Reconsidering Overthinking: Penalizing Internal and External Redundancy in CoT Reasoning

重新审视过度思考: 在链式思维推理中惩罚内部和外部冗余

Jialiang Hong [1*] , Taihang Zhen [2*] , Kai Chen [2] , Jiaheng Liu [2] , Wenpeng Zhu [1†] , Jing Huo [2†] , Yang Gao[2†] , Depeng Wang [1] , Haitao Wan [1] , Xi Yang [1] , Boyan Wang [2] , Fanyu Meng [3]

洪嘉良 [1*] , 甄太行 [2*] , 陈凯 [2] , 刘嘉恒 [2] , 朱文鹏 [1†] , 霍晶 [2†] , 杨 Gao[2†] , 王德鹏 [1] , 万海涛 [1] , 杨曦 [1] , 王博岩 [2] , 孟凡宇 [3]

[1] China Mobile (Suzhou) Software Technology Co., Ltd.

[1] 中国移动 (苏州) 软件技术有限公司

[2] State Key Laboratory for Novel Software Technology, Nanjing University

[2] 南京大学新型软件技术国家重点实验室

[3] China Mobile Research Institute

[3] 中国移动研究院

Abstract

摘要

Large Reasoning Models (LRMs) often produce excessively verbose reasoning traces, a phenomenon known as over-thinking, which hampers both efficiency and interpretability. Prior works primarily address this issue by reducing response length, without fully examining the underlying semantic structure of the reasoning process. In this paper, we revisit overthinking by decomposing it into two distinct forms: internal redundancy, which consists of low-contribution reasoning steps within the first correct solution (FCS), and external redundancy, which refers to unnecessary continuation after the FCS. To mitigate both forms, we propose a dual-penalty reinforcement learning framework. For internal redundancy, we adopt a sliding-window semantic analysis to penalize low-gain reasoning steps that contribute little toward reaching the correct answer. For external redundancy, we penalize its proportion beyond the FCS to encourage earlier termination. Our method significantly compresses reasoning traces with minimal accuracy loss, and generalizes effectively to out-of-domain tasks such as question answering and code generation. Crucially, we find that external redundancy can be safely removed without degrading performance, whereas internal redundancy must be reduced more cautiously to avoid impairing correctness. These findings suggest that our method not only improves reasoning efficiency but also enables implicit, semantic-aware control over Chain-of-Thought length, paving the way for more concise and interpretable LRMs.

大型推理模型 (Large Reasoning Models, LRMs) 常常产生过于冗长的推理轨迹，这种现象被称为过度思考，影响了效率和可解释性。以往工作主要通过缩短响应长度来解决该问题，但未充分探讨推理过程的语义结构。本文重新审视过度思考，将其分解为两种不同形式: 内部冗余，指首次正确解 (First Correct Solution, FCS) 内贡献较低的推理步骤；外部冗余，指 FCS 之后的不必要延续。为缓解这两种冗余，我们提出了双重惩罚的强化学习框架。针对内部冗余，采用滑动窗口语义分析惩罚对正确答案贡献较小的推理步骤；针对外部冗余，惩罚 FCS 之后的比例以鼓励更早终止。该方法显著压缩推理轨迹且准确率损失极小，并能有效泛化至问答和代码生成等领域外任务。关键发现是外部冗余可安全去除而不影响性能，而内部冗余需谨慎减少以避免正确性受损。研究表明，我们的方法不仅提升了推理效率，还实现了对链式思维长度的隐式语义感知控制，为更简洁且可解释的 LRMs 开辟了新路径。

Code - https://github.com/HenryZhen97/Reconsidering-Overthinking

代码 - https://github.com/HenryZhen97/Reconsidering-Overthinking

Introduction

引言

Large reasoning models (LRMs), such as OpenAI's ol (Jaech et al. 2024), DeepSeek-R1 (Guo et al. 2025), and QwQ (Team 2025), demonstrate strong performance on complex reasoning tasks. A key factor behind their success is the generation of dense Chain-of-Thought (CoT) sequences, which decompose complex problems into stepwise reasoning that guides models toward correct answers. However, such lengthy reasoning chains often contain substantial redundancy, which can hinder the overall reasoning efficiency and reduce the readability of LRMs' outputs (Chen et al. 2024; Sui et al. 2025).

大型推理模型 (LRMs)，如 OpenAI 的 ol(Jaech 等，2024)、DeepSeek-R1(Guo 等，2025) 和 QwQ(团队，2025)，在复杂推理任务中表现出色。其成功的关键因素之一是生成密集的链式思维 (Chain-of-Thought, CoT) 序列，将复杂问题分解为逐步推理，引导模型得出正确答案。然而，这些冗长的推理链常包含大量冗余，影响整体推理效率并降低 LRMs 输出的可读性 (Chen 等，2024；Sui 等，2025)。

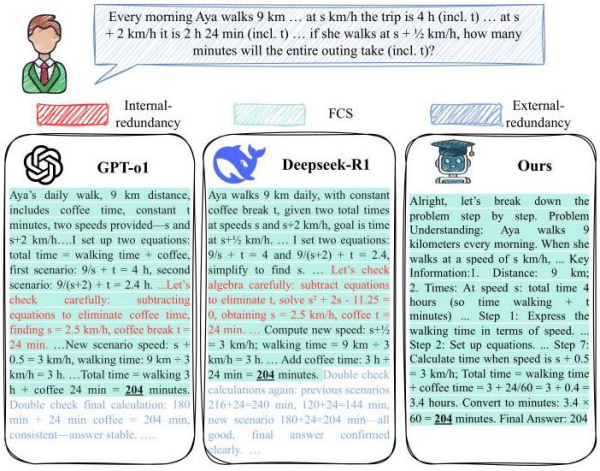

Figure 1: Response examples in AIME24 from o1, R1, and our method. The underlined number ”204” marks the first correct answer, which splits the reasoning into two parts: the preceding span is the First Correct Solution (FCS), which may contain internal redundancy; the continuation beyond it constitutes external redundancy.

图 1: 来自 o1、R1 及我们方法的 AIME24 响应示例。下划线标记的数字“204”为首次正确答案，将推理分为两部分: 前半部分为首次正确解 (FCS)，可能包含内部冗余；其后的延续构成外部冗余。

Recent efforts (Liu et al. 2025a; Sheng et al. 2025; Wang et al. 2025a, b) investigate overthinking in LRMs, aiming to uncover its root causes and develop targeted compression techniques for CoT outputs. Among them, a line of work grounded in reinforcement learning (RL) further explores how to suppress redundant reasoning by constraining token budgets (Luo et al. 2025a; Arora and Zanette 2025). However, these methods mainly target the symptom of redundancy by limiting the length of reasoning traces, without addressing its underlying semantic causes. In contrast, our work analyzes the semantic nature of redundancy and decomposes it into two distinct types for more precise detection and suppression.

近期研究 (Liu 等，2025a；Sheng 等，2025；Wang 等，2025a, b) 探讨 LRMs 中的过度思考，旨在揭示其根源并开发针对性压缩链式思维输出的技术。其中，一些基于强化学习 (RL) 的工作进一步研究如何通过限制 token 预算抑制冗余推理 (Luo 等，2025a；Arora 和 Zanette，2025)。然而，这些方法主要针对冗余的表象，通过限制推理轨迹长度来应对，而未解决其语义根因。相比之下，我们的工作分析了冗余的语义本质，并将其分解为两种不同类型，以实现更精准的检测和抑制。

The first correct solution (FCS) (Chen et al. 2024) refers to the earliest complete reasoning trace that leads to a correct answer. Theoretically, there exists a minimal subset within the FCS that is sufficient to solve the task. Based on this assumption, any content beyond this minimal subset, either superfluous steps within the FCS or additional content generated afterward, can be regarded as redundancy. As shown in Figure 1, we analyze outputs from two mainstream LRMs on AIME24. By identifying the FCS in each response, we distinguish between two types of redundancy: internal redundancy, which refers to unnecessary steps embedded within the FCS itself, and external redundancy, which arises after the correct answer has already been produced. These observations motivate a semantic-aware compression framework aimed at removing both types of redundancy rather than merely shortening sequences. The third subfigure in Figure 1 illustrates the effect of our approach, where both internal and external redundancy have been successfully removed.

*Equal contribution

* 同等贡献

† Corresponding author: Wenpeng Zhu (zhuwen-peng@cmss.chinamobile.com), Jing Huo (huojing@nju.edu.cn) and Yang Gao (gaoy@nju.edu.cn)

† 通讯作者: 朱文鹏 (zhuwen-peng@cmss.chinamobile.com)，霍晶 (huojing@nju.edu.cn) 和高阳 (gaoy@nju.edu.cn)

首次正确解 (FCS)(Chen 等，2024) 指的是导致正确答案的最早完整推理轨迹。从理论上讲，FCS 中存在一个最小子集足以解决任务。基于此假设，超出该最小子集的任何内容，无论是 FCS 内多余的步骤还是之后生成的额外内容，都可视为冗余。如图 1 所示，我们分析了两种主流大规模语言模型 (LRMs) 在 AIME24 上的输出。通过识别每个回答中的 FCS，我们区分了两类冗余: 内部冗余，指嵌入 FCS 本身的不必要步骤；外部冗余，指在正确答案产生后出现的冗余。这些观察促使我们提出了一个语义感知压缩框架，旨在去除这两类冗余，而不仅仅是缩短序列。图 1 的第三个子图展示了我们方法的效果，成功去除了内部和外部冗余。

Internal Redundancy Compression LRMs often repeat semantically similar content, such as reiterating premises or reassessing intermediate steps, which we define as internal redundancy. To quantify this, we apply a sliding-window semantic similarity measure to detect repetitive spans before the first correct answer. A penalty is then incorporated into the RL objective to promote stepwise utility and reduce informational overlap.

