac{1}{|C(q)|}\sum_{i\in C(q)} L_i$ their mean; then $D(q) = \frac{1}{|C(q)|}\sum_{i\in C(q)} |L_i - \bar{L}| \,/\, \bar{L}$. A lower $D(q)$ indicates that the model converges to a more consistent reasoning depth when correctly solving that question. Questions are partitioned into four difficulty subsets based on the base model's per-question Pass@1: $[0,\ 0.25]$, $(0.25,\ 0.5]$, $(0.5,\ 0.75]$, and $(0.75,\ 1.0]$. A dash (–) in the tables indicates that no question falls within that difficulty subset under the base model's Pass@1 distribution.

E.2 Experiment Results

Tables 11 – 13 present the full Pass@1 and dispersion $D(q)$ results. In these three models, CARE attains lower $D(q)$ than GRPO in most subsets while maintaining comparable or higher Pass@1, although several exceptions appear on AIME 2025, AIME 2026, and MATH500.

Table 11: Pass@1 (%) and $D(q)$ (%) for **Qwen3-1.7B** across four benchmarks.

Difficulty Subset	AIME 2025				AIME 2026			
	GRPO		CARE		GRPO		CARE	
	Pass@1	$D(q)$	Pass@1	$D(q)$	Pass@1	$D(q)$	Pass@1	$D(q)$
$[0,\ 0.25]$	8.04	14.48	11.61	12.27	2.78	4.85	4.86	9.81
$(0.25,\ 0.5]$	37.50	8.95	37.50	10.81	40.62	15.63	37.50	4.65
$(0.5,\ 0.75]$	77.08	14.49	91.67	16.30	76.56	29.15	73.44	29.05
$(0.75,\ 1.0]$	95.00	13.10	92.50	12.48	80.21	17.48	85.42	15.21
Difficulty Subset	HMMT 2026				MATH500			
	GRPO		CARE		GRPO		CARE	
	Pass@1	$D(q)$	Pass@1	$D(q)$	Pass@1	$D(q)$	Pass@1	$D(q)$
$[0,\ 0.25]$	5.32	22.12	8.33	17.15	16.55	20.78	13.85	29.85
$(0.25,\ 0.5]$	–	–	–	–	44.85	16.29	50.37	16.15
$(0.5,\ 0.75]$	34.38	30.40	62.50	24.79	70.22	20.95	70.22	22.95
$(0.75,\ 1.0]$	92.19	21.80	92.19	17.35	97.35	14.45	96.77	13.83

Table 12: Pass@1 (%) and $D(q)$ (%) for **Qwen3-4B** across four benchmarks.

Difficulty Subset	AIME 2025				AIME 2026			
	GRPO		CARE		GRPO		CARE	
	Pass@1	$D(q)$	Pass@1	$D(q)$	Pass@1	$D(q)$	Pass@1	$D(q)$
[0, 0.25]	12.02	12.53	14.90	9.83	16.35	12.58	18.27	13.17
(0.25, 0.5]	46.88	12.35	53.12	10.96	47.92	24.12	56.25	15.63
(0.5, 0.75]	65.00	31.16	67.50	21.36	–	–	–	–
(0.75, 1.0]	90.62	11.25	90.62	10.06	87.50	12.47	88.84	9.84
Difficulty Subset	HMMT 2026				MATH500			
	GRPO		CARE		GRPO		CARE	
	Pass@1	$D(q)$	Pass@1	$D(q)$	Pass@1	$D(q)$	Pass@1	$D(q)$
[0, 0.25]	7.25	14.99	7.75	14.48	22.95	15.67	24.29	14.81
(0.25, 0.5]	–	–	–	–	62.89	21.09	62.89	19.41
(0.5, 0.75]	56.25	22.46	81.25	21.56	77.26	19.73	83.85	18.83
(0.75, 1.0]	91.07	27.47	90.18	18.60	98.69	12.05	98.60	11.54

Table 13: Pass@1 (%) and $D(q)$ (%) for **Qwen3-8B** across four benchmarks.

