From "Aha Moments" to Controllable Thinking: Toward Meta-Cognitive Reasoning in LRMs via Decoupled Reasoning and Control

Rui Ha[1], Rui Pu[1], Chaozhuo Li[1], Li Sun[1], Sen Su[1,2,†]

[1]Beijing University of Posts and Telecommunications, China
[2]Chongqing University of Posts and Telecommunications, China
{harry, puruirui, lichaozhuo, lsun, susen}@bupt.edu.cn

Abstract

Large Reasoning Models (LRMs) can exhibit step-by-step reasoning, reflection, and backtracking, but these behaviors are often unregulated, leading to overthinking. As a result, LRMs continue generating redundant reasoning even after reaching high-confidence conclusions. This increases inference cost and latency, limiting practical deployment. The root cause is the absence of an intrinsic mechanism to monitor the reasoning state and decide when to continue, backtrack, or stop. We propose MERA, a meta-cognitive reasoning framework that decouples reasoning from control to enable independent optimization of control strategies. MERA constructs high-quality reasoning–control supervision data via a takeover-based pipeline, and transforms long-horizon traces into structured reasoning–control alternating sequences for training. The model is trained with supervised fine-tuning to internalize the structured separation, and further optimized with Control-Segment Policy Optimization (CSPO), which combines segment-wise GRPO with control masking to focus learning on control segments. Experiments across reasoning benchmarks show that MERA improves both efficiency and accuracy.

1 Introduction

Large Reasoning Models (LRMs) have achieved significant advancements in complex tasks such as mathematical problem-solving and symbolic reasoning by integrating cognitive operations including step-by-step reasoning, reflection, and backtracking (Li et al., 2025b; Xu et al., 2025; Wang et al., 2026). These emergent capabilities, often described as "Aha Moments," demonstrate the model's latent ability to engage in complex test-time reasoning (Guo et al., 2025a; Li et al., 2025a). However, such cognitive behaviors remain unregulated and uncontrolled, frequently resulting in over-

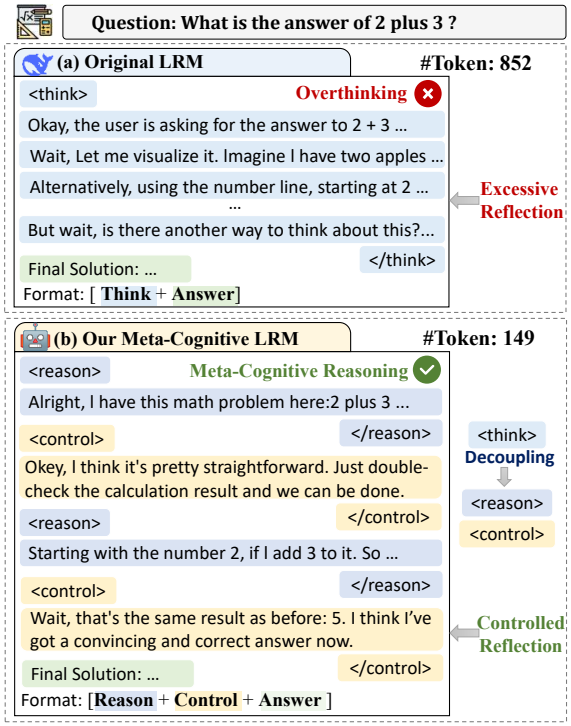

Figure 1: The comparison of reasoning answers between Original LRM and our proposed Meta-Cognitive LRM.

thinking, where the model continues generating redundant reasoning content even after reaching high-confidence conclusions (Chen et al., 2024; Wu et al., 2025; Zhao et al., 2022). This leads to substantial computational overhead and increased response latency, thereby limiting the practical deployment of LRMs (Chang et al., 2025; Qu et al., 2025; Liu et al., 2026).

The current approaches to addressing this challenge can be classified into three main categories. Methods that directly shorten the reasoning length often lack adaptability to problems of varying difficulty, which may result in excessive shortening and a subsequent decline in performance (Luo et al., 2025a; Wang et al., 2025). Some methods introduce preset budget mechanisms, which either rely

[†]The corresponding author.

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 6698–6710

on coarse-grained control between fast and slow thinking modes (Zhang et al., 2025a; Lou et al., 2025), or impose length constraints prior to answering (Shen et al., 2025; Aggarwal and Welleck, 2025), both lacking the flexibility to adjust the model's state during the reasoning process. Dynamic early-stopping methods, while capable of determining the optimal termination point of reasoning, still fundamentally rely on external evaluation metrics such as confidence thresholds to make decisions (Yang et al., 2025a; Qiao et al., 2025).

The limitation of these approaches lies in their treatment of length control as an external intervention. Specifically, they rely on predefined rules or external metrics, rather than enabling the model to decide when to stop based on its own reasoning state. In contrast, humans can flexibly allocate cognitive resources in real time during problem solving (Erickson and Heit, 2013). This leads to a critical question: *Are LRMs capable of effectively regulating their own behavior during reasoning?*

As shown in Figure 1(a), even when reliable intermediate conclusions have already been reached, LRMs tend to repeatedly perform cognitive operations such as reflection and backtracking, resulting in substantial redundant generation. This indicates that LRMs lack an internal self-regulation mechanism during reasoning: they often struggle to decide when to continue, whether reflection or backtracking is warranted, or when to terminate reasoning. Inspired by the concept of meta-cognition in cognitive psychology (Wagner, 2017; Liu et al., 2025), which refers to the awareness and regulation of one's own cognitive processes, we characterize this issue as a deficiency in the model's meta-cognitive capacity.

In current frameworks, control behaviors are often entangled with the reasoning process and jointly optimized under the same objective function. Control behaviors often degrade into static strategies, and models trained under this framework prioritize generating correct answers rather than optimizing the reasoning path, further exacerbating redundant generation in the reasoning process.

To address the above issue, we propose the Meta-cognitive Reasoning Framework (MERA). Unlike conventional methods that treat the thinking process as a unified whole, MERA's core innovation lies in decoupling the thinking process within LRMs into two distinct components: reasoning and control. By employing structurally separated training and optimization, MERA facilitates authentic meta-cognitive regulation. As illustrated in Figure 1(b), unlike prior models that blindly repeat cognitive operations without self-awareness, the models trained with MERA first assesse the current reasoning state. Upon recognizing that the reasoning output is accurate and reliable, they promptly issue a definitive control signal to terminate the reasoning process.

