EffiReason-Bench: A Unified Benchmark for Evaluating and Advancing Efficient Reasoning in Large Language Models

Junquan Huang[1,*], Haotian Wu[2,*], Yubo Gao[1,*], Yibo Yan[1,3],

Junyan Zhang[1], Yonghua Hei[1], Song Dai[1,3], Jie Zhang[2], Puay Siew Tan [4], Xuming Hu[1,3,†]

[1]Hong Kong University of Science and Technology (Guangzhou),
[2]Nanyang Technological University,
[3]The Hong Kong University of Science and Technology,
[4]Singapore Institute of Manufacturing Technology, A*STAR,
xuminghu@hkust-gz.edu.cn

Abstract

Large language models (LLMs) with Chain-of-Thought (CoT) prompting achieve strong reasoning but frequently generate unnecessarily verbose explanations, increasing costs and reducing accuracy. Despite the proliferation of efficiency-oriented approaches, fragmented evaluation practices hinder systematic comparison and obscure which strategies are effective under varying conditions. We present **EffiReason-Bench**, the first unified benchmark enabling rigorous cross-paradigm evaluation of efficient reasoning methods organized into three categories: *Reasoning Blueprints*, *Dynamic Execution*, and *Post-hoc Refinement*. To enable comprehensive step-by-step reasoning evaluation, we construct verified CoT annotations for CommonsenseQA and LogiQA through a rigorous pipeline enforcing standardized reasoning structures, comprehensive option-wise analysis, and human verification. We evaluate 7 methods across 6 LLMs (1B-70B) on 4 reasoning datasets, covering mathematical, commonsense, and logical reasoning, and propose the E^3**-Score**, a principled metric inspired by economic trade-off modeling that provides smooth, stable evaluation without the discontinuities and over-reliance on heuristic dependencies plaguing prior measures. Experimental results demonstrate that no single method universally dominates: optimal strategies depend sensitively on backbone scale, task complexity, and architecture.

1 Introduction

Large Language Models (LLMs) have shown strong capabilities in complex reasoning (Chen et al., 2025; Yan et al., 2024), largely driven by the Chain-of-Thought (CoT) paradigm (Wei et al., 2022), where LLMs generate explicit step-by-step rationales to solve problems. This approach improves performance on logic-intensive tasks by providing structured intermediate steps (Wang et al., 2022; Yao et al., 2023; Besta et al., 2024). However, it often leads to the "overthinking phenomenon" (Chen et al., 2024; Team et al., 2025; Feng et al., 2025), where unnecessarily long and redundant reasoning chains are produced even for simple tasks. Such verbosity not only increases computational cost but can also accumulate errors and reduce accuracy (Sui et al., 2025).

To mitigate these challenges, a growing body of research on *efficient reasoning* (Sui et al., 2025; Wang et al., 2025a; Yue et al., 2025) has proposed diverse strategies. These methods intervene at different stages of the reasoning process. Some approaches act before reasoning, serving as *reasoning blueprints* that guide the LLM through reinforcement learning with length-aware rewards (Shen et al., 2025b; Luo et al., 2025) or by using concise prompts (Xu et al., 2025a). Others modify the generation process itself, applying *dynamic execution* techniques such as reasoning in continuous space (Zhang et al., 2025c) or restructuring decoding (Ning et al., 2023). A third category, *post-hoc refinement*, operates after generation, pruning, or compressing verbose outputs (Xia et al., 2025). Though promising, the lack of a unified evaluation framework makes systematic comparison difficult. Key questions remain open: *Which strategies provide the best accuracy–efficiency trade-off? How do they scale with LLM backbone size? Do their benefits transfer across reasoning domains?*

Existing benchmarks only partially address these questions. Some focus narrowly on specific method families, such as Sys2Bench (Parashar et al., 2025) for inference-time search strategies, or on compression methods like quantization (Liu et al., 2025) and pruning (Zhang et al., 2025b). Others analyze isolated phenomena, such as the effect of reasoning step length (Jin et al., 2024) or the tendency to overthink in agentic tasks (Cuadron et al., 2025). Despite their contributions, these efforts remain

---

fragmented. Moreover, current evaluation metrics for efficiency-effectiveness trade-off, such as Accuracy per Computation Unit (ACU) (Ma et al., 2025) and Accuracy–Efficiency Score (AES) (Luo et al., 2025) are limited: the former misrepresents improvements near strong baselines and the latter introduces discontinuities, and depends on excessive heuristic hyperparameters. As a result, both fair comparison and principled assessment of efficiency-effectiveness balance for efficient reasoning remain open challenges.

To fill these gaps, we introduce **EffiReason-Bench**, the first comprehensive benchmark for evaluating efficient reasoning methods across paradigms. EffiReason-Bench systematically compares seven representative methods under a consistent setup, spanning six open-source LLM backbones of varying scales and four datasets covering mathematics (*GSM8K*, *MATH500*), commonsense (*CommonsenseQA*), and logic (*LogiQA*) reasoning. Considering that the datasets CommonsenseQA and LogiQA do not provide the solution processes, to enable transparent evaluation of step-by-step reasoning processes beyond answer accuracy alone, we construct high-quality CoT annotations for CommonsenseQA and LogiQA through a rigorous three-stage pipeline: standardized reasoning structures, explicit comparative analysis across all options ensuring strict alignment with ground-truth answers, and dual human verification to eliminate logical inconsistencies.