内部冗余压缩大规模语言模型常常重复语义相似的内容，如重复前提或重新评估中间步骤，我们将其定义为内部冗余。为量化这一点，我们采用滑动窗口语义相似度度量来检测首次正确答案之前的重复片段。随后在强化学习目标中引入惩罚，以促进逐步效用并减少信息重叠。

External Redundancy Compression Once the correct answer is produced, further continuation (e.g., re-deriving the answer or verifying previous steps) contributes little to problem-solving. We define this as external redundancy. we use a normalized redundancy proportion, the ratio of trailing length to total reasoning length, as a penalty in the RL reward. This encourages the model to stop reasoning promptly once the correct answer is reached.

外部冗余压缩一旦产生正确答案，后续继续推理 (如重新推导答案或验证先前步骤) 对解决问题贡献甚微。我们将其定义为外部冗余。我们使用归一化冗余比例，即尾部长度与总推理长度的比率，作为强化学习奖励中的惩罚项，鼓励模型在达到正确答案后及时停止推理。

To validate our approach, we conduct extensive reinforcement learning experiments across three mathematical benchmarks. Results show that both internal and external redundancy steadily decrease during training, while accuracy is largely preserved. Notably, our method achieves comparable semantic conciseness to human-written solutions without relying on hard length constraints, confirming the effectiveness of our dual-penalty framework in guiding efficient reasoning behavior. In conclusion, our contributions are summarized as follows:

为验证我们的方法，我们在三个数学基准上进行了大量强化学习实验。结果显示，训练过程中内部和外部冗余均稳步下降，同时准确率基本保持不变。值得注意的是，我们的方法在不依赖严格长度约束的情况下，实现了与人类书写解答相当的语义简洁性，验证了双重惩罚框架在引导高效推理行为方面的有效性。总之，我们的贡献总结如下:

- We provide the first systematic decomposition of CoT redundancy into internal and external components, offering a novel semantic-aware perspective on overthinking in LRMs.

  - 我们首次系统地将链式思维 (CoT) 冗余分解为内部和外部两部分，提供了对大规模语言模型中过度思考的新颖语义感知视角。

- We develop an innovative sliding-window semantic similarity approach to detect and penalize low-informative reasoning steps within the FCS, enabling fine-grained internal redundancy mitigation.

- 我们开发了一种创新的滑动窗口语义相似度方法，用于检测并惩罚 FCS 内信息量低的推理步骤，实现了细粒度的内部冗余缓解。

- We introduce a normalized proportion-based metric to quantify external redundancy beyond the FCS, and apply a targeted penalty to discourage unnecessary continuation.

- 我们引入了一种基于归一化比例的指标来量化 FCS 之外的外部冗余，并应用针对性惩罚以抑制不必要的继续推理。

- Extensive experiments demonstrate that our method significantly compresses reasoning traces with minimal accuracy degradation. Further ablation studies reveal that the slight accuracy loss primarily stems from removing internal redundancy, while external redundancy can be reduced safely without harming performance.

- 大量实验表明，我们的方法显著压缩了推理轨迹，且准确率仅有极小下降。进一步的消融研究显示，轻微的准确率损失主要源于内部冗余的去除，而外部冗余的减少则不会影响性能。

Related Work

相关工作

CoT reasoning (Wei et al. 2022) has become a core technique for enhancing the step-by-step reasoning capabilities of LLMs. By decomposing complex questions into intermediate reasoning steps, CoT improves answer accuracy and transparency (Qiao et al. 2022). However, with increasing task complexity, generated CoT traces often become unnecessarily lengthy and redundant, reducing inference efficiency (Chen et al. 2024; Team et al. 2025).

链式思维推理 (CoT)(Wei 等，2022) 已成为提升大型语言模型 (LLMs) 逐步推理能力的核心技术。通过将复杂问题分解为中间推理步骤，CoT 提升了答案准确性和透明度 (Qiao 等，2022)。然而，随着任务复杂度增加，生成的 CoT 轨迹往往变得冗长且重复，降低了推理效率 (Chen 等，2024；Team 等，2025)。

Recent studies have attempted to compress CoT length through reinforcement learning with explicit length-based reward functions (Team et al. 2025; Arora and Zanette 2025; Shen et al. 2025; Qu et al. 2025; Yang, Lin, and Yu 2025; She et al. 2025; Hou et al. 2025). While effective to some extent, these approaches treat redundancy as a monolithic problem, overlook the root cause of redundancy and risk removing essential reasoning steps.

近期研究尝试通过强化学习结合显式基于长度的奖励函数来压缩 CoT 长度 (Team 等，2025；Arora 和 Zanette，2025；Shen 等，2025；Qu 等，2025；Yang、Lin 和 Yu，2025；She 等，2025；Hou 等，2025)。尽管在一定程度上有效，这些方法将冗余视为单一问题，忽视了冗余的根本原因，且存在移除关键推理步骤的风险。

In contrast to prior works, our approach introduces a semantic-aware dual-penalty framework that structurally decomposes overthinking into internal redundancy and external redundancy. By applying targeted penalties to each, our method provides finer control over reasoning compression and interpretability. To our knowledge, this is the first work to explicitly isolate and address these two forms of redundancy in LLM reasoning traces.

与以往工作不同，我们的方法引入了语义感知的双重惩罚框架，结构性地将过度思考分解为内部冗余和外部冗余。通过对两者施加针对性惩罚，我们的方法在推理压缩和可解释性上提供了更细致的控制。据我们所知，这是首个明确区分并解决 LLM 推理轨迹中这两类冗余的工作。

CoT Redundancy Detection

CoT 冗余检测

Redundant reasoning in LRMs can occur at different stages of the CoT process (Han et al. 2024; Liu et al. 2024; Ma et al. 2025). As shown in Figure 1, we observe that redundancy clusters either before or after the first correct answer, motivating a segmentation-based analysis.

大规模语言模型中的冗余推理可能发生在 CoT 过程的不同阶段 (Han 等，2024；Liu 等，2024；Ma 等，2025)。如图 1 所示，我们观察到冗余聚集在首次正确答案之前或之后，这促使我们采用基于分段的分析方法。

We adopt a regular-expression-based strategy to locate the earliest sentence containing the final correct answer. This sentence serves as a boundary to divide the CoT sequence into two segments: a pre-answer segment comprising all reasoning steps up to the answer, and a post-answer segment starting after the sentence where the answer first appears. This segmentation enables separate analysis of repetition patterns within the FCS (internal redundancy) and beyond it (external redundancy).

我们采用基于正则表达式的策略定位包含最终正确答案的最早句子。该句作为边界，将 CoT 序列划分为两个部分: 答案前段包含所有至答案的推理步骤，答案后段从答案首次出现句子之后开始。此分段使我们能够分别分析 FCS 内的重复模式 (内部冗余) 和 FCS 之外的重复模式 (外部冗余)。

Internal Redundancy Degree (IRD)

内部冗余度 (IRD)

In Figure 2a, we design a pipeline that first splits each CoT output into N discrete sentences $\{s_1, s_2, \ldots, s_N\}$. A sentence-level sliding window is then applied over this sequence to detect local redundancy. To adapt to CoT outputs of varying lengths, we dynamically set the window size and stride as fixed proportions of the total sentence count N. Specifically, the window size is defined as $w = \lfloor \alpha N \rfloor$, and the stride as $t = \lfloor \beta N \rfloor$, where $\alpha \in (0, 1), \beta \in (0, \alpha)$. The dynamic window size enables the method to scale naturally with different reasoning lengths, while the constraint $\beta < \alpha$ ensures overlapping windows, which smooths the semantic similarity signal and improves the robustness of redundancy estimation. Each window represents a contiguous reasoning segment, preserving local semantic coherence. For each segment, we compute sentence-level embeddings and calculate cosine similarity between adjacent windows:

在图 2a 中，我们设计了一个流程，首先将每个 CoT 输出拆分成 N 个离散句子 $\{s_1, s_2, \ldots, s_N\}$。然后对该序列应用句子级滑动窗口以检测局部冗余。为了适应不同长度的 CoT 输出，我们动态设置窗口大小和步幅为总句子数 N 的固定比例。具体地，窗口大小定义为 $w = \lfloor \alpha N \rfloor$，步幅为 $t = \lfloor \beta N \rfloor$，其中 $\alpha \in (0,1), \beta \in (0,\alpha)$。动态窗口大小使该方法能够自然适应不同推理长度，而约束条件 $\beta < \alpha$ 确保窗口重叠，这平滑了语义相似度信号并提升了冗余估计的鲁棒性。每个窗口代表一个连续的推理片段，保持局部语义连贯。对于每个片段，我们计算句子级嵌入并计算相邻窗口之间的余弦相似度:

$$\mathbf{e}_i = f_{\text{embedding}}\left(\mathcal{W}_i\right) \tag{1}$$

$$\text{sim}_i = \max\left(0, \cos\left(\mathbf{e}_i, \mathbf{e}_{i+1}\right)\right) \tag{2}$$

$$\text{IRD} = \frac{1}{M} \sum_{i=1}^{M} \text{sim}_i \tag{3}$$

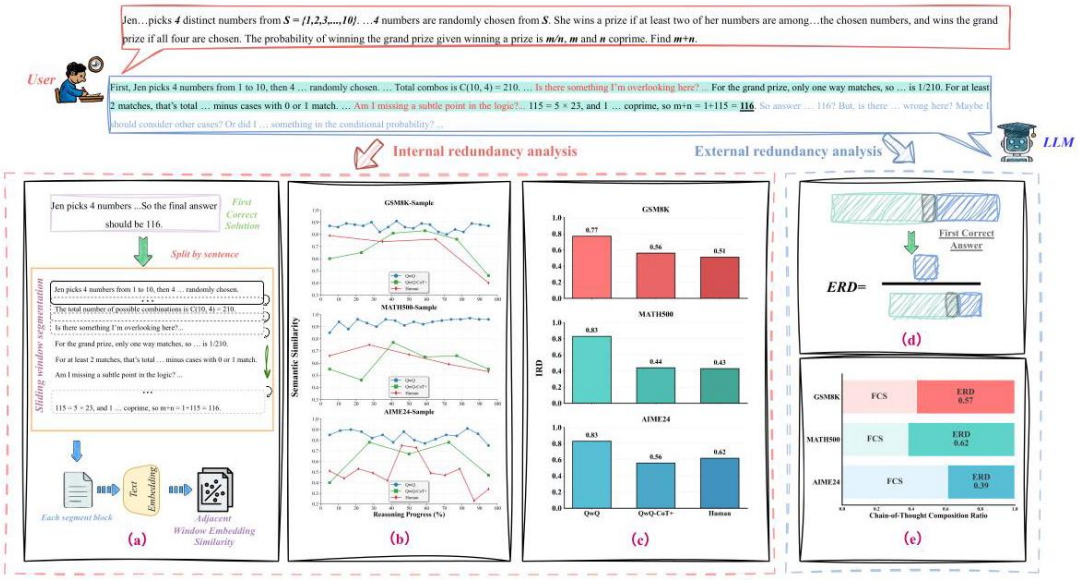

Figure 2: Analysis of CoT redundancy. (a) Internal redundancy detection using sliding windows. (b) Semantic similarities of QwQ responses, QwQ-CoT+, and human-written solutions. (c) The Internal Redundancy Degree (IRD) of QwQ responses, QwQ-CoT+ and human references across GSM8K, MATH500 and AIME24. (d) The External Redundancy Degree (ERD) measurement using length proportion. (e) The ERD of QwQ responses across GSM8K, MATH500, and AIME24.