However, introducing an independent control component brings three new challenges. First, there is a severe scarcity of reasoning-control data, as existing training datasets generally lack high-quality meta-cognitive control annotations, and manually labeling such fine-grained data is extremely costly. Second, distinguishing control behaviors from reasoning content presents a notable challenge, as the reasoning traces in LRMs are often lengthy and lack explicit structural boundaries, making it difficult to clearly separate the two. Third, control instructions are often dispersed across multiple segments of reasoning, which causes policy optimization signals to be diluted by non-critical content during training.

To address these challenges, our framework introduces three key mechanisms. First, to mitigate the scarcity of high-quality reasoning-control data, a control-takeover mechanism is designed, which identifies critical moments during reasoning where meta-cognitive control is required, and delegates control generation at these points to auxiliary LLMs. Second, a structured decoupling mechanism is implemented via supervised fine-tuning, enabling the model to generate explicit reasoning and control tags, thereby achieving clear separation between reasoning and control processes. Third, a Control-Segment Policy Optimization (CSPO) method is proposed, which combines segment-wise GRPO and control masking to enable targeted optimization of control behavior with minimal interference from irrelevant content.

This paper makes the following contributions:

- We frame overthinking as a deficiency in fine-grained internal control, and we propose a reasoning-control decoupling strategy to endow LRMs with meta-cognitive regulatory capabilities.

- We propose MERA, a framework that integrates control-takeover, structural separation, and CSPO to overcome key challenges in independent control optimization.

- Comprehensive experiments show that MERA significantly improves both accuracy and efficiency on various reasoning benchmarks.

2 Methodology

To address the issue of overthinking exhibited by LRMs in complex tasks, we propose the **Meta-cognitive Reasoning Framework (MERA)**. This architecture injects structured, self-regulatory meta-cognitive capabilities into LRMs. As illustrated in Figure 2, MERA comprises three interrelated components: a decoupled modeling mechanism that separates reasoning and control, a control-driven data construction pipeline, and a training paradigm that integrates supervised fine-tuning with CSPO. Together, these modules enable the model to explicitly monitor, evaluate, and regulate its internal reasoning processes.

2.1 Decoupled Modeling of Thought Processes

2.1.1 Decoupling Definition

Traditional LRMs typically treat the entire reasoning process as a monolithic text generation stream, lacking built-in mechanisms for introspection and self-regulation. To enable finer-grained control over the internal cognitive process, we propose a structural decoupling of **reasoning** and **control** during generation. Specifically, we decompose the overall cognitive process into the following two functional modules:

- $r_k \in \mathcal{R}$: a **reasoning statement**, which represents a logical expression or step used to solve the task.
- $c_k \in \mathcal{C}$: a **control statement**, responsible for assessing and regulating the reasoning process.

The model output is composed of an alternating sequence of reasoning and control segments, formalized as:

$$\tau = \{(r_1, c_1),\ (r_2, c_2),\ \ldots,\ (r_K, c_K)\}\,. \quad (1)$$

Here, each pair (r_k, c_k) represents the k-th round of reasoning–control interaction.

Reasoning segments are denoted using the `<reason>` tag, while control segments are marked with the `<control>` tag. In line with prior work (Yang et al., 2025b), transitions between reasoning and control commonly occur at *cognitive turning points*, such as "wait", "hmm", or "alternative", which act as natural linguistic cues that signal the need for meta-cognitive intervention.

2.1.2 Problem Formalization

Under this structural framework, we formalize the meta-cognitive reasoning task as a conditional generation problem. Given an input query x, the model is required to generate an alternating reasoning–control sequence τ, followed by the final answer y. The overall generation probability can be factorized as:

$$\pi_\theta(\tau, y \mid x) = \pi_\theta(y \mid \tau, x) \cdot \pi_\theta(\tau \mid x), \quad (2)$$

where $\pi_\theta(\tau \mid x)$ denotes the probability of generating the structured reasoning–control sequence conditioned on the input, and $\pi_\theta(y \mid \tau, x)$ represents the probability of producing the final answer based on both the input and the generated reasoning trajectory.

2.2 Meta-cognitive Data Construction

To develop reasoning models with autonomous regulation capabilities, we construct a dataset enriched with explicit meta-cognitive signals, enabling the model not only to generate reasoning traces but also to dynamically monitor and control the reasoning process. The data construction process is divided into three structured stages: identification of control takeover points, generation of control signals, and construction of alternating sequences.

2.2.1 Identification of Control Takeover

We design a control takeover mechanism to automatically identify key takeover points in LRM reasoning. Long-horizon traces often exhibit step-by-step reasoning, reflection, and backtracking, which naturally form discrete segments. Segment transitions are typically marked by turning expressions such as "wait" and"alternatively", indicating hesitation, self-reflection, or shifts in reasoning. We treat these markers as control takeover signals, grounded in prior findings that such turning expressions consistently align with reasoning-state transitions and can serve as reliable anchors for trace segmentation (Yang et al., 2025a; Chen et al., 2024; Qian et al., 2025). Using these anchors, we insert explicit control instructions to enable a structured separation of reasoning and control.

2.2.2 Generation of Control Signals

After locating the takeover points, we use the Llama-3.3-70B-Instruct model (Grattafiori et al.,

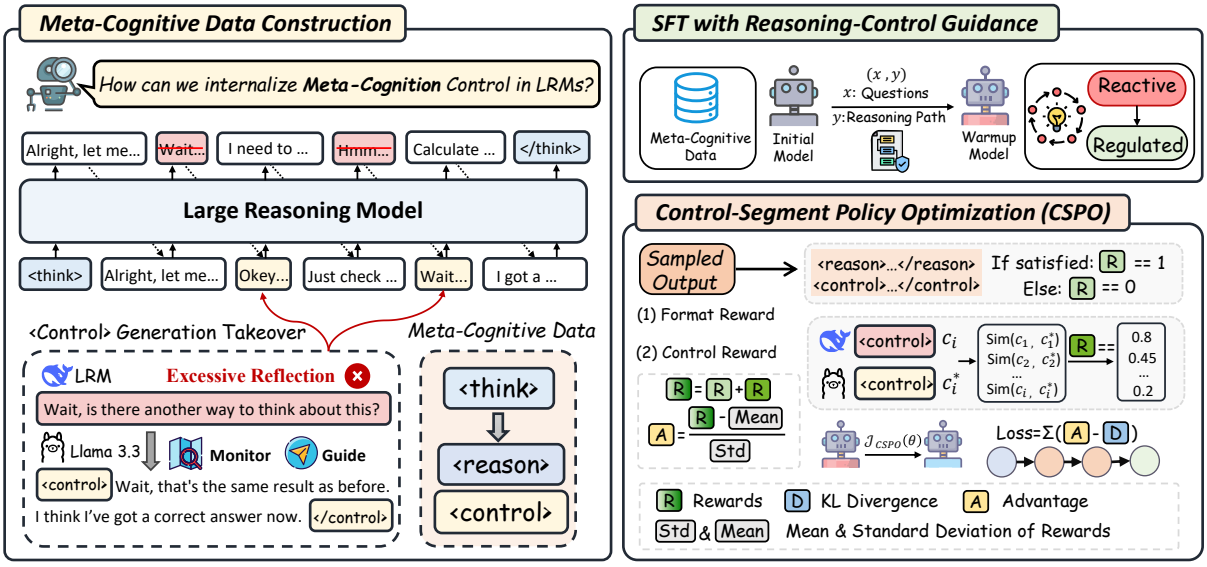

Figure 2: Overview of the proposed Meta-cognitive Reasoning Framework (MERA). The framework consists of three key components: Meta-cognitive Data Construction, SFT with Reasoning-Control Guidance, and Control-Segment Policy Optimization.