Difficulty Subset	AIME 2025				AIME 2026			
	GRPO		CARE		GRPO		CARE	
	Pass@1	$D(q)$	Pass@1	$D(q)$	Pass@1	$D(q)$	Pass@1	$D(q)$
[0, 0.25]	16.74	14.57	14.73	12.33	14.49	18.03	15.91	13.21
(0.25, 0.5]	62.50	20.27	65.62	16.31	60.94	20.92	63.54	17.84
(0.5, 0.75]	70.62	12.80	72.50	12.72	–	–	–	–
(0.75, 1.0]	96.56	13.41	92.81	13.39	95.43	13.60	92.79	13.10
Difficulty Subset	HMMT 2026				MATH500			
	GRPO		CARE		GRPO		CARE	
	Pass@1	$D(q)$	Pass@1	$D(q)$	Pass@1	$D(q)$	Pass@1	$D(q)$
[0, 0.25]	7.07	14.90	8.83	12.74	23.40	14.44	23.26	19.57
(0.25, 0.5]	18.75	24.35	21.88	22.71	69.06	20.74	71.88	21.40
(0.5, 0.75]	92.19	24.67	95.31	23.20	84.01	15.89	84.38	14.42
(0.75, 1.0]	–	–	–	–	98.80	11.62	98.98	11.90

E.2.1 Benchmark-Specific Analysis

For Qwen3-1.7B, the reduction in $D(q)$ varies across benchmarks. The clearest improvement appears on HMMT 2026, where CARE lowers dispersion in all three available subsets, including a reduction from 30.40% to 24.79% on $(0.5,\ 0.75]$ together with a Pass@1 improvement. AIME 2026 also shows a decrease on $(0.25,\ 0.5]$, from 15.63% to 4.65%. However, on MATH500, CARE slightly reduces $D(q)$ on $(0.25,\ 0.5]$ and the easiest subset, while dispersion increases on the other two subsets.

A stronger and more consistent pattern emerges for Qwen3-4B. On AIME 2025, $D(q)$ decreases across all four difficulty subsets, while Pass@1 is maintained or improved in most cases. The $(0.25,\ 0.5]$ subset of AIME 2026 is particularly notable, with dispersion dropping from 24.12% to 15.63% as Pass@1 rises from 47.92% to 56.25%. CARE likewise achieves lower $D(q)$ throughout MATH500. For HMMT 2026, the largest accuracy improvement occurs on $(0.5,\ 0.75]$, where Pass@1 increases from 56.25% to 81.25% while $D(q)$ decreases slightly.

The Qwen3-8B results further show that this behavior persists at a larger model size. CARE lowers $D(q)$ across all available subsets on both AIME 2025 and AIME 2026, with clear reductions in $D(q)$ on the partially solvable subsets. HMMT 2026 exhibits a similarly consistent trend, where lower dispersion is accompanied by higher Pass@1 in all three available subsets. In contrast, MATH500 exhibits a different pattern, with $D(q)$ decreasing on $(0.5,\ 0.75]$ but increasing in the other subsets despite largely comparable or improved Pass@1.

F ACCURACY UNDER VARYING BUDGET CONSTRAINTS

This section evaluates the robustness of CARE under varying inference budgets and reports Pass@1 on AIME 2025, AIME 2026, HMMT 2026, and MATH500 for Qwen3-1.7B, Qwen3-4B and Qwen3-8B.

F.1 EVALUATION PROTOCOL

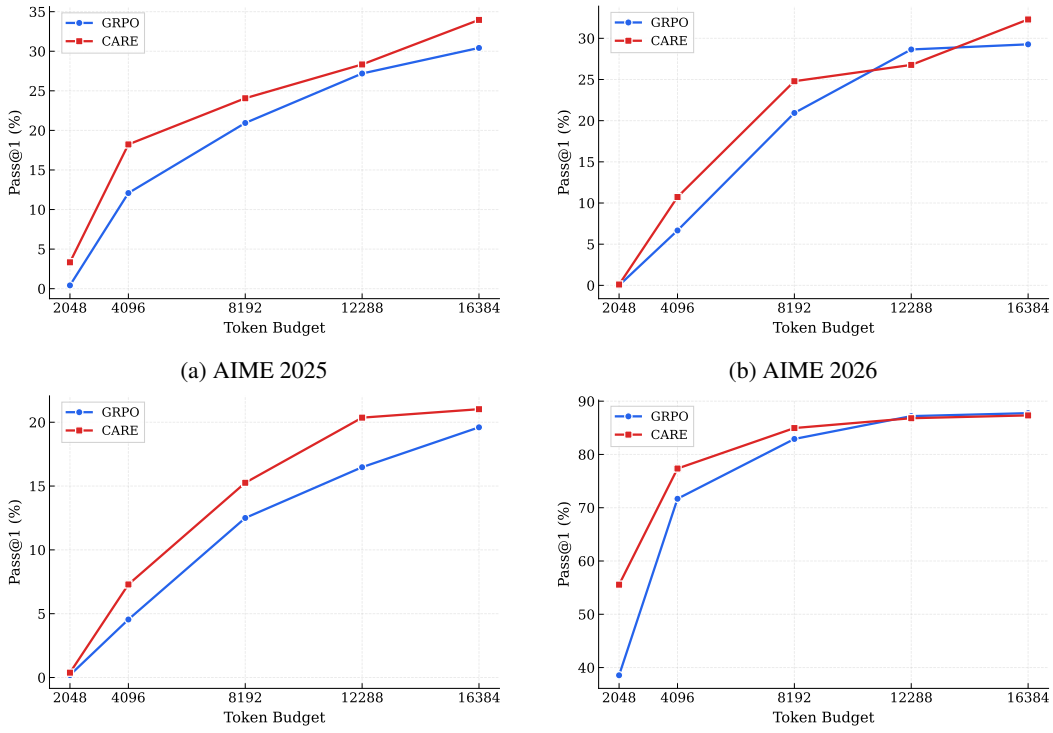
We evaluate both CARE and GRPO on Qwen3-1.7B, Qwen3-4B and Qwen3-8B by sweeping the maximum reasoning length across five token budgets: $\{2048, 4096, 8192, 12288, 16384\}$.

F.2 EXPERIMENT RESULTS

As shown in Figures 25– 27, CARE mostly achieves comparable or higher Pass@1 than GRPO under restricted token budget, while the gap tends to diminish as the budget increases, although the pattern varies in different models and benchmarks. This suggests that CARE prioritizes token-efficient reasoning paths, an advantage more apparent under limited computation budgets.

F.2.1 BENCHMARK-SPECIFIC ANALYSIS

On these benchmarks, these models exhibit a shared pattern but differ in Pass@1. For Qwen3-1.7B, CARE begins to outperform GRPO starting from $2{,}048$ tokens on both AIME 2025 and AIME 2026, and this gap remains relatively stable through $16{,}384$. Qwen3-4B follows a similar trajectory over the same budget range. On AIME 2026, CARE leads GRPO by approximately 4% at $8{,}192$ tokens, while on AIME 2025 the gap is slightly smaller but equally persistent from $4{,}096$ onward. For Qwen3-8B, the improvement is also more visible under restricted budgets. CARE achieves higher Pass@1 on both AIME benchmarks at $4{,}096$ and $8{,}192$ tokens, while the difference becomes small at $12{,}288$ and $16{,}384$ tokens. On HMMT 2026, CARE consistently maintains an advantage under these five token budgets and all three model sizes, whereas the curves on MATH500 become increasingly close as the token budget grows, particularly for Qwen3-8B.