To quantify the trade-off between efficiency and effectiveness, we further propose the **E^3-Score** (Efficiency–Effectiveness Equilibrium Score), a smooth and stable metric inspired by economic trade-off modeling. E^3-Score avoids discontinuities and heuristic tuning, emphasizes accuracy gains near strong baselines, and penalizes simultaneous degradation of efficiency and accuracy. Together, EffiReason-Bench and E^3-Score provide the first rigorous foundation for cross-paradigm evaluation of efficient reasoning.

Our contributions can be summarized as follows:

❶ We propose **EffiReason-Bench**, the first benchmark to enable systematic and fair comparison of efficient reasoning methods across different paradigms, backbone scales, and reasoning domains.

❷ We introduce the **E^3-Score**, a principled metric that provides smooth and reliable evaluation of efficiency–effectiveness trade-offs, addressing the limitations of prior measures.
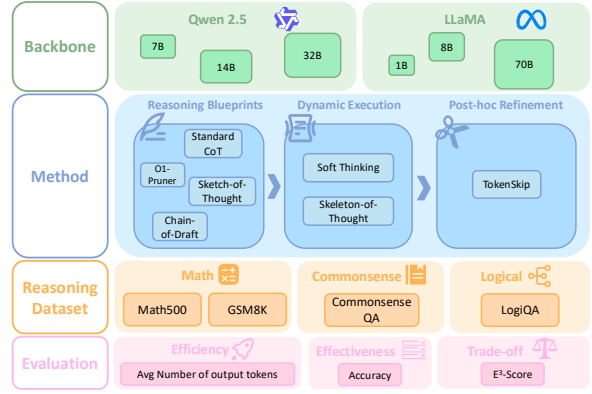

Figure 1: Overview of EffiReason-Bench, which compares efficient reasoning methods across diverse paradigms, backbones, and datasets.

❸ We conduct extensive experiments with 7 methods, 6 backbones, and 4 datasets, revealing that optimal efficiency strategies depend on both backbones size and task complexity, and we will release all implementations for reproducibility.

2 Related Work

The Overthinking Phenomenon The CoT paradigm (Wei et al., 2022) enables LLMs to generate explicit intermediate reasoning steps, which has led to significant gains on complex reasoning tasks. However, subsequent studies have identified the tendency of models to produce excessively long and redundant reasoning chains, even for trivial problems (Han et al., 2025; Shen et al., 2025c; Xu et al., 2025b). This "overthinking" increases computational cost and inference latency, and in some cases reduces accuracy due to the accumulation of errors. These observations have motivated a line of work on improving the efficiency of reasoning.

Paradigms of Efficient Reasoning Research on efficient reasoning can be broadly grouped into three paradigms depending on when efficiency interventions are applied. **Reasoning blueprints** modify the model's behavior before inference, for example by fine-tuning with length-sensitive rewards (Shen et al., 2025b; Luo et al., 2025; Team et al., 2025; Yeo et al., 2025) or by designing concise prompts (Xu et al., 2025a; Wang et al., 2025b; Zhang et al., 2025a). **Dynamic execution** methods adjust the reasoning process during inference, such as reasoning in latent space (Hao et al., 2024; Cheng and Van Durme, 2024; Shen et al., 2025c,a; Xu et al., 2025b; Zhang et al., 2025c) or restructuring decoding with intermediate skeletons (Ning et al., 2023). **Post-hoc refinement** tech-

niques operate after generation, e.g. pruning redundant steps (Xia et al., 2025). These lines of work highlight complementary approaches to reducing reasoning cost, yet their effectiveness varies substantially across models and tasks.

Benchmarks for Reasoning Efficiency Several benchmarks have been proposed to evaluate reasoning or efficiency, but they are limited in scope. Sys2Bench (Parashar et al., 2025) provides systematic evaluation for inference-time search strategies, while other studies focus on model-level compression such as quantization and pruning (Zhang et al., 2025b; Liu et al., 2025). Additional efforts investigate isolated phenomena, including the effect of reasoning length (Jin et al., 2024), overthinking in agentic settings (Cuadron et al., 2025), or latent-space reasoning (Hagendorff and Fabi, 2025). Although valuable, these works remain siloed, making it difficult to compare across paradigms or to establish general principles for efficient reasoning. Nonetheless, even with available benchmarks, achieving a fair comparison of efficiency–effectiveness balance crucially depends on the choice of evaluation metric. Prior measures exhibit clear limitations. *Accuracy per Computation Unit (ACU)* (Ma et al., 2025) normalizes accuracy by computational cost, but it undervalues small yet meaningful gains in high-accuracy regimes, where marginal improvements are most difficult to achieve. *Accuracy–Efficiency Score (AES)* (Luo et al., 2025) combines accuracy and efficiency through a piecewise formulation, but it relies on a series of manually chosen thresholds and heuristic hyperparameters, making results sensitive and often unstable across settings.

Our study differs from existing efforts in two key aspects. First, we establish **EffiReason-Bench**, the first benchmark that systematically compares efficient reasoning methods across paradigms, model scales, and reasoning domains under a unified experimental setup. Second, we propose the **E^3-Score**, a principled metric that provides smooth and fair evaluation of efficiency–effectiveness trade-offs, addressing the limitations of prior measures. Together, these contributions offer a rigorous and reproducible foundation for advancing research on efficient reasoning.

3 Dataset, Task, and Pipeline Setup

EffiReason-Bench addresses three key limitations in current evaluation practices. First, heterogeneous backbones and training strategies across prior studies make it challenging to isolate the specific contributions of efficiency methods from inherent LLM capabilities. Second, cross-domain generalization remains insufficiently explored, as most evaluations concentrate on specific task families. Third, existing metrics measuring the efficiency–effectiveness trade-off, suffer from discontinuities and heuristic dependencies, limiting their reliability. To address these challenges, EffiReason-Bench provides a unified evaluation platform enabling systematic *cross-paradigm*, *cross-backbone*, and *cross-domain* assessment.

3.1 Datasets

To ensure coverage of diverse reasoning types, we select four established datasets spanning mathematical, commonsense, and logical reasoning: **GSM8K** (Cobbe et al., 2021), a collection of grade school math word problems requiring multi-step arithmetic; **MATH500**, a curated subset of the MATH dataset (Hendrycks et al., 2021) comprising 500 competition-level problems designed to balance difficulty with computational feasibility for large-scale benchmarking; **CommonsenseQA** (Talmor et al., 2018), a multiple-choice dataset evaluating everyday commonsense reasoning; and **LogiQA** (Liu et al., 2020), a benchmark derived from national logic examinations requiring conditional and deductive reasoning.

Since CommonsenseQA and LogiQA provide only final answers without explanatory reasoning, we construct verified CoT solutions to enable comprehensive evaluation of step-by-step reasoning. To address the logical gaps and contradictions against ground-truth commonly found in auto-generated CoT data, we propose a rigorous construction pipeline with three core components. **First**, we adopt a standardized four-stage reasoning structure: premise identification, relationship formalization, condition evaluation, and option mapping. **Second**, we enforce explicit comparative analysis across all options: each solution must explain why the correct answer holds and why distractors fail within a unified logical framework, ensuring strict alignment between reasoning chains and ground-truth answers. **Third**, all annotations undergo dual human verification to eliminate logical inconsistencies and unsupported inferences. This pipeline yields high-quality CoT annotations that enable transparent cross-domain evaluation of reasoning processes beyond answer accuracy alone.

3.2 Efficient Reasoning Methods

We include 7 representative methods that span the three major paradigms of efficient reasoning, with Standard CoT serving as the reference substance.

Reasoning blueprints (Pre-process). *O1-Pruner* fine-tunes models using a length-aware reward function, encouraging concise reasoning while preserving correctness. *Sketch-of-Thought (SoT)* introduces a lightweight router that maps inputs to high-level reasoning patterns ("sketches"), guiding the model to follow compressed reasoning templates. *Chain-of-Draft* employs prompt instructions that restrict intermediate steps to very short drafts, explicitly constraining verbosity.

Dynamic execution (In-process). *Soft Thinking* enables reasoning in a continuous space by generating "concept tokens," which represent probability-weighted mixtures of token embeddings. *Skeleton-of-Thought (SloT)* restructures decoding into a two-stage process: first generating a high-level outline (skeleton) and then expanding details for each component, potentially reducing redundancy through parallel generation.

Post-hoc refinement (Post-process). *TokenSkip* operates on completed reasoning chains by removing tokens with low semantic contribution to the final answer, producing shorter rationales.

3.3 Backbone Models

All methods are evaluated on 6 widely used open-source LLM backbones to ensure reproducibility and fair comparison across scales. We adopt the Qwen2.5-Instruct series (7B, 14B, 32B) and the LLaMA-Instruct series (1B, 8B, 70B), which enables controlled analysis of how efficiency strategies interact with model capacity.

3.4 Training Criteria

3.4.1 Main Experiments

We consider two regimes. In the *Train-Free* setting, models perform direct inference without any parameter updates. In the *Train-based* setting, models are first trained on the full training set and then evaluated by inference. For all datasets and backbones, the evaluation prompt consists of a fixed instruction header followed by the test instance. Answer formatting is standardized: the prediction must appear on a single line beginning with "Final Answer:"; multiple-choice tasks require a single option letter (A/B/C/D); open-ended math expects a numeric or short span. This unified protocol is consistently applied across all methods and both training regimes to ensure fair comparison.

3.4.2 Few-shot Setting.

Few-shot analysis is reported only when explicitly stated. We evaluate $k \in \{1, 4, 8, 12\}$ shots. *Exemplar construction and reuse.* Few-shot exemplars are drafted by GPT-4o and human-audited for clarity and correctness. For each dataset and each k, we finalize exemplar lists once with a fixed global seed and reuse them verbatim across methods, backbones, and both regimes to eliminate sampling variance. Each exemplar follows Question → Reasoning (numbered steps) → Final Answer. For mathematical tasks, the reasoning enumerates key intermediate arithmetic/transformations; for commonsense/logical tasks, it adopts [Premises] → [Inference rule] → [Conclusion] and, for multiple-choice data, includes a brief justification for distractors. Answer formatting follows the same rule as in the main experiments.

Variants. Unless otherwise noted, exemplars are in-domain (same dataset as the evaluation task). We also probe a cross-domain variant (exemplars from a source dataset, evaluation on a target dataset from a different reasoning domain). To assess robustness, we introduce noisy exemplars via controlled perturbations covering: (i) reasoning-step noise (deletion/permutation/repetition), (ii) semantic noise (e.g., mild numeric or connector perturbations). Details are provided in Appendix A.

3.5 Evaluation Strategy

We evaluate models on both *effectiveness* (accuracy) and *efficiency* (average output tokens). To evaluate the efficiency-effectiveness trade-off, several prior metrics attempt to combine these two dimensions, but each has limitations. **ACU** linearly scales accuracy by computation, undervaluing small but crucial improvements in high-accuracy regimes. **AES** considers relative gains in accuracy and tokens, but it is piecewise and relies on heuristic hyperparameters, leading to instability.

To overcome these issues, we propose the **Efficiency–Effectiveness Equilibrium Score (E^3-Score)**, a smooth metric inspired by the Constant Elasticity of Substitution (CES) formulation (Mc-

---

[1]OOM errors occurred during O1-Pruner's required, memory-intensive fine-tuning phase, which exceeded the 4x 80G A800 capacity for 32B/70B models, even though we set batch size = 1

Table 1: Accuracy (%) and average output tokens comparison of reasoning methods on *GSM8K* dataset. Best performance is indicated in bold, and runner-up is underlined. Train-free methods are highlighted in green, while train-based methods are highlighted in red. "OOM" is the abbreviation for out-of-memory [1].

Category	Methods	Qwen 2.5 (Instruct)									LLaMA(Instruct)								
		7B			14B			32B			1B			8B			70B		
		Acc ↑	Tokens ↓	E³ ↑	Acc ↑	Tokens ↓	E³ ↑	Acc ↑	Tokens ↓	E³ ↑	Acc ↑	Tokens ↓	E³ ↑	Acc ↑	Tokens ↓	E³ ↑	Acc ↑	Tokens ↓	E³ ↑
Reference Substance	CoT	90.60	292.86	-	93.63	278.61	-	94.39	277.36	-	22.44	227.37	-	78.70	271.57	-	95.68	240.65	-
Reasoning Blueprints	CoD	64.82	**45.94**	0.21	79.23	**48.67**	0.28	86.35	**56.09**	0.40	1.82	**138.96**	0.25	58.45	140.66	0.46	88.63	**80.98**	0.37
	SoT	72.02	103.81	0.29	89.46	89.28	0.61	89.69	101.81	0.54	12.89	300.46	0.68	61.03	**125.14**	0.51	92.42	84.81	0.57
	O1-Pruner	79.15	206.12	0.42	86.58	233.79	0.46	OOM	OOM	OOM	36.39	170.94	1.44	73.24	231.61	0.80	OOM	OOM	OOM
Dynamic Execution	Soft Thinking	90.67	287.09	1.01	94.39	272.54	1.14	94.47	276.39	1.01	35.10	208.84	1.20	84.31	248.25	1.36	95.53	238.12	0.97
	SloT	70.05	109.78	0.27	85.67	134.29	0.43	86.88	127.25	0.41	4.55	209.63	0.48	53.68	324.74	0.36	93.86	206.63	0.70
Post-hoc Refinement	TokenSkip	88.63	214.91	0.84	93.10	219.10	0.93	95.38	224.93	1.23	43.67	165.48	1.54	81.88	149.88	1.31	95.68	172.69	1.01

Fadden, 1963), which has long been used to model trade-offs with controlled substitutability. Given baseline accuracy A_0 and tokens T_0, and method values A, T, we define

$$r_{\text{acc}} = \left(\tfrac{A}{1-A}\right)/\left(\tfrac{A_0}{1-A_0}\right), \quad r_{\text{tok}} = \frac{T_0}{T}. \quad (1)$$

The E³-Score is then given by

$$\mathrm{E}^3(A, T) = \left(w \cdot r_{\text{acc}}^{\rho} + (1 - w) \cdot r_{\text{tok}}^{\rho} \right)^{\frac{1}{\rho}}, \quad (2)$$

where the weight w calibrates the *relative importance* of accuracy versus efficiency to task difficulty; setting $w = A_0$ (the baseline accuracy) makes accuracy more prominent when the baseline is already strong and avoids extra tuning. The parameter ρ governs *substitutability*: with $\rho < 0$ the two factors behave as complements so that efficiency gains cannot fully offset accuracy losses. We adopt $\rho = -1$ (a weighted harmonic mean) as the default, providing a conservative balance that penalizes asymmetric improvements and prevents excessive token savings from overshadowing accuracy degradation; without this knob (e.g., implicitly fixing $\rho = 0$ or 1) the metric becomes too permissive and less reliable. This CES-inspired aggregation ensures a smooth trade-off: accuracy gains at high baselines are emphasized, efficiency is rewarded but cannot fully offset accuracy drops, and two-way degradation is strongly penalized.

Theorem 1 (Global sensitivity of E_3 to ρ). *Let $a = r_{acc} > 0$, $b = r_{tok} > 0$, and $w \in (0, 1)$. Define $\Delta := |\log(a/b)|$. For any $\rho_1, \rho_2 \in \mathbb{R}$ (interpreting $\rho = 0$ by continuity of the power mean),*

$$\left| \log E_3(A, T; \rho_1) - \log E_3(A, T; \rho_2) \right| \le \frac{\Delta^2}{8} |\rho_1 - \rho_2|. \quad (3)$$

Equivalently, $\left| \partial_\rho \log E_3(A, T; \rho) \right| \le \Delta^2/8$ for all ρ, and

$$e^{-\frac{\Delta^2}{8} |\rho_1 - \rho_2|} \le \frac{E_3(A, T; \rho_1)}{E_3(A, T; \rho_2)} \le e^{\frac{\Delta^2}{8} |\rho_1 - \rho_2|}. \quad (4)$$

The bound is independent of w.

4 Experiments and Analysis

4.1 Mathematical Reasoning Tasks

We evaluate performance on two mathematical reasoning datasets: GSM8K and MATH500. Tables 1 and 2 reveal that efficiency strategy effectiveness is highly contingent on task complexity, backbone scale, and architectural choices.

Within the **Reasoning Blueprints category**, we observe a fundamental accuracy-efficiency trade-off. The train-based O1-Pruner consistently prioritizes accuracy preservation, often surpassing the CoT baseline on MATH500 for LLaMA 1B and 8B, while achieving more modest compression ratios. Conversely, train-free methods such as CoD and SoT attain maximum token compression but at significant accuracy cost, demonstrating the inverse tendency. This pattern suggests that learned pruning strategies better preserve reasoning integrity than heuristic compression approaches.

The **Dynamic Execution** paradigm exhibits striking performance divergence. Soft Thinking maintains near-identical accuracy to the CoT baseline on GSM8K and consistently matches or exceeds it on MATH500, demonstrating remarkable robustness across backbone scales. In contrast, Skeleton-of-Thought induces substantial accuracy degradation in most configurations, particularly on the more challenging MATH500 benchmark. This disparity indicates that adaptive token generation mechanisms prove more reliable than parallel decoding strategies for complex reasoning tasks.

Post-hoc Refinement through TokenSkip reveals a critical backbone-dependent phenomenon. While the method achieves balanced accuracy-efficiency trade-offs on GSM8K across most of the backbones, its behavior on MATH500 diverges dramatically by architecture. TokenSkip induces catastrophic accuracy collapse on Qwen, with degradations ranging from 34.2% to 37.0 %, yet proves remarkably effective on LLaMA backbones, im-

Table 2: Accuracy (%) and average output tokens comparison of reasoning methods on *MATH 500* dataset.

Category	Methods	Qwen 2.5 (Instruct)									LLaMA (Instruct)								
		7B			14B			32B			1B			8B			70B		
		Acc ↑	Tokens ↓	E³ ↑	Acc ↑	Tokens ↓	E³ ↑	Acc ↑	Tokens ↓	E³ ↑	Acc ↑	Tokens ↓	E³ ↑	Acc ↑	Tokens ↓	E³ ↑	Acc ↑	Tokens ↓	E³ ↑
Reference Substance	CoT	75.00	579.00	–	77.00	558.31	–	**81.40**	544.64	–	13.00	701.48	–	33.20	739.55	–	75.20	582.59	–
Reasoning Blueprints	CoD	14.80	**145.34**	0.08	45.00	**71.31**	0.31	55.40	**93.40**	0.34	2.80	**246.15**	1.02	23.40	**363.00**	**1.15**	50.00	**249.99**	0.42
	SoT	56.80	328.58	0.54	62.60	177.73	**0.62**	68.60	192.79	**0.59**	8.80	556.54	1.12	26.20	486.41	1.11	64.80	277.70	0.74
	O1-Pruner	59.20	455.38	0.57	64.00	514.32	0.60	OOM	OOM	OOM	17.40	364.58	**1.84**	37.00	464.14	1.43	OOM	OOM	OOM
Dynamic Execution	Soft Thinking	75.40	612.63	**1.00**	80.20	582.17	**1.14**	81.00	555.23	0.98	24.00	704.77	1.07	**48.00**	660.48	1.29	73.60	555.74	**0.95**
	SloT	46.20	157.77	0.37	57.00	184.93	0.49	58.80	172.56	0.39	5.40	579.19	0.94	22.60	587.14	0.91	59.00	312.02	0.58
Post-hoc Refinement	TokenSkip	40.80	314.40	0.29	42.20	318.73	0.27	44.40	314.16	0.22	**27.60**	404.98	1.81	46.00	355.32	1.94	66.60	372.25	0.77

proving accuracy on 1B and 8B variants while incurring only an 8.6 % loss on the 70B variants. This architectural sensitivity suggests that TokenSkip's pruning criteria align with LLaMA's internal reasoning representations but fundamentally misalign with Qwen's computational structure, highlighting the importance of architecture-aware optimization.

4.2 Commonsense Reasoning Tasks

On the CommonsenseQA dataset (Table 3), we observe a stark contrast to the mathematical reasoning tasks. Commonsense reasoning demonstrates high robustness to token compression, with many efficiency methods maintaining stable or even superior accuracy while drastically reducing token counts.

Reasoning Blueprints exhibit extreme efficiency. The train-free CoD and SoT methods successfully compress token counts by over 90% (e.g., Qwen 7B's 286.47 tokens reduced to 7.59 by CoD). Under this extreme compression, CoD largely maintains accuracy on Qwen (even slightly improving it on 7B and 14B) but shows a performance drop on LLaMA. The train-based O1-Pruner again demonstrates its ability to enhance weaker baselines, significantly improving LLaMA 8B's accuracy from 62.72% to 71.28%.

Among **Dynamic Execution** methods, Soft Thinking again proves its robustness. Its accuracy and token count remain highly consistent with the CoT baseline across all backbones (e.g., on Qwen 7B and LLaMA 70B) and deliver a notable improvement on LLaMA 8B (62.72% to 69.47%). Conversely, SloT performs poorly on this task. While it significantly reduces tokens, it also incurs a consistent accuracy drop across most of the backbones, culminating in a performance collapse on LLaMA 8B (62.72% to 25.50%).

Post-hoc Refinement: TokenSkip emerges as an exceptionally safe and effective strategy for commonsense reasoning. While moderately reducing tokens, it either maintains or improves accuracy on all backbones. It shows positive effects on both

mid-performance baselines (like LLaMA 8B, from 62.72% to 73.58%) and high-performance ones (like Qwen 32B, from 84.20% to 85.60%). This performance stands in sharp contrast to its catastrophic impact on Qwen in the MATH500 task.

4.3 Logical Reasoning Tasks

Results on the LogiQA dataset presented in Table 4 reveal the heightened sensitivity of logical reasoning to compression strategies. In contrast to commonsense reasoning, logical inference demands greater preservation of reasoning step integrity, rendering aggressive compression approaches substantially more precarious. The **Reasoning Blueprints** category yields mixed results. The train-free CoD method, while achieving the most extreme token compression, consistently degrades accuracy across all backbones. SoT is similarly aggressive in compression but exhibits unstable performance: it marginally improves accuracy on Qwen 7B and 32B but harms performance on LLaMA backbones and other Qwen. In contrast, the train-based O1-Pruner again proves highly effective at improving weak backbones, substantially lifting LLaMA 1B accuracy from 14.29% to 24.27% and LLaMA 8B from 33.95% to 43.16%.

Within the **Dynamic Execution** category, Soft Thinking is a standout performer for accuracy, achieving performance gains across all backbones and notably on LLaMA 8B (33.95% →44.09%) and Qwen 14B (56.37% → 60.37%). This accuracy enhancement, however, comes at the cost of minimal efficiency gains, as its token consumption is nearly identical to the CoT baseline. Conversely, SloT performs poorly on this task. Despite a neutral result on Qwen 14B, it degrades accuracy on almost all other backbones and causes a performance collapse on LLaMA 1B.

Post-hoc Refinement via TokenSkip emerges as a balanced and robust strategy for logical reasoning. It achieves moderate token compression while universally maintaining or improving accu-

Table 3: Accuracy (%) and average output tokens comparison of reasoning methods on *Commonsense* dataset.

| Category | Methods | Qwen 2.5 (Instruct) | | | | | | | | | LLaMA (Instruct) | | | | | | | | |
| | | 7B | | | 14B | | | 32B | | | 1B | | | 8B | | | 70B | | |
		Acc ↑	Tokens ↓	E³ ↑	Acc ↑	Tokens ↓	E³ ↑	Acc ↑	Tokens ↓	E³ ↑	Acc ↑	Tokens ↓	E³ ↑	Acc ↑	Tokens ↓	E³ ↑	Acc ↑	Tokens ↓	E³ ↑
Reference Substance	CoT	80.00	286.47	-	81.23	204.10	-	84.20	220.43	-	18.52	216.88	-	62.72	249.77	-	83.37	332.19	-
Reasoning Blueprints	CoD	82.47	7.59	1.46	81.40	7.58	1.23	83.79	10.06	1.14	4.61	117.40	0.76	56.38	56.25	1.11	80.33	10.14	0.97
	SoT	78.35	15.72	1.12	80.99	26.62	1.18	84.53	28.08	1.19	12.84	245.68	0.83	55.31	179.47	0.89	80.25	36.27	1.14
	O1-Pruner	78.68	117.94	1.05	80.66	198.85	0.97	OOM	OOM	OOM	39.75	184.61	1.32	71.28	100.87	1.74	OOM	OOM	OOM
Dynamic Execution	Soft Thinking	80.00	273.83	1.01	79.84	180.40	0.95	83.62	205.00	0.97	23.87	157.51	1.38	69.47	212.05	1.28	83.54	318.85	1.02
	SloT	72.35	48.13	0.80	77.04	119.52	0.86	76.95	132.36	0.69	4.28	163.31	0.64	25.50	236.54	0.29	79.51	180.51	0.86
Post-hoc Refinement	TokenSkip	79.34	209.53	1.02	81.98	147.99	1.10	85.60	145.66	1.16	53.25	134.61	1.84	73.58	159.44	1.62	83.21	241.30	1.04

Table 4: Accuracy (%) and average output tokens comparison of reasoning methods on *LogiQA* dataset.

| Category | Methods | Qwen 2.5(Instruct) | | | | | | | | | LLaMA(Instruct) | | | | | | | | |
| | | 7B | | | 14B | | | 32B | | | 1B | | | 8B | | | 70B | | |
		Acc ↑	Tokens ↓	E³ ↑	Acc ↑	Tokens ↓	E³ ↑	Acc ↑	Tokens ↓	E³ ↑	Acc ↑	Tokens ↓	E³ ↑	Acc ↑	Tokens ↓	E³ ↑	Acc ↑	Tokens ↓	E³ ↑
Reference Substance	CoT	51.92	471.95	–	56.37	422.58	–	63.44	399.54	–	14.29	461.17	–	33.95	534.32	–	63.13	530.36	–
Reasoning Blueprints	CoD	49.77	27.45	1.68	55.76	20.35	1.67	60.52	51.35	1.31	8.60	105.99	2.22	30.11	280.66	1.33	53.00	35.66	1.02
	SoT	53.46	83.58	1.74	55.61	85.50	1.49	63.90	96.43	1.41	11.98	673.31	0.70	32.26	510.20	1.00	58.22	142.67	1.14
	O1-Pruner	51.92	417.19	1.06	55.61	463.90	0.94	OOM	OOM	OOM	24.27	320.35	1.49	43.16	303.59	1.65	OOM	OOM	OOM
Dynamic Execution	Soft Thinking	55.15	475.83	1.06	60.37	405.13	1.12	65.44	393.80	1.06	17.67	371.33	1.25	44.09	426.06	1.34	63.13	506.40	1.02
	SloT	46.54	90.56	1.36	56.84	145.03	1.42	62.06	156.49	1.23	7.07	231.50	1.35	25.50	376.36	1.03	58.06	238.99	1.06
Post-hoc Refinement	TokenSkip	51.61	365.02	1.11	58.53	305.96	1.20	66.05	294.23	1.20	27.19	262.90	1.81	44.09	278.12	1.77	63.59	357.94	1.15

racy across all backbones and scales. Its benefits are particularly pronounced on the LLaMA backbone, where it dramatically improves the LLaMA 1B (14.29% → 27.19%) and LLaMA 8B (33.95% → 44.09%) baselines, and it also delivers a strong gain on Qwen 32B (63.44% → 66.05%).

4.4 Few-shot Learning Setting

In the few-shot experiments, we compare four representative methods: SloT, SoT, CoT, and CoD. Other approaches, such as TokenSkip and O1-Pruner, were excluded, as they did not learn effective patterns under limited supervision. Results are reported under 1-, 4-, 8-, and 12-shot conditions.

Clean few-shot evaluation. Key obversions are shown as follows: **First**, as illustrated by the solid lines, CoT serves as a high-accuracy, high-token baseline across most of the tasks. **Second**, the most significant finding is the superior scalability of SloT (solid green line). On MATH500, GSM8K, and CommonsenQA, SloT's accuracy improves with more shots, even approaching CoT. On Commonsense, SloT's accuracy surpasses that of CoT at 4, 8, or 12 shots. A potential mechanism is that SloT's structured nature (skeleton-then-expansion) allows it to effectively learn the structural patterns of reasoning from exemplars; thus, more shots lead to a more robust learned reasoning structure. **Third**, conversely, the Reasoning Blueprint methods (CoD and SoT) exhibit rigidity. While CoD (solid orange line) and SoT (solid red line) achieve extreme token compression, their accuracy shows more instability than CoT and SloT across all tasks

and shows minimal improvement on GSM8K with more shots. This suggests these methods primarily learn a surface-level pattern ("be short"), and this rigid compression inhibits their ability to learn complex reasoning from additional exemplars.

Noise injection setting. We further examine robustness by injecting noise into few-shot exemplars. In most cases, test methods degrade under noisy demonstrations, but the extent differs. SoT and CoD are most vulnerable, especially on mathematically intensive tasks, while SloT shows comparatively stable performance. CoT remains the most consistent across domains, with minimal losses and sometimes even shorter outputs. This suggests that blueprint methods depend heavily on surface-level exemplar patterns, so perturbations in reasoning steps, semantics, or formatting directly disrupt their effectiveness. In contrast, SloT extracts higher-level structural organization, which is less sensitive to local corruption, and CoT's unconstrained generation helps maintain baseline stability. *Implication.* In practice, exemplar quality cannot always be guaranteed. Therefore, efficiency methods must consider robustness: structure-aware decoding (SloT) offers a safer choice under noisy supervision, while blueprint compression strategies need additional safeguards to remain reliable.

4.5 Cross-domain Reasoning Setting

We further evaluate cross-domain transfer in the few-shot setting, where exemplars are drawn from a source domain and evaluated on a target task from a different domain. The results, shown in Figure
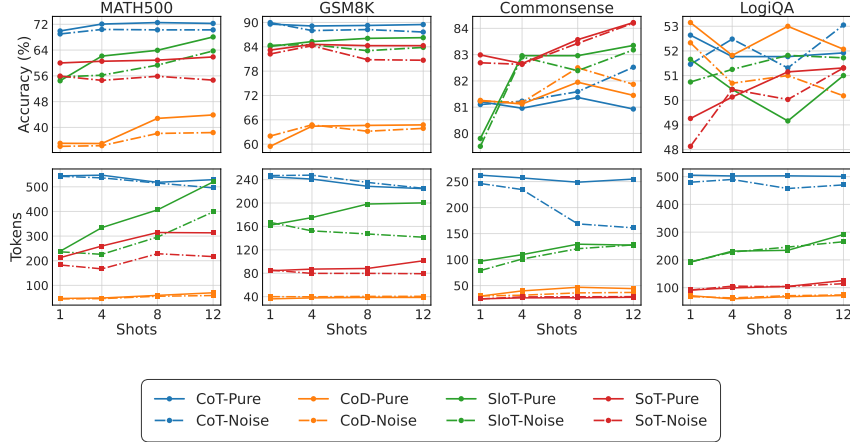
Figure 2: Few-shot performance (accuracy on top, tokens on bottom) of four reasoning methods on Qwen2.5-7B across four datasets, comparing clean vs. noisy settings.

3 and Figure 4, indicate that cross-domain transfer presents a significant robustness challenge, with performance varying considerably across methods, transfer paths, and shot counts.
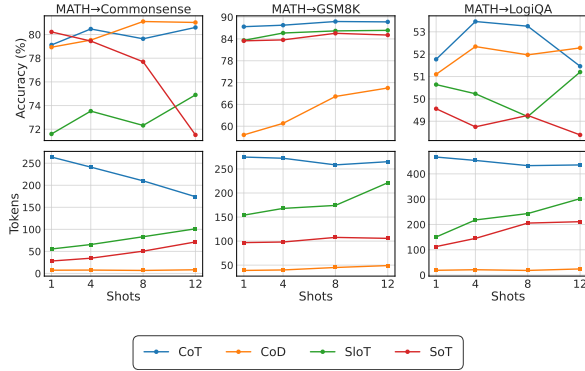

Figure 3: Comparison of few-shot transfer from **MATH500** to Commonsense, GSM8K, and LogiQA, in terms of accuracy (top) and tokens (bottom).

First, CoT generally demonstrates the strongest stability. In most transfer scenarios, CoT (blue line) maintains a high level of accuracy that is relatively less perturbed by the number of shots or the change in domain, serving as a robust cross-domain baseline. **Second**, SloT shows shot-dependent cross-domain reasoning transfer. SloT's (green line) performance is not always strong at 1-shot, as seen in the LOGIQA->MATH transfer, where it starts far below CoT. However, SloT is highly sensitive to exemplar count: its accuracy rises substantially with more shots in multiple settings (e.g., LOGIQA->MATH and MATH->Commonsense). This suggests SloT may be capable of learning transferable reasoning structures from out-of-domain exemplars, but this learning is more data-dependent than

CoT's. **Third**, Reasoning Blueprint methods (CoD, SoT) transfer poorly and unreliably. CoD (orange line) exhibits low accuracy in most cross-domain settings, though it shows modest improvement with more shots in near-domain transfer (e.g., MATH->GSM8K). SoT's (red line) behavior is particularly volatile: in the MATH->Commonsense transfer, its accuracy plummets as shots increase, while in the LOGIQA->GSM8K transfer, it remains consistently low. **Forth**, domain similarity appears to be a key factor. Performance in near-domain transfers (e.g., MATH->GSM8K, both math tasks) is generally more stable for most methods than in far-domain transfers (e.g., LOGIQA->Commonsense). In most of the far-domain scenarios, we observe more significant volatility in accuracy for nearly all methods, including CoT and SloT. This indicates that methods reliant on surface patterns (CoD, SoT) or specific structures (SloT) may be particularly brittle when faced with a substantial domain shift.
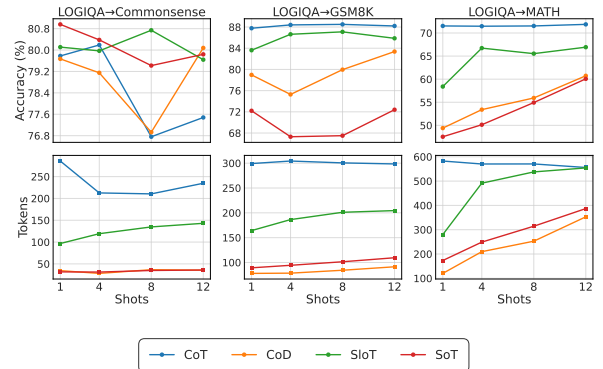

Figure 4: Comparison of few-shot transfer from **LogiQA** to Commonsense, GSM8K, and MATH500 in terms of accuracy (top) and tokens (bottom).

5 Conclusion

We present EffiReason-Bench, a unified benchmark for efficient reasoning with E^3-Score, a CES-inspired metric that jointly measures accuracy and token efficiency. Evaluating six open-source backbones across mathematics, commonsense, and logic tasks, we find that train-free blueprint methods (CoD/SoT) achieve extreme compression but frequently sacrifice accuracy, while the train-based O1-Pruner preserves or improves accuracy with modest efficiency gains. Post-hoc TokenSkip refinement delivers favorable trade-offs on commonsense, logic, and LLaMA backbones, yet degrades MATH500 accuracy substantially on Qwen, revealing strong architecture and domain sensitivity. Among dynamic execution methods, latent-space Soft Thinking exhibits consistent robustness, whereas structure-first SloT shows instability in zero-shot settings but scales effectively with additional shots and curated exemplars. By standardizing evaluation and releasing reproducible implementations, EffiReason-Bench establishes a rigorous foundation for fair comparison and development of adaptive, efficient reasoning strategies.

References

Maciej Besta, Nils Blach, Ales Kubicek, Robert Gerstenberger, Michal Podstawski, Lukas Gianinazzi, Joