Aware First, Think Less: Dynamic Boundary Self-Awareness Drives Significant Gains in Reasoning Efficiency in Large Language Models

Qiguang Chen[1*], Dengyun Peng[1*], Jinhao Liu[1], Huikang Su[1], Jiannan Guan[1], Libo Qin[2†], Wanxiang Che[1†]

[1]Research Center for Social Computing and Interactive Robotics, Harbin Institute of Technology
[2]School of Computer Science and Engineering, Central South University
{qgchen,dypeng}@ir.hit.edu.cn, lbqin@csu.edu.cn, car@ir.hit.edu.cn

Abstract

Recent advancements in large language models (LLMs) have greatly improved their ability to perform complex reasoning tasks through Long Chain-of-Thought (CoT). However, this approach often results in substantial redundancy, impairing computational efficiency and causing significant delays in real-time applications. To improve the efficiency, current methods often rely on human-defined difficulty priors, which do not align with the LLM's self-awared difficulty, leading to inefficiencies. In this paper, we introduce the Dynamic Reasoning-Boundary Self-Awareness Framework (DR. SAF), which enables LLMs to dynamically assess and adjust their reasoning depth in response to problem complexity. DR. SAF integrates three key components: Boundary Self-Awareness Alignment, Adaptive Reward Management, and a Boundary Preservation Mechanism. These components allow models to optimize their reasoning processes, balancing efficiency and accuracy without compromising performance. Our experimental results demonstrate that DR. SAF achieves a 49.27% reduction in total response tokens with minimal loss in accuracy. The framework also delivers a 6.59x gain in token efficiency and a 5x reduction in training time, making it well-suited to resource-limited settings. During extreme training, DR. SAF can even surpass traditional instruction-based models in token efficiency with more than 16% accuracy improvement.

Code — https://github.com/sfasfaffa/DR_SAF

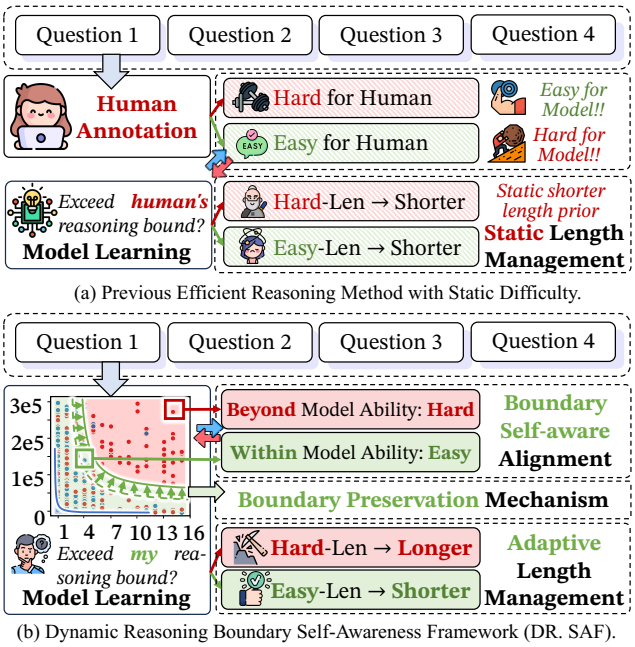


Figure 1: Traditional efficient reasoning training methods (a) primarily determine the difficulty of questions based on human-defined priors, while our dynamic reasoning boundary self-awareness framework (b) judged the difficulty of questions based on model self-awared reasoning boundary.

1 Introduction

Recent advancements in large language models (LLMs) have demonstrated their remarkable ability to tackle complex reasoning tasks, particularly through the use of Long Chain-of-Thought (Long CoT) techniques (Wei et al. 2022; Guo et al. 2025; Chen et al. 2025; Li et al. 2025b). In contrast to the shorter chain-of-thought (Short CoT) typically employed in traditional LLMs, Long CoT reasoning involves a more detailed and iterative process of exploration and reflection within a given problem space. This process is facilitated by inference-time scaling (Guo et al. 2025; Zhang et al. 2025b; Jaech et al. 2024). As a result, Long CoT has significantly advanced areas such as mathematical and logical reasoning. Moreover, it has provided new insights into the role of supervised fine-tuning (SFT) and reinforcement learning (RL) in enhancing the learning and exploration of extended reasoning chains (Qin et al. 2024; Min et al. 2024).

While achieving promising performance, the Long CoT paradigm generates many redundant tokens, which severely harms computational efficiency and leads to high application latency (Wu et al. 2025; Sui et al. 2025; Feng et al. 2025). To mitigate this issue, several approaches focus on optimizing reasoning length (Tu et al. 2025; Shi et al. 2025). Specifically, recent studies apply compression techniques, including static pruning thresholds that remove intermediate

*Equal contribution
†Corresponding Author

tokens (Luo et al. 2025b) and adaptive routing to more efficient modules (Lu et al. 2025; Ling et al. 2025; Liang et al. 2025). Other approaches improve the model's inherent ability to produce concise reasoning paths (Tu et al. 2025; Shi et al. 2025). For instance, AdaptThink (Zhang et al. 2025a) and DAST (Shen et al. 2025) dynamically adjust reasoning depth based on predefined measures of problem complexity, and Huang et al. (2025) extend these paradigms to human-designed adaptive budgeting. However, as illustrated in Figure 1(a), current methods still depend on manually designed static priors of difficulty and target length, while overlooking each LLM's evolving reasoning boundaries during training (Chen et al. 2024a). As a result, tasks initially labeled as "simple" may still require long exploration for an LLM with limited capability, whereas those labeled as "complex" may later be solved intuitively via shorter reasoning, yielding inefficient reasoning and suboptimal performance.

To tackle this challenge, as shown in Figure 1 (b), we present the Dynamic Reasoning-Boundary Self-Awareness Framework (DR. SAF), which assesses problem difficulty relative to a model's reasoning capacity. Specifically, DR. SAF consists of three key components: (1) **Boundary Self-Awareness Alignment** enables LLMs to recognize their real-time reasoning boundaries. This self-awareness allows the model to assess the difficulty of a given question based on its own capabilities, prompting self-guided adjustments in reasoning depth and answer length. (2) **Adaptive Length Management** further refines efficiency by adapting the reward according to the model's real-time boundaries. It encourages longer exploration beyond the Completely Infeasible Reasoning Boundary (CIRB) and shorter reasoning within the Completely Feasible Reasoning Boundary (CFRB), ensuring that the model does not oversimplify and compromise quality. (3) **Boundary Preservation Mechanism** maintains stability by preventing the collapse of real-time reasoning boundaries during training, ensuring that all correct responses receive non-negative reinforcement. These innovations address the traditional trade-off issue between efficiency and accuracy, enabling models to dynamically adjust their reasoning depth based on their capabilities.

DR. SAF enhances a model's boundary self-awareness, enforces boundary-driven length adaptation, and preserves these boundaries, enabling real-time control of reasoning depth without degrading performance. When evaluated on six public benchmarks, applying DR. SAF to the distilled Qwen-2.5 model reduces total response tokens by 49.27% and achieves state-of-the-art token efficiency. Compared with distilled model, DR. SAF delivers a 6.59x gain in token efficiency. After additional continual training, the extremely compressed DR. SAF model can even surpass traditional instruction-based models in token efficiency and increase accuracy by more than 16%. On the distilled Qwen-3 model, DR. SAF reduces training steps by 80% compared with previous reinforcement-learning methods, making it attractive for deployments with limited computational resources.