图 2:CoT 冗余分析。(a) 使用滑动窗口检测内部冗余。(b) QwQ 响应、QwQ-CoT+ 及人工解答的语义相似度。(c) QwQ 响应、QwQ-CoT+ 及人工参考在 GSM8K、MATH500 和 AIME24 上的内部冗余度 (IRD)。(d) 使用长度比例测量的外部冗余度 (ERD)。(e) QwQ 响应在 GSM8K、MATH500 和 AIME24 上的 ERD 表现。

where $\mathcal{W}_i = \{s_i, s_{i+1}, \ldots, s_{i+w-1}\}$ denotes the i -th windowed segment, M is the total number of windows. To focus on semantic redundancy, we clip negative similarity values to zero during computation. Since CoT

reasoning inherently relies on maintaining semantic continuity, elevated local similarity often signals insufficient informational progression. We thus define the IRD as the average similarity across adjacent segments, where higher values indicate less efficient reasoning within the FCS.

其中 $\mathcal{W}_i = \{s_i, s_{i+1}, \ldots, s_{i+w-1}\}$ 表示第 i 个窗口片段，M 为窗口总数。为聚焦语义冗余，计算时将负相似度值截断为零。由于 CoT 推理本质上依赖于保持语义连续性，较高的局部相似度通常表明信息进展不足。因此，我们将 IRD 定义为相邻片段间相似度的平均值，较高值表示 FCS 内推理效率较低。

To empirically validate our proposed IRD metric, we construct a high-quality CoT dataset, QwQ-CoT+, by refining the original QwQ-32B model responses with a well-designed prompt. The resulting traces retain correctness while substantially reducing redundancy, producing concise multi-step reasoning. We evaluate the correct outputs of QwQ, QwQ-CoT+, and human-written solutions using OpenAI's text-3-embedding-large (OpenAI 2024) model to compute the IRD (with $\alpha = 0.1, \beta = 0.05$). As shown in Figure 2b and 2c, QwQ-CoT+ and human references consistently exhibit lower IRD values than QwQ responses across the GSM8K, MATH500, and AIME24 benchmarks, confirming the IRD metric's ability to detect redundancy in the FCS. Notably, the IRD scores of QwQ-CoT+ and human references remain significantly above zero, indicating that a moderate redundancy may be necessary for effective reasoning.

为实证验证我们提出的 IRD 指标, 我们构建了高质量 CoT 数据集 QwQ-CoT+, 通过精心设计的提示对原始 QwQ-32B 模型响应进行优化。生成的推理轨迹在保持正确性的同时显著减少冗余, 产生简洁的多步推理。我们使用 OpenAI 的 text-3-embedding-large (OpenAI 2024) 模型计算 QwQ、QwQ-CoT+ 及人工解答的 IRD(带 $\alpha = 0.1, \beta = 0.05$)。如图 2b 和 2c 所示，QwQ-CoT+ 和人工参考在 GSM8K、MATH500 和 AIME24 基准上持续表现出低于 QwQ 响应的 IRD 值，验证了 IRD 指标检测 FCS 冗余的能力。值得注意的是，QwQ-CoT+ 和人工参考的 IRD 分数仍显著高于零，表明适度冗余可能对有效推理是必要的。

External Redundancy Degree (ERD)

外部冗余度 (ERD)

External redundancy, defined as content beyond the FCS in CoT, typically does not contribute to deriving the correct answer and can be considered uninformative. To quantify this, we propose the External Redundancy Degree (ERD), which measures the proportion of redundant content rather than absolute length, avoiding bias against longer CoT outputs. Specifically, ERD is calculated as the ratio of the length of the external segment after the FCS to the total length of the CoT reasoning trace:

外部冗余定义为 CoT 中 FCS 之外的内容，通常不助于得出正确答案，可视为无信息量。为量化此点，我们提出外部冗余度 (ERD)，衡量冗余内容的比例而非绝对长度，避免对较长 CoT 输出的偏见。具体地，ERD 计算为 FCS 之后外部片段长度与 CoT 推理轨迹总长度的比值:

$$\text{ERD} = \frac{L_{\text{external}}}{L_{\text{total}}} \tag{4}$$

As illustrated in Figure 2d, a higher ERD indicates greater external redundancy in the CoT reasoning process.

Since QwQ-CoT+ and human-written solutions exhibit no external redundancy, we report only QwQ's ERD performance in Figure 2e. This metric effectively quantifies unnecessary reasoning produced after reaching the first correct answer, providing a clear measure of model efficiency in CoT reasoning.

如图 2d 所示，ERD 越高表示 CoT 推理过程中的外部冗余越大。由于 QwQ-CoT+ 和人工解答无外部冗余，我们仅报告图 2e 中 QwQ 的 ERD 表现。该指标有效量化了达到首个正确答案后产生的不必要推理，清晰衡量模型在 CoT 推理中的效率。

Dual-Redundancy Penalty

双重冗余惩罚

To mitigate internal and external redundancy, we augment the reinforcement learning objective by applying two penalty terms to the accuracy-based reward. These penalties are computed from the internal and external redundancy degrees and encourage the model to generate concise and efficient reasoning without unnecessary repetition or continuation.

为减轻内部和外部冗余，我们通过在基于准确率的奖励中加入两个惩罚项来增强强化学习目标。这些惩罚项基于内部和外部冗余度计算，鼓励模型生成简洁高效的推理，避免不必要的重复或冗长。

Internal Redundancy Penalty To reduce the redundancy in FCS, we apply a normalized internal redundancy penalty upon the accuracy reward. The total reward function is as follows:

内部冗余惩罚为了减少 FCS 中的冗余，我们在准确率奖励上施加了归一化的内部冗余惩罚。总奖励函数如下:

$$\mathbb{E}_{x\sim D}\left[\mathbb{1}\left\{y_{\text{pred}} = y_{\text{GT}}\right\}\cdot\frac{\sigma_k\left(1-\text{IRD}\right)-\sigma_k\left(0\right)}{\sigma_k\left(1\right)-\sigma_k\left(0\right)}\right], \tag{5}$$

where $\sigma_k\left(x\right) = \frac{1}{1+e^{-k(x-c)}}$ is a sharpened sigmoid function with slope k and center c. As shown in our earlier analysis of QwQ-CoT+ and human references, a moderate amount of internal redundancy is often essential for coherent reasoning. To preserve this desirable property, we calibrate the penalty function such that it only becomes active when the IRD exceeds a threshold. Specifically, we set $k = 20$ and $c = 0.3$ in this paper, so that the penalty is negligible when IRD < 0.5 , and increases rapidly beyond this point. This allows the model to tolerate reasonable redundancy while still discouraging excessive repetition.

其中 $\sigma_k\left(x\right) = \frac{1}{1+e^{-k(x-c)}}$ 是一个斜率为 k 、中心为 c 的锐化 sigmoid 函数。如我们之前对 QwQ-CoT+ 和人工参考的分析所示，适度的内部冗余通常是连贯推理的必要条件。为了保留这一理想特性，我们校准了惩罚函数，使其仅在 IRD 超过阈值时才生效。具体而言，本文设置了 $k = 20$ 和 $c = 0.3$ ，使得当 IRD < 0.5 时惩罚可忽略不计，且在此点之后迅速增加。这允许模型容忍合理的冗余，同时抑制过度重复。

External Redundancy Penalty Similarly, to discourage unnecessary continuation after the answer is found, we apply a normalized linear penalty based on the ERD:

外部冗余惩罚同样地，为了防止在找到答案后不必要的继续，我们基于 ERD 施加了归一化的线性惩罚:

$$\mathbb{E}_{x\sim D}\left[\mathbb{1}\{y_{\text{pred}} = y_{\text{GT}}\} \cdot (1 - \text{ERD})\right] \tag{6}$$

Experiment

实验

In this section, we detail the experimental setup and comparative results to assess how well our method reduces over-thinking compared to existing RL-based length compression approaches. The overall training procedure is summarized in Algorithm 1.

本节详细介绍实验设置及对比结果，以评估我们的方法在减少过度思考方面相较于现有基于强化学习的长度压缩方法的效果。整体训练流程总结于算法 1。

Training Setup

训练设置

We adopt verl (Sheng et al. 2024), a scalable and high-throughput reinforcement learning library tailored for LLMs. Training is conducted using Group Relative Policy Optimization (GRPO) (Shao et al. 2024) across 64 NVIDIA A800 GPUs. We fine-tune DeepSeek-R1-Distilled-Qwen- 1.5B and DeepSeek-R1-Distill-Qwen-7B on a subset of DeepScaleR (Luo et al. 2025b), originally containing 40k math questions. This subset is curated by selecting 10k samples whose answers are purely numeric and contain at least two digits to ensure accurate extraction of FCS from CoT traces. Notably, We choose DeepScaleR as our training corpus because it has been widely adopted in prior works on RL-based CoT compression and is also used, fully or partially, by several baselines we compare against. This ensures a fair and consistent training setup for evaluating the effectiveness of our proposed method. Our training is performed with temperature 0.6, top-p 1.0, 8 samples per input, and a batch size of 128 .