2024) to generate key control statements. Specifically, we design structured prompt templates that simulate a "meta-cognitive monitor" observing the model's reasoning process and request the following two tasks: evaluating the current reasoning and providing control suggestions. Based on the assessment of the current reasoning state, the model generates control statements and inserts them after a reasoning segment.

2.2.3 Construction of Alternating Sequences

After generating the control statements, we return the generation phase to the original LRMs to continue producing subsequent content. This process ensures that the reasoning chain alternates naturally between "control intervention–continue thinking" in a structured manner. Finally, we integrate the complete reasoning trajectory with the generated control statements, constructing the **reasoning–control alternating sequence** for model training. Each training sample also includes the final answer, allowing the model to simultaneously learn process regulation and task completion. The entire process automatically transforms the original reasoning data into structured samples in the form of triples (x_i, τ_i, y_i), where x_i is the input query, τ_i is the alternating reasoning–control sequence, and y_i is the final answer.

2.3 SFT with Reasoning-Control Guidance

To effectively guide the model in mastering the structured reasoning–control pattern, we design a unified supervised fine-tuning (SFT) mechanism based on the constructed dataset, allowing the model to simultaneously learn incremental reasoning and meta-cognitive regulation abilities during the generation process.

2.3.1 Joint Generation Modeling

During the training phase, each sample is modeled as a triplet (x_i, τ_i, y_i), where x_i represents the input query, τ_i is the structured intermediate process formed by alternating reasoning and control segments, and y_i is the final answer. We adopt the standard conditional language modeling objective to model the answer path:

$$\mathcal{L}_{\text{SFT}}(\theta) = -\sum_{i=1}^{N} \log \pi_\theta \left(\tau_i, y_i \mid x_i\right). \quad (3)$$

After the initial SFT, the model acquires the ability to decouple the original reasoning process by appropriately using the `<reason>` and `<control>` tags. It also learns to generate effective control content that includes both evaluative feedback and directive signals to guide the reasoning.

2.4 Control-Segment Policy Optimization

To further enhance the self-regulatory capabilities of LRMs during reasoning, we propose **Control Segment Policy Optimization (CSPO)**, a training framework designed to address two major challenges in reinforcement learning. First, control

directives are often distributed across multiple reasoning segments, making it difficult for standard GRPO to attribute rewards with fine granularity; Second, control content is highly sparse within sequences, causing policy updates to be easily diluted by non-critical positions. CSPO addresses these issues through the following three mechanisms: **(1) Segmented GRPO Modeling**, which provides independent reward feedback for each reasoning–control unit; **(2) Control Reward Modeling**, combining semantic and structural signals to guide learning; **(3) Control Masking**, which restricts optimization to control-relevant tokens, thus improving learning efficiency and stability.

2.4.1 Segmented GRPO Modeling

Inspired by prior work (Guo et al., 2025b), we partition the generated sequence into multiple segments (τ_i) to capture fine-grained control features within each segment. For each segment, we sample an output o_i from the policy and independently evaluate its corresponding reward. We adopt the GRPO policy optimization method (Shao et al., 2024), wherein G complete outputs are sampled from the previous policy $\pi_{\theta_{\text{old}}}$, followed by reward normalization and computation of the advantage function for each segment:

$$\hat{A}_{i,t} = \frac{r(o_i) - \text{mean}(r(o_1), \ldots, r(o_G)\})}{\text{std}(\{r(o_1), \ldots, r(o_G)\})}. \quad (4)$$

Here, $r(o_i)$ is the segment-level reward function defined below, computed from independent feedback per segment.

2.4.2 Control Reward Modeling

We design a control reward function $r(o_i)$ composed of two complementary components that evaluate both semantic correctness and structural conformity of the generated control content:

Control Reward (R_{ctrl}): This term measures whether the generated control segment c_k semantically aligns with the reference control target c_k^*. We compute semantic similarity using the GPT-4o model (OpenAI et al., 2024):

$$R_{\text{ctrl}}(c_k) = \text{Score}_{\text{Sim}}(c_k, c_k^*). \quad (5)$$

Format Reward (R_{format}): This term encourages the model to follow a standardized structure, particularly the `<reason>` and `<control>` format, which improves structural consistency and interpretability. The overall reward for each segment is computed as:

$$r(o_i) = R_{\text{ctrl}} + R_{\text{format}}. \quad (6)$$

This mask ensures that only tokens within control spans receive gradient updates, which significantly improves the precision and efficiency of policy optimization in control-sensitive tasks. The final objective function for CSPO is:

$$\mathcal{J}_{\text{CSPO}}(\theta) = \mathbb{E}_{x,o} \frac{1}{Z} \sum_{k=1}^{K} \frac{1}{G} \sum_{i=1}^{G} \frac{1}{|o_i|} \sum_{t=1}^{|o_i|} \left[M_k \min \left(r_t(\theta) \hat{A}_{i,t}, \text{clip}(r_t(\theta), 1-\epsilon, 1+\epsilon) \hat{A}_{i,t} \right) - \beta D_{\text{KL}}(\pi_\theta \| \pi_{\text{ref}}) \right], \quad (7)$$

where $r_t(\theta)$ denotes the policy ratio at time step t, D_{KL} penalizes the divergence from the reference policy π_{ref} and $Z = \sum_{k=1}^{K} M_k$ is the normalization factor. This objective ensures that optimization is exclusively directed at control tokens, with high-quality structural and semantic signals driving stable and efficient policy learning.

3 Experiment

3.1 Experimental Setup

Training datasets. We construct our training set using approximately 5,000 question–answer pairs selected from the DeepScaleR Preview-Dataset (Luo et al., 2025b). This dataset is a challenging collection of mathematics problems, covering a wide range of difficulty levels and drawing from sources such as AIME (1984–2023), AMC (prior to 2023), and the MATH training set. Consequently, the resulting training set has no overlap with our benchmark dataset.