Our contributions can be summarized as follows:

- We first point out the limitations of existing efficient reasoning methods, which often rely on human-annotated difficulty priors that do not align with LLMs' reasoning requirements. This misalignment leads to inefficient reasoning processes and suboptimal performance.
- We propose a novel DR. SAF framework, which enables models to dynamically assess their own reasoning boundaries, adaptively manage length reward signals based on problem feasibility, and prevent models from reasoning boundary collapse.
- We demonstrate the effectiveness of DR. SAF through extensive experiments on challenging benchmarks, revealing substantial improvements in efficiency. The extreme speedup can even enable LLMs to surpass the token efficiency of instruction models, while maintaining a 16% improvement in accuracy.

2 Preliminaries

2.1 The Efficient Reasoning Objective

Given an input x, the model generates an output $y = \{S, a\}$, which consists of a reasoning step trajectory $S = (s_1, s_2, \ldots, s_T)$ and a final answer a.

The objective of efficient reasoning is to develop a policy that minimizes the reasoning path length while preserving accuracy. Formally, the efficient reward is expressed as:

$$R_{\text{Eff}}(y|x) = R_{\text{Acc}}(y|x) + \gamma R_{\text{Len}}(y|x), \tag{1}$$

where $R_{\text{Acc}}(y|x)$ provides a reward of 1 when y is correct, $R_{\text{Len}}(y|x) \propto -\ell_y$ is the length reward, which is negatively correlated with the response length ℓ_y, and γ is a constant hyperparameter.

2.2 Group Relative Policy Optimization

We utilize Group Relative Policy Optimization (GRPO) to optimize LLMs, an efficient, critic-free reinforcement learning method that reduces memory and computational costs. Specifically, GRPO operates through group-wise advantage estimation. For a given input, the policy model π_θ generates a group of k outputs, $\mathcal{Y} = \{y_1, \ldots, y_k\}$, evaluated by a reward function to yield reward group $\mathcal{R}_{\text{Eff}} = \{R_{\text{Eff}}(y_i|x)\}_{i=1}^k$. The advantage for each output is computed by normalizing its reward against the group's statistics:

$$\mathcal{A}(\mathcal{Y}|x) = \frac{\mathcal{R}_{\text{Eff}} - \mu_{\mathcal{R}}}{\sigma_{\mathcal{R}} + \epsilon} \tag{2}$$

where $\mu_{\mathcal{R}}$ and $\sigma_{\mathcal{R}}$ are the mean and standard deviation of the rewards group $\mathcal{R}_{\text{Eff}}$, respectively. ϵ is a small constant to prevent division by zero. The policy is refined by minimizing the GRPO loss $\mathcal{L}_{\text{GRPO}}$, which amplifies actions with high advantage and penalizes those with low advantage.

3 Methodology

To compress efficiently without sacrificing accuracy, we introduce a three-module framework (see Figure 2): (1) Boundary Self-Awareness Alignment enables model to gauge question difficulty. (2) Adaptive Length Management applies a discrete length-reward schedule that scales with difficulty and reasoning bounds, preventing harmful over-compression. (3) Boundary Preservation Mechanism stabilizes optimization through advantage reshaping. Together,

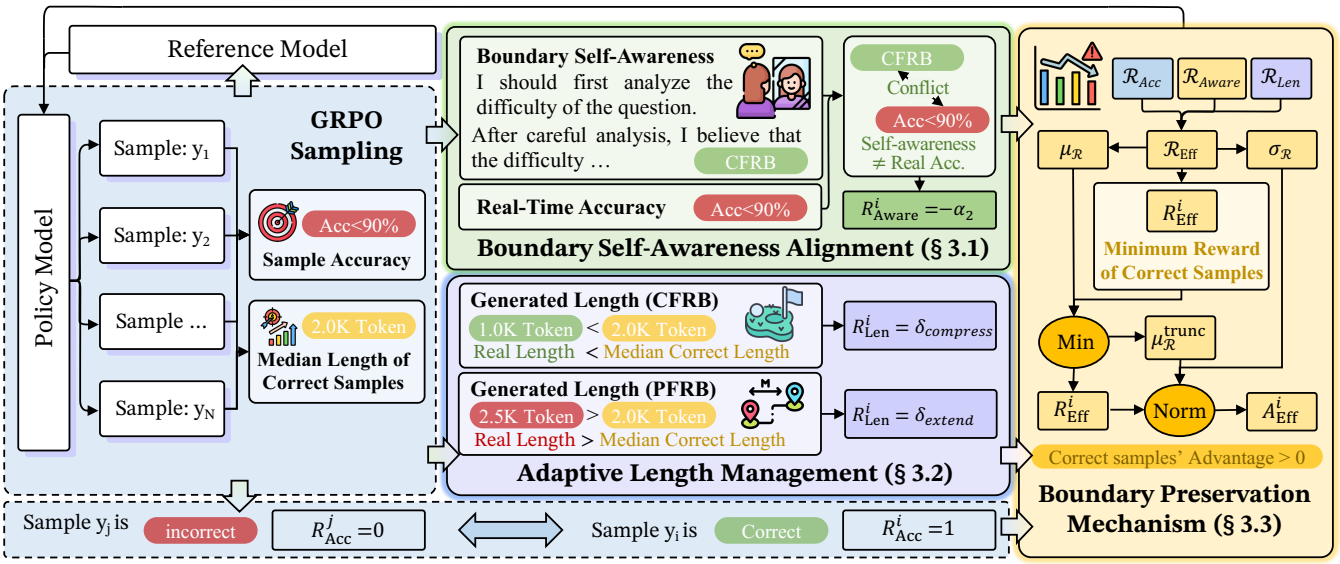


Figure 2: Main pipeline of Dynamic Reasoning-Boundary Self-Awareness Framework (DR. SAF), including Boundary Self-Awareness Alignment (BSA), Adaptive Length Management (ALM), and Boundary Preservation Mechanism (BPM).

these modules, collectively called DR. SAF, offer a theory-guided solution to balance efficiency and accuracy. Formal proofs of each module's effectiveness are in the Appendix.

3.1 Boundary Self-Awareness Alignment

First, the model develops self-awareness of whether a given task falls within its reasoning capabilities. As shown in the green area of Figure 2, the model calibrates perceived task difficulty against its real-time accuracy; any gap between expected and observed performance incurs a reward penalty. Specifically, we employ Boundary Self-Awareness Alignment (BSA) with a format-based reward. Guided by carefully designed prompts, the model evaluates its proficiency on each problem: Inspired by Chen et al. (2024a), if it determines a problem to be fully mastered, it classifies it as within its Completely Feasible Reasoning Boundary (CFRB) and provides more concise solution; otherwise, it assigns the problem to its Partially Feasible Reasoning Boundary (PFRB), initiating a deeper reasoning process.