我们采用 ver1(Sheng 等，2024)，一款针对大型语言模型 (LLMs) 设计的可扩展高吞吐量强化学习库。训练使用 64 块 NVIDIA A800 GPU，基于群体相对策略优化 (GRPO)(Shao 等，2024) 进行。我们在 DeepScaleR(Luo 等，2025b) 的子集上微调 DeepSeek-R1-Distilled-Qwen-1.5B 和 DeepSeek-R1-Distill-Qwen-7B，该子集原含 4 万道数学题。该子集通过筛选答案纯数字且至少含两位数的 10k 样本构成，以确保能准确从 CoT 轨迹中提取 FCS。值得注意的是，我们选择 DeepScaleR 作为训练语料库，是因为其在先前基于强化学习的 CoT 压缩研究中被广泛采用，且多个对比基线也全部或部分使用该数据，确保了评估我们方法有效性的公平一致训练环境。训练参数为温度 0.6，top-p 1.0，每输入 8 个样本，批量大小 128。

Algorithm 1: Training with Dual-Redundancy Penalty

算法 1: 双重冗余惩罚训练

Require: Model $\mathcal{M}$, Dataset $\mathcal{D}$, Reward function $\mathcal{R}$, Opti-

输入: 模型 $\mathcal{M}$ ，数据集 $\mathcal{D}$ ，奖励函数 $\mathcal{R}$ ，优化器

mizer $\mathcal{O}$, Window size w

$\mathcal{O}$ ，窗口大小 w

for each batch of prompts $\{x_i\}_{i=1}^{N}$ from $\mathcal{D}$ do

对于来自 $\mathcal{D}$ 的每个批次提示 $\{x_i\}_{i=1}^{N}$ ，执行

Sample K responses $\left\{\widehat{y}_i^{(k)}\right\}_{k=1}^{K}$ from $\mathcal{M}\left(x_i\right)$ for each

从 $\mathcal{M}\left(x_i\right)$ 中为每个提示采样 K 个响应 $\left\{\widehat{y}_i^{(k)}\right\}_{k=1}^{K}$

x_i

for each response $\widehat{y}_i^{(k)}$ do

对于每个响应 $\widehat{y}_i^{(k)}$ ，执行

if final answer is incorrect then

如果最终答案错误，则

Assign reward $r_i^{(k)} \leftarrow 0$

赋予奖励 $r_i^{(k)} \leftarrow 0$

else

否则

Locate first correct solution (FCS) in $\widehat{y}_i^{(k)}$

定位第一个正确解 (FCS) 于 $\widehat{y}_i^{(k)}$

Split into [FCS, post-FCS]

拆分为 [FCS，后 FCS]

Compute internal redundancy penalty: p_{int}

计算内部冗余惩罚: p_{int}

Compute external redundancy penalty: p_{ext}

计算外部冗余惩罚: p_{ext}

Assign reward: $r_i^{(k)} \leftarrow r_i^{(k)} \cdot p_{\text{int}} \cdot p_{\text{ext}}$

分配奖励: $r_i^{(k)} \leftarrow r_i^{(k)} \cdot p_{\text{int}} \cdot p_{\text{ext}}$

end if

结束条件判断

end for

结束循环

Compute GRPO policy loss with $\left\{r_i^{(k)}\right\}$

使用 $\left\{r_i^{(k)}\right\}$ 计算 GRPO 策略损失

Update model $\mathcal{M}$ via optimizer $\mathcal{O}$

通过优化器 $\mathcal{O}$ 更新模型 $\mathcal{M}$

end for

结束循环

Baselines

基线方法

For a fair comparison, we benchmark our method against high-performing RL-based approaches that are representative of recent advances in length compression.

为了公平比较，我们将方法与代表近期长度压缩进展的高性能基于强化学习 (RL) 的方案进行基准测试。

ThinkPrune (Hou et al. 2025) imposes a maximum generation length constraint during reinforcement learning, compelling the model to complete reasoning within a fixed token budget. This encourages concise reasoning behavior under pressure.

ThinkPrune(Hou 等，2025) 在强化学习过程中施加最大生成长度限制，迫使模型在固定的令牌预算内完成推理。这鼓励模型在压力下进行简洁推理。

LC-R1 (Cheng et al. 2025) builds upon a length-penalty reward by introducing an additional compression reward, which is computed by using an auxiliary LLM to generate a compressed version of the original response. The reward is proportional to the reduction in length compared to the original.

LC-R1(Cheng 等,2025) 基于长度惩罚奖励,额外引入压缩奖励,该奖励通过辅助大型语言模型 (LLM) 生成原始回答的压缩版本计算。奖励与相较原文的长度缩减成正比。

Laser-DE (Liu et al. 2025b) avoids hard truncation by setting a context window significantly larger than the target length, and encourages brevity by assigning extra rewards to correct outputs whose lengths fall below the target length.

Laser-DE(Liu 等，2025b) 避免硬截断，通过设置远大于目标长度的上下文窗口，并通过对长度低于目标的正确输出赋予额外奖励来鼓励简洁。

Training (Arora and Zanette 2025) leverages the multiple-sampling nature of reinforcement learning. It applies differentiated rewards based on the length of generated answers, favoring shorter completions that maintain correctness.

Training(Arora 和 Zanette，2025) 利用强化学习的多样本特性，根据生成答案的长度施加差异化奖励，偏好保持正确性的更短回答。

All baselines are evaluated using their publicly released models except for ThinkPrune's DeepSeek-R1- Distill-Qwen-7B model. As this model is not publicly available, we reproduced it following the implementation details described in their paper.

除 ThinkPrune 的 DeepSeek-R1-Distill-Qwen-7B 模型外，所有基线均使用其公开发布的模型进行评估。由于该模型未公开，我们根据其论文中描述的实现细节进行了复现。

Evaluation Setup

评估设置

We conduct all evaluations under a unified experimental setup across three mathematical reasoning benchmarks of varying difficulty: GSM8K (Cobbe et al. 2021), MATH500 (Hendrycks et al. 2021), and AIME24. We use Pass@1 as the primary metric for reasoning accuracy. During inference, we allow a maximum response length of 16k tokens. For GSM8K and MATH500, we sample 4 responses per instance with a temperature of 0.6, while for AIME24, we use 64 samples due to the limited number of available problems. To ensure a fair comparison across different CoT compression methods, some of which may shorten or omit the conclusion, we exclude the final answer statement (conclusion) part from token length statistics. This allows us to more precisely measure the efficiency of the reasoning process itself.

我们在三个难度不同的数学推理基准测试上，采用统一的实验设置进行所有评估:GSM8K(Cobbe 等，2021)、MATH500(Hendrycks 等，2021) 和 AIME24。我们以 Pass@1 作为推理准确率的主要指标。推理时，允许最大响应长度为 16k 个标记。对于 GSM8K 和 MATH500，每个实例采样 4 个响应，温度为 0.6；而对于 AIME24，由于可用问题数量有限，采样 64 个响应。为确保不同链式推理 (CoT) 压缩方法间的公平比较，其中一些方法可能缩短或省略结论部分，我们在标记长度统计中排除了最终答案陈述 (结论) 部分，从而更精确地衡量推理过程本身的效率。

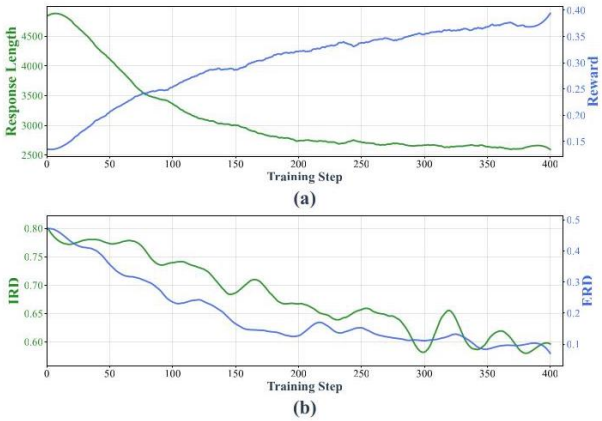

Figure 3: Training process. (a) Response length and reward trends. (b) IRD and ERD trends.

图 3: 训练过程。(a) 响应长度和奖励趋势。(b) 内部冗余度 (IRD) 和外部冗余度 (ERD) 趋势。

Token Efficiency Metric

标记效率指标

In addition, to quantitatively evaluate the trade-off between reasoning accuracy and conciseness, we propose the Token Efficiency (TE) metric, defined as:

此外，为了定量评估推理准确性与简洁性之间的权衡，我们提出了标记效率 (Token Efficiency，TE) 指标，定义如下:

$$\mathrm{TE} = \frac{A}{\sqrt{L}} \tag{7}$$

This metric is inspired by the per-token accuracy formulation A/L , which captures how much each token contributes to the overall correctness. To reflect a stronger preference for accuracy over brevity during evaluation, we apply a square root to the token count L , dampening the penalty for longer but correct traces. A higher TE indicates a better balance between precision and efficiency.

该指标灵感来源于每标记准确率的公式 A/L ，反映了每个标记对整体正确性的贡献。为了在评估中更加强调准确性而非简洁性，我们对标记数 L 取平方根，减弱了对较长但正确推理路径的惩罚。TE 值越高，表示精度与效率的平衡越好。

Main Results

主要结果

We first present the training process of our dual-penalty method. Figure 3 shows that the reward steadily increases throughout training, while both IRD and ERD consistently decrease, demonstrating the effectiveness of the proposed penalty mechanism. Specifically, ERD converges to approximately 0.1, and IRD stabilizes around 0.6, comparable to the levels observed in QwQ-CoT+ and human-written solutions. This reduction is also reflected in the declining average response length, suggesting that our method encourages more efficient reasoning without relying on explicit length constraints. These findings provide a key insight: by targeting the semantic characteristics of redundant reasoning content, our approach enables LRMs to suppress overthinking in a principled and interpretable manner.