Evaluation Datasets. We conduct comprehensive evaluations of our method on several widely recognized benchmarks for mathematical reasoning. Specifically, we assess performance on the test sets of GSM8K (Cobbe et al., 2021), MATH-500 (Patel et al., 2021), AMC2023, as well as AIME2024 and AIME2025 (MAA Committees, 2024). These benchmarks cover a diverse range of problem types and difficulty levels, providing a rigorous testbed for evaluating mathematical reasoning capabilities. Additionally, we evaluate our models on MMLU-Pro (Wang et al., 2024) to assess their generalization beyond the mathematical domain, selecting 100 questions from each of three different fields.

Base Models. We adopt the open-source DeepSeek-R1-Distill model series (Guo et al.,

Method	GSM8K		MATH-500		AMC 2023		AIME 2024		AIME 2025		Overall	
	Acc↑	Tokens↓	Acc↑	Tokens↓	Acc↑	Tokens↓	Acc↑	Tokens↓	Acc↑	Tokens↓	Acc↑	Tokens↓
DeepSeek-R1-Distill-Qwen-1.5B												
Original	86.1	2,245	83.0	3,978	67.7	7,160	29.3	13,832	26.9	14,680	58.6	8,379
O1-Pruner	85.1	1,535	82.3	2,446	69.5	5,622	27.5	12,155	24.1	12,701	57.7	6,892
No Wait	84.9	1,955	81.6	2,894	68.5	6,422	26.1	9,167	20.3	11,601	56.3	6,408
DAST	85.6	1,783	83.4	3,155	70.2	5,583	29.5	10,042	20.6	10,647	57.9	6,242
FCS+Ref.	87.7	1,397	82.9	2,883	72.1	4,449	29.7	9,898	28.6	11,624	60.2	6,050
LCPO	83.5	1,890	80.5	2,684	63.9	6,694	23.5	9,692	24.7	10,075	55.2	6,207
DEER	86.2	1,205	84.1	2,398	70.2	4,179	26.6	9,732	20.2	9,890	57.5	5,481
MERA(Ours)	**89.3**	**1,108**	**86.0**	**2,239**	**74.5**	**3,180**	**31.2**	**8,425**	**30.9**	**9,357**	**62.4**	**4,862**
DeepSeek-R1-Distill-Qwen-7B												
Original	90.2	1,819	86.9	3,422	77.4	6,738	53.1	12,185	48.2	13,276	71.2	7,488
O1-Pruner	92.5	1,052	89.0	2,678	82.8	7,501	52.2	9,412	49.4	10,973	73.2	6,323
No Wait	89.8	1,733	87.2	2,579	75.7	5,478	42.5	10,048	35.2	11,827	66.1	6,333
DAST	92.7	1,558	88.6	2,876	80.3	4,601	52.6	10,240	49.5	9,721	72.7	5,799
FCS+Ref.	92.9	1,218	87.8	2,909	81.5	5,143	55.1	9,212	49.8	9,844	73.4	5,665
LCPO	88.0	1,488	84.7	2,539	73.6	4,821	45.3	9,408	40.6	9,904	66.4	5,632
DEER	89.0	1,082	88.9	1,908	82.2	5,194	46.4	9,932	39.6	9,302	69.2	5,484
MERA(Ours)	**93.7**	**822**	**91.0**	**1,739**	**85.7**	**3,711**	**56.1**	**8,398**	**53.6**	**8,732**	**76.0**	**4,680**
DeepSeek-R1-Distill-Qwen-14B												
Original	92.3	1,917	87.4	2,832	83.4	7,925	60.6	11,155	56.4	12,752	76.0	7,316
O1-Pruner	91.9	1,263	90.2	2,015	84.2	7,381	55.2	9,003	53.1	9,356	74.9	5,804
No Wait	92.6	1,152	88.7	1,931	79.6	5,709	53.8	9,692	45.3	8,280	72.0	5,353
DAST	94.8	1,778	88.9	1,943	84.9	4,338	55.6	7,768	56.2	10,298	76.1	5,225
FCS+Ref.	94.7	1,141	89.3	1,937	83.0	5,281	56.4	7,749	53.7	8,309	75.4	4,883
LCPO	91.2	1,645	86.8	2,426	78.1	5,533	56.1	9,460	50.2	10,405	72.5	5,894
DEER	93.5	975	87.7	1,729	77.2	3,291	60.7	8,388	47.3	9,723	73.3	4,821
MERA(Ours)	**96.1**	**835**	**92.5**	**1,359**	**88.0**	**3,109**	**63.3**	**6,695**	**59.2**	**7,320**	**79.8**	**3,864**

Table 1: Performance comparison of MERA and baseline methods on five mathematical reasoning benchmarks.

2025a), including DeepSeek-R1-Distill-Qwen-1.5B, DeepSeek-R1-Distill-Qwen-7B, and DeepSeek-R1-Distill-Qwen-14B, as our base models. These models are obtained through supervised fine-tuning on reasoning data generated by the DeepSeek-R1 model.

Baselines. The baselines include in our comparison fall into three distinct categories:(1) Methods that directly reduce reasoning length, including O1-Pruner (Luo et al., 2025a) and No Wait (Wang et al., 2025) ; (2) Methods that rely on preset computational budgets before inference, including DAST (Shen et al., 2025), FCS+Ref. (Chen et al., 2024) and LCPO (Aggarwal and Welleck, 2025); (3) Methods that dynamically determine termination, including DEER (Yang et al., 2025a).

Evaluation Metrics. We evaluate the proposed method using two key metrics: Accuracy(ACC) and Generation Length(Tokens). ACC is calculated as $\text{ACC} = \frac{1}{N}\sum_{i=1}^{N}\mathbb{I}\{\mathcal{M}(\mathcal{LLM}(x_i)) = y_i\}$, where x_i is the input question, y_i is the ground-truth answer, $\mathcal{LLM}(\cdot)$ denotes the model's output, $\mathcal{M}(\cdot)$ extracts the predicted answer according to a predefined format (e.g., starting with "The answer is…"). Tokens measures the average generated tokens, computed as $\text{Tokens} = \frac{1}{N}\sum_{i=1}^{N}|\mathcal{LLM}(x_i)|$, where $|\cdot|$ counts the tokens of generated words.

Implementation Details. All experiments are conducted using the `vLLM` framework under a zero-shot chain-of-thought (CoT) setting. The prompt used is: *"Please reason step by step and put the final answer in \boxed{}."* To ensure reliability and statistical validity, each model and configuration is evaluated across five sampling runs. To ensure complete reasoning traces are captured, the maximum number of new tokens is set to 16384 for AIME and 8192 for all other datasets.