Next, to enable adaptive boundary awareness, BSA assesses the model's accuracy across multiple runs of the same problem. Following Chen et al. (2024a), we label problems with accuracy above 90% as CFRB and those below 90% as PFRB. When the model's boundary classification aligns with the problem's true difficulty, resulting in a correct CFRB or PFRB judgment followed by a correct answer, it receives a positive reward. Conversely, if it mislabels a PFRB problem as CFRB and then answers incorrectly, it incurs a negative reward.

Formally, the reward function is defined as:

$$R_{\text{Aware}}(y|x) = \begin{cases} +\alpha_1, & \text{if } \text{Acc}(\mathcal{Y}|x) \geq 90\% \wedge \\ & \text{Aware}(x) < \texttt{CFRB}; \\ +\alpha_1, & \text{if } \text{Acc}(\mathcal{Y}|x) < 90\% \wedge \\ & \texttt{CFRB} \leq \text{Aware}(x) \leq \texttt{PFRB}; \\ -\alpha_2, & \text{otherwise}, \end{cases} \tag{3}$$

where α_1 and α_2 are positive constants that scale the rewards, $\text{Acc}(\mathcal{Y}|x)$ is the model's empirical accuracy on input x, and $\text{Aware}(x)$ denotes the model's self-assessed difficulty for x. This framework continuously calibrates the model's self-awareness of reasoning boundaries based on performance feedback. We first establish golden RB based on the actual accuracy of the rollout group. Then, each response is assigned a predicted RB, with a positive reward for a match and a negative reward for a mismatch. As a result, rewards for different responses to the same problem vary.

3.2 Adaptive Length Management

Unlike traditional compression tasks, which focus on unified length penalties, Adaptive Length Management (ALM) introduces staged incentives to generate suitable response lengths. As illustrated in the purple area of Figure 2, tasks in CFRB that LLM already masters receive compression rewards, driving concise reasoning. For low-accuracy tasks beyond Completely Infeasible Reasoning Boundary (CIRB), we give extension rewards to longer incorrect answers, prompting the model to elaborate for deeper exploration.

Specifically, based on k sampling results, we select a correct sample set $\mathcal{C}$. First, we determine the minimum number of tokens required to maintain a question within the CFRB. Formally, we define $\bar{\ell}_{\texttt{CFRB}}$, the median response length for correct samples in $\mathcal{C}$, as the model's required length for mastery within CFRB. Based on this, we then define two reward

types: (1) the ***compression reward*** δ_{comp} for fully mastered questions. It rewards answers under CFRB whose length ℓ should be below the CFRB mean $\bar{\ell}_{\texttt{CFRB}}$. (2) the ***extension reward*** δ_{ext} for exploration-needed questions, those beyond CIRB. Here the answer length ℓ should exceed $\bar{\ell}_{\texttt{CFRB}}$ (if no correct sample exists, the length threshold will degrade to the average length of all samples $\bar{\ell}_{\texttt{All}}$). Formally, the adaptive reward for ALM is:

$$R_{\text{Len}}(y|x) = \begin{cases} \delta_{\text{comp}} & \text{if Acc}(\mathcal{Y}|x){>}90\% \wedge \ell \leq \bar{\ell}_{\texttt{CFRB}} \\ \delta_{\text{ext}} & \text{if Acc}(\mathcal{Y}|x){<}10\% \wedge \ell > \bar{\ell}_{\texttt{CFRB}} \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

where $\delta_{\text{comp}} < R_{\text{Acc}}$ and $\delta_{\text{ext}} < R_{\text{Acc}}$ are positive constants smaller than the accuracy reward R_{Acc}.

3.3 Boundary Preservation Mechanism

Training models with only length penalties and correctness rewards often leads to a common issue known as boundary collapse. In this case, the model excessively compresses reasoning chains, causing the advantage of correct responses to fall below zero. As a result, valid reasoning boundaries collapse, and infeasible ones arise, both undermining performance and destabilizing training. As illustrated in yellow part of Figure 2, we address this by enforcing non-negative advantages for all correct responses.

Let $\mathcal{C}$ represent the set of correct responses, which defines the feasible region. This region includes responses that meet all correctness, length, and awareness criteria within the CFRB, as well as correct responses that may violate secondary preferences (such as length) within the PFRB. Both categories are considered feasible, ensuring that every output $y_i \in \mathcal{C}$ receives a non-negative advantage. Formally, for a given input x, we sample a group of k outputs, $\mathcal{Y} = \{y_1, \ldots, y_k\}$, and calculate the total efficient rewards $\mathcal{R}_{\text{Eff}} = \{R_{\text{Eff}}(y_i|x)\}_{i=1}^{k}$ as follows:

$$R_{\text{Eff}}(y_i|x) = R_{\text{Acc}}(y_i|x) + R_{\text{Len}}(y_i|x) + R_{\text{Aware}}(y_i|x). \quad (5)$$

The Boundary Preservation Mechanism (BPM) ensures that correct responses, regardless of length, are not unduly suppressed, thereby preventing boundary collapse. Specifically, we utilize the efficient reward function and the method for calculating boundary preservation advantages. To achieve this, truncated-mean normalization is applied to the output sample group. The advantages of boundary preservation mechanism A are computed as follows:

$$\mu_{\mathcal{R}}^{\text{trunc}} = \min(\mu_{\mathcal{R}}, \min_{y_i \in \mathcal{C}} R_{\text{Eff}}(y_i|x)), \quad (6)$$

where $\mathcal{C}$ is the set of correct responses. This step ensures that the decision boundary for correct responses is preserved, preventing the model from assigning negative advantages to correct but length-variant answers. Next, the boundary preservation advantages are computed as:

$$\mathcal{A}_{\text{Pre}}(\mathcal{Y}|x) = \frac{\mathcal{R}_{\text{Eff}} - \mu_{\mathcal{R}}^{\text{trunc}}}{\sigma_{\mathcal{R}} + \epsilon}, \quad (7)$$

where $\mu_{\mathcal{R}}$ and $\sigma_{\mathcal{R}}$ are the untruncated mean and standard deviation of reward group $\mathcal{R}_{\text{Eff}}$. By bounding the group mean as $\mu_{\mathcal{R}}^{\text{trunc}}$, we ensure:

$$\forall y_i \in \mathcal{C}: \quad \mathcal{A}_{\text{Pre}}(y_i|x) = \frac{R_{\text{Eff}(y_i|x)} - \mu_{\mathcal{R}}^{\text{trunc}}}{\sigma_{\mathcal{R}} + \epsilon} \geq 0. \quad (8)$$

This safeguard guarantees that correct responses, regardless of length variation, always receive a non-negative advantage. By enforcing this, the Boundary Preservation Mechanism ensures that valid outputs never receive suppressed advantages, thus preventing boundary collapse.

4 Experiments

4.1 Experimental Setup

We utilize verl (Sheng et al. 2024) as the reinforcement learning framework on 8 A100-80G GPUs. We randomly sample 5,000 instances from the DeepMath103K (He et al. 2025) as training set, and trained DR. SAF based on two LLMs, R1-distill-Qwen-2.5-7B (Guo et al. 2025) and R1-distill-Qwen-3-8B (Guo et al. 2025). We validate the effectiveness of strategies on AIME24 (AIME 2024), GSM8K (Cobbe et al. 2021), Math-500 (Lightman et al. 2024), AMC23 (AMC 2023), OlympiadBench (He et al. 2024), and AIME25 (AIME 2025). We report three metrics: Accuracy (ACC in %), average response token length (LEN), and token efficiency (EFF). The Token Efficiency (EFF) is defined as the ratio of Accuracy to Length (EFF = ACC / LEN in %), serving as an indicator of the correctness and reasoning efficiency trade-off.

For comprehensive comparison, we adopt three representative paradigms as baselines for efficient reasoning: (1) **Prompting Strategies:** Dynasor-CoT (Fu et al. 2025b) and DEER (Yang et al. 2025) activate early-exit mechanisms during reasoning; ThinkSwitcher (Liang et al. 2025) trains a switcher to dynamically choose between long and short CoT. (2) **Offline Strategies:** OverThink (Chen et al. 2024b) fine-tunes on the shortest generated answers. Spirit (Cui et al. 2025) and ConCISE-SimPO (Qiao et al. 2025) prune tokens based on confidence scores via supervised fine-tuning or direct preference training. AdaptThink (Zhang et al. 2025a) and DAST (Shen et al. 2025) incorporate human-defined difficulty priors to learn efficient reasoning trajectories. **Online Strategies:** Length-Penalty (Arora and Zanette 2025) encourages compress all outputs; FEDH (Ling et al. 2025) applies a human-defined length prior to promote concise reasoning process. Additionally, we compare the token efficiency of instruction models, like Qwen2.5-Ins (Yang et al. 2024a) and Qwen2.5-Math (Yang et al. 2024b).

4.2 Experimental Results

Offline methods yield high accuracy but are less token-efficient than online methods. Offline approaches use gold-standard reasoning trajectories, raising accuracy by more than 6% on AIME24 (see in Table 1); however, their longer reasoning chains increase token consumption. In contrast, online methods are more economical, forcing fewer tokens and shorter reasoning paths while maintaining competitive accuracy. Thus, whereas offline methods maximize precision, online methods achieve a superior balance between accuracy and token efficiency.

Model Name	GSM8K			MATH500			AIME24			AMC			OlymBench			AIME25		
	ACC	LEN	EFF	ACC	LEN	EFF	ACC	LEN	EFF	ACC	LEN	EFF	ACC	LEN	EFF	ACC	LEN	EFF
Qwen2.5-7B-Ins	90.9	279	32.58	74.2	567	13.09	12.0	1016	1.18	47.5	801	5.93	39.2	827	4.74	7.6	1240	0.61
Qwen2.5-7B-Math	93.2	439	21.23	63.4	740	8.57	19.0	1429	1.33	62.5	1022	6.12	31.5	1037	3.04	4.0	2562	0.16
Qwen2.5-7B-Math-Ins	95.2	323	29.47	81.4	670	12.15	10.3	1363	0.76	60.0	1029	5.83	38.9	1027	3.79	9.3	2087	0.45
R1-Distill-Qwen2.5-7B	92.4	1833	5.04	90.8	3854	2.36	49.2	10200	0.48	90.0	6476	1.39	66.1	7789	0.85	35.0	10518	0.33
+ ThinkSwitcher	**92.5**	1389	6.66	91.3	3495	2.61	48.3	7936	0.61	–	–	–	57.0	5147	1.11	37.5	6955	0.54
+ Dynasor-CoT	89.6	1285	6.97	89.4	2661	3.36	46.7	12695	0.37	85.0	5980	1.42	–	–	–	–	–	–
+ DEER	90.6	917	9.88	89.8	2143	4.19	49.2	9839	0.50	85.0	4451	1.91	–	–	–	–	–	–
+ OverThink	91.4	879	10.39	**92.9**	2405	3.86	50.0	9603	0.52	–	–	–	–	–	–	–	–	–
+ Spirit	87.2	687	12.68	90.8	1765	5.14	38.3	6926	0.55	–	–	–	–	–	–	–	–	–
+ ConCISE-SimPO	92.1	715	12.88	91.0	1945	4.68	48.3	7745	0.62	–	–	–	–	–	–	–	–	–
+ DAST	86.7	459	18.89	89.6	2162	4.14	45.6	7578	0.60	–	–	–	–	–	–	–	–	–
+ AdaptThink	91.0	309	29.45	92.0	1875	4.91	**55.6**	8599	0.65	85.0*	4265*	1.99*	58.4*	5988*	0.98*	**38.3***	10380*	0.37*
+ Length-Penalty	87.2	263	33.16	89.1	2121	4.20	51.9	7464	0.70	82.5	4411	1.87	**59.8**	4919	1.22	33.3	8902	0.37
+ FEDH	90.1	218	41.33	88.5	1306	6.50	42.3	7242	0.58	–	–	–	–	–	–	–	–	–
+ **DR. SAF**	88.1	**162**	**54.38**	88.3	**1061**	**8.32**	50.6	**6288**	**0.80**	**90.0**	**3096**	**2.91**	59.4	**3259**	**1.82**	38.2	**6764**	**0.56**
R1-Distill-Qwen3-8B	94.2	2135	4.41	90.6	7051	1.28	**67.9**	20155	0.34	83.5	11931	0.70	60.1	12895	0.47	**62.9**	20992	0.30
+ FEDH*	**94.4**	2014	4.69	92.6	6761	1.37	61.3	13463	0.42	94.7	11928	0.79	63.5	12353	0.51	46.7	14730	0.32
+ Length-Penalty*	93.3	604	15.45	92.4	2581	3.58	63.7	12303	0.52	89.2	6166	1.45	68.4	7383	0.93	54.7	12446	0.44
+ **DR. SAF**	92.3	**521**	**17.72**	**93.3**	**2168**	**4.30**	66.0	**9807**	**0.67**	**95.6**	**4003**	**2.39**	**71.3**	**5766**	**1.24**	57.9	**10692**	**0.54**

Table 1: Performance comparison. **Bold** marks the best baseline score per metric. For each method we report its most token-efficient variant. Here, "": prompting strategies, "": offline strategies, "": online strategies. Rows are ordered by token efficiency on GSM8K. "∗" indicates results reproduced in this study.

DR. SAF performs state-of-the-art token efficiency with minimal accuracy degradation. We next report the main results on token efficiency, overall efficiency, and accuracy. As shown in Table 1, DR. SAF demonstrates superior performance in reasoning length and token efficiency. DR. SAF reduces the average response token count by 26.53% compared with Length Penalty.

The gains are most pronounced on GSM8K with all data within CFRB, where DR. SAF delivers over **90%** shorter reasoning and nearly **10x** higher token efficiency than the distilled backbone.

DR. SAF markedly increases LLM token efficiency relative to static difficulty-based reasoning. As shown in Table 1, DR. SAF is evaluated against two representative static baselines, AdaptThink and DAST, whose difficulty and length are fixed based on human prior. Because these baselines cannot adapt to variations in complexity relative to the model's capabilities, they often allocate more tokens than necessary. In contrast, DR. SAF dynamically updates its efficiency during reasoning, reducing the average token count by 34.33%. On GSM8K in particular, it achieves a token-efficiency rate of 54.38%, outperforming AdaptThink by more than 20%. This adaptive mechanism consistently delivers higher token efficiency across all benchmarks.

DR. SAF shows significant performance improvements on stronger LLMs. As shown in Table 1, compared to R1-Distill-Qwen3-8B, DR. SAF achieves comprehensive token efficiency gains: on GSM8K, token efficiency improves from 4.41 to 17.72 (**302%** increase) with minimal accuracy trade-off; on MATH500, both accuracy (90.6% to 93.3%) and token efficiency (1.28 to 4.30, **236%** improvement) increase; on AMC, accuracy rises from 83.5% to 95.6% while token efficiency improves by **241%**. These results demonstrate that DR. SAF achieves an excellent balance between computational efficiency and performance.

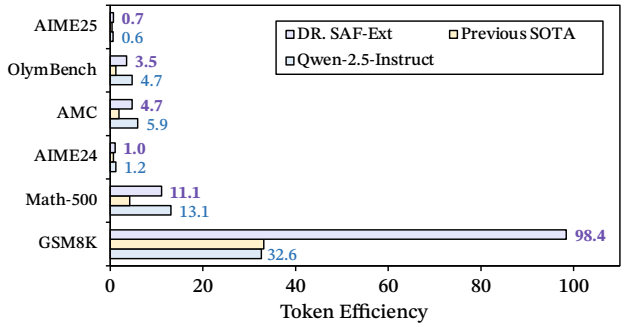


Figure 3: Comparing the extreme efficiency of DR. SAF (DR. SAF-Ext) with traditional instruction models and current SOTA reasoning efficient techniques.

4.3 Feature Analysis of DR. SAF

In this section, we analyze the key feature of DR. SAF by addressing three central questions: (1) Can DR. SAF outperform its original instruction backbone in token efficiency? (2) Does DR. SAF offer improved training speed? (3) Does DR. SAF focus solely on reasoning compression without enhancing overall performance?

Answer1: DR. SAF can achieve comparable, or even superior, token efficiency to traditional instruction models across all benchmarks. As shown in Figure 3, earlier techniques match instruction models only on simple datasets such as GSM8K, falling short by over 40% on more complex tasks. In contrast, the further compressed variant DR. SAF-Ext consistently maintains, and even exceeds, the token efficiency of instruction models across varying task complexities. On the CFRB benchmarks (GSM8K and MATH-500), it doubles the token efficiency of prior methods while matching instruction models. On average, DR. SAF improves token efficiency by 211% over previous SOTA approaches and

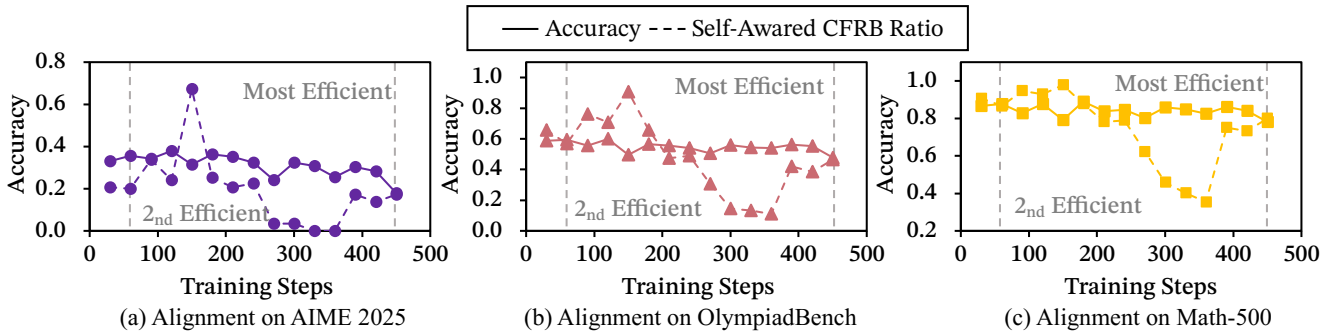


Figure 4: Training trajectory of BSA, shown as the predicted CFRB ratio plotted against the training steps.

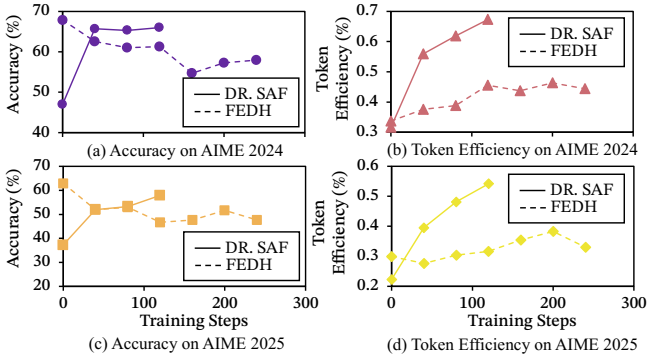


Figure 5: Training efficiency comparison of DR. SAF vs. FEDH on R1-Distill-Qwen-3-8B.

Model Name	ACC_{AVG}	LEN_{AVG}	EFF_{AVG}
DR. SAF	75.28	2773.2	13.65
w/o BSA	74.98	3219.9	8.04
w/o ALM	67.54	2105.1	11.96
w/o BPM	67.87	2543.6	13.65

Table 2: Ablation analysis of the model's average accuracy, length, and efficiency scores across GSM8K, MATH500, AIME24, AMC, and OlymBench.

increases accuracy by 16.15% relative to instruction models.

Answer2: DR. SAF achieves significant compression training speedup. Compared to previous RL methods, as shown in Figure 5, DR. SAF reduces training time by up to 5-6 times while maintaining high efficiency. This speedup is particularly evident in large-scale datasets, where DR. SAF minimizes the computational cost associated with model training. Benchmarks indicate that DR. SAF's compression strategy not only enhances training speed but also ensures minimal loss in model performance, with accuracy even improved. These results demonstrate DR. SAF's ability to achieve fast and efficient training, making it highly scalable for practical applications.

Answer3: DR. SAF achieves performance improvement during compression while length penalty-based methods decrease the performance. In contrast to length penalty-based methods, which often lead to performance degradation during compression, as shown in Figure 5 (a,c), DR. SAF demonstrates a unique ability to enhance model performance even as it compresses reasoning length. As a result, DR. SAF not only maintains high accuracy but also improves it in many cases.

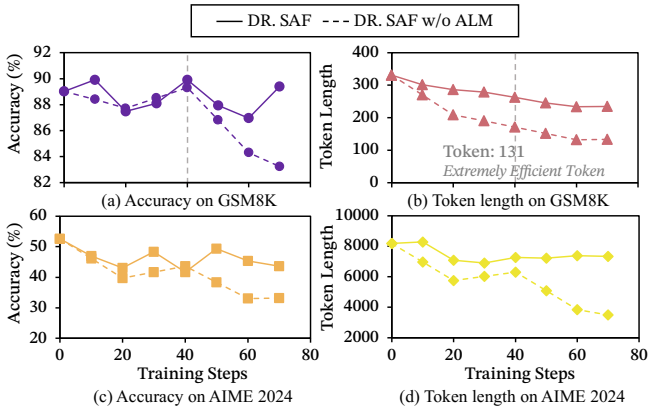
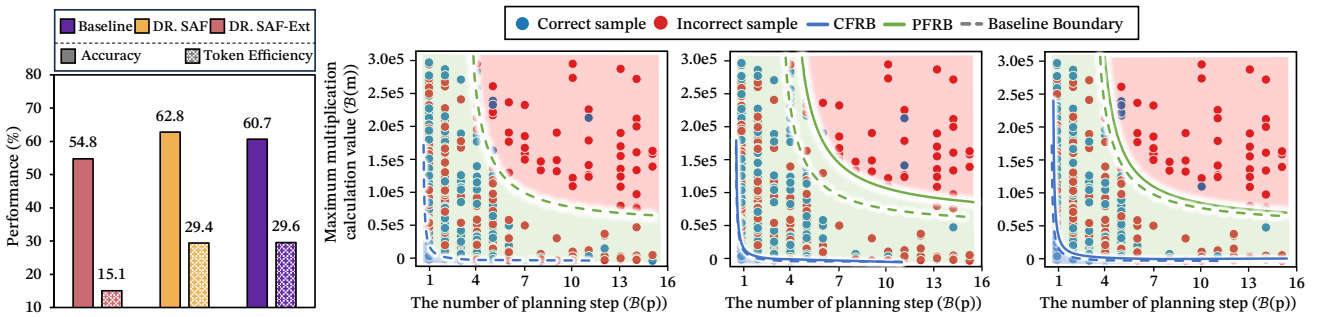


Figure 6: Trends in accuracy and response length during training with Adaptive Length Management.

4.4 Module Effect Analysis of DR. SAF

This section evaluates the key effects of each module in DR. SAF with three central questions: (1) Does BSA enhance the model's boundary self-awareness, thereby improving token efficiency? (2) Does ALM adaptively control token length to simultaneously improve accuracy and efficiency? (3) Does BPM prevent reasoning boundary collapse during compression training, thus preserving model performance?

Answer1: Boundary Self-Awareness Alignment is crucial for DR. SAF efficiency. We assess the effectiveness of the Boundary Self-Awareness Mechanism by ablating it from the DR. SAF. As shown in Table 2, token efficiency decreases by more than 40% without this component. Further, Figure 4 reveals that, during training, the model's predicted task difficulty progressively aligns with its actual reasoning accuracy. Notably, the models with the highest and



(a) The accuracy and token efficiency comparison across three reasoning models on BigGSM.

(b) The reasoning boundaries of R1-Distill-Qwen-2.5-7B (**Baseline**) on BigGSM. (CFRG: 0.096; PFRG: 0.43)

(c) The reasoning boundaries of **DR. SAF** on BigGSM. (CFRG: 0.112; PFRG: 0.45)

(d) The reasoning boundaries of DR. SAF with extreme efficiency (**DR. SAF-Ext**) on BigGSM. (CFRG: 0.098; PFRG: 0.51)

Figure 7: Boundary Preservation Mechanism's impact on reasoning boundaries on BigGSM (Chen et al. 2024a).

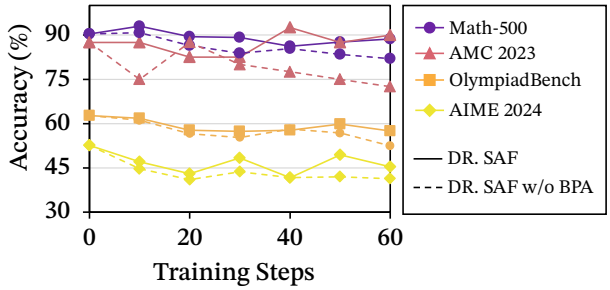


Figure 8: Accuracy trend produced during training by the boundary-preservation mechanism

second-highest alignment scores also achieve the highest and second-highest levels of token efficiency, respectively.

Answer2: Adaptive Length Management is crucial for both accuracy and efficiency in DR. SAF. Replacing Adaptive Length Management with a simple length penalty lowers average response length by 7.74% and token efficiency score by 1.69 (Table 2). Figure 6 further shows that, although the penalty initially shortens reasoning in CFRB (GSM8K), efficiency declines once responses fall below a critical threshold or when tackling harder problems such as AIME. These results confirm that ALM is necessary to sustain the optimal trade-off between brevity and efficiency.

Answer3: The Boundary Preservation Mechanism effectively mitigates reasoning boundary collapse in DR. SAF. As shown by the ablation results in Table 2, removing this module reduces accuracy by 7.41% without affecting token efficiency, indicating that BPM primarily preserves reasoning boundaries rather than enhancing compression. This effect is further supported by Figure 7, where BPM increases Out-of-Domain accuracy on BigGSM from 54.8% to 60.7%, and DR. SAF maintains superior performance even under extreme token compression (DR. SAF-Ext), highlighting BPM's robustness across compression levels. Consequently, BPM effectively prevents substantial performance degradation across diverse tasks and benchmarks.

5 Related Work

Existing work on efficient reasoning largely falls into two categories. The first, confidence-aware methods, aims to allocate compute only when needed. This is achieved either through during-reasoning techniques like dynamic early exiting based on output probabilities or structural confidence (Lu et al. 2025; Yang et al. 2025; Qiao et al. 2025; Eo et al. 2025; Ding et al. 2025; Wu et al. 2025), or via pre-reasoning triggers that determine the necessity of reasoning in the first place (Ding et al. 2024; Ong et al. 2025; Saha et al. 2025; Pan et al. 2024; Luo et al. 2025a; Zhang et al. 2025c; Li et al. 2025a; Zhang et al. 2025a; Huang et al. 2025). The second category focuses on optimizing reasoning path length. Early reinforcement learning (RL) approaches used direct length penalties (Lou et al. 2025; Team et al. 2025; Yeo et al. 2025; Luo et al. 2025b; Ling et al. 2025), while more recent work explores training-free inference-time compression (Fu et al. 2025a; Lin et al. 2025) and adaptive rewards that link Chain-of-Thought (CoT) length to problem difficulty (Tu et al. 2025; Shi et al. 2025; Qu et al. 2025; An et al. 2025; Ling et al. 2025).

However, these traditional methods predominantly rely on fixed, human-defined difficulty levels. In contrast, DR.SAF introduces a self-aware system capable of dynamically adjusting the depth of reasoning according to the model's internal capabilities and the real-time complexity of the task. This approach enhances both efficiency and accuracy.

6 Conclusion

In this work, we introduce the Dynamic Reasoning-Boundary Self-Awareness Framework (DR.SAF) to incorporate a self-aware system that adjusts the reasoning depth according to the model's internal capabilities and real-time task complexity, thereby improving both efficiency and accuracy. Extensive experiments on benchmarks such as Math-500 and AIME demonstrate that DR.SAF cuts response tokens 50% and trains 5x faster than long chain-of-thought baselines, yet keeps top-tier accuracy across 6 benchmarks. This framework sets the foundation for more scalable, efficient, and reliable LLMs in real-world applications, balancing reasoning depth with performance.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (NSFC) via grant 62236004, 62206078, 62476073, 92570120 and 62306342. This work was sponsored by the CCF-Zhipu Large Model Innovation Fund (NO.CCF-Zhipu202406).

References

AIME. 2024. American Invitational Mathematics Examination (AIME) AIME 2024-I & II.

AIME. 2025. American Invitational Mathematics Examination (AIME) 2025-I & II.

AMC. 2023. American Mathematics Competitions.

An, S.; Wang, R.; Zhou, T.; and Hsieh, C.-J. 2025. Don't Think Longer, Think Wisely: Optimizing Thinking Dynamics for Large Reasoning Models. *arXiv preprint arXiv:2505.21765*.

Arora, D.; and Zanette, A. 2025. Training language models to reason efficiently. *arXiv preprint arXiv:2502.04463*.

Chen, Q.; Qin, L.; Liu, J.; Peng, D.; Guan, J.; Wang, P.; Hu, M.; Zhou, Y.; Gao, T.; and Che, W. 2025. Towards reasoning era: A survey of long chain-of-thought for reasoning large language models. *arXiv preprint arXiv:2503.09567*.

Chen, Q.; Qin, L.; Wang, J.; Zhou, J.; and Che, W. 2024a. Unlocking the capabilities of thought: A reasoning boundary framework to quantify and optimize chain-of-thought. *Advances in Neural Information Processing Systems*, 37: 54872–54904.

Chen, X.; Xu, J.; Liang, T.; He, Z.; Pang, J.; Yu, D.; Song, L.; Liu, Q.; Zhou, M.; Zhang, Z.; et al. 2024b. Do not think that much for 2+ 3=? on the overthinking of o1-like llms. *arXiv preprint arXiv:2412.21187*.

Cobbe, K.; Kosaraju, V.; Bavarian, M.; Chen, M.; Jun, H.; Kaiser, L.; Plappert, M.; Tworek, J.; Hilton, J.; Nakano, R.; et al. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Cui, Y.; He, P.; Zeng, J.; Liu, H.; Tang, X.; Dai, Z.; Han, Y.; Luo, C.; Huang, J.; Li, Z.; Wang, S.; Xing, Y.; Tang, J.; and He, Q. 2025. Stepwise Perplexity-Guided Refinement for Efficient Chain-of-Thought Reasoning in Large Language Models. In Che, W.; Nabende, J.; Shutova, E.; and Pilehvar, M. T., eds., *Findings of the Association for Computational Linguistics: ACL 2025*, 18581–18597. Vienna, Austria: Association for Computational Linguistics. ISBN 979-8-89176-256-5.

Ding, D.; Mallick, A.; Wang, C.; Sim, R.; Mukherjee, S.; Ruhle, V.; Lakshmanan, L. V. S.; and Awadallah, A. H. 2024. Hybrid LLM: Cost-Efficient and Quality-Aware Query Routing. arXiv:2404.14618.

Ding, Y.; Jiang, W.; Liu, S.; Jing, Y.; Guo, J.; Wang, Y.; Zhang, J.; Wang, Z.; Liu, Z.; Du, B.; Liu, X.; and Tao, D. 2025. Dynamic Parallel Tree Search for Efficient LLM Reasoning. arXiv:2502.16235.

Eo, S.; Moon, H.; Zi, E. H.; Park, C.; and Lim, H. 2025. Debate Only When Necessary: Adaptive Multiagent Collaboration for Efficient LLM Reasoning. arXiv:2504.05047.

Feng, S.; Fang, G.; Ma, X.; and Wang, X. 2025. Efficient reasoning models: A survey. *arXiv preprint arXiv:2504.10903*.

Fu, Y.; Chen, J.; Zhu, S.; Fu, Z.; Dai, Z.; Zhuang, Y.; Ma, Y.; Qiao, A.; Rosing, T.; Stoica, I.; and Zhang, H. 2025a. Efficiently Scaling LLM Reasoning with Certaindex. arXiv:2412.20993.

Fu, Y.; Chen, J.; Zhuang, Y.; Fu, Z.; Stoica, I.; and Zhang, H. 2025b. Reasoning Without Self-Doubt: More Efficient Chain-of-Thought Through Certainty Probing. In *ICLR 2025 Workshop on Foundation Models in the Wild*.

Guo, D.; Yang, D.; Zhang, H.; Song, J.; Zhang, R.; Xu, R.; Zhu, Q.; Ma, S.; Wang, P.; Bi, X.; et al. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

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

He, Z.; Liang, T.; Xu, J.; Liu, Q.; Chen, X.; Wang, Y.; Song, L.; Yu, D.; Liang, Z.; Wang, W.; et al. 2025. Deepmath-103k: A large-scale, challenging, decontaminated, and verifiable mathematical dataset for advancing reasoning. *arXiv preprint arXiv:2504.11456*.

Huang, S.; Wang, H.; Zhong, W.; Su, Z.; Feng, J.; Cao, B.; and Fung, Y. R. 2025. AdaCtrl: Towards Adaptive and Controllable Reasoning via Difficulty-Aware Budgeting. *arXiv preprint arXiv:2505.18822*.

Jaech, A.; Kalai, A.; Lerer, A.; Richardson, A.; El-Kishky, A.; Low, A.; Helyar, A.; Madry, A.; Beutel, A.; Carney, A.; et al. 2024. Openai o1 system card. *arXiv preprint arXiv:2412.16720*.

Li, Z.-Z.; Liang, X.; Tang, Z.; Ji, L.; Wang, P.; Xu, H.; W, X.; Huang, H.; Deng, W.; Wu, Y. N.; Gong, Y.; Guo, Z.; Liu, X.; Yin, F.; and Liu, C.-L. 2025a. TL;DR: Too Long, Do Re-weighting for Efficient LLM Reasoning Compression. arXiv:2506.02678.

Li, Z.-Z.; Zhang, D.; Zhang, M.-L.; Zhang, J.; Liu, Z.; Yao, Y.; Xu, H.; Zheng, J.; Wang, P.-J.; Chen, X.; et al. 2025b. From system 1 to system 2: A survey of reasoning large language models. *arXiv preprint arXiv:2502.17419*.

Liang, G.; Zhong, L.; Yang, Z.; and Quan, X. 2025. Thinkswitcher: When to think hard, when to think fast. *arXiv preprint arXiv:2505.14183*.

Lightman, H.; Kosaraju, V.; Burda, Y.; Edwards, H.; Baker, B.; Lee, T.; Leike, J.; Schulman, J.; Sutskever, I.; and Cobbe, K. 2024. Let's Verify Step by Step. In *The Twelfth International Conference on Learning Representations*.

Lin, W.; Li, X.; Yang, Z.; Fu, X.; Zhen, H.-L.; Wang, Y.; Yu, X.; Liu, W.; Li, X.; and Yuan, M. 2025. TrimR: Verifier-based Training-Free Thinking Compression for Efficient Test-Time Scaling. arXiv:2505.17155.

Ling, Z.; Chen, D.; Zhang, H.; Jiao, Y.; Guo, X.; and Cheng, Y. 2025. Fast on the Easy, Deep on the Hard: Efficient Reasoning via Powered Length Penalty. *arXiv preprint arXiv:2506.10446*.

Lou, C.; Sun, Z.; Liang, X.; Qu, M.; Shen, W.; Wang, W.; Li, Y.; Yang, Q.; and Wu, S. 2025. AdaCoT: Pareto-Optimal Adaptive Chain-of-Thought Triggering via Reinforcement Learning. arXiv:2505.11896.

Lu, J.; Yu, H.; Xu, S.; Ran, S.; Tang, G.; Wang, S.; Shan, B.; Fu, T.; Feng, H.; Tang, J.; Wang, H.; and Huang, C. 2025. Prolonged Reasoning Is Not All You Need: Certainty-Based Adaptive Routing for Efficient LLM/MLLM Reasoning. arXiv:2505.15154.

Luo, F.; Chuang, Y.-N.; Wang, G.; Le, H. A. D.; Zhong, S.; Liu, H.; Yuan, J.; Sui, Y.; Braverman, V.; Chaudhary, V.; and Hu, X. 2025a. AutoL2S: Auto Long-Short Reasoning for Efficient Large Language Models. arXiv:2505.22662.

Luo, H.; Shen, L.; He, H.; Wang, Y.; Liu, S.; Li, W.; Tan, N.; Cao, X.; and Tao, D. 2025b. O1-Pruner: Length-Harmonizing Fine-Tuning for O1-Like Reasoning Pruning. arXiv:2501.12570.

Min, Y.; Chen, Z.; Jiang, J.; Chen, J.; Deng, J.; Hu, Y.; Tang, Y.; Wang, J.; Cheng, X.; Song, H.; et al. 2024. Imitate, explore, and self-improve: A reproduction report on slow-thinking reasoning systems. *arXiv preprint arXiv:2412.09413*.

Ong, I.; Almahairi, A.; Wu, V.; Chiang, W.-L.; Wu, T.; Gonzalez, J. E.; Kadous, M. W.; and Stoica, I. 2025. RouteLLM: Learning to Route LLMs with Preference Data. arXiv:2406.18665.

Pan, J.; Zhang, Y.; Zhang, C.; Liu, Z.; Wang, H.; and Li, H. 2024. DynaThink: Fast or Slow? A Dynamic Decision-Making Framework for Large Language Models. In Al-Onaizan, Y.; Bansal, M.; and Chen, Y.-N., eds., *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 14686–14695. Miami, Florida, USA: Association for Computational Linguistics.

Qiao, Z.; Deng, Y.; Zeng, J.; Wang, D.; Wei, L.; Meng, F.; Zhou, J.; Ren, J.; and Zhang, Y. 2025. ConCISE: Confidence-guided Compression in Step-by-step Efficient Reasoning. *arXiv preprint arXiv:2505.04881*.

Qin, Y.; Li, X.; Zou, H.; Liu, Y.; Xia, S.; Huang, Z.; Ye, Y.; Yuan, W.; Liu, H.; Li, Y.; et al. 2024. O1 Replication Journey: A Strategic Progress Report–Part 1. *arXiv preprint arXiv:2410.18982*.

Qu, Y.; Yang, M. Y. R.; Setlur, A.; Tunstall, L.; Beeching, E. E.; Salakhutdinov, R.; and Kumar, A. 2025. Optimizing Test-Time Compute via Meta Reinforcement Fine-Tuning. arXiv:2503.07572.

Saha, S.; Prasad, A.; Chen, J. C.-Y.; Hase, P.; Stengel-Eskin, E.; and Bansal, M. 2025. System-1.x: Learning to Balance Fast and Slow Planning with Language Models. arXiv:2407.14414.

Shen, Y.; Zhang, J.; Huang, J.; Shi, S.; Zhang, W.; Yan, J.; Wang, N.; Wang, K.; Liu, Z.; and Lian, S. 2025. Dast: Difficulty-adaptive slow-thinking for large reasoning models. *arXiv preprint arXiv:2503.04472*.

Sheng, G.; Zhang, C.; Ye, Z.; Wu, X.; Zhang, W.; Zhang, R.; Peng, Y.; Lin, H.; and Wu, C. 2024. HybridFlow: A Flexible and Efficient RLHF Framework. *arXiv preprint arXiv: 2409.19256*.

Shi, T.; Wu, Y.; Song, L.; Zhou, T.; and Zhao, J. 2025. Efficient Reinforcement Finetuning via Adaptive Curriculum Learning. arXiv:2504.05520.

Sui, Y.; Chuang, Y.-N.; Wang, G.; Zhang, J.; Zhang, T.; Yuan, J.; Liu, H.; Wen, A.; Zhong, S.; Chen, H.; et al. 2025. Stop overthinking: A survey on efficient reasoning for large language models. *arXiv preprint arXiv:2503.16419*.

Team, K.; Du, A.; Gao, B.; Xing, B.; Jiang, C.; Chen, C.; Li, C.; Xiao, C.; Du, C.; Liao, C.; et al. 2025. Kimi k1.5: Scaling reinforcement learning with llms. *arXiv preprint arXiv:2501.12599*.

Tu, S.; Lin, J.; Zhang, Q.; Tian, X.; Li, L.; Lan, X.; and Zhao, D. 2025. Learning When to Think: Shaping Adaptive Reasoning in R1-Style Models via Multi-Stage RL. *arXiv preprint arXiv:2505.10832*.

Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Xia, F.; Chi, E.; Le, Q. V.; Zhou, D.; et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35: 24824–24837.

Wu, Y.; Wang, Y.; Du, T.; Jegelka, S.; and Wang, Y. 2025. When More is Less: Understanding Chain-of-Thought Length in LLMs.

Yang, A.; Yang, B.; Zhang, B.; Hui, B.; Zheng, B.; Yu, B.; Li, C.; Liu, D.; Huang, F.; Wei, H.; et al. 2024a. Qwen2.5 Technical Report. *arXiv preprint arXiv:2412.15115*.

Yang, A.; Zhang, B.; Hui, B.; Gao, B.; Yu, B.; Li, C.; Liu, D.; Tu, J.; Zhou, J.; Lin, J.; et al. 2024b. Qwen2.5-math technical report: Toward mathematical expert model via self-improvement. *arXiv preprint arXiv:2409.12122*.

Yang, C.; Si, Q.; Duan, Y.; Zhu, Z.; Zhu, C.; Li, Q.; Lin, Z.; Cao, L.; and Wang, W. 2025. Dynamic Early Exit in Reasoning Models. *arXiv preprint arXiv:2504.15895*.

Yeo, E.; Tong, Y.; Niu, M.; Neubig, G.; and Yue, X. 2025. Demystifying Long Chain-of-Thought Reasoning in LLMs. arXiv:2502.03373.

Zhang, J.; Lin, N.; Hou, L.; Feng, L.; and Li, J. 2025a. AdaptThink: Reasoning Models Can Learn When to Think. arXiv:2505.13417.

Zhang, Q.; Lyu, F.; Sun, Z.; Wang, L.; Zhang, W.; Hua, W.; Wu, H.; Guo, Z.; Wang, Y.; Muennighoff, N.; et al. 2025b. A Survey on Test-Time Scaling in Large Language Models: What, How, Where, and How Well? *arXiv preprint arXiv:2503.24235*.

Zhang, S.; Wu, J.; Chen, J.; Zhang, C.; Lou, X.; Zhou, W.; Zhou, S.; Wang, C.; and Wang, J. 2025c. OThink-R1: Intrinsic Fast/Slow Thinking Mode Switching for Over-Reasoning Mitigation. arXiv:2506.02397.