我们首先展示了双重惩罚方法的训练过程。图 3 显示，训练过程中奖励稳步提升，而内部冗余度 (IRD) 和外部冗余度 (ERD) 持续下降，证明了所提惩罚机制的有效性。具体而言，ERD 收敛至约 0.1，IRD 稳定在 0.6 左右，与 QwQ-CoT+ 及人工编写解答的水平相当。这一减少也反映在平均响应长度的下降上，表明我们的方法在不依赖显式长度约束的情况下，促进了更高效的推理。这一发现提供了关键见解: 通过针对冗余推理内容的语义特征，我们的方法使大规模语言模型 (LRMs) 能够以原则性且可解释的方式抑制过度思考。

As shown in Table 1, our method achieves the best overall performance in terms of the token efficiency on both DeepSeek-R1-Distill-Qwen-1.5B and 7B models. Especially, on the more challenging MATH500 and AIME24 benchmarks, our approach significantly outperforms all baselines. In addition, we find that some existing length-compression methods still exhibit considerable internal and external redundancy in the reasoning traces, indicating that our method can be applied on top of existing compression techniques to further refine the CoT process. Among the baselines, those with higher TE scores also demonstrate marked reductions in both types of redundancy, suggesting a convergent trend: models effectively learn to compress CoT reasoning by minimizing internal and external redundancy. This observation indirectly validates that our method successfully captures the fundamental nature of redundancy in LRMs.

如表 1 所示，我们的方法在 DeepSeek-R1-Distill-Qwen-1.5B 和 7B 模型上均实现了标记效率的最佳整体表现。尤其在更具挑战性的 MATH500 和 AIME24 基准上，我们的方法显著优于所有基线。此外，我们发现一些现有的长度压缩方法在推理路径中仍存在较大程度的内部和外部冗余，表明我们的方法可以叠加于现有压缩技术之上，进一步优化链式推理过程。在基线中，标记效率较高的方法也表现出内部和外部冗余的显著减少，显示出一种趋同趋势: 模型通过最小化内部和外部冗余，有效学习压缩链式推理。这一观察间接验证了我们的方法成功捕捉了大规模语言模型中冗余的本质。

Cross-Domain Generalization

跨领域泛化

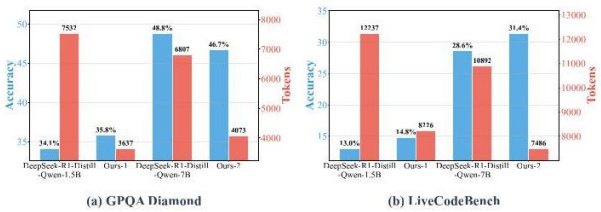

Figure 4: Performance on GPQA Diamond and Live-CodeBench. Our method generalizes well to out-of-domain (OOD) reasoning tasks.

图 4:GPQA Diamond 和 Live-CodeBench 上的表现。我们的方法在领域外 (OOD) 推理任务中表现良好。

Some findings (Meng et al. 2025; Chu et al. 2025; Xie et al. 2025) suggest that reinforcement learning can endow large models with strong generalization capabilities on out-of-domain (OOD) data. Motivated by this, we investigate whether the redundancy compression patterns learned through RL on mathematical reasoning tasks can generalize to non-mathematical domains. Specifically, we evaluate our trained models on two OOD benchmarks: GPQA Diamond (Rein et al. 2024), which emphasizes graduate-level factual reasoning, and LiveCodeBench (Jain et al. 2024), which targets multi-hop reasoning in program understanding. Models are evaluated on LiveCodeBench with test data collected between May 2023 and January 2025.

一些研究 (Meng 等，2025；Chu 等，2025；Xie 等，2025) 表明，强化学习可以赋予大模型在领域外 (OOD) 数据上的强泛化能力。受此启发，我们探讨了通过强化学习在数学推理任务中学到的冗余压缩模式是否能泛化到非数学领域。具体而言，我们在两个 OOD 基准上评估训练好的模型:GPQA Diamond(Rein 等，2024)，侧重研究生级事实推理；以及 LiveCodeBench(Jain 等，2024)，聚焦程序理解中的多跳推理。LiveCodeBench 的测试数据收集时间为 2023 年 5 月至 2025 年 1 月。

As shown in Figure 4, our method remains effective in compressing CoT traces across both datasets. This indicates that the model has internalized a domain-agnostic paradigm for generating concise and efficient reasoning traces, beyond merely overfitting to the training distribution. The results underscore the transferability of our RL-based compression framework and suggest its broader applicability to various reasoning-intensive tasks.

如图 4 所示，我们的方法在两个数据集上均能有效压缩链式推理路径，表明模型已内化了一种领域无关的范式，能够生成简洁高效的推理路径，而非仅仅对训练分布进行过拟合。结果强调了我们基于强化学习的压缩框架的迁移能力，并暗示其在各种推理密集型任务中的广泛适用性。

Ablation Studies

消融研究

Composition To rigorously evaluate the individual and combined contributions of internal and external redundancy penalties to CoT length compression, we perform an ablation study under controlled conditions. As a baseline, we apply a fixed truncation of 6k tokens to each response during training. Building upon this, we separately introduce the internal redundancy penalty and the external redundancy penalty to assess their isolated impact on the reduction of response length over training steps. As shown in Figure 5a, both penalties independently lead

to a substantial decrease in response length, reaching a minimal length around 3100 tokens. When both penalties are applied jointly, the response length is further compressed to approximately 2600 tokens, indicating a complementary and synergistic effect. Moreover, in Figure 5 b and 5c , we observe that the application of one penalty does not significantly influence the redundancy degree targeted by the other, suggesting that internal and external redundancy reflect orthogonal aspects of reasoning traces.

组成为了严格评估内部和外部冗余惩罚对链式思维 (CoT) 长度压缩的单独及联合贡献，我们在受控条件下进行了消融研究。作为基线，我们在训练过程中对每个响应应用固定截断 6k 个标记。在此基础上，我们分别引入内部冗余惩罚和外部冗余惩罚，以评估它们对训练步骤中响应长度减少的独立影响。如图 5a 所示，两种惩罚均能独立显著减少响应长度，最低可达约 3100 个标记。当两种惩罚联合应用时，响应长度进一步压缩至约 2600 个标记，表明二者具有互补和协同效应。此外，在图 5 b 和 5c 中，我们观察到一种惩罚的应用并未显著影响另一种惩罚所针对的冗余程度，表明内部和外部冗余反映了推理轨迹的正交方面。

Model	GSM8K					MATH500					AIME24				
	Accuracy	Tokens	IRD	ERD	TE	Accuracy	Tokens	IRD	ERD	TE	Accuracy	Tokens	IRD	ERD	TE
DeepSeek-R1-Distill-Qwen-1.5B															
Original Model	84.1	1555	73.7	43.0	2.13	82.2	3549	77.5	55.2	1.38	28.5	8681	71.4	28.6	0.31
ThinkPrune-4k	86.1	910	77.9	40.1	2.85	83.7	2101	73.2	39.8	1.83	28.6	6431	75.2	21.0	0.36
LC-R1	82.5	507	67.2	19.3	3.66	79.6	1673	75.8	22.5	1.95	24.2	5075	79.6	20.4	0.34
Laser-DE	86.4	971	74.3	37.5	2.77	83.6	2282	78.0	36.3	1.75	32.7	7268	73.5	22.2	0.38
Training	81.0	292	61.6	7.8	4.74	82.8	1543	65.5	14.5	2.11	28.5	7049	73.2	17.4	0.34
Ours	84.9	513	49.6	5.7	3.75	83.8	1505	51.0	7.9	2.16	34.0	6077	72.5	10.9	0.44
DeepSeek-R1-Distill-Owen-7B															
Original Model	91.1	844	70.0	36.0	3.14	91.2	2836	78.1	51.6	1.71	52.3	7241	77.8	31.1	0.61
ThinkPrune-4k	92.8	716	70.5	36.0	3.47	89.7	1683	77.9	36.1	2.19	50.4	5723	79.2	14.6	0.67
LC-R1	87.5	152	61.8	4.9	7.10	87.5	1201	65.8	7.0	2.52	52.7	6087	79.1	10.2	0.68
Laser-DE	93.3	637	68.2	31.1	3.70	92.1	1402	77.0	30.1	2.46	52.7	5061	80.5	11.8	0.74
Training	91.2	387	65.1	14.6	4.64	91.0	2090	76.3	38.3	1.99	50.8	6669	78.8	23.1	0.62
Ours	90.9	318	51.8	6.5	5.10	89.8	1200	58.7	6.1	2.59	53.2	5025	77.4	3.7	0.75

模型	GSM8K					MATH500					AIME24				
	准确率	标记数	IRD	ERD	TE	准确率	标记数	IRD	ERD	TE	准确率	标记数	IRD	ERD	TE
DeepSeek-R1-Distill-Qwen-1.5B															
原始模型	84.1	1555	73.7	43.0	2.13	82.2	3549	77.5	55.2	1.38	28.5	8681	71.4	28.6	0.31
ThinkPrune-4k	86.1	910	77.9	40.1	2.85	83.7	2101	73.2	39.8	1.83	28.6	6431	75.2	21.0	0.36
LC-R1	82.5	507	67.2	19.3	3.66	79.6	1673	75.8	22.5	1.95	24.2	5075	79.6	20.4	0.34
Laser-DE	86.4	971	74.3	37.5	2.77	83.6	2282	78.0	36.3	1.75	32.7	7268	73.5	22.2	0.38
训练	81.0	292	61.6	7.8	4.74	82.8	1543	65.5	14.5	2.11	28.5	7049	73.2	17.4	0.34
Ours	84.9	513	49.6	5.7	3.75	83.8	1505	51.0	7.9	2.16	34.0	6077	72.5	10.9	0.44
DeepSeek-R1-Distill-Owen-7B															
原始模型	91.1	844	70.0	36.0	3.14	91.2	2836	78.1	51.6	1.71	52.3	7241	77.8	31.1	0.61
ThinkPrune-4k	92.8	716	70.5	36.0	3.47	89.7	1683	77.9	36.1	2.19	50.4	5723	79.2	14.6	0.67
LC-R1	87.5	152	61.8	4.9	7.10	87.5	1201	65.8	7.0	2.52	52.7	6087	79.1	10.2	0.68
Laser-DE	93.3	637	68.2	31.1	3.70	92.1	1402	77.0	30.1	2.46	52.7	5061	80.5	11.8	0.74
训练	91.2	387	65.1	14.6	4.64	91.0	2090	76.3	38.3	1.99	50.8	6669	78.8	23.1	0.62
Ours	90.9	318	51.8	6.5	5.10	89.8	1200	58.7	6.1	2.59	53.2	5025	77.4	3.7	0.75

Table 1: Comparison with CoT compression baselines. Accuracy denotes Pass@1 accuracy; Tokens indicates average CoT length. Our method achieves the best balance of accuracy and efficiency. We time IRD, ERD and TE with a scaling factor 100 for readability.

表 1: 与 CoT 压缩基线的比较。准确率表示 Pass@1 准确率；Tokens 表示平均 CoT 长度。我们的方法在准确率和效率之间实现了最佳平衡。为了便于阅读，我们对 IRD、ERD 和 TE 的时间进行了 100 倍的缩放。

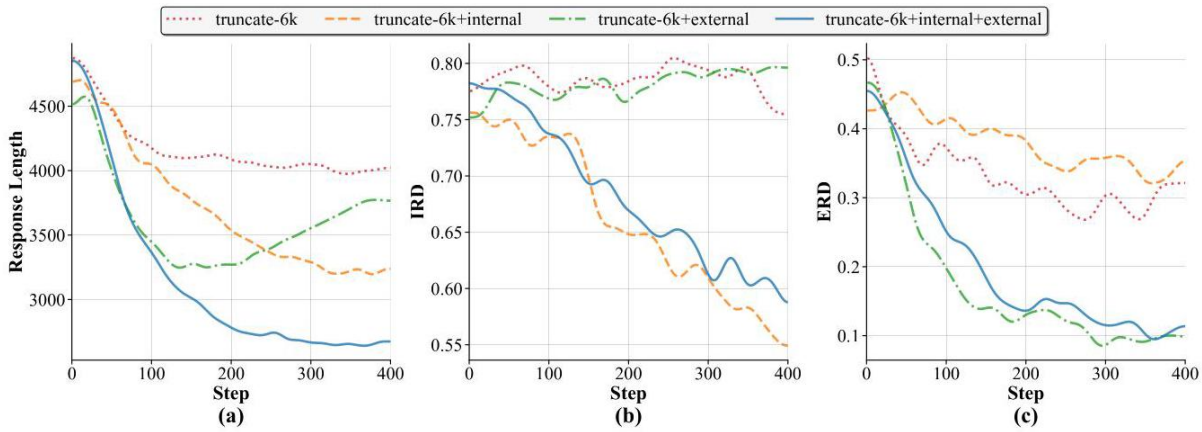

Figure 5: Impact of internal and external redundancy penalties on CoT compression. These two penalties operate independently with minimal interference, yet their combined use enhances compression efficiency beyond individual applications.

图 5: 内部和外部冗余惩罚对 CoT 压缩的影响。这两种惩罚独立作用，干扰极小，但联合使用能提升压缩效率，优于单独应用。

Accuracy To understand the source of accuracy degradation during CoT compression, we conduct a controlled ablation study that separately examines the effects of internal and external redundancy reduction. Unlike our main experiment where the base model (DeepSeek-R1-Distill-Qwen- 1.5B) still retains room for accuracy improvement during reinforcement learning, we deliberately choose a saturated model, DeepScaleR-1.5B-Preview, whose performance has plateaued. This allows us to eliminate potential reward-induced accuracy gains, thereby studying the true impact of redundancy compression.

准确率为了理解 CoT 压缩过程中准确率下降的原因，我们进行了受控消融研究，分别考察内部和外部冗余减少的影响。与主实验中基础模型 (DeepSeek-R1-Distill-Qwen-1.5B) 在强化学习期间仍有提升空间不同，我们特意选择了性能已达瓶颈的模型 DeepScaleR-1.5B-Preview。这样可以排除奖励引起的准确率提升，进而研究冗余压缩的真实影响。

We first apply external redundancy penalty alone during RL training, using a 16 k maximum response length. Once the ERD converges, we proceed to a second training stage where only the internal redundancy penalty is applied. In this phase, we limit the response length to 8k tokens to avoid the model mistakenly adapting to longer outputs. All other training and evaluating configurations are kept consistent with the main experiment. We report the accuracy associated with various IRD and ERD checkpoints in Table 2.

我们首先在强化学习训练中仅应用外部冗余惩罚，使用 16 k 最大响应长度。ERD 收敛后，进入第二阶段训练，仅施加内部冗余惩罚。此阶段我们将响应长度限制为 8k 个 tokens，以避免模型错误适应更长输出。其他训练和评估配置与主实验保持一致。表 2 报告了不同 IRD 和 ERD 检查点对应的准确率。

The results show that when ERD converges to 0.09, the model maintains nearly identical accuracy to its initial state across all three benchmarks (GSM8K, MATH500, and AIME24). In contrast, as the IRD is progressively reduced, accuracy drops substantially, especially on the more Table 2: Accuracy Changes under IRD and ERD Reduction. When ERD converges, the accuracy remains largely unaffected. In contrast, reducing IRD leads to notable accuracy drop, especially on the more challenging AIME24 benchmark. complex AIME24 dataset. These findings suggest that accuracy degradation during CoT compression is primarily attributable to the removal of internal redundancy.

结果显示，当 ERD 收敛至 0.09 时，模型在三个基准测试 (GSM8K、MATH500 和 AIME24) 上的准确率几乎与初始状态相同。相比之下，随着 IRD 逐步减少，准确率显著下降，尤其是在更复杂的 AIME24 数据集上。表 2:IRD 和 ERD 减少下的准确率变化。当 ERD 收敛时，准确率基本不受影响；而减少 IRD 则导致明显的准确率下降，特别是在更具挑战性的 AIME24 基准上。这些发现表明，CoT 压缩过程中准确率下降主要归因于内部冗余的去除。

Model	GSM8K			MATH500			AIME24		
	Accuracy	Tokens	Accuracy Drop	Accuracy	Tokens	Accuracy Drop	Accuracy	Tokens	Accuracy Drop
DeepScaleR-1.5B-Preview	87.8	1437	-	87.7	2593	-	41.1	7561	-
+ External penalty (ERD=0.09)	87.2	959	↓ 0.68%	87.6	1970	↓0.11%	41.0	6681	↓0.24%
+ Internal penalty (IRD=0.76)	87.2	762	↓ 0.68%	86.7	1758	↓1.14%	38.4	6555	↓ 6.57%
+ Internal penalty (IRD=0.68)	86.5	445	↓ 1.48%	84.0	1454	↓ 4.22%	38.2	5504	↓ 7.06%
+ Internal penalty (IRD=0.60)	85.4	345	↓ 2.73%	83.5	1247	↓ 4.79%	34.7	4810	↓ 15.57%

模型	GSM8K			MATH500			AIME24		
	准确率	标记数	准确率下降	准确率	标记数	准确率下降	准确率	标记数	准确率下降
DeepScaleR-1.5B-预览版	87.8	1437	-	87.7	2593	-	41.1	7561	-
+ 外部惩罚 (ERD=0.09)	87.2	959	↓ 0.68%	87.6	1970	↓0.11%	41.0	6681	↓0.24%
+ 内部惩罚 (IRD=0.76)	87.2	762	↓ 0.68%	86.7	1758	↓1.14%	38.4	6555	↓ 6.57%
+ 内部惩罚 (IRD=0.68)	86.5	445	↓ 1.48%	84.0	1454	↓ 4.22%	38.2	5504	↓ 7.06%
+ 内部惩罚 (IRD=0.60)	85.4	345	↓ 2.73%	83.5	1247	↓ 4.79%	34.7	4810	↓ 15.57%

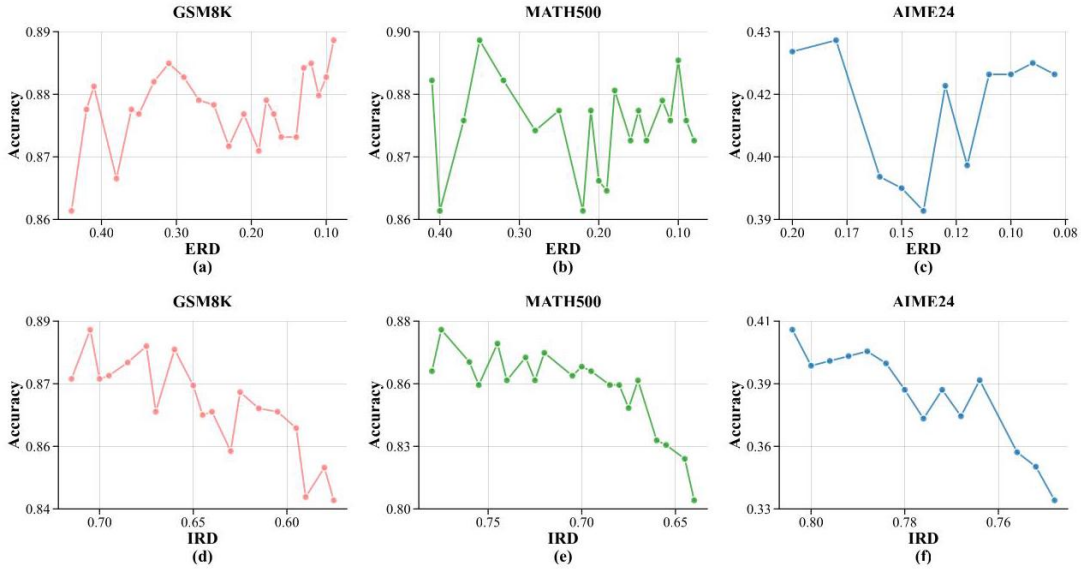

Figure 6: Impact of IRD and ERD reduction on accuracy. Reducing IRD consistently lowers accuracy, whereas penalizing external redundancy does not harm performance.

图 6:IRD 和 ERD 减少对准确率的影响。减少 IRD 会持续降低准确率，而惩罚外部冗余则不会损害性能。

To further investigate this relationship, we examine how accuracy varies as a function of ERD reduction during the first stage of training. As shown in Figure 6a, 6b, and 6c, while a slight accuracy drop of approximately 2 percentage points is observed at the beginning of the training, the accuracy eventually returns to and remains at the initial level across GSM8K, MATH500, and AIME24. Given the inherent variance in evaluation accuracy, these results indicate that reducing external redundancy does not degrade the model's reasoning capability. This suggests that the portion of the reasoning trace following the FCS contributes little to final prediction correctness, highlighting the safety and effectiveness of external redundancy removal.

为了进一步探究这一关系，我们考察了训练第一阶段中准确率随 ERD 减少的变化。如图 6a、6b 和 6c 所示，训练初期观察到约 2 个百分点的轻微准确率下降，但准确率最终回升并保持在 GSM8K、MATH500 和 AIME24 的初始水平。鉴于评估准确率的固有波动，这些结果表明减少外部冗余不会削弱模型的推理能力。这表明 FCS(First-Chain Segment) 之后的推理轨迹部分对最终预测正确性贡献甚微，凸显了外部冗余移除的安全性和有效性。

Similarly, we assess how accuracy responds to decreasing IRD in the second stage. As results shown in Figure 6d, 6e, and 6f, accuracy drops significantly as IRD decreases. This demonstrates that overly compressing the internal reasoning trace within the FCS directly harms model performance on tasks requiring fine-grained, multi-step reasoning. We hypothesize that during internal redundancy compression, the model is encouraged to eliminate intermediate reasoning steps, which increases the semantic gap between adjacent segments. This disrupts local coherence and leads to discontinuous reasoning trajectories or even CoT leaps that exceed the model's inference capability, ultimately resulting in reduced accuracy. This aligns with recent findings in (Xu et al. 2025). These observations highlight the critical importance of preserving a minimal level of internal reasoning structure and point to the need for adaptive compression strategies that balance brevity and coherence.

同样，我们评估了第二阶段中准确率对 IRD 减少的响应。如图 6d、6e 和 6f 所示，随着 IRD 的减少，准确率显著下降。这表明过度压缩 FCS 内的内部推理轨迹会直接损害模型在需要细粒度、多步推理任务上的表现。我们假设在内部冗余压缩过程中，模型被鼓励去除中间推理步骤，导致相邻片段之间的语义间隙增大。这破坏了局部连贯性，导致推理轨迹不连续，甚至出现超出模型推理能力的链式跳跃 (CoT leaps)，最终导致准确率下降。这与 (Xu et al. 2025) 的最新研究结果一致。这些观察强调了保持最低限度内部推理结构的重要性，并指出需要平衡简洁性与连贯性的自适应压缩策略。

Conclusion

结论

In this paper, we introduce a novel view of overthinking in LRMs by decomposing it into internal and external redundancy, and propose a dual-penalty reinforcement learning framework to reduce both. Experiments show that this approach significantly reduces reasoning length with minimal accuracy loss, and that external redundancy can be safely removed without harming performance. We believe these insights offer a promising direction for developing more efficient and interpretable reasoning in LRMs. References

本文提出了一种关于大型推理模型 (LRMs) 过度思考的新视角，将其分解为内部和外部冗余，并提出了双重惩罚强化学习框架以同时减少两者。实验表明，该方法显著缩短了推理长度且准确率损失极小，且外部冗余的安全移除不会影响性能。我们认为这些见解为开发更高效且可解释的 LRM 推理提供了有前景的方向。参考文献

Arora, D.; and Zanette, A. 2025. Training Language Models to Reason Efficiently. arXiv preprint arXiv:2502.04463.

Arora, D.; and Zanette, A. 2025. Training Language Models to Reason Efficiently. arXiv preprint arXiv:2502.04463.

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

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

Cheng, Z.; Chen, D.; Fu, M.; and Zhou, T. 2025. Optimizing Length Compression in Large Reasoning Models. arXiv preprint arXiv:2506.14755.

Cheng, Z.; Chen, D.; Fu, M.; and Zhou, T. 2025. Optimizing Length Compression in Large Reasoning Models. arXiv preprint arXiv:2506.14755.

Chu, T.; Zhai, Y.; Yang, J.; Tong, S.; Xie, S.; Schuurmans, D.; Le, Q. V.; Levine, S.; and Ma, Y. 2025. Sft memorizes, rl generalizes: A comparative study of foundation model post-training. arXiv preprint arXiv:2501.17161.

Chu, T.; Zhai, Y.; Yang, J.; Tong, S.; Xie, S.; Schuurmans, D.; Le, Q. V.; Levine, S.; and Ma, Y. 2025. Sft memorizes, rl generalizes: A comparative study of foundation model post-training. arXiv preprint arXiv:2501.17161.

Cobbe, K.; Kosaraju, V.; Bavarian, M.; Chen, M.; Jun, H.; Kaiser, L.; Plappert, M.; Tworek, J.; Hilton, J.; Nakano, R.; Hesse, C.; and Schulman, J. 2021. Training Verifiers to Solve Math Word Problems. arXiv preprint arXiv:2110.14168.

Cobbe, K.; Kosaraju, V.; Bavarian, M.; Chen, M.; Jun, H.; Kaiser, L.; Plappert, M.; Tworek, J.; Hilton, J.; Nakano, R.; Hesse, C.; and Schulman, J. 2021. Training Verifiers to Solve Math Word Problems. arXiv preprint arXiv:2110.14168.

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

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

Han, T.; Wang, Z.; Fang, C.; Zhao, S.; Ma, S.; and Chen, Z. 2024. Token-budget-aware llm reasoning. arXiv preprint arXiv:2412.18547.

Han, T.; Wang, Z.; Fang, C.; Zhao, S.; Ma, S.; and Chen, Z. 2024. Token-budget-aware llm reasoning. arXiv preprint arXiv:2412.18547.

Hendrycks, D.; Burns, C.; Kadavath, S.; Arora, A.; Basart, S.; Tang, E.; Song, D.; and Steinhardt, J. 2021. Measuring mathematical problem solving with the math dataset. arXiv preprint arXiv:2103.03874.

Hendrycks, D.; Burns, C.; Kadavath, S.; Arora, A.; Basart, S.; Tang, E.; Song, D.; and Steinhardt, J. 2021. Measuring mathematical problem solving with the math dataset. arXiv preprint arXiv:2103.03874.

Hou, B.; Zhang, Y.; Ji, J.; Liu, Y.; Qian, K.; Andreas, J.; and Chang, S. 2025. Thinkprune: Pruning long chain-of-thought of llms via reinforcement learning. arXiv preprint arXiv:2504.01296.

Hou, B.; Zhang, Y.; Ji, J.; Liu, Y.; Qian, K.; Andreas, J.; 和 Chang, S. 2025. Thinkprune: 通过强化学习修剪大型语言模型 (LLMs) 中的长链式思维。arXiv 预印本 arXiv:2504.01296。

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

Jaech, A.; Kalai, A.; Lerer, A.; Richardson, A.; El-Kishky, A.; Low, A.; Helyar, A.; Madry, A.; Beutel, A.; Carney, A.; 等. 2024. OpenAI O1 系统卡。arXiv 预印本 arXiv:2412.16720。

Jain, N.; Han, K.; Gu, A.; Li, W.-D.; Yan, F.; Zhang, T.; Wang, S.; Solar-Lezama, A.; Sen, K.; and Stoica, I. 2024. Livecodebench: Holistic and contamination free evaluation of large language models for code. arXiv preprint arXiv:2403.07974.

Jain, N.; Han, K.; Gu, A.; Li, W.-D.; Yan, F.; Zhang, T.; Wang, S.; Solar-Lezama, A.; Sen, K.; 和 Stoica, I. 2024. Livecodebench: 对大型语言模型代码能力的整体且无污染评估。arXiv 预印本 arXiv:2403.07974。

Liu, K.; Shen, C.; Zhang, Z.; Liu, J.; Yuan, X.; and ye, J. 2025a. Efficient Reasoning Through Suppression of Self-Affirmation Reflections in Large Reasoning Models. arXiv:2506.12353.

Liu, K.; Shen, C.; Zhang, Z.; Liu, J.; Yuan, X.; 和 Ye, J. 2025a. 通过抑制大型推理模型中的自我肯定反思实现高效推理。arXiv:2506.12353。

Liu, T.; Guo, Q.; Hu, X.; Jiayang, C.; Zhang, Y.; Qiu, X.; and Zhang, Z. 2024. Can language models learn to skip steps? arXiv preprint arXiv:2411.01855.

Liu, T.; Guo, Q.; Hu, X.; Jiayang, C.; Zhang, Y.; Qiu, X.; 和 Zhang, Z. 2024. 语言模型能学会跳过步骤吗？arXiv 预印本 arXiv:2411.01855。

Liu, W.; Zhou, R.; Deng, Y.; Huang, Y.; Liu, J.; Deng, Y.; Zhang, Y.; and He, J. 2025b. Learn to Reason Efficiently with Adaptive Length-based Reward Shaping. arXiv preprint arXiv:2505.15612.