3.2 Experimental Results

3.2.1 Main Results.

Table 1 presents the performance of MERA across five reasoning benchmarks, evaluated by answer accuracy (Acc) and generation length (Tokens). Compared to existing methods, MERA achieves higher

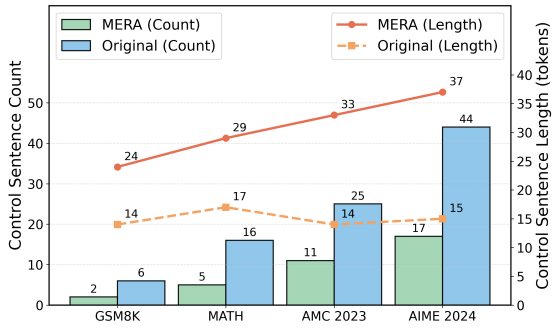

Figure 3: Analysis of control sentence before and after the application of MERA on DeepSeek-R1-Distill-Qwen-7B.

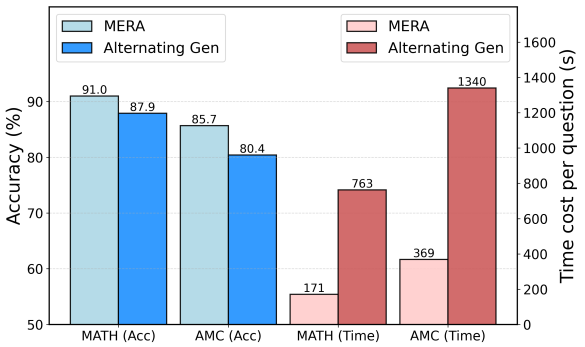

Figure 4: Comparison between MERA and the alternating generation setting in terms of accuracy and runtime using DeepSeek-R1-Distill-Qwen-7B.

Method	Law ACC	Law LEN	Engineering ACC	Engineering LEN	Physics ACC	Physics LEN
Original	23.5	1528	26.4	5412	34.1	4978
O1-Pruner	22.8	1054	25.6	4981	35.6	4587
DAST	21.4	1249	27.1	5821	33.5	5243
FCS+Ref.	24.1	973	27.9	5518	34.9	5098
MERA (Ours)	**25.7**	**814**	**28.5**	**4092**	**36.9**	**3854**

Table 2: Evaluation on MMLU-Pro to validate generalization using DeepSeek-R1-Distill-Qwen-7B.

accuracy while significantly reducing reasoning length across all datasets. For instance, on the DeepSeek-R1-Distill-Qwen-1.5B model, MERA reduces the average length from 8,379 tokens to 4,583, while improving accuracy from 58.6% to 62.5%. In contrast, MERA not only reduces reasoning length but also improves accuracy, achieving the best efficiency overall. These improvements are attributed to MERA's explicit enablement of self-regulation during reasoning, which allows the model to dynamically perceive its current cognitive state and perform fine-grained, stage-wise control over the reasoning trajectory. These results demonstrate that structured meta-cognition effectively mitigates overthinking and enhances the model's efficiency in handling problems of varying complexity.

3.2.2 Analysis of Control Behavior.

To further examine the behavioral changes induced by MERA's structured meta-cognitive design, we analyze control statement usage during reasoning, focusing on two aspects: the total number of control statements and the average length of each control statement. As shown in Figure 3, MERA significantly reduces the frequency of control statements across all evaluated datasets. For instance, on AIME 2024, the original model produces an average of 44 control sentences, whereas MERA reduces this number to just 17. This substantial decline indicates that, after explicit decoupling and independent reinforcement, the model learns to suppress unnecessary or ineffective cognitive behaviors, thereby promoting more concise and coherent reasoning trajectories. In contrast, the average length of individual control statements increases notably under MERA, rising from 15 tokens to 37 tokens on AIME 2024. This trend becomes more pronounced as task difficulty increases. This suggests that the optimized control statements produced by the model tend to be more informative and carry stronger regulatory intent. Overall, these results demonstrate that MERA not only reduces redundant control interventions but also enhances the quality and functional value of retained control, enabling more deliberate and efficient self-regulation.

3.2.3 Generalization to different domains.

As shown in Table 2, MERA achieves the highest reasoning efficiency across all three domains in the MMLU-Pro benchmark: Law, Engineering, and Physics, while maintaining accuracy on par with baseline methods. Specifically, MERA reduces the average token length by a large margin, such as from 5,412 to 4,092 in the Engineering, without sacrificing answer correctness. This indicates that the proposed meta-cognitive framework not only improves reasoning within mathematical tasks but also generalizes effectively to broader open-domain contexts. The consistent length reduction suggests the model can monitor its internal process and reduce redundancy, even on non-mathematical tasks.

Method	GSM8K		MATH-500		AMC 2023		AIME 2024		AIME 2025		Overall	
	Acc↑	Tokens↓	Acc↑	Tokens↓	Acc↑	Tokens↓	Acc↑	Tokens↓	Acc↑	Tokens↓	Acc↑	Tokens↓
DeepSeek-R1-Distill-Qwen-7B												
Original	90.2	1,819	86.9	3,422	77.4	6,738	53.1	12,185	48.2	13,276	71.2	7,488
+SFT	91.2	1,575	88.1	2,340	80.5	4,881	53.6	9,171	52.7	9,934	73.2	5,580
+SFT+GRPO	91.9	1,785	89.5	2,614	83.7	5,193	55.4	9,740	52.9	11,059	74.7	6,078
+total MERA	**93.7**	**822**	**91.0**	**1,739**	**85.7**	**3,711**	**56.1**	**8,398**	**53.6**	**8,732**	**76.0**	**4,680**

Table 3: Ablation studies comparing different training strategies using DeepSeek-R1-Distill-Qwen-7B.

3.2.4 Comparison Between Internalized and Alternating Control Generation.

To evaluate the runtime efficiency and effectiveness of MERA's internalized control mechanism, we compare it against a dual-model alternating generation setting, where a reasoning model and a separate auxiliary model jointly generate reasoning and control segments in turn. This setup mirrors the process used during data construction, in which control decisions are externally injected by a helper model. As shown in Figure 4, the alternating generation strategy results in significantly higher time cost per question, reaching 763 seconds on MATH and 1,340 seconds on AMC 2023, compared to MERA's 171 and 369 seconds respectively. Moreover, MERA also achieves higher accuracy, outperforming the alternating method by 3.1% on MATH and 5.3% on AMC 2023. These results highlight the limitations of external control reliance at inference time. While such methods may provide high-quality control during data annotation, they introduce substantial latency and fail to match the coordination quality of a model with internalized control. In contrast, MERA achieves better efficiency and accuracy by integrating control and reasoning within a unified architecture, enabling coherent and cost-effective decisions.

3.2.5 Ablation Study

Table 3 presents the ablation study results. The results indicate that incorporating supervised SFT leads to a modest improvement in accuracy across all benchmark tasks, while also reducing token usage. When SFT is combined with standard GRPO, a slight increase in accuracy is observed, but at the cost of increased token generation. In contrast, the most substantial performance gains are achieved when the full MERA is applied. It not only yields higher accuracy but also enables more efficient reasoning. This shows that CSPO outperforms standard GRPO by optimizing control behavior.

4 Related Work

To mitigate overthinking in Large Reasoning Models (LRMs), prior work mainly follows three strategies. First, some methods directly compress reasoning trajectories: O1-Pruner (Luo et al., 2025a) trims LLM-generated traces for supervised distillation, and No Wait (Wang et al., 2025) suppresses hesitation tokens to shorten generation. While effective, such compression is often task-insensitive and can over-truncate. Second, budget-based controls constrain inference with predefined limits; for example, LCPO (Aggarwal and Welleck, 2025) sets a static token-length budget before decoding, but this coarse constraint cannot react to the model's evolving reasoning state. Third, dynamic early-exit methods stop based on intermediate signals; DEER (Yang et al., 2025a) uses the confidence of partial answers to trigger termination, yet still relies on external evaluation rather than intrinsic self-monitoring.

In contrast, our proposed MERA explicitly separates reasoning and control within a structured framework and optimizes them independently. MERA enables the model to autonomously decide whether to continue, revise, or terminate the reasoning process. This intrinsic regulatory capability allows for more adaptive and efficient reasoning by reducing unnecessary steps without relying on external heuristics or predefined constraints. Additional related work is provided in the appendix D.

5 Conclusion

We propose MERA, a meta-cognitive framework that equips LRMs with structured control to mitigate overthinking. MERA separates reasoning from control, constructs high-quality control supervision via a takeover mechanism, and improves adaptive regulation with CSPO. Experiments show that MERA reduces redundant reasoning while improving accuracy across diverse benchmarks.

Limitations

MERA delivers consistent gains in both efficiency and accuracy, but two aspects warrant further improvement. First, support for user-side preference personalization remains limited, as different application contexts and users may favor different trade-offs in reasoning depth, verification intensity, and early-stopping aggressiveness; providing explicit preference controls would better tailor the control policy to these needs. Second, control-decision interpretability still has room for improvement: while the model can provide some rationale for actions such as stopping, continuing, or backtracking, scenarios that require higher transparency would benefit from making these rationales more fine-grained and auditable, and more explicitly tied to the key evidence and judgments in the reasoning process.

Acknowledgements

This work is supported by the National Key Research and Development Program of China (2024YFF0907401), the National Natural Science-Foundation of China (62072052) and Beijing Natural Science Foundation (L251037).

References

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

Edward Y. Chang, Yuxuan Tong, Morry Niu, Graham Neubig, and Xiang Yue. 2025. Demystifying long chain-of-thought reasoning in llms. *CoRR*, abs/2502.03373.

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

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *Preprint*, arXiv:2110.14168.

Mehul Damani, Idan Shenfeld, Andi Peng, Andreea Bobu, and Jacob Andreas. 2025. Learning how hard to think: Input-adaptive allocation of LM computation. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net.

Shanna Erickson and Evan Heit. 2013. Math and metacognition: Resolving the paradox. In *Proceedings of the 35th Annual Meeting of the Cognitive Science Society, CogSci 2013, Berlin, Germany, July 31 - August 3, 2013*. cognitivesciencesociety.org.

Gongfan Fang, Xinyin Ma, and Xinchao Wang. 2025. Thinkless: LLM learns when to think. *CoRR*, abs/2505.13379.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and et al. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

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

Yiran Guo, Lijie Xu, Jie Liu, Dan Ye, and Shuang Qiu. 2025b. Segment policy optimization: Effective segment-level credit assignment in rl for large language models. *Preprint*, arXiv:2505.23564.

Lingjie Jiang, Xun Wu, Shaohan Huang, Qingxiu Dong, Zewen Chi, Li Dong, Xingxing Zhang, Tengchao Lv, Lei Cui, and Furu Wei. 2025. Think only when you need with large hybrid-reasoning models. *CoRR*, abs/2505.14631.

Chaozhuo Li, Pengbo Wang, Chenxu Wang, Litian Zhang, Zheng Liu, Qiwei Ye, Yuanbo Xu, Feiran Huang, Xi Zhang, and Philip S Yu. 2025a. Loki's dance of illusions: A comprehensive survey of hallucination in large language models. *arXiv preprint arXiv:2507.02870*.

Zhong-Zhi Li, Duzhen Zhang, Ming-Liang Zhang, Jiaxin Zhang, Zengyan Liu, Yuxuan Yao, Haotian Xu, Junhao Zheng, Pei-Jie Wang, Xiuyi Chen, Yingying Zhang, Fei Yin, Jiahua Dong, Zhijiang Guo, Le Song, and Cheng-Lin Liu. 2025b. From system 1 to system 2: A survey of reasoning large language models. *CoRR*, abs/2502.17419.

Songyang Liu, Chaozhuo Li, Jiameng Qiu, Xi Zhang, Feiran Huang, Litian Zhang, Yiming Hei, and Philip S Yu. 2025. The scales of justitia: A comprehensive survey on safety evaluation of llms. *arXiv preprint arXiv:2506.11094*.

Songyang Liu, Chaozhuo Li, Chenxu Wang, Jinyu Hou, Zejian Chen, Litian Zhang, Zheng Liu, Qiwei Ye, Yiming Hei, Xi Zhang, et al. 2026. Clawkeeper: Comprehensive safety protection for openclaw agents through skills, plugins, and watchers. *arXiv preprint arXiv:2603.24414*.

Chenwei Lou, Zewei Sun, Xinnian Liang, Meng Qu, Wei Shen, Wenqi Wang, Yuntao Li, Qingping Yang, and Shuangzhi Wu. 2025. Adacot: Pareto-optimal adaptive chain-of-thought triggering via reinforcement learning. *arXiv preprint arXiv:2505.11896*.

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

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

Wenjie Ma, Jingxuan He, Charlie Snell, Tyler Griggs, Sewon Min, and Matei Zaharia. 2025. Reasoning models can be effective without thinking. *CoRR*, abs/2504.09858.

MAA Committees. 2024. Aime problems and solutions. `https://artofproblemsolving.com/wiki/index.php/AIME_Problems_and_Solutions`.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, and et al. Lama Ahmad. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Arkil Patel, Satwik Bhattamishra, and Navin Goyal. 2021. Are nlp models really able to solve simple math word problems? *Preprint*, arXiv:2103.07191.

Chen Qian, Dongrui Liu, Haochen Wen, Zhen Bai, Yong Liu, and Jing Shao. 2025. Demystifying reasoning dynamics with mutual information: Thinking tokens are information peaks in LLM reasoning. *CoRR*, abs/2506.02867.

Ziqing Qiao, Yongheng Deng, Jiali Zeng, Dong Wang, Lai Wei, Fandong Meng, Jie Zhou, Ju Ren, and Yaoxue Zhang. 2025. Concise: Confidence-guided compression in step-by-step efficient reasoning. *arXiv preprint arXiv:2505.04881*.

Yuxiao Qu, Matthew YR Yang, Amrith Setlur, Lewis Tunstall, Edward Emanuel Beeching, Ruslan Salakhutdinov, and Aviral Kumar. 2025. Optimizing test-time compute via meta reinforcement fine-tuning. *arXiv preprint arXiv:2503.07572*.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *Preprint*, arXiv:2402.03300.

Yi Shen, Jian Zhang, Jieyun Huang, Shuming Shi, Wenjing Zhang, Jiangze Yan, Ning Wang, Kai Wang, and Shiguo Lian. 2025. Dast: Difficulty-adaptive slow-thinking for large reasoning models. *arXiv preprint arXiv:2503.04472*.

Katherine Wagner. 2017. Biological and artificial perspectives on metacognition. In *Proceedings of the 39th Annual Meeting of the Cognitive Science Society, CogSci 2017, London, UK, 16-29 July 2017*. cognitivesciencesociety.org.

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

Chenxu Wang, Chaozhuo Li, Songyang Liu, Zejian Chen, Jinyu Hou, Ji Qi, Rui Li, Litian Zhang, Qiwei Ye, Zheng Liu, et al. 2026. The devil behind moltbook: Anthropic safety is always vanishing in self-evolving ai societies. *arXiv preprint arXiv:2602.09877*.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, Tianle Li, Max Ku, Kai Wang, Alex Zhuang, Rongqi Fan, Xiang Yue, and Wenhu Chen. 2024. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. *Preprint*, arXiv:2406.01574.

Xingyu Wu, Yuchen Yan, Shangke Lyu, Linjuan Wu, Yiwen Qiu, Yongliang Shen, Weiming Lu, Jian Shao, Jun Xiao, and Yueting Zhuang. 2025. Lapo: Internalizing reasoning efficiency via length-adaptive policy optimization. *arXiv preprint arXiv:2507.15758*.

Haotian Xu, Xing Wu, Weinong Wang, Zhongzhi Li, Da Zheng, Boyuan Chen, Yi Hu, Shijia Kang, Jiaming Ji, Yingying Zhang, Zhijiang Guo, Yaodong Yang, Muhan Zhang, and Debing Zhang. 2025. Redstar: Does scaling long-cot data unlock better slow-reasoning systems? *CoRR*, abs/2501.11284.

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

Wang Yang, Xiang Yue, Vipin Chaudhary, and Xiaotian Han. 2025b. Speculative thinking: Enhancing small-model reasoning with large model guidance at inference time. *Preprint*, arXiv:2504.12329.

Jiajie Zhang, Nianyi Lin, Lei Hou, Ling Feng, and Juanzi Li. 2025a. Adaptthink: Reasoning models can learn when to think. *arXiv preprint arXiv:2505.13417*.

Jinghan Zhang, Xiting Wang, Fengran Mo, Yeyang Zhou, Wanfu Gao, and Kunpeng Liu. 2025b. Entropy-based exploration conduction for multi-step reasoning. In *Findings of the Association for Computational Linguistics, ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pages 3895–3906. Association for Computational Linguistics.

Jianan Zhao, Meng Qu, Chaozhuo Li, Hao Yan, Qian Liu, Rui Li, Xing Xie, and Jian Tang. 2022. Learning on large-scale text-attributed graphs via variational inference. *arXiv preprint arXiv:2210.14709*.

A Evaluation Dataset Descriptions

We conduct experiments on a range of publicly available benchmark datasets. These datasets span various difficulty levels—from elementary arithmetic to Olympiad-style problems—and include cross-domain general knowledge evaluations. Detailed descriptions of each dataset are as follows:

GSM8K (Cobbe et al., 2021) consists of approximately 8,500 high-quality elementary school math word problems designed to evaluate step-by-step numerical reasoning. Each problem typically involves multiple arithmetic operations and emphasizes accuracy, decomposition, and logical consistency.

MATH-500 (Patel et al., 2021) is a curated subset of 500 challenging problems sampled from the full MATH dataset. It covers a wide range of topics in high school and early college mathematics, including algebra, combinatorics, geometry, and number theory. MATH-500 is widely used to assess formal mathematical reasoning, requiring symbolic manipulation, inductive strategies, and structured derivations. It is particularly suited to evaluating the rigor and reliability of multi-hop mathematical.

AMC2023 refers to the complete set of 40 problems from the 2023 American Mathematics Competition (AMC). The problems range from moderate to high difficulty and test creative decomposition, numerical estimation, and structural understanding. All problems are converted to a free-form generation format to align with our evaluation framework.

AIME. (MAA Committees, 2024) We collect problems from the 2024 and 2025 editions of the AIME (American Invitational Mathematics Examination), forming two small but highly challenging evaluation sets. Compared to AMC, AIME problems exhibit higher complexity and frequently involve algebraic constructions, advanced factorization techniques, and recursive formula analysis.

MMLU-Pro. (Wang et al., 2024) To evaluate generalization beyond mathematical domains, we use a subset of MMLU-Pro as our cross-domain benchmark. We select 300 problems in total—100 each from the domains of Physics, Law, and Engineering. This dataset evaluates the model's ability to transfer structured reasoning and control skills to non-mathematical contexts, testing the robustness and broad applicability of our method.

B Algorithm

Algorithm 1 outlines the Control Segment Policy Optimization (CSPO) procedure. The model first samples multiple reasoning–control trajectories and partitions them into segments. Each segment receives a control-specific reward based on semantic alignment and structural correctness. A masking mechanism ensures that only control-relevant tokens influence gradient updates. Finally, the policy is optimized using masked advantage-weighted updates, enabling more precise and stable learning of meta-cognitive control.