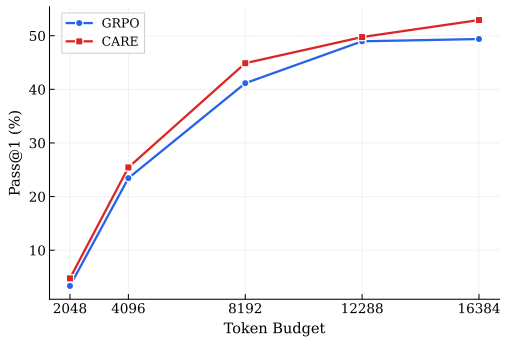
(a) AIME 2025

(b) AIME 2026

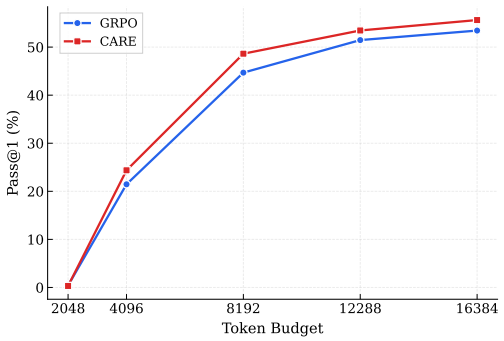
(c) HMMT 2026

(d) MATH500

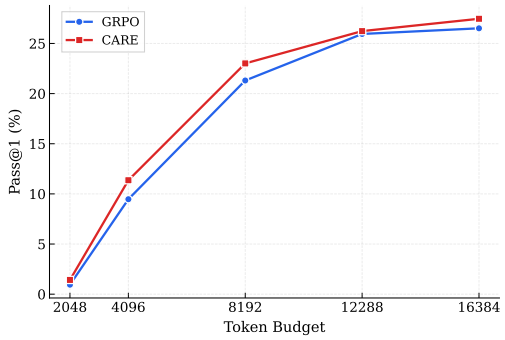
Figure 25: Pass@1 under varying token budgets for **Qwen3-1.7B**.



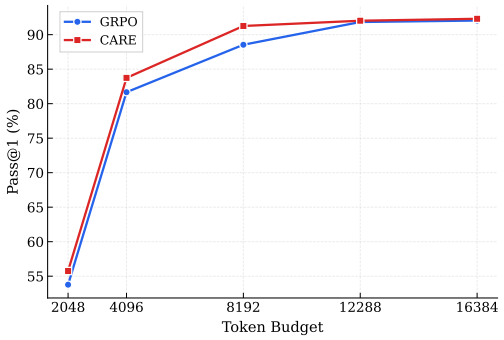
(a) AIME 2025



(b) AIME 2026

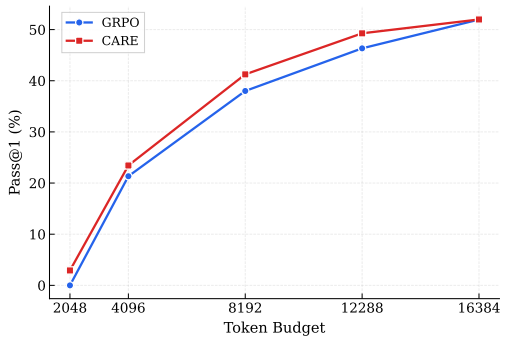


(c) HMMT 2026

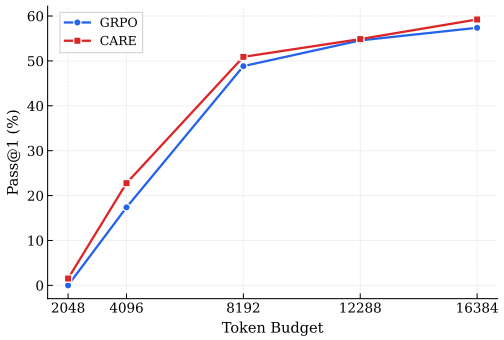


(d) MATH500

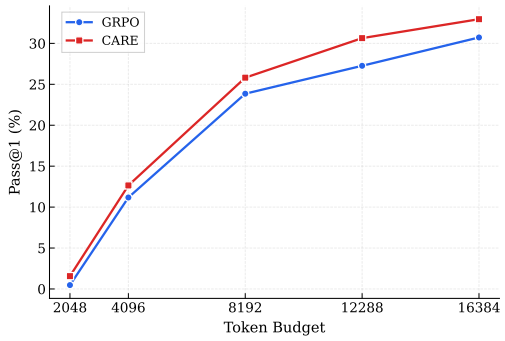
Figure 26: Pass@1 under varying token budgets for **Qwen3-4B**.



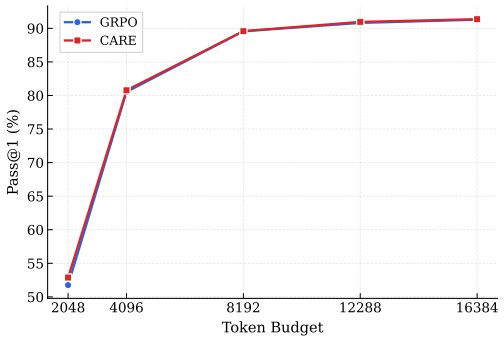
(a) AIME 2025



(b) AIME 2026



(c) HMMT 2026



(d) MATH500

Figure 27: Pass@1 under varying token budgets for **Qwen3-8B**.

G Statistical Significance Analysis

This section analyzes the main results across random seeds. We additionally include ALP (Xiang et al., 2025) and DAST (Shen et al., 2025), two efficient-reasoning baselines discussed in Section 2, and compare them with Base, GRPO, and CARE across Qwen3-1.7B, Qwen3-4B, and Qwen3-8B.

G.1 Evaluation Protocol

For each model–method combination, we conduct three independent evaluation runs using different random seeds on AIME 2025, AIME 2026, HMMT 2026, and MATH500 under the same evaluation settings as Section 6.1. Each entry reports the mean Pass@1, sample standard deviation, and the corresponding 95% Student-t confidence interval.

G.2 Experiment Results

Table 14 presents the complete multi-seed evaluation results. CARE achieves the highest mean Pass@1 in nine of the twelve model–benchmark combinations and remains competitive in the remaining evaluated settings. In particular, CARE consistently obtains the best mean performance on all four benchmarks for Qwen3-4B, and on AIME 2026, HMMT 2026, and MATH500 for Qwen3-8B. For Qwen3-1.7B, CARE performs best on AIME 2026 and HMMT 2026, while remaining close to the strongest baseline on AIME 2025. These results show that the performance of CARE remains robust across different random seeds and model sizes.

Table 14: Pass@1 (%) across three random seeds for Qwen3 family. Each entry reports mean ± sample SD [95% Student-t CI]. The best mean result is shown in bold.

Model Size	Method	AIME 2025	AIME 2026	HMMT 2026	MATH500
Qwen3-1.7B	Base	28.68 ± 1.65 [24.59, 32.77]	31.63 ± 1.29 [28.42, 34.85]	16.41 ± 0.88 [14.23, 18.60]	85.42 ± 2.02 [80.41, 90.44]
	GRPO	29.48 ± 0.81 [27.46, 31.50]	30.21 ± 0.81 [28.19, 32.23]	19.16 ± 1.38 [15.73, 22.59]	**87.15 ± 1.03 [84.60, 89.70]**
	ALP	**32.29 ± 0.72 [30.50, 34.08]**	30.21 ± 1.83 [25.66, 34.76]	20.14 ± 0.67 [18.47, 21.81]	85.90 ± 0.16 [85.51, 86.29]
	DAST	26.22 ± 0.22 [25.68, 26.75]	28.96 ± 1.09 [26.26, 31.66]	16.22 ± 0.36 [15.33, 17.12]	83.14 ± 0.13 [82.82, 83.46]
	CARE	32.26 ± 1.50 [28.54, 35.97]	**32.64 ± 0.87 [30.47, 34.81]**	**21.28 ± 0.44 [20.19, 22.36]**	86.66 ± 1.09 [83.96, 89.37]
Qwen3-4B	Base	46.88 ± 0.63 [45.30, 48.45]	51.49 ± 0.59 [50.02, 52.96]	24.27 ± 0.85 [22.17, 26.38]	89.57 ± 1.11 [86.80, 92.33]
	GRPO	51.70 ± 2.03 [46.65, 56.75]	53.44 ± 0.31 [52.66, 54.21]	27.71 ± 1.04 [25.12, 30.31]	92.16 ± 0.18 [91.71, 92.61]
	ALP	51.56 ± 1.17 [48.65, 54.48]	51.28 ± 1.33 [47.99, 54.58]	28.03 ± 0.16 [27.62, 28.44]	90.67 ± 0.06 [90.53, 90.82]
	DAST	34.48 ± 1.93 [29.69, 39.27]	37.36 ± 0.51 [36.08, 38.64]	20.68 ± 0.39 [19.70, 21.65]	86.79 ± 0.08 [86.59, 86.99]
	CARE	**52.22 ± 0.60 [50.73, 53.72]**	**54.90 ± 0.79 [52.94, 56.85]**	**28.38 ± 0.20 [27.89, 28.87]**	**92.27 ± 0.13 [91.95, 92.60]**
Qwen3-8B	Base	46.42 ± 0.64 [44.84, 48.00]	52.26 ± 0.12 [51.96, 52.56]	26.23 ± 0.57 [24.82, 27.64]	88.16 ± 0.10 [87.92, 88.40]
	GRPO	**56.04 ± 0.91 [53.79, 58.30]**	57.50 ± 0.95 [55.13, 59.87]	32.67 ± 0.49 [31.45, 33.89]	90.65 ± 0.18 [90.20, 91.09]
	ALP	53.54 ± 0.45 [52.41, 54.67]	57.15 ± 0.75 [55.28, 59.02]	32.20 ± 0.43 [31.12, 33.27]	90.65 ± 0.13 [90.34, 90.97]
	DAST	52.08 ± 0.48 [50.90, 53.27]	57.26 ± 0.80 [55.28, 59.23]	29.55 ± 0.09 [29.31, 29.78]	90.33 ± 0.04 [90.23, 90.43]
	CARE	54.93 ± 0.63 [53.37, 56.49]	**58.30 ± 0.66 [56.66, 59.94]**	**33.18 ± 0.67 [31.52, 34.83]**	**92.72 ± 0.08 [92.52, 92.93]**

H Other Experimental Details

This section provides additional implementation details that supplement the configurations reported in Section 6 and Appendix A, including training prompts and answer extraction.

H.1 Training Prompt

Training Prompt

Let's think step by step, and put your final answer within \boxed{}.

H.2 Answer Extraction

We extract the final answer from each model response by locating the last `\boxed{...}` expression and extracting its content, following the extraction protocol from Hendrycks et al. (2021).

I CASE STUDY

This section presents representative examples to illustrate the behaviors discussed in the paper.

I.1 LENGTH COLLAPSE IN QWEN3-1.7B UNDER SHORT-REWARD

Example: Length Collapse *AIME 2025, Problem_idx 1*

Question: Find the sum of all integer bases $b > 9$ for which 17_b is a divisor of 97_b. **Answer:** 70

Qwen3-1.7B (Base) 7,162 Tokens ✓

<think>
Okay, so I need to find all integer bases b greater than 9 where 17_b is a divisor of 97_b. ···
So, $17_b = b + 7$ and $97_b = 9b + 7$. We need $(9b + 7)/(b + 7)$ to be an integer. ···
Let me write $9b + 7 = 9(b + 7) - 56$. Ah! So this is a key insight. ···
So $b + 7$ must be a positive divisor of 56, and $b + 7 > 16$. ···
...
Therefore $b = 21$ or $b = 49$. The answer is 70.
</think>
Final Answer
\boxed{70}

Qwen3-1.7B (GRPO + Short Reward) 8 Tokens ✗

</think>
\boxed{6}

I.2 COMPARISON OF GRPO AND CARE RESPONSES

Example: Reasoning Trace in GRPO and CARE *HMMT 2026, Problem_idx 1*

Question: A line intersects the graph of $y = x^2 + \frac{2}{x}$ at three distinct points. Given that the x-coordinates of two of the points are 6 and 7, compute the x-coordinate of the third point. **Answer:** $-1/21$

GRPO 13,569 Tokens ✗

<think>
Okay, so I need to find the x-coordinate of the ···
...
Therefore, the third root is 5/21. Therefore, the ···
...
Wait, that seems straightforward. Let me check ···
...
Wait, that's not zero. So something is wrong here. That means my calculation is wrong. Hmm. So ···
...
But when I plug $x = 7$ into the cubic equation, it gives -42, but according to Vieta's formula, the ···
...
But how? This is very confusing. I must have ···
...
Alternatively, maybe $x = 6$ and $x = 7$ are not ···
...
I will trust Vieta's formula here. The x-coordinate of the third point is $5/21$.
</think>
...
Final Answer
\boxed{5/21}

CARE 1,591 Tokens ✓

<think>
Okay, so I need to find the x-coordinate of the third point where a line intersects the graph of y ···
...
Let's say the line has equation $y = mx + b$. Then, setting this equal to the function $nmx + b = x^2$ ···
...
Then, by Vieta's formula, the sum of roots of the cubic equation is equal to the coefficient of x^2 term ···
...
Multiply both sides by x: $x^3 - mx^2 - bx + 2 = 0$. The product of the roots is -2. Therefore, $6 \cdot 7 \cdot r$ ···
...
Let me verify. If the product of the roots is -2, ···
...
Therefore, the answer is $-1/21$.
</think>
...
Final Answer
\boxed{-1/21}

I.3 Cyclic Reasoning: Representative Cases

In this part, we present three representative cases from the unstable interval of Qwen3-1.7B and Qwen3-4B under long-reward training, where keyword-flagged sentences correspond to semantic repetition. Cyclic keywords are marked in **red bold**.

Example 1 *Qwen3-1.7B · Long-Reward*

Question: Simplify $90r - 44r$. **Answer:** $46r$

Selection rationale. The unstable response re-derives the *identical* subtraction $90 - 44 = 46$ immediately after stating it, triggered by "Alternatively."

Unstable Interval — Step 40 — 4,139 Tokens

<think>
...
Okay, so I need to simplify the expression 90r minus 44r. Let me start by thinking about what this means. ···
...
Hmm, 90 minus 40 is 50, and then minus 4 more would be 46. So, 90 minus 44 is 46. Therefore, 90r minus 44r is 46r.
Alternatively, maybe I can use algebraic manipulation. Let me write it as: $90r - 44r = (90 - 44)r$. Because when you have the same variable, you can combine the coefficients. So, $90 - 44$ is 46, so it's $46r$. ···
...
</think>
...
Final Answer
The simplified form of $90r - 44r$ is \boxed{$46r$}.

Example 2 *Qwen3-1.7B · Long-Reward*

Question: A room is 19 feet long and 11 feet wide. Find the ratio of the length of the room to its perimeter. Express your answer in the form $a : b$. **Answer:** $19:60$

Selection rationale. The unstable response re-derives the *identical* perimeter calculation $2\times19+2\times11 = 60$ immediately after stating it, triggered by "Wait."

Unstable Interval — Step 43 — 6,144 Tokens

<think>
Okay, so I need to find the ratio of the length of the room to its perimeter. The room is 19 feet long and 11 feet wide. ···
...
So the perimeter is 60 feet?
Wait, let me check that again. 2×19 is 38, 2×11 is 22. $38 + 22$. Hmm, $38 + 22$. Yeah, $38 + 20$ is 58, plus 2 is 60. ···
...
For example, if the question was about the ratio of the width to the perimeter, then it would be $11:60$, but the problem says length to perimeter. **Alternatively**, maybe I need to think if there's a mistake in the problem ···

Example 3 *Qwen3-4B · Long-Reward*

Question: The operation $\#$ is defined as $a\#b = a + \frac{a}{b}$. What is the value of $6\#2$? **Answer:** 9

Selection rationale. The unstable response returns to the same substitution-and-calculation path $6\#2 = 6 + \frac{6}{2} = 6 + 3 = 9$ after considering an alternative interpretation, triggered by "But."

Unstable Interval — Step 37 — 5,193 Tokens

<think>
Okay, so I need to figure out what $6\#2$ is, given that the operation $\#$ is defined as $a\#b = a + (a/b)$. ···
...
So, it's definitely a plus (a divided by b). So, in this case, with a being 6 and b being 2, it's 6 plus 6 divided by 2.
But that would be different. Let me check the original problem again.··· **Alternatively**, if it were written without parentheses, ··· So, going back to the original calculation: $6\#2$ is $6 + (6/2)$.···
...
</think>
...
Final Answer
\boxed{9}

J COMPUTE RESOURCES

Each training run (one model × one reward configuration) required approximately 120 hours on 8×A100 80GB GPUs. All evaluations were performed on 8×NVIDIA RTX 4080 Super GPUs.

K LIMITATIONS

Our evaluation is limited to mathematical reasoning tasks. The generalizability of CARE to other reasoning domains, such as code generation or commonsense reasoning, remains to be explored. In addition, the current direction signal $d(q)$ is a discrete ternary value $(-1, 0, +1)$. A continuous variant may provide finer-grained control but is left for future work.