Liu, W.; Zhou, R.; Deng, Y.; Huang, Y.; Liu, J.; Deng, Y.; Zhang, Y.; 和 He, J. 2025b. 通过基于长度的自适应奖励塑造学习高效推理。arXiv 预印本 arXiv:2505.15612。

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

Luo, H.; Shen, L.; He, H.; Wang, Y.; Liu, S.; Li, W.; Tan, N.; Cao, X.; 和 Tao, D. 2025a. O1-Pruner: 面向 O1 类推理剪枝的长度协调微调。arXiv 预印本 arXiv:2501.12570。

Luo, M.; Tan, S.; Wong, J.; Shi, X.; Tang, W. Y.; Roongta, M.; Cai, C.; Luo, J.; Zhang, T.; Li, L. E.; et al. 2025b. Deep-scaler: Surpassing o1-preview with a 1.5 b model by scaling rl. Notion Blog.

Luo, M.; Tan, S.; Wong, J.; Shi, X.; Tang, W. Y.; Roongta, M.; Cai, C.; Luo, J.; Zhang, T.; Li, L. E.; 等. 2025b. Deep-scaler: 通过扩展强化学习超越 O1-preview 的 1.5 b 模型。Notion 博客。

Ma, X.; Wan, G.; Yu, R.; Fang, G.; and Wang, X. 2025. CoT-Valve: Length-Compressible Chain-of-Thought Tuning. arXiv preprint arXiv:2502.09601.

Ma, X.; Wan, G.; Yu, R.; Fang, G.; 和 Wang, X. 2025. CoT-Valve: 可压缩长度的链式思维调优。arXiv 预印本 arXiv:2502.09601。

Meng, F.; Du, L.; Liu, Z.; Zhou, Z.; Lu, Q.; Fu, D.; Han, T.; Shi, B.; Wang, W.; He, J.; et al. 2025. Mm-eureka: Exploring the frontiers of multimodal reasoning with rule-based reinforcement learning. arXiv preprint arXiv:2503.07365.

Meng, F.; Du, L.; Liu, Z.; Zhou, Z.; Lu, Q.; Fu, D.; Han, T.; Shi, B.; Wang, W.; He, J.; 等. 2025. Mm-eureka: 通过基于规则的强化学习探索多模态推理前沿。arXiv 预印本 arXiv:2503.07365。

OpenAI. 2024. text-embedding-3-large (Embedding Model). https://platform.openai.com/docs/models/text-embedding-3-large.

OpenAI. 2024. text-embedding-3-large(嵌入模型)。https://platform.openai.com/docs/models/text-embedding-3-large。

Qiao, S.; Ou, Y.; Zhang, N.; Chen, X.; Yao, Y.; Deng, S.; Tan, C.; Huang, F.; and Chen, H. 2022. Reasoning with language model prompting: A survey. arXiv preprint arXiv:2212.09597.

Qiao, S.; Ou, Y.; Zhang, N.; Chen, X.; Yao, Y.; Deng, S.; Tan, C.; Huang, F.; 和 Chen, H. 2022. 语言模型提示推理综述。arXiv 预印本 arXiv:2212.09597。

Qu, Y.; Yang, M. Y.; Setlur, A.; Tunstall, L.; Beeching, E. E.; Salakhutdinov, R.; and Kumar, A. 2025. Optimizing test-time compute via meta reinforcement fine-tuning. arXiv preprint arXiv:2503.07572.

Qu, Y.; Yang, M. Y.; Setlur, A.; Tunstall, L.; Beeching, E. E.; Salakhutdinov, R.; 和 Kumar, A. 2025. 通过元强化微调优化测试时计算。arXiv 预印本 arXiv:2503.07572。

Rein, D.; Hou, B. L.; Stickland, A. C.; Petty, J.; Pang, R. Y.; Dirani, J.; Michael, J.; and Bowman, S. R. 2024. GPQA: A Graduate-Level Google-Proof Q&A Benchmark. In First Conference on Language Modeling.

Rein, D.; Hou, B. L.; Stickland, A. C.; Petty, J.; Pang, R. Y.; Dirani, J.; Michael, J.; 和 Bowman, S. R. 2024. GPQA: 研究生级别的 Google-proof 问答基准。首届语言建模会议论文集。

Shao, Z.; Wang, P.; Zhu, Q.; Xu, R.; Song, J.; Bi, X.; Zhang, H.; Zhang, M.; Li, Y.; Wu, Y.; et al. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. arXiv preprint arXiv:2402.03300.

Shao, Z.; Wang, P.; Zhu, Q.; Xu, R.; Song, J.; Bi, X.; Zhang, H.; Zhang, M.; Li, Y.; Wu, Y.; 等. 2024. Deepseekmath: 推动开放语言模型中数学推理的极限。arXiv 预印本 arXiv:2402.03300。

She, J.; Li, Z.; Huang, Z.; Li, Q.; Xu, P.; Li, H.; and Ho, Q. 2025. Hawkeye: Efficient reasoning with model collaboration. arXiv preprint arXiv:2504.00424.

She, J.; Li, Z.; Huang, Z.; Li, Q.; Xu, P.; Li, H.; 和 Ho, Q. 2025. Hawkeye: 通过模型协作实现高效推理。arXiv 预印本 arXiv:2504.00424。

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

Shen, Y.; Zhang, J.; Huang, J.; Shi, S.; Zhang, W.; Yan, J.; Wang, N.; Wang, K.; Liu, Z.; 和 Lian, S. 2025. Dast: 面向大型推理模型的难度自适应慢思考。arXiv 预印本 arXiv:2503.04472。

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

Sheng, G.; Zhang, C.; Ye, Z.; Wu, X.; Zhang, W.; Zhang, R.; Peng, Y.; Lin, H.; 和 Wu, C. 2024. HybridFlow: 一种灵活高效的 RLHF 框架。arXiv 预印本 arXiv:2409.19256。

Sheng, L.; Zhang, A.; Wu, Z.; Zhao, W.; Shen, C.; Zhang, Y.; Wang, X.; and Chua, T.-S. 2025. On Reasoning Strength Planning in Large Reasoning Models. arXiv preprint arXiv:2506.08390.

Sheng, L.; Zhang, A.; Wu, Z.; Zhao, W.; Shen, C.; Zhang, Y.; Wang, X.; 和 Chua, T.-S. 2025. 关于大型推理模型中的推理强度规划。arXiv 预印本 arXiv:2506.08390。

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

Sui, Y.; Chuang, Y.-N.; Wang, G.; Zhang, J.; Zhang, T.; Yuan, J.; Liu, H.; Wen, A.; Zhong, S.; Chen, H.; 等. 2025. 停止过度思考: 大型语言模型高效推理综述。arXiv 预印本 arXiv:2503.16419。

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

Team, K.; Du, A.; Gao, B.; Xing, B.; Jiang, C.; Chen, C.; Li, C.; Xiao, C.; Du, C.; Liao, C.; 等. 2025. Kimi k1.5: 利用大型语言模型扩展强化学习。arXiv 预印本 arXiv:2501.12599。

Team, Q. 2025. QwQ-32B: Embracing the Power of Reinforcement Learning.

Team, Q. 2025. QwQ-32B: 拥抱强化学习的力量。

Wang, C.; Feng, Y.; Chen, D.; Chu, Z.; Krishna, R.; and Zhou, T. 2025a. Wait, We Don't Need to" Wait"! Removing Thinking Tokens Improves Reasoning Efficiency. arXiv preprint arXiv:2506.08343.

Wang, C.; Feng, Y.; Chen, D.; Chu, Z.; Krishna, R.; 和 Zhou, T. 2025a. 等等，我们不需要“等待”！移除思考标记提升推理效率。arXiv 预印本 arXiv:2506.08343。

Wang, Y.; Liu, Q.; Xu, J.; Liang, T.; Chen, X.; He, Z.; Song, L.; Yu, D.; Li, J.; Zhang, Z.; et al. 2025b. Thoughts Are All Over the Place: On the Underthinking of o1-Like LLMs. arXiv preprint arXiv:2501.18585.

Wang, Y.; Liu, Q.; Xu, J.; Liang, T.; Chen, X.; He, Z.; Song, L.; Yu, D.; Li, J.; Zhang, Z.; 等. 2025b. 思绪纷乱: 关于 o1 类大型语言模型的欠思考问题。arXiv 预印本 arXiv:2501.18585。

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

Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Xia, F.; Chi, E.; Le, Q. V.; Zhou, D.; 等. 2022. 链式思维提示激发大型语言模型的推理能力。神经信息处理系统进展, 35: 24824-24837。

Xie, Y.; Ma, Y.; Lan, S.; Yuille, A.; Xiao, J.; and Wei, C. 2025. Play to Generalize: Learning to Reason Through Game Play. arXiv preprint arXiv:2506.08011.

Xie, Y.; Ma, Y.; Lan, S.; Yuille, A.; Xiao, J.; 和 Wei, C. 2025. 通过游戏学习泛化推理能力。arXiv 预印本 arXiv:2506.08011。

Xu, H.; Yan, Y.; Shen, Y.; Zhang, W.; Hou, G.; Jiang, S.; Song, K.; Lu, W.; Xiao, J.; and Zhuang, Y. 2025. Mind the Gap: Bridging Thought Leap for Improved Chain-of-Thought Tuning. arXiv preprint arXiv:2505.14684.

Xu, H.; Yan, Y.; Shen, Y.; Zhang, W.; Hou, G.; Jiang, S.; Song, K.; Lu, W.; Xiao, J.; 和 Zhuang, Y. 2025. 注意差距: 弥合思维跳跃以改进链式思维调优。arXiv 预印本 arXiv:2505.14684。

Yang, J.; Lin, K.; and Yu, X. 2025. Think when you need: Self-adaptive chain-of-thought learning. arXiv preprint arXiv:2504.03234.

Yang, J.; Lin, K.; 和 Yu, X. 2025. 需要时思考: 自适应链式思维学习。arXiv 预印本 arXiv:2504.03234。