Algorithm 1 Control Segment Policy Optimization

Require: Input-query x, Current policy π_θ, Old policy $\pi_{\theta_{\text{old}}}$
Ensure: Updated policy with improved control precision
Step 1: Segment Sampling
Sample G complete outputs $\{o_1, o_2, \ldots, o_G\}$ from $\pi_{\theta_{\text{old}}}$
Partition each output into reasoning–control segments $\{\tau_1, \ldots, \tau_K\}$
Step 2: Segment-wise Reward Estimation
for each segment o_i **do**
    Compute control reward $r(o_i)$ based on semantic and format signals
    Normalize across samples to obtain $\hat{A}_{i,t}$
end for
Step 3: Control Masking
Identify control-relevant spans $\{c_1, \ldots, c_K\}$ and construct mask M_k
Step 4: Objective Update
for each control token t in o_i **do**
    if $M_k = 1$ **then**
        Accumulate policy into $\mathcal{J}_{\text{CSPO}}(\theta)$
    end if
end for
Step 5: Policy Optimization
Update parameters via gradient descent on $\mathcal{J}_{\text{CSPO}}(\theta)$

C Preliminary: Group Relative Policy Optimization (GRPO)

Group Relative Policy Optimization (GRPO) (Guo et al., 2025a) is a reinforcement learning algorithm that estimates token-level advantages by comparing sampled sequences within the same input group. Unlike traditional methods such as PPO, GRPO avoids the use of a separate value function and

instead derives advantage estimates based solely on normalized group-level rewards.

Given an input x, the reference policy $\pi_{\theta_{\text{old}}}$ generates a set of G candidate outputs $\{y^{(i)}\}_{i=1}^{G}$. For each response $y^{(i)}$, we compute a final reward $\mathcal{R}(x, y^{(i)})$, and define the group-normalized advantage $\hat{A}_i$ as:

$$\hat{A}_i = \frac{\mathcal{R}(x, y^{(i)}) - \text{mean}(\{\mathcal{R}(x, y^{(j)})\}_{j=1}^{G})}{\text{std}(\{\mathcal{R}(x, y^{(j)})\}_{j=1}^{G})}. \quad (8)$$

This scalar advantage $\hat{A}_i$ is uniformly assigned to each token in trajectory $y^{(i)}$, such that $\hat{A}_{i,t} = \hat{A}_i$ for all $t \in \{1, \ldots, |y^{(i)}|\}$.

The overall GRPO training objective incorporates a clipped policy loss and a KL regularization term with respect to a reference policy π_{ref}, and is formulated as:

$$\mathcal{J}_{\text{GRPO}}(\theta) = \mathbb{E}_{x\sim\mathcal{D},\ \{y^{(i)}\}_{i=1}^{G}\sim\pi_\theta(\cdot|x)} \Bigg\{ \frac{1}{G}\sum_{i=1}^{G} \frac{1}{|y^{(i)}|} \sum_{t=1}^{|y^{(i)}|} \Big[\min\Big(r_{i,t}(\theta)\,\hat{A}_{i,t},\ \text{clip}(r_{i,t}(\theta), 1-\epsilon,\ 1+\epsilon)\,\hat{A}_{i,t}\Big) - \beta\, D_{\text{KL}}(\pi_\theta \,\|\, \pi_{\text{ref}}) \Big] \Bigg\}, \quad (9)$$

where the policy ratio $r_{i,t}(\theta)$ is given by:

$$r_{i,t}(\theta) = \frac{\pi_\theta(a_{i,t} \mid s_{i,t})}{\pi_{\theta_{\text{old}}}(a_{i,t} \mid s_{i,t})}. \quad (10)$$

This token-level formulation supports trajectory-wise normalization while preserving fine-grained optimization, making GRPO well-suited for open-ended sequence generation scenarios such as mathematical reasoning and control-sensitive tasks.

D Extended Discussion on Related Work

Mitigating Overthinking in Large Reasoning Models (LRMs). In complex multi-step tasks such as mathematical reasoning, Large Reasoning Models (LRMs) often exhibit excessively long, repetitive, or inefficient reasoning trajectories—a phenomenon referred to as "overthinking." Prior efforts to address this issue can be categorized into three major paradigms: (1) Direct trajectory shortening, (2) Budget-aware inference control, and (3) Output-based dynamic early termination.

Direct Trajectory Shortening

This class of methods seeks to compress existing reasoning trajectories by removing redundant steps while preserving correctness. For instance, O1-Pruner (Luo et al., 2025a) selects supervision signals from LLM-generated reasoning trajectories using a length-harmonized score that balances correctness and brevity. No Wait (Wang et al., 2025) reduces the probability of generating delay-inducing tokens such as "wait," while No Think (Ma et al., 2025) directly instructs the model to skip intermediate reasoning and provide an answer immediately. Although effective at reducing output length, these approaches often lack adaptivity to problem complexity. Their uniform truncation strategies tend to harm performance on harder, more demanding problems.

Budget-Aware Inference Control

This line of work imposes length constraints or computational budgets before inference begins. For example, DAST (Shen et al., 2025) applies fixed token-length budgets during preference learning, encouraging shorter outputs for correct responses and longer ones for incorrect cases. FCS+Ref (Chen et al., 2024) constructs preference pairs based on identifying the first correct solution and reflection point within the reasoning trace. LCPO (Aggarwal and Welleck, 2025) adopts a straightforward reinforcement learning framework that jointly optimizes accuracy and user-specified length bounds. Moreover, Ada-Bok (Damani et al., 2025), Thinkless (Fang et al., 2025), HGPO (Jiang et al., 2025) explore coarse-grained "fast–slow thinking" modes by dynamically deciding whether to engage in full CoT reasoning based on problem difficulty. A common limitation of these methods is that their control signals are predefined and fixed prior to generation, lacking responsiveness to the model's evolving internal reasoning state.

Dynamic Early Exit

The third category monitors intermediate outputs during inference to dynamically determine when to terminate reasoning. DEER (Yang et al., 2025a) is a representative method in this direction, triggering early exits based on confidence scores of partial answers to improve efficiency while maintaining accuracy. Other methods leverage uncertainty signals such as entropy. For example, Entropy-Based Adaptive Think (Zhang et al., 2025b) stops the generation process once marginal information gain drops below a threshold, avoiding inefficient stagnation. While these approaches introduce elements of self-regulation, they primarily rely on external

scorers or auxiliary prediction heads and lack true meta-cognitive control within the model itself.

In contrast to the above paradigms, our proposed MERA framework explicitly separates reasoning and control within a structured modeling framework. MERA endows the model with an intrinsic self-regulatory mechanism that enables it to determine whether to continue, revise, or terminate its reasoning trajectory. Instead of relying on compression, budget, or external signals, MERA monitors internal reasoning states and inserts explicit `<control>` directives. These serve as fine-grained, adaptive interventions, enabling meta-cognitive regulation. Our training pipeline combines supervised fine-tuning with segment-level policy optimization (CSPO), allowing the model to develop both expressive reasoning capabilities and dynamic self-control.