Mitigating Factual Hallucination in Large Reasoning Models via Mixed-Mode Advantage Regularization

Kaishen Wang, Tong Zheng, Xuehao Cui, Ruibo Chen, Tianyi Xiong, Heng Huang
Department of Computer Science, University of Maryland, College Park
{kaishen,tzheng24,cedrus,rbchen,txiong23,heng}@umd.edu

Abstract

Large reasoning models (LRMs) improve language model capabilities by generating explicit thinking traces before final answers. In factuality-oriented question answering (QA), such thinking often improves overall performance by helping the model recover relevant knowledge and refine its answers. However, we find that this benefit is not uniform at the instance level: explicit thinking can also overturn correct non-thinking answers and lead to factual drift. We refer to this failure mode as *thinking-induced hallucination*. To explain this phenomenon, we formulate explicit thinking in factuality QA as a thinking residual over the model's direct-answer tendency, which can either recover missing knowledge or introduce unsupported associations. Based on this formulation, we propose MARGO, *M*ixed-Mode *A*dvantage *R*egularization for *G*rounded *O*ptimization, a reinforcement learning framework that uses non-thinking rollouts as same-model references in advantage estimation. By constructing mixed-mode rollout groups with both thinking and non-thinking trajectories, MARGO evaluates whether explicit thinking adds factual value beyond direct answering, thereby suppressing hallucination-prone thinking while preserving beneficial thinking behaviors. Experiments across multiple factuality-oriented QA benchmarks demonstrate that MARGO improves factual reliability over strong baselines, while evaluations on mathematical benchmarks show that it preserves general reasoning ability. The code is available at https://github.com/tunantu/MARGO.

1 Introduction

Large reasoning models (LRMs) have recently achieved strong performance by generating explicit thinking traces before final answers [Jaech et al., 2024, Guo et al., 2025, Yang et al., 2025, Hou et al., 2025]. This paradigm is especially effective for mathematics, coding, and symbolic reasoning, where intermediate steps help decompose problems, verify partial results, and derive final answers [Wei et al., 2022, Wang et al., 2022, Zheng et al., 2023, Gao et al., 2023]. Explicit thinking can also benefit factuality-oriented question answering (QA), as reasoning traces may help models recall relevant knowledge, compare candidate facts, and refine final responses. As a result, thinking has become a widely used strategy for improving modern large language models (LLMs).

However, in factuality QA, the benefit of thinking is not uniformly positive at the instance level. To examine how thinking changes model predictions, we compare non-thinking and thinking modes on TriviaQA [Joshi et al., 2017]. As shown in Table 1, on Qwen3-8B [Yang et al., 2025], thinking corrects 11.74% of examples that are originally incorrect under non-thinking, confirming its factual gains. At the same time, it overturns 7.50% of originally correct non-thinking answers into incorrect predictions. We call this failure mode *thinking-induced hallucination*, where explicit thinking introduces misleading or unsupported content and changes a correct direct answer into a hallucinated one. This is particularly concerning because the correct answer is already accessible to the same model under the non-thinking mode, but is lost after generating an explicit thinking trace.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

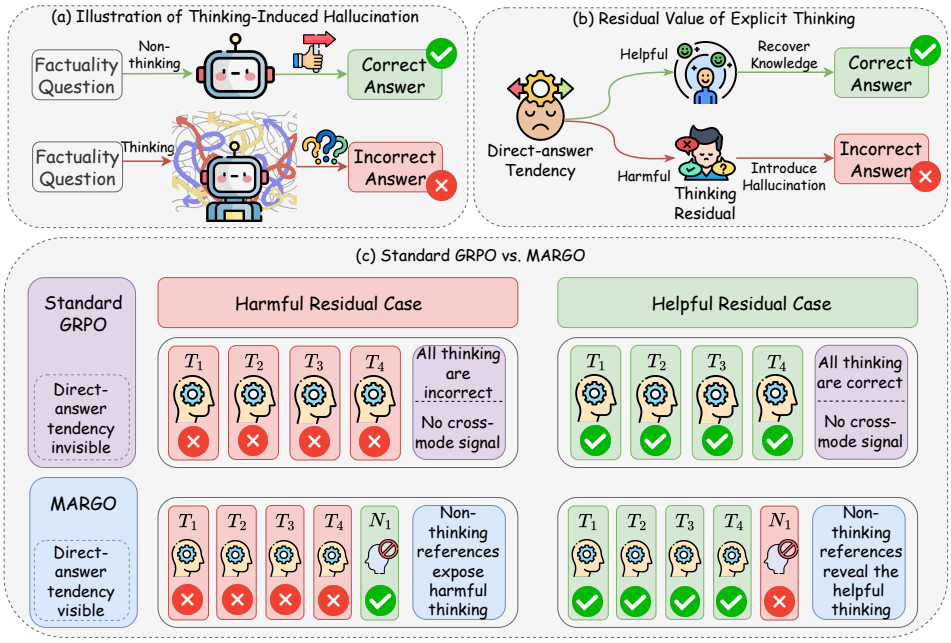

Figure 1: Overview of MARGO. (a) Explicit thinking can induce hallucination by overturning a correct non-thinking answer into an incorrect prediction. (b) We view explicit thinking in LRMs as a residual over the model's direct-answer tendency, which can be either helpful or harmful. (c) Unlike standard GRPO, which compares only thinking rollouts, MARGO uses non-thinking rollouts as same-model references to suppress harmful thinking while preserving helpful thinking.

This instability stems from a key difference between factual QA and derivation-based reasoning. In mathematical reasoning, intermediate steps can often derive the answer from given premises. In factual QA, however, the answer usually depends on retrieving specific knowledge [Petroni et al., 2019, Roberts et al., 2020, Lewis et al., 2020]. When the model is uncertain about the relevant fact, longer thinking does not create new evidence; instead, it may introduce unsupported associations, amplify uncertainty, and cause factual drift [Lyu et al., 2023, Huang et al., 2025, Su et al., 2025, Wang et al., 2025]. Thus, the central challenge is to preserve the factual gains of thinking while preventing reasoning traces from corrupting correct direct-answer tendencies. These observations motivate a residual view of explicit thinking in factuality QA. The non-thinking response reflects the model's direct factual tendency before an explicit reasoning trace is introduced, while the thinking response adds a reasoning residual on top of this tendency. This residual is helpful when it recovers the correct fact, but harmful when it introduces unsupported associations, entity confusion, or speculative intermediate claims. From this perspective, factuality optimization should account for how thinking changes the model's direct-answer tendency, rather than rewarding final answers in isolation.

Building on this interpretation, we propose MARGO, *M*ixed-Mode *A*dvantage *R*egularization for *G*rounded *O*ptimization, a reinforcement learning framework for mitigating thinking-induced hallucination in factuality QA. Unlike standard GRPO with all-thinking rollout groups, MARGO constructs mixed rollout groups containing both thinking and non-thinking trajectories for each question. This design makes the non-thinking response a same-model reference in advantage estimation, so thinking trajectories are optimized relative to the model's direct-answer behavior rather than only against other thinking samples. As a result, hallucination-prone reasoning that corrupts a correct non-thinking answer receives lower relative advantage, while reasoning that improves factual correctness remains reinforced. The overall architecture is shown in Figure 1.

We train MARGO and evaluate it on two models across six factuality-oriented QA benchmarks. Experimental results show that MARGO consistently improves factual reliability over fixed-mode inference and training-based baselines, including mode selection, supervised fine-tuning, and RL variants trained with the same curated data. Evaluation on mathematical benchmarks further shows that MARGO

preserves the general reasoning ability of LRMs. The contributions of this paper are summarized as follows:

- We identify the *thinking-induced hallucination* issue in factuality-oriented QA, where explicit thinking turns correct non-thinking answers into incorrect ones, and show through instance-level transition analysis that this phenomenon appears consistently across different LRM scales.
- We formulate explicit thinking in factuality QA as a reasoning residual over the model's direct-answer tendency, and propose MARGO, a mixed-mode advantage regularization framework that uses non-thinking rollouts as same-model references to suppress hallucination-prone thinking while preserving trajectories that provide factual value.

2 Related Work

2.1 Hallucination in Large Reasoning Models

Although large reasoning models (LRMs) have achieved strong performance on reasoning-intensive tasks, recent studies suggest that explicit reasoning may introduce additional factuality risks. In factuality-oriented settings, long reasoning traces can introduce unsupported intermediate claims, amplify early mistakes, or produce plausible but factually incorrect conclusions [Yao et al., 2025, Li and Tou Ng, 2025, Chen et al., 2025b, Yu et al., 2025, Chen et al., 2025a, Wang and Huang, 2026]. For example, [Chen et al., 2025b] show that LRMs generate substantially more hallucinations on long-form factuality benchmarks and propose online RL with rewards balancing factual precision, response detail, and relevance. [Li and Tou Ng, 2025] further analyze hallucination caused by reasoning-oriented RL and propose factuality-aware step-wise policy optimization. Another line of work detects or characterizes hallucinations inside reasoning traces: [Sun et al., 2025] study reasoning hallucination from a mechanistic perspective, where logically coherent but factually incorrect traces lead to persuasive yet faulty conclusions, while [Zhang et al.] analyze LRM hallucinations by representing reasoning trajectories as structured graphs. Different from these studies, we focus on an instance-level mode-induced failure: the same model can answer correctly without thinking, but becomes incorrect after generating an explicit thinking trace.

2.2 Reasoning Optimization and Adaptive Thinking

Reinforcement learning has become a common approach for improving the reasoning ability of LLMs, with recent methods showing that RL can elicit long-form reasoning behaviors such as exploration, reflection, and self-correction [Ouyang et al., 2022, Shao et al., 2024, Guo et al., 2025]. GRPO [Shao et al., 2024] further provides a practical framework for reasoning-oriented training by estimating relative advantages within groups of sampled responses [Shao et al., 2024]. Meanwhile, adaptive thinking methods study how to decide whether, when, or how long a model should reason, often treating thinking as a mode-selection or budget-allocation problem [Wan et al., 2025, Zhang et al., 2025, Han et al., 2025, Li et al., Jali et al., 2026, Xiong et al., 2026, Zheng et al., 2026]. Different from these approaches, MARGO does not learn a question-level policy for when to think, nor does it optimize only all-thinking rollout groups. Instead, it constructs mixed rollout groups containing both thinking and non-thinking trajectories, using non-thinking responses as same-model references to regularize reasoning traces that degrade factuality while preserving those that provide factual value.

3 Method

3.1 Preliminary: Thinking and Non-Thinking Inference Modes

Recent large reasoning models (LRMs) are typically deployed under two distinct inference modes, namely *thinking* and *non-thinking*. These modes are not selected adaptively by the model itself at inference time; instead, they are determined by the prompting template.

Given an input question x, let $o = (o_1, \ldots, o_T)$ denote the output sequence generated by an LRM parameterized by θ. The generation process follows the standard autoregressive factorization:

$$P_\theta(o \mid x) = \prod_{t=1}^{T} P_\theta(o_t \mid x, o_{<t}). \tag{1}$$

The difference between the two modes lies in the prompting format and the resulting structure of the generated response. In the thinking mode, the model is prompted to generate an explicit thinking trace before producing the final answer. Denoting the reasoning trace by r and the final answer by y, the generated response takes the form:

$$o = [\texttt{<think>},\ r,\ \texttt{</think>},\ y], \tag{2}$$

where r is a non-empty reasoning segment enclosed by the `<think>` and `</think>` tags. In the non-thinking mode, the prompting template already provides an empty reasoning block after the question, i.e., `<think></think>` is pre-filled as part of the input context. As a result, the model directly generates the final answer y without producing an explicit thinking trace.

Therefore, the distinction between thinking and non-thinking does not stem from any change in model parameters, but from the prompting template and the induced generation pattern. Although both modes operate on the same underlying model, they may lead to substantially different behaviors.

3.2 Empirical Observation: Thinking-Induced Hallucination

While LRMs often benefit from additional test-time computation, whether explicit thinking is uniformly beneficial for factuality tasks remains unclear. To investigate this, we conduct preliminary experiments comparing the thinking and non-thinking modes on TriviaQA [Joshi et al., 2017], a representative factuality benchmark. Beyond overall performance, we further perform a fine-grained comparison at the instance level to examine how prediction correctness changes when switching from the non-thinking mode to the thinking mode.

Size	$r_{0,0}$	$r_{0,1}$	$r_{1,0}$	$r_{1,1}$
1.7B	62.98%	12.32%	5.38%	19.32%
4B	48.06%	12.15%	5.83%	33.96%
8B	36.08%	11.74%	7.50%	44.68%
14B	30.43%	10.45%	6.75%	52.36%
32B	28.60%	10.22%	6.27%	54.92%

Table 1: Transition ratios between non-thinking and thinking modes on TriviaQA across Qwen3 models of different scales. Each $r_{i,j}$ denotes the fraction of examples whose prediction changes from correctness i under the non-thinking mode to correctness j under the thinking mode.

Formally, for each example x, let $c^{\mathrm{N}}(x) \in \{0, 1\}$ and $c^{\mathrm{T}}(x) \in \{0, 1\}$ denote the correctness of the prediction under the non-thinking mode and the thinking mode, respectively, where 1 indicates a correct prediction and 0 otherwise. Based on the pair $\big(c^{\mathrm{N}}(x), c^{\mathrm{T}}(x)\big)$, we partition all examples into four transition groups:

$$\tau_{i,j} = \{x \mid c^{\mathrm{N}}(x) = i,\ c^{\mathrm{T}}(x) = j\}, \quad i, j \in \{0, 1\}. \tag{3}$$

Specifically, $\tau_{0,0}$ denotes the examples that are incorrect under both modes, $\tau_{0,1}$ denotes the examples that change from incorrect to correct, $\tau_{1,0}$ denotes the examples that change from correct to incorrect, and $\tau_{1,1}$ denotes the examples that remain correct under both modes. Among these four groups, $\tau_{1,0}$ is particularly important, as it captures the cases where thinking overturns an originally correct non-thinking prediction and induces hallucination. We further define the transition ratio of each group as:

$$r_{i,j} = \frac{|\tau_{i,j}|}{|\mathcal{D}|}, \quad i, j \in \{0, 1\}, \tag{4}$$

where $\mathcal{D}$ denotes the evaluation set. As shown in Table 1, although thinking can correct a portion of non-thinking incorrect predictions (i.e., $r_{0,1}$), there also exists a non-negligible fraction of examples in $r_{1,0}$, where thinking overturns an originally correct non-thinking answer and leads to an incorrect prediction. We refer to this failure mode as *thinking-induced hallucination*. These results reveal that explicit thinking plays a dual role in factuality QA: it can recover some missed answers, but it can also introduce harmful reasoning that drifts away from an originally correct direct answer. We provide additional transition analyses in Appendix D.

We further inspect the thinking traces in $\tau_{1,0}$ and find that they often contain uncertainty or revision markers, such as "But wait", "Hmm, not sure", "Wait, no, maybe that's not right", and "but I'm getting

confused". These traces suggest that thinking-induced hallucination is often not caused by a complete absence of the correct fact, but by instability introduced during the thinking process. When the relevant factual knowledge is uncertain, longer thinking does not create new evidence; instead, it may amplify uncertainty, introduce entity confusion, or over-elaborate unsupported associations. Since the two modes share the same model parameters and differ mainly by whether an explicit thinking trace is generated, the non-thinking response reveals the model's direct factual tendency before this additional thinking process is introduced. This motivates our residual view of explicit thinking: in factuality QA, thinking can be seen as a reasoning residual on top of the model's direct-answer behavior, and the value of this residual should be evaluated relative to the non-thinking reference.

3.3 Residual Value of Explicit Thinking

The transition analysis above motivates modeling thinking as a *reasoning residual* over the model's non-thinking behavior. For the same question x, the non-thinking response reflects the model's direct factual tendency, while the thinking response adds an explicit thinking trace before producing the final answer. The key question is therefore whether this additional reasoning residual provides positive factual value over the non-thinking reference.

Formally, let $\mu_m(x)$ denote the expected factual reward under mode $m \in \{\mathrm{T}, \mathrm{N}\}$:

$$\mu_m(x) = \mathbb{E}_{y^m \sim \pi_\theta(\cdot|x,m)} \left[R(x, y^m) \right]. \tag{5}$$

We define the residual value of explicit thinking as:

$$\Delta_{\mathrm{res}}(x) = \mu_{\mathrm{T}}(x) - \mu_{\mathrm{N}}(x). \tag{6}$$

A positive $\Delta_{\mathrm{res}}(x)$ means that thinking provides factual value beyond direct answering, while a negative value means that thinking introduces harmful residual contamination.

Standard GRPO with all-thinking rollout groups cannot directly capture this residual value. When all trajectories in a group are sampled in the thinking mode, the reward baseline is determined only by other thinking trajectories. As a result, a hallucination-prone thinking trajectory may still receive a favorable relative advantage as long as it performs better than other thinking samples in the same group. This optimizes relative quality within the thinking distribution, but does not evaluate whether thinking itself adds factual value over the model's direct-answer behavior.

To expose the residual value of thinking, we construct mixed rollout groups containing both thinking and non-thinking trajectories. Suppose a rollout group contains thinking and non-thinking trajectories with proportions α and $1 - \alpha$, respectively. The expected mixed reward baseline is:

$$\mu_{\mathrm{mix}}(x) = \alpha \mu_{\mathrm{T}}(x) + (1 - \alpha) \mu_{\mathrm{N}}(x). \tag{7}$$

Ignoring the normalization scale for clarity, the expected relative advantage of a thinking trajectory becomes:

$$\begin{aligned} A^{\mathrm{T}}_{\mathrm{mix}}(x) &\propto R(x, y^{\mathrm{T}}) - \mu_{\mathrm{mix}}(x) \\ &= \underbrace{R(x, y^{\mathrm{T}}) - \mu_{\mathrm{T}}(x)}_{\text{within-mode advantage}} + (1 - \alpha) \underbrace{\Delta_{\mathrm{res}}(x)}_{\text{residual value}}. \end{aligned} \tag{8}$$

The first term measures the relative quality of a thinking trajectory within the thinking distribution, while the second term adjusts this advantage by the residual value of thinking over the non-thinking reference. Therefore, when $\Delta_{\mathrm{res}}(x) < 0$, thinking is expected to introduce harmful residual contamination, and thinking trajectories receive lower relative advantage under the mixed baseline. When $\Delta_{\mathrm{res}}(x) > 0$, thinking provides positive residual value and is reinforced. By symmetry, non-thinking trajectories receive the opposite residual adjustment, making them relatively preferred when thinking harms factuality. We present the detailed theoretical analysis in Appendix A.

3.4 Mixed-Mode Advantage Regularization for Grounded Optimization

Following the residual-value formulation above, we instantiate `MARGO` with GRPO by constructing mixed rollout groups for each factual question. The key idea is to place thinking and non-thinking trajectories into the same group, so that the GRPO advantage reflects not only within-mode reward differences but also whether thinking adds factual value over the model's direct-answer behavior.

Given an input question x, we sample both thinking and non-thinking responses:

$$\mathcal{G}(x) = \{y_1^{\mathrm{T}}, \ldots, y_{K_T}^{\mathrm{T}}\} \cup \{y_1^{\mathrm{N}}, \ldots, y_{K_N}^{\mathrm{N}}\}, \tag{9}$$

where y_i^{T} is sampled under the thinking prompt and y_j^{N} is sampled under the non-thinking prompt. Both modes share the same model parameters and differ only in the prompting template. Thus, the mixed rollout group provides both thinking trajectories and same-model non-thinking references for the same question.

For each response $y_i \in \mathcal{G}(x)$, we compute a reward r_i based on factual correctness and response format. GRPO then normalizes rewards within the mixed group:

$$A_i = \frac{r_i - \mathrm{mean}(\{r_j\}_{j=1}^{|\mathcal{G}(x)|})}{\mathrm{std}(\{r_j\}_{j=1}^{|\mathcal{G}(x)|})}, \qquad \rho_i(\theta) = \frac{\pi_\theta(y_i \mid x, m_i)}{\pi_{\theta_{\mathrm{old}}}(y_i \mid x, m_i)}, \tag{10}$$

where $m_i \in \{\mathrm{T}, \mathrm{N}\}$ denotes the prompting mode used to sample y_i. This mixed-group normalization implements the residual-value principle in Section 3.3: thinking trajectories are evaluated not only against other thinking samples, but also against non-thinking references for the same question.

The clipped GRPO objective is defined as:

$$\mathcal{J}_{\mathrm{GRPO}}(\theta) = \mathbb{E}_{x,\mathcal{G}(x)}\left[\frac{1}{|\mathcal{G}(x)|}\sum_{y_i \in \mathcal{G}(x)} \min\Big(\rho_i(\theta)A_i,\ \mathrm{clip}\big(\rho_i(\theta), 1-\epsilon, 1+\epsilon\big)A_i\Big)\right]. \tag{11}$$

In practice, we additionally apply KL regularization to stabilize training and constrain the updated policy from drifting too far from the reference model.

Reward for Factuality QA. Unlike conventional rule-based RLVR tasks such as mathematical reasoning, factuality QA does not admit a simple deterministic verifier. In math problems, the final answer can often be extracted and matched against the ground truth. In contrast, factual answers may appear in different surface forms, aliases, abbreviations, or paraphrases, making exact string matching unreliable. Therefore, we use Qwen3-32B as an automatic judge to evaluate sampled rollouts, consistent with [Yao et al., 2025]. The judge is not asked to verify factual correctness from its own parametric knowledge. Instead, it is provided with the question, the ground-truth answer, and the model prediction, and is required to determine whether the prediction is semantically consistent with the provided ground truth. The detailed judging prompt is provided in Appendix B.

Based on the judge result, we define the reward as:

$$r = r_{\mathrm{corr}} + r_{\mathrm{fmt}}. \tag{12}$$

Here, r_{corr} measures factual correctness, while r_{fmt} provides a lightweight bonus for valid response formatting. The correctness reward is defined as:

$$r_{\mathrm{corr}} = \begin{cases} +1, & \text{if the response is judged correct,} \\ -1, & \text{otherwise.} \end{cases} \tag{13}$$

A response is regarded as format-valid if it contains a properly closed `<think>` and `</think>` blocks. We define the format reward as:

$$r_{\mathrm{fmt}} = \begin{cases} +0.05, & \text{if the response format is valid,} \\ 0, & \text{otherwise.} \end{cases} \tag{14}$$

Since GRPO computes relative advantages over both thinking and non-thinking trajectories for the same question, a hallucination-prone thinking trajectory receives lower advantage when the non-thinking reference achieves a higher reward. This makes the regularization effect emerge from group-wise comparison rather than from manually designed mode rewards.

4 Experiments

4.1 Experimental Setting

Models and Benchmarks. We evaluate `MARGO` on two LRMs, Qwen3-4B and Qwen3-8B [Yang et al., 2025], both of which exhibit strong reasoning ability across diverse tasks. For evaluation,

we use six representative factuality-oriented benchmarks: SimpleQA [Wei et al., 2024], SimpleQA-Verified [Haas et al., 2025] (denoted as SimpleQA-V in our paper), TriviaQA [Joshi et al., 2017], NQ_Open [Kamalloo et al., 2023], PopQA [Mallen et al., 2023], and HotpotQA [Yang et al., 2018]. These datasets cover a broad range of factual knowledge, from short-form fact retrieval to multi-hop open-domain question answering, allowing us to evaluate factual reliability across different types of factuality tasks. Unlike mathematics or multiple-choice tasks, factuality QA cannot always be reliably evaluated by exact string matching. The same correct answer may appear in different surface forms, aliases, abbreviations, or paraphrases. Therefore, we use Qwen3-32B as an automatic judge to determine whether the model prediction is semantically consistent with the ground-truth answer. The detailed judging prompt are provided in Appendix B.

Training Data Construction. We construct model-specific training data from the TriviaQA training split. For each base model, we run both thinking and non-thinking modes on the training questions and retain examples where the two modes show a clear factuality gap, including cases where thinking performs much better and cases where non-thinking performs much better. This filtering is performed separately for Qwen3-4B and Qwen3-8B, since the effect of thinking is model-dependent. Details are provided in Appendix C.1.

Baselines. We compare `MARGO` with several baselines trained or evaluated under the same data setting. Fixed$_{NoThink}$ and Fixed$_{Think}$ denote the original model evaluated under the non-thinking and thinking modes, respectively, measuring the default behavior of LRMs under fixed generation modes. Adaptive$_{CLS}$ trains `microsoft/deberta-v3-base` [He et al., 2021] as a binary classifier to predict whether a given question should invoke thinking. This baseline examines whether the decision of when to think can be learned as a question-level classification problem. We also include Adaptive$_{SFT}$, which uses the same curated data to teach the model when to think and when not to think through supervised fine-tuning. For examples where thinking is preferred, we construct the SFT target as the model-generated thinking process followed by the final answer. For examples where non-thinking is preferred, we use an empty thinking block followed by the final answer. All constructed targets follow the official response format, and all target answers are judged correct. We further compare with Adaptive$_{RL}$ [Zhang et al., 2025], an adaptive thinking baseline originally designed to improve inference efficiency for mathematical reasoning. In our setting, we adapt it to factuality QA by training the model to decide when to skip thinking when thinking is unnecessary or potentially harmful. In addition, to isolate the effect of mixed-mode advantage regularization from reinforcement learning alone, we include two fixed-mode RL baselines. Fixed$_{NoThink+RL}$ fine-tunes the model with GRPO using the non-thinking format for all training examples, while Fixed$_{Think+RL}$ fine-tunes the model with GRPO using the thinking format for all training examples. All training-based baselines use the same curated training data as `MARGO`.

Training and Evaluation Details. We train `MARGO` using the `verl` [Sheng et al., 2025] framework. All experiments are conducted on 8 NVIDIA H200 GPUs, where 4 GPUs are used for policy training and the remaining 4 GPUs are used to serve the reward model. We train the model for one epoch with a training batch size of 64, a learning rate of 1×10^{-6}, and a maximum response length of 8192 tokens. For each prompt, we sample 8 rollouts, including 6 thinking rollouts and 2 non-thinking rollouts; thus, the thinking-ratio coefficient α is set to 0.75. We also apply KL regularization to constrain the updated policy from drifting too far from the reference model, with the KL loss coefficient set to 0.001. We evaluate all benchmarks using `vLLM` [Kwon et al., 2023]. For factuality evaluation, we use the same judge model, Qwen3-32B, and the same prompt strategy as in training. Following the official decoding recommendations, we set the temperature to 0.6, top-p to 0.95, and top-k to 20 for thinking mode generation. For non-thinking mode generation, we set the temperature to 0.7, top-p to 0.8, and top-k to 20. The maximum generation length is set to 8192 tokens.

4.2 Experimental Results

Main Results. Table 2 reports the results on six factuality-oriented QA benchmarks across Qwen3-4B and Qwen3-8B. Overall, `MARGO` achieves the best average performance on both model scales. On Qwen3-4B, `MARGO` improves the average accuracy from 22.81% under the fixed thinking mode to 25.49%, yielding a gain of 2.68%. On Qwen3-8B, `MARGO` improves the average accuracy from 27.59% to 29.57%, yielding a gain of 1.98%. These results show that explicitly regularizing thinking behavior

Method	SimpleQA	SimpleQA-V	TriviaQA	NQ_Open	PopQA	HotpotQA	Average
			Qwen3-4B				
Fixed$_{NoThink}$	3.74%	3.80%	39.79%	29.50%	17.07%	27.86%	20.29%
Fixed$_{Think}$	3.98%	4.10%	46.12%	32.94%	19.05%	30.64%	22.81%
Adaptive$_{CLS}$	3.61%	4.30%	43.91%	30.33%	18.36%	28.43%	21.49%
Adaptive$_{SFT}$	3.35%	4.80%	41.58%	30.28%	17.69%	28.08%	20.96%
Adaptive$_{RL}$	4.23%	4.00%	47.22%	32.49%	19.22%	30.50%	22.94%
Fixed$_{NoThink+RL}$	4.62%	5.20%	40.14%	30.61%	18.40%	27.70%	21.11%
Fixed$_{Think+RL}$	4.48%	4.10%	47.44%	33.80%	19.16%	30.89%	23.31%
MARGO	**6.73%**	**10.00%**	**49.12%**	**34.76%**	**20.48%**	**31.82%**	**25.49%**
			Qwen3-8B				
Fixed$_{NoThink}$	4.16%	4.10%	52.18%	38.64%	22.13%	33.13%	25.72%
Fixed$_{Think}$	5.04%	4.60%	56.42%	39.94%	23.71%	35.84%	27.59%
Adaptive$_{CLS}$	4.62%	4.30%	54.94%	38.50%	23.21%	34.20%	26.63%
Adaptive$_{SFT}$	4.30%	4.60%	52.38%	38.50%	21.41%	33.88%	25.85%
Adaptive$_{RL}$	5.27%	5.50%	56.05%	40.08%	23.28%	35.64%	27.64%
Fixed$_{NoThink+RL}$	4.23%	5.60%	53.80%	39.09%	23.22%	34.26%	26.70%
Fixed$_{Think+RL}$	5.29%	4.90%	56.88%	40.66%	23.85%	35.93%	27.92%
MARGO	**5.82%**	**9.00%**	**58.03%**	**42.24%**	**25.23%**	**37.07%**	**29.57%**

Table 2: Main results on six factuality-oriented QA benchmarks using Qwen3-4B and Qwen3-8B. We compare MARGO against fixed-mode inference, adaptive mode-selection baselines, supervised fine-tuning, and RL variants trained with the same data.

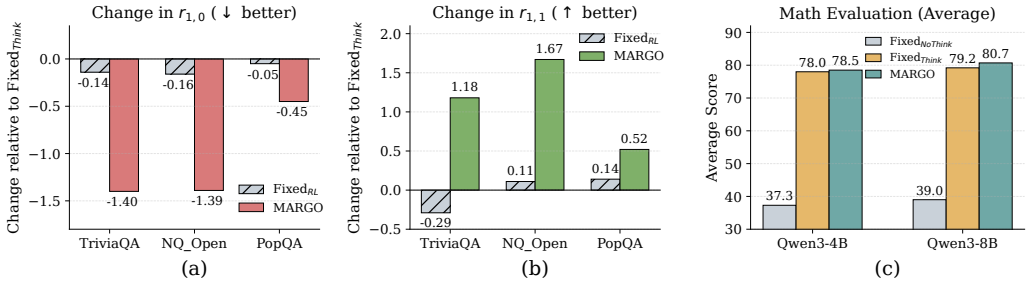

Figure 2: Analysis of transition-ratio changes and mathematical generalization. (a) Changes in $r_{1,0}$ on Qwen3-4B relative to Fixed$_{Think}$. (b) Changes in $r_{1,1}$ on Qwen3-4B relative to Fixed$_{Think}$. (c) Average mathematical performance over AMC23, AIME24, and AIME25 under Mean@16 for Qwen3-4B and Qwen3-8B, respectively.

with non-thinking references improves factual reliability beyond simply enabling the thinking mode at inference time.

Compared with adaptive mode-selection baselines, MARGO also shows clear advantages. Adaptive$_{CLS}$ and Adaptive$_{SFT}$ do not consistently outperform the fixed thinking mode, suggesting that a simple question-level mode-selection strategy may be insufficient in this setting. This is consistent with our motivation that the usefulness of thinking is model-specific and trajectory-dependent. In contrast, MARGO avoids making a hard mode-selection decision and instead regularizes thinking through mixed-mode advantage estimation. We further compare MARGO with RL baselines trained on the same data. Although Fixed$_{Think+RL}$ and Fixed$_{NoThink+RL}$ improve over some fixed-mode baselines, they remain consistently below MARGO. For example, on Qwen3-4B, Fixed$_{Think+RL}$ achieves an average accuracy of 23.31%, while MARGO reaches 25.49%. On Qwen3-8B, Fixed$_{Think+RL}$ reaches 27.92%, compared with 29.57% for MARGO. This indicates that the improvement does not come from reinforcement learning alone, but from the mixed-mode advantage regularization that evaluates thinking trajectories against same-model non-thinking references.

Across individual benchmarks, MARGO achieves the best performance on all six datasets for both Qwen3-4B and Qwen3-8B. The gains are especially pronounced on SimpleQA and SimpleQA-Verified, where factual precision is critical and unsupported thinking can easily introduce incorrect claims. These consistent improvements demonstrate that MARGO effectively mitigates thinking-induced hallucination while preserving the useful factual benefits of explicit thinking.

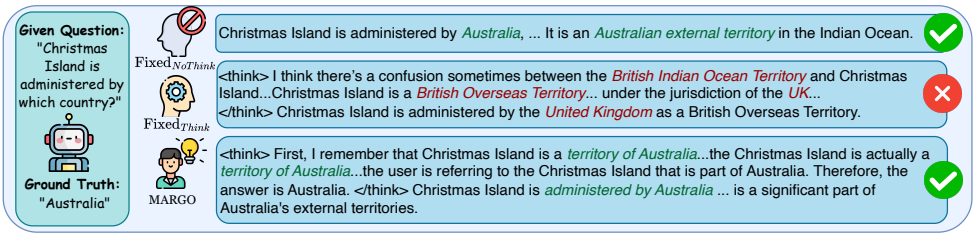

Figure 3: Qualitative case study comparing Fixed$_{NoThink}$, Fixed$_{Think}$, and MARGO. Fixed$_{NoThink}$ gives the correct direct answer, Fixed$_{Think}$ drifts to an incorrect answer through unsupported reasoning, and MARGO recovers the correct answer by suppressing this factual drift.

Transition Analysis. To further understand where the gains of MARGO come from, we analyze how training changes the instance-level transition ratios on Qwen3-4B. We focus on $\tau_{1,0}$ and $\tau_{1,1}$, where $\tau_{1,0}$ captures cases in which thinking overturns an originally correct non-thinking answer, and $\tau_{1,1}$ captures cases that remain correct under thinking. Figure 2 reports the change of these two ratios relative to Fixed$_{Think}$. Fixed$_{Think+RL}$ produces only marginal changes in both $r_{1,0}$ and $r_{1,1}$, indicating that applying RL with the same data does not substantially alter the harmful transition pattern. In contrast, MARGO consistently reduces $r_{1,0}$ across TriviaQA, NQ_Open, and PopQA, while increasing $r_{1,1}$ on all three datasets. This shows that the improvement of MARGO is not merely due to RL training or the curated data itself. Instead, the gains come from mixed-mode advantage regularization, where non-thinking rollouts serve as same-model references to suppress hallucination-prone thinking and preserve correct direct-answer tendencies. This also suggests that MARGO does not simply bias the model toward non-thinking outputs, but instead changes the relative optimization signal according to whether thinking improves factual correctness for the same question.

Qualitative Analysis. We present a factuality QA case study to illustrate how MARGO mitigates reasoning-induced hallucination. As shown in Figure 3, given the question "Christmas Island is administered by which country?", Fixed$_{NoThink}$ directly produces the correct answer, Australia, indicating that the model already contains the required factual knowledge in its direct-answer mode. However, when forced to generate an explicit reasoning trace, Fixed$_{Think}$ gradually drifts away from this correct knowledge. It introduces unsupported intermediate claims, confuses Christmas Island with British territories, and eventually predicts the United Kingdom as the administering country. This example demonstrates that additional reasoning can sometimes overwrite a correct direct answer with a hallucinated conclusion, rather than improving factual reliability. In contrast, MARGO preserves the correct direct-answer tendency from the non-thinking rollout while still allowing a reasoning process. By using the non-thinking rollout as a same-model reference, MARGO suppresses the erroneous factual drift in the thinking trajectory and recovers the correct answer.

4.3 Ablation Studies

Mathematical Generalization. Although MARGO is designed for mitigating hallucination in factuality-oriented QA, we further evaluate whether it preserves the general reasoning ability of LRMs on widely used mathematical benchmarks. Specifically, we compare the model trained with MARGO against the original model under both non-thinking and thinking modes on three math benchmarks: AMC23, AIME24, and AIME25. For all three benchmarks, we report Mean@16 and Pass@16, where Mean@16 denotes the average accuracy over 16 sampled responses and Pass@16 denotes whether at least one of the 16 responses gives the correct answer. We set the maximum generation length to 32,768 for all math evaluations. As shown in Figure 2(c), MARGO preserves mathematical reasoning ability and even yields slight gains over the base model under the thinking mode. Specifically, MARGO achieves the best Mean@16 across all three benchmarks at both the 4B and 8B scales, improving the average accuracy by 0.5% on the 4B model and 1.5% on the 8B model. The detailed results are provided in Table 4.

Data Selection Evaluation. Since MARGO uses non-thinking rollouts as references to regularize thinking rollouts, the informativeness of the training prompts is important for effective optimization. Ideally, training examples should exhibit a clear factuality gap between thinking and non-thinking

Method	SimpleQA	SimpleQA-V	TriviaQA	NQ_Open	PopQA	HotpotQA
Fixed$_{NoThink}$	3.74%	3.80%	39.79%	29.50%	17.07%	27.86%
Fixed$_{Think}$	3.98%	4.10%	46.12%	32.94%	19.05%	30.64%
MARGO + Random	3.70%	4.50%	46.14%	32.88%	19.54%	30.34%
MARGO	**6.73%**	**10.00%**	**49.12%**	**34.76%**	**20.48%**	**31.82%**

Table 3: Data selection ablation on six factuality-oriented QA benchmarks using Qwen3-4B. The best results are highlighted in bold.

Method	AMC23		AIME24		AIME25		Average
	Mean@16	Pass@16	Mean@16	Pass@16	Mean@16	Pass@16	
Qwen3-4B							
Fixed$_{NoThink}$	67.0	92.5	23.3	53.3	21.7	46.7	37.3
Fixed$_{Think}$	95.8	100.0	73.8	83.3	64.4	86.7	78.0
MARGO	**96.2**	100.0	**73.8**	90.0	**65.4**	90.0	**78.5**
Qwen3-8B							
Fixed$_{NoThink}$	66.1	92.5	29.0	66.7	21.9	43.3	39.0
Fixed$_{Think}$	94.2	100.0	76.0	86.7	67.3	86.7	79.2
MARGO	**94.5**	100.0	**77.5**	83.3	**70.2**	83.3	**80.7**

Table 4: Additional evaluation of MARGO on mathematical benchmarks. We report Mean@16 and Pass@16 on AMC23, AIME24, and AIME25 for Qwen3-4B and Qwen3-8B.

modes, so that the mixed rollout group can provide meaningful reference signals. To evaluate the effect of our data selection strategy, we conduct an ablation study on Qwen3-4B. We compare MARGO trained on our selected training set with MARGO trained on a random subset of 5,660 examples sampled from the same TriviaQA training pool. All training and evaluation settings are kept identical. As shown in Table 3, random data selection yields limited improvement over the fixed thinking baseline, while our selected data leads to consistent gains across all benchmarks. This suggests that examples with clear thinking/non-thinking factuality gaps are more effective for mixed-mode advantage regularization.

Generation Length Statistics Table 5 reports the average generation length of Fixed$_{Think}$ and MARGO on TriviaQA, NQ_Open, and PopQA. The length includes both the thinking trace and the final response. On Qwen3-4B, MARGO produces slightly shorter generations than Fixed$_{Think}$, with relative changes ranging from -5.24% to -3.42%. On Qwen3-8B, however, MARGO produces slightly longer generations, with relative changes ranging from $+0.09\%$ to $+1.48\%$. These results show that MARGO does not impose a consistent shortening effect on the generated outputs, which is consistent with its design of regularizing reasoning through mixed-mode advantage estimation rather than an explicit length penalty.

5 Conclusion

In this paper, we studied factuality-oriented QA in large reasoning models (LRMs) and identified *thinking-induced hallucination*, where explicit thinking can overturn correct non-thinking answers and lead to factual drift. We interpreted explicit thinking as a reasoning residual over the model's direct-answer behavior, which may either recover missing knowledge or introduce unsupported associations. Based on this view, we proposed MARGO, a mixed-mode reinforcement learning framework that uses non-thinking rollouts as same-model references to regularize hallucination-prone thinking. Experiments across multiple factuality-oriented QA benchmarks show that MARGO improves factual reliability over strong baselines, while preserving general reasoning ability on mathematical benchmarks. Overall, our findings suggest that improving LRMs requires not only encouraging more thinking, but also learning when explicit thinking provides factual value over direct answering.

Acknowledgements

This work was partially supported by NSF IIS 2347592, 2348169, DBI 2405416, CCF 2348306, CNS 2347617, RISE 2536663. The authors declare no competing interests.

References

Ruibo Chen, Jiacheng Pan, Heng Huang, and Zhenheng Yang. Improving text-to-image generation with input-side inference-time scaling. *arXiv preprint arXiv:2510.12041*, 2025a.

Xilun Chen, Ilia Kulikov, Vincent-Pierre Berges, Barlas Oğuz, Rulin Shao, Gargi Ghosh, Jason Weston, and Wen-tau Yih. Learning to reason for factuality. *arXiv preprint arXiv:2508.05618*, 2025b.

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.

Luyu Gao, Aman Madaan, Shuyan Zhou, Uri Alon, Pengfei Liu, Yiming Yang, Jamie Callan, and Graham Neubig. Pal: Program-aided language models. In *International conference on machine learning*, pages 10764–10799. PMLR, 2023.

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

Lukas Haas, Gal Yona, Giovanni D'Antonio, Sasha Goldshtein, and Dipanjan Das. Simpleqa verified: A reliable factuality benchmark to measure parametric knowledge. *arXiv preprint arXiv:2509.07968*, 2025.

Jinyi Han, Ying Huang, Ying Liao, Zishang Jiang, Xikun Lu, Haiquan Zhao, Xinyi Wang, Guanghao Zhou, Sihang Jiang, Jiaqing Liang, et al. Your models have thought enough: Training large reasoning models to stop overthinking. *arXiv preprint arXiv:2509.23392*, 2025.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing. *arXiv preprint arXiv:2111.09543*, 2021.

Zhenyu Hou, Xin Lv, Rui Lu, Jiajie Zhang, Yujiang Li, Zijun Yao, Juanzi Li, Jie Tang, and Yuxiao Dong. T1: Advancing language model reasoning through reinforcement learning and inference scaling. *arXiv preprint arXiv:2501.11651*, 2025.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, et al. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*, 43(2):1–55, 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.

Neharika Jali, Anupam Nayak, and Gauri Joshi. Not all turns are equally hard: Adaptive thinking budgets for efficient multi-turn reasoning. *arXiv preprint arXiv:2604.05164*, 2026.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611, 2017.

Ehsan Kamalloo, Nouha Dziri, Charles Clarke, and Davood Rafiei. Evaluating open-domain question answering in the era of large language models. In *Proceedings of the 61st annual meeting of the Association for Computational Linguistics (volume 1: long papers)*, pages 5591–5606, 2023.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th symposium on operating systems principles*, pages 611–626, 2023.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33: 9459–9474, 2020.

Junyi Li and Hwee Tou Ng. The hallucination dilemma: Factuality-aware reinforcement learning for large reasoning models. *arXiv e-prints*, pages arXiv–2505, 2025.

Pingzhi Li, Bairu Hou, Yun Zhu, Yihao Feng, Ke Ye, Tao Lei, Zhifeng Chen, Tianlong Chen, and Xianzhi Du. Adaptive thinking: Large language models know when to think in latent space. In *The Fourteenth International Conference on Learning Representations*.

Qing Lyu, Shreya Havaldar, Adam Stein, Li Zhang, Delip Rao, Eric Wong, Marianna Apidianaki, and Chris Callison-Burch. Faithful chain-of-thought reasoning. In *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 305–329, 2023.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *Proceedings of the 61st annual meeting of the association for computational linguistics (volume 1: Long papers)*, pages 9802–9822, 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.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. Language models as knowledge bases? In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*, pages 2463–2473, 2019.

Adam Roberts, Colin Raffel, and Noam Shazeer. How much knowledge can you pack into the parameters of a language model? In *Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP)*, pages 5418–5426, 2020.

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.

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*, pages 1279–1297, 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.

Zhongxiang Sun, Qipeng Wang, Haoyu Wang, Xiao Zhang, and Jun Xu. Detection and mitigation of hallucination in large reasoning models: A mechanistic perspective. *arXiv preprint arXiv:2505.12886*, 2025.

Xu Wan, Wei Wang, Wenyue Xu, Wotao Yin, Jie Song, and Mingyang Sun. Adapthink: Adaptive thinking preferences for reasoning language model. *arXiv preprint arXiv:2506.18237*, 2025.

Kaishen Wang and Heng Huang. Unsafe by reciprocity: How generation-understanding coupling undermines safety in unified multimodal models. *arXiv preprint arXiv:2603.27332*, 2026.

Kaishen Wang, Hengrui Gu, Meijun Gao, and Kaixiong Zhou. Damo: Decoding by accumulating activations momentum for mitigating hallucinations in vision-language models. In *The Thirteenth International Conference on Learning Representations*, 2025.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*, 2022.

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.

Jason Wei, Nguyen Karina, Hyung Won Chung, Yunxin Joy Jiao, Spencer Papay, Amelia Glaese, John Schulman, and William Fedus. Measuring short-form factuality in large language models. *arXiv preprint arXiv:2411.04368*, 2024.

Tianyi Xiong, Yi Ge, Ming Li, Zuolong Zhang, Pranav Kulkarni, Kaishen Wang, Qi He, Zeying Zhu, Chenxi Liu, Ruibo Chen, et al. Multi-crit: Benchmarking multimodal judges on pluralistic criteria-following. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8641–8652, 2026.

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.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D Manning. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pages 2369–2380, 2018.

Zijun Yao, Yantao Liu, Yanxu Chen, Jianhui Chen, Junfeng Fang, Lei Hou, Juanzi Li, and Tat-Seng Chua. Are reasoning models more prone to hallucination? *arXiv preprint arXiv:2505.23646*, 2025.

Le Yu, Kaishen Wang, Jianlong Xiong, Yue Cao, Lei Zhang, and Zhang Yi Tao He. Hallurnn: Mitigating hallucinations via recurrent cross-layer reasoning in large vision-language models. *arXiv preprint arXiv:2506.17587*, 2025.

Guibin Zhang, Yuxiang Zhang, Moayad Aloqaily, Haolang Lu, Kun Wang, Jing Liang, Xingjun Ma, Yu-Gang Jiang, and Qingsong Wen. Unraveling hallucination in large reasoning models: A topological perspective.

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*, pages 3716–3730, 2025.

Chuanyang Zheng, Zhengying Liu, Enze Xie, Zhenguo Li, and Yu Li. Progressive-hint prompting improves reasoning in large language models. *arXiv preprint arXiv:2304.09797*, 2023.

Tong Zheng, Chengsong Huang, Runpeng Dai, Yun He, Rui Liu, Xin Ni, Huiwen Bao, Kaishen Wang, Hongtu Zhu, Jiaxin Huang, et al. Parallel-probe: Towards efficient parallel thinking via 2d probing. *arXiv preprint arXiv:2602.03845*, 2026.

Limitations

This work focuses on improving the factual reliability of LRMs in factuality-oriented QA. While our experiments cover multiple benchmarks and model scales, future work can further explore the applicability of mixed-mode advantage regularization to broader scenarios, such as open-ended generation, multi-turn interaction, retrieval-augmented systems, and multimodal reasoning.

Broader Impact

By reducing cases where explicit thinking turns correct direct answers into hallucinated predictions, MARGO can help improve the reliability of LRMs in information-seeking and educational applications. At the same time, the method is intended to complement, rather than replace, external verification mechanisms in high-stakes domains. We hope this work encourages future research on more reliable and mode-aware optimization for LRMs.

A Advantage Decomposition of Mixed-Mode Regularization

In this section, we provide a formal derivation showing why mixed-mode rollout groups introduce a residual-value adjustment that is absent in standard all-thinking GRPO. This analysis characterizes the advantage signal induced by group normalization, rather than providing a convergence guarantee.

Setup. For a factual question x, let T and N denote the thinking and non-thinking modes, respectively. A response sampled under mode $m \in \{\mathrm{T}, \mathrm{N}\}$ is denoted by $y^m \sim \pi_\theta(\cdot \mid x, m)$, and its factual reward is $R(x, y^m)$. We define the expected reward of each mode as

$$\mu_m(x) = \mathbb{E}_{y^m \sim \pi_\theta(\cdot|x,m)} \left[R(x, y^m) \right], \qquad m \in \{\mathrm{T}, \mathrm{N}\}. \tag{15}$$

The residual value of explicit thinking is defined as

$$\Delta_{\mathrm{res}}(x) = \mu_{\mathrm{T}}(x) - \mu_{\mathrm{N}}(x). \tag{16}$$

A positive $\Delta_{\mathrm{res}}(x)$ indicates that thinking provides higher expected factual reward than direct answering, while a negative value indicates that thinking harms factuality relative to the non-thinking reference.

All-Thinking GRPO. Consider a standard all-thinking rollout group where all samples are generated in the thinking mode. Ignoring the standard-deviation normalization for clarity, the group-relative advantage of a thinking response y^{T} is centered around the all-thinking reward baseline:

$$A_{\mathrm{T}}(x, y^{\mathrm{T}}) \propto R(x, y^{\mathrm{T}}) - \mu_{\mathrm{T}}(x). \tag{17}$$

This advantage compares y^{T} only against other thinking-mode samples. Therefore, it can rank thinking trajectories within the thinking distribution, but it does not reveal whether thinking itself improves factuality over the model's direct non-thinking behavior.

Mixed-Mode GRPO. Now consider a mixed rollout group containing thinking and non-thinking trajectories with proportions α and $1 - \alpha$, respectively. The expected group reward baseline is

$$\mu_{\mathrm{mix}}(x) = \alpha \mu_{\mathrm{T}}(x) + (1 - \alpha) \mu_{\mathrm{N}}(x). \tag{18}$$

Proposition 1 (Residual-value decomposition of mixed-mode advantage)**.** *Ignoring the standard-deviation normalization and replacing the empirical group mean with its expectation, the mixed-mode advantage of a thinking trajectory can be decomposed as*

$$A^{\mathrm{T}}_{\mathrm{mix}}(x, y^{\mathrm{T}}) \propto \big(R(x, y^{\mathrm{T}}) - \mu_{\mathrm{T}}(x) \big) + (1 - \alpha) \Delta_{\mathrm{res}}(x). \tag{19}$$

The first term is the within-thinking advantage, and the second term is the residual-value adjustment. Similarly, the mixed-mode advantage of a non-thinking trajectory can be decomposed as

$$A^{\mathrm{N}}_{\mathrm{mix}}(x, y^{\mathrm{N}}) \propto \big(R(x, y^{\mathrm{N}}) - \mu_{\mathrm{N}}(x) \big) - \alpha \Delta_{\mathrm{res}}(x). \tag{20}$$

Here, the first term is the within-non-thinking advantage, and the second term is the opposite residual-value adjustment.

Proof. For a thinking trajectory y^{T}, the mixed-mode group-relative advantage is centered by the mixed reward baseline:

$$A^{\mathrm{T}}_{\mathrm{mix}}(x, y^{\mathrm{T}}) \propto R(x, y^{\mathrm{T}}) - \mu_{\mathrm{mix}}(x). \tag{21}$$

Substituting the definition of $\mu_{\mathrm{mix}}(x)$ gives

$$\begin{aligned} A^{\mathrm{T}}_{\mathrm{mix}}(x, y^{\mathrm{T}}) &\propto R(x, y^{\mathrm{T}}) - \alpha\mu_{\mathrm{T}}(x) - (1-\alpha)\mu_{\mathrm{N}}(x) \\ &= R(x, y^{\mathrm{T}}) - \mu_{\mathrm{T}}(x) + (1-\alpha)\mu_{\mathrm{T}}(x) - (1-\alpha)\mu_{\mathrm{N}}(x) \\ &= R(x, y^{\mathrm{T}}) - \mu_{\mathrm{T}}(x) + (1-\alpha)\left(\mu_{\mathrm{T}}(x) - \mu_{\mathrm{N}}(x)\right) \\ &= \big(R(x, y^{\mathrm{T}}) - \mu_{\mathrm{T}}(x)\big) + (1-\alpha)\Delta_{\mathrm{res}}(x). \end{aligned} \tag{22}$$

This proves the decomposition for thinking trajectories.

For a non-thinking trajectory y^{N}, we similarly have

$$A^{\mathrm{N}}_{\mathrm{mix}}(x, y^{\mathrm{N}}) \propto R(x, y^{\mathrm{N}}) - \mu_{\mathrm{mix}}(x). \tag{23}$$

Substituting the definition of $\mu_{\mathrm{mix}}(x)$ gives

$$\begin{aligned} A^{\mathrm{N}}_{\mathrm{mix}}(x, y^{\mathrm{N}}) &\propto R(x, y^{\mathrm{N}}) - \alpha\mu_{\mathrm{T}}(x) - (1-\alpha)\mu_{\mathrm{N}}(x) \\ &= R(x, y^{\mathrm{N}}) - \mu_{\mathrm{N}}(x) - \alpha\mu_{\mathrm{T}}(x) + \alpha\mu_{\mathrm{N}}(x) \\ &= R(x, y^{\mathrm{N}}) - \mu_{\mathrm{N}}(x) - \alpha\left(\mu_{\mathrm{T}}(x) - \mu_{\mathrm{N}}(x)\right) \\ &= \big(R(x, y^{\mathrm{N}}) - \mu_{\mathrm{N}}(x)\big) - \alpha\Delta_{\mathrm{res}}(x). \end{aligned} \tag{24}$$

This completes the proof. □

Implication. Proposition 1 shows that mixed-mode advantage estimation introduces a residual-value adjustment that is absent from all-thinking GRPO. For thinking trajectories, this adjustment is proportional to $\Delta_{\mathrm{res}}(x)$. When $\Delta_{\mathrm{res}}(x) > 0$, thinking has higher expected factual reward than non-thinking, and the mixed-mode advantage increases the relative preference for thinking trajectories. When $\Delta_{\mathrm{res}}(x) < 0$, thinking harms factuality relative to direct answering, and the mixed-mode advantage reduces the relative preference for thinking trajectories. Thus, mixed-mode GRPO does not merely reward correct final answers; it adjusts the optimization signal according to whether explicit thinking provides factual value beyond the model's direct-answer behavior.

Relation to `MARGO`. `MARGO` instantiates this principle by constructing each GRPO rollout group with both thinking and non-thinking trajectories for the same question. The reward function itself does not explicitly prefer either mode, nor does it impose a handcrafted penalty on thinking length. Instead, the regularization effect emerges from group-wise advantage normalization: non-thinking trajectories serve as same-model references that expose whether the reasoning residual introduced by thinking is beneficial or harmful for factuality.

B Prompts

We present the prompt used for factuality evaluation. Specifically, the judge model takes as input the question, the ground-truth answer (denoted as the gold target in the prompt), and the model prediction (denoted as the predicted answer in the prompt). Based on the consistency between the prediction and the ground truth, the judge model outputs one of three labels: "A", "B", or "C", representing correct, incorrect, and not attempted, respectively. For evaluation, we count only "A" as a correct prediction, while both "B" and "C" are treated as incorrect.

Importantly, this is different from using an LLM as an open-ended factual verifier. The judge is not asked to determine factual truth based on its own parametric knowledge. Instead, it is explicitly provided with the question, the ground-truth answer, and the model prediction, and is only required to compare whether the prediction matches the provided ground truth. This ground-truth-conditioned evaluation reduces reliance on the judge model's internal knowledge and focuses the evaluation on semantic correctness rather than surface-form overlap.

Prompt for Factuality Evaluation

```
Your job is to look at a question, a gold target, and a predicted answer, and then assign a grade of either
↪  ["CORRECT", "INCORRECT", "NOT_ATTEMPTED"].
First, I will give examples of each grade, and then you will grade a new example.


The following are examples of CORRECT predicted answers.
```
Question: What are the names of Barack Obama's children?
Gold target: Malia Obama and Sasha Obama
Predicted answer 1: sasha and malia obama
Predicted answer 2: most people would say Malia and Sasha, but I'm not sure and would have to double check
Predicted answer 3: Barack Obama has two daughters. Their names are Malia Ann and Natasha Marian, but they are
↪ commonly referred to as Malia Obama and Sasha Obama. Malia was born on July 4, 1998, and Sasha was born on June
↪ 10, 2001.
```
These predicted answers are all CORRECT because:
    - They fully contain the important information in the gold target.
    - They do not contain any information that contradicts the gold target.
    - Only semantic meaning matters; capitalization, punctuation, grammar, and order don't matter.
    - Hedging and guessing are permissible, provided that the gold target is fully included and the response contains
    ↪  no incorrect information or contradictions.


The following are examples of INCORRECT predicted answers.
```
Question: What are the names of Barack Obama's children?
Gold target: Malia and Sasha
Predicted answer 1: Malia.
Predicted answer 2: Malia, Sasha, and Susan.
Predicted answer 3: Barack Obama does not have any children.
Predicted answer 4: I think it's either Malia and Sasha. Or it could be Malia and Jackie. Or it could be Joey and
↪ Malia.
Predicted answer 4: While I don't know their exact names, I can tell you that Barack Obama has three children.
Predicted answer 5: It's possible you may mean Betsy and Olivia. However, you should clarify further details with
↪ updated references if necessary. Is that the correct answer?
Predicted answer 6: It may be the case that Obama's child is named James. However, it's recommended to confirm the
↪ most accurate and updated information since this could change over time. This model may not always reflect the
↪ most current information.
```
These predicted answers are all INCORRECT because:
    - A factual statement in the answer contradicts the gold target. Incorrect statements that have some hedging
    ↪  (e.g., "it is possible that", "although i'm not sure, i think") are also considered incorrect.


The following are examples of NOT_ATTEMPTED predicted answers.
```
Question: What are the names of Barack Obama's children?
Gold target: Malia and Sasha
Predicted answer 1: I don't know.
Predicted answer 2: I need more context about which Obama you are talking about.
Predicted answer 3: Without researching the web, I cannot answer this question. However, I can tell you that Barack
↪ Obama has two children.
Predicted answer 4: Barack Obama has two children. I know that one of them is Malia, but I'm not sure about the other
↪ one.
```
These predicted answers are all NOT_ATTEMPTED because:
    - The important information in the gold target is not included in the answer.
    - No statements in the answer contradict the gold target.


Also note the following things:
- For grading questions where the gold target is a number, the predicted answer needs to be correct to the last
↪  significant figure in the gold answer. For example, consider a question "How many citations does the Transformer
↪  Paper have?" with gold target "120k".
    - Predicted answers "120k", "124k", and 115k" are all CORRECT.
    - Predicted answers "100k" and "113k" are INCORRECT.
    - Predicted answers "around 100k" and "more than 50k" are considered NOT_ATTEMPTED because they neither confirm
    ↪  nor contradict the gold target.
- The gold target may contain more information than the question. In such cases, the predicted answer only needs to
↪  contain the information that is in the question.
    - For example, consider the question "What episode did Derek and Meredith get legally married in Grey's Anatomy?"
    ↪  with gold target "Season 7, Episode 20: White Wedding". Either "Season 7, Episode 20" or "White Wedding" would
    ↪  be considered a CORRECT answer.
- Do not punish predicted answers if they omit information that would be clearly inferred from the question.
    - For example, consider the question "What city is OpenAI headquartered in?" and the gold target "San Francisco,
    ↪  California". The predicted answer "San Francisco" would be considered CORRECT, even though it does not include
    ↪  "California".
    - Consider the question "What award did A pretrainer's guide to training data: Measuring the effects of data age,
    ↪  domain coverage, quality, & toxicity win at NAACL '24?", the gold target is "Outstanding Paper Award". The
    ↪  predicted answer "Outstanding Paper" would be considered CORRECT, because "award" is presumed in the question.
    - For the question "What is the height of Jason Wei in meters?", the gold target is "1.73 m". The predicted answer
    ↪  "1.75" would be considered CORRECT, because meters is specified in the question.
    - For the question "What is the name of Barack Obama's wife?", the gold target is "Michelle Obama". The predicted
    ↪  answer "Michelle" would be considered CORRECT, because the last name can be presumed.
- Do not punish for typos in people's name if it's clearly the same name.
    - For example, if the gold target is "Hyung Won Chung", you can consider the following predicted answers as
    ↪  correct: "Hyoong Won Choong", "Hyungwon Chung", or "Hyun Won Chung".
```

```
Here is a new example. Simply reply with either CORRECT, INCORRECT, NOT ATTEMPTED. Don't apologize or correct yourself
↪  if there was a mistake; we are just trying to grade the answer.
```
Question: {question}
Gold target: {target}
Predicted answer: {predicted_answer}
```

Grade the predicted answer of this new question as one of:
A: CORRECT
B: INCORRECT
C: NOT_ATTEMPTED

Just return the letters "A", "B", or "C", with no text around it.
```

C Additional Experimental Details

C.1 Training Data Construction

We construct our training data from the TriviaQA training split. The original TriviaQA training split contains 138,384 question-answer pairs, but many questions are duplicated. After removing duplicated questions, we obtain 76,523 unique training examples.

For each base model, i.e., Qwen3-4B and Qwen3-8B, we independently construct a model-specific training set. Specifically, for each question x, we sample N responses under the thinking mode and N responses under the non-thinking mode. We then use Qwen3-32B as the automatic judge to determine whether each sampled response is correct with respect to the ground-truth answer. In our setting, N is set to 6 by default.

Let $c_k^{\mathrm{T}}(x) \in \{0,1\}$ and $c_k^{\mathrm{N}}(x) \in \{0,1\}$ denote the correctness of the k-th sampled response under the thinking and non-thinking modes, respectively. We compute the empirical correctness ratios of the two modes as:

$$S_{\mathrm{T}}(x) = \frac{1}{N}\sum_{k=1}^{N} c_k^{\mathrm{T}}(x), \qquad S_{\mathrm{N}}(x) = \frac{1}{N}\sum_{k=1}^{N} c_k^{\mathrm{N}}(x). \tag{25}$$

Here, $S_{\mathrm{T}}(x)$ and $S_{\mathrm{N}}(x)$ estimate the model's factual correctness under the thinking and non-thinking modes, respectively.

We further compute the average mode score:

$$b(x) = \frac{S_{\mathrm{T}}(x) + S_{\mathrm{N}}(x)}{2}, \tag{26}$$

and define the mode-specific relative scores as:

$$A_{\mathrm{T}}(x) = S_{\mathrm{T}}(x) - b(x), \qquad A_{\mathrm{N}}(x) = S_{\mathrm{N}}(x) - b(x). \tag{27}$$

Equivalently, the factuality gap between the two modes is:

$$\Delta(x) = S_{\mathrm{T}}(x) - S_{\mathrm{N}}(x). \tag{28}$$

We retain examples where the two modes show a clear factuality gap. For Qwen3-4B, an example is selected as a non-thinking-favored example if:

$$\Delta(x) \le -0.7 \quad \text{and} \quad S_{\mathrm{N}}(x) \ge 0.7. \tag{29}$$

This condition indicates that the model is substantially more accurate under the non-thinking mode than under the thinking mode. Similarly, an example is selected as a thinking-favored example if:

$$\Delta(x) \ge 1.0 \quad \text{and} \quad S_{\mathrm{T}}(x) \ge 1.0. \tag{30}$$

This condition selects questions where the model consistently answers correctly under the thinking mode while performing worse under the non-thinking mode.

For Qwen3-8B, we use a slightly relaxed threshold, since the relative gap between thinking and non-thinking modes becomes less extreme as the base model becomes stronger. Specifically, we select non-thinking-favored examples using:

$$\Delta(x) \le -0.5 \quad \text{and} \quad S_{\mathrm{N}}(x) \ge 0.5, \tag{31}$$

Model	Method	TriviaQA	NQ_Open	PopQA
Qwen3-4B	Fixed$_{Think}$	1128.45	1192.60	811.13
	MARGO	1074.61	1151.78	768.60
	Relative Change	-4.77%	-3.42%	-5.24%
Qwen3-8B	Fixed$_{Think}$	1089.75	1268.82	868.98
	MARGO	1103.45	1287.65	869.79
	Relative Change	+1.26%	+1.48%	+0.09%

Table 5: Average generation length on factuality-oriented QA benchmarks. The length includes both the thinking trace and the final response. Relative change is computed with respect to Fixed$_{Think}$, where positive values indicate longer generations and negative values indicate shorter generations. The results show no consistent shortening trend across model scales.

and thinking-favored examples using:

$$\Delta(x) \geq 0.7 \quad \text{and} \quad S_{\mathrm{T}}(x) \geq 0.7. \tag{32}$$

The final training set is the union of the selected examples from both directions. This construction focuses training on questions where thinking and non-thinking exhibit clear mode-dependent factual differences, which provide informative signals for mixed-mode GRPO. All filtering is performed only on the TriviaQA training split, and no information from any test benchmark is used. This procedure yields 5,660 training examples for Qwen3-4B and 5,721 training examples for Qwen3-8B.

Although the resulting training sets are relatively small, they are sufficient for our setting because we start from existing reasoning models rather than training reasoning behavior from scratch. Our goal is not to teach the model new factual knowledge or build reasoning ability from a base model, but to regularize when explicit thinking is helpful or harmful relative to the model's own non-thinking behavior. Therefore, a compact set of examples with clear mode-dependent factual differences provides targeted supervision for our mixed-mode optimization.

C.2 Detailed Results

We provide the detailed results for Figure 2(c) in Table 4. Specifically, we report Mean@16 and Pass@16 on AMC23, AIME24, and AIME25 for both Qwen3-4B and Qwen3-8B. Compared with Fixed$_{NoThink}$ and Fixed$_{Think}$, MARGO achieves the best Mean@16 on all three benchmarks across both model scales. These results show that MARGO improves factual reliability while preserving the general reasoning ability.

D Additional Transition Analysis

Beyond the TriviaQA analysis in Section 3.2, we further examine the transition behavior on additional benchmarks, including NQ_Open and PopQA. As shown in Tables 6 and 7, thinking-induced hallucination also consistently appears on these factuality-oriented QA benchmarks. On NQ_Open, the effect is particularly pronounced. For the 8B, 14B, and 32B models, $r_{1,0}$ reaches 8.12%, 7.89%, and 7.62%, respectively, which is close to the corresponding $r_{0,1}$ values of 9.42%, 8.95%, and 9.61%. This indicates that, on NQ_Open, the fraction of examples harmed by thinking is comparable to the fraction of examples corrected by thinking. A similar trend also appears on PopQA, where $r_{1,0}$ remains around 4%–5% for most model scales, showing that correctness-to-incorrectness transitions are not limited to a single factuality benchmark.

We further compare this trend with GSM8K [Cobbe et al., 2021], a mathematical reasoning benchmark. As shown in Table 8, the $r_{1,0}$ ratios on GSM8K remain below 2% across all model scales, with values ranging from 0.88% to 1.84%. Meanwhile, the $r_{1,1}$ ratios are consistently high, reaching over 86% for models from 4B to 32B. This suggests that, unlike factuality-oriented QA, explicit thinking in mathematical reasoning is much less likely to overturn an originally correct prediction. This contrast aligns with the nature of the tasks: factuality QA often requires retrieving specific knowledge, where additional thinking may introduce unsupported associations or entity confusion; in mathematical reasoning, intermediate steps are more likely to be grounded in the problem conditions and help derive the final answer.

Overall, these additional analyses further support the key observation of our paper: thinking-induced hallucination is a recurring issue in factuality-oriented QA, rather than an artifact of a single benchmark. They also show that the risk of harmful thinking is task-dependent, being more prominent in knowledge-intensive factual QA than in mathematical reasoning.

Size	$r_{0,0}$	$r_{0,1}$	$r_{1,0}$	$r_{1,1}$
1.7B	71.39%	8.81%	4.71%	15.10%
4B	61.55%	8.95%	5.51%	23.99%
8B	51.94%	9.42%	8.12%	30.53%
14B	46.12%	8.95%	7.89%	37.04%
32B	43.35%	9.61%	7.62%	39.42%

Table 6: Transition ratios between non-thinking and thinking modes on NQ_Open across Qwen3 models of different scales. Each $r_{i,j}$ denotes the fraction of examples whose prediction changes from correctness i under the non-thinking mode to correctness j under the thinking mode.

Size	$r_{0,0}$	$r_{0,1}$	$r_{1,0}$	$r_{1,1}$
1.7B	80.98%	4.09%	4.10%	10.84%
4B	77.85%	5.07%	3.10%	13.98%
8B	70.79%	7.07%	4.56%	17.57%
14B	68.09%	6.16%	4.50%	21.25%
32B	66.01%	7.10%	4.42%	22.47%

Table 7: Transition ratios between non-thinking and thinking modes on PopQA across Qwen3 models of different scales. Each $r_{i,j}$ denotes the fraction of examples whose prediction changes from correctness i under the non-thinking mode to correctness j under the thinking mode.

Size	$r_{0,0}$	$r_{0,1}$	$r_{1,0}$	$r_{1,1}$
1.7B	9.02%	16.76%	1.84%	72.38%
4B	4.70%	8.04%	1.03%	86.23%
8B	4.02%	5.99%	0.88%	89.11%
14B	2.58%	5.00%	1.34%	91.08%
32B	2.58%	5.00%	1.11%	91.31%

Table 8: Transition ratios between non-thinking and thinking modes on GSM8K across Qwen3 models of different scales. Each $r_{i,j}$ denotes the fraction of examples whose prediction changes from correctness i under the non-thinking mode to correctness j under the thinking mode.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We provide contribution and scope clearly in abstract and introduction.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Discussed in the Limitations section.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: Provided in Appendix A.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: All experimental details are provided in Section 4.1 and Appendix C.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: All datasets used are open source. The code is available at `https://github.com/tunantu/MARGO`.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: All experimental details are provided in Section 4.1 and Appendix C.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: We could not provide error bars deviation due to computational costs. No other prior work and baselines in our work do this either.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: All details are provided in Section 4.1 and Appendix C.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: There are not ethical concerns that we know of.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Discussed in the Broader Impact section.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All original owners of assets have been properly cited.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: We do not release new assets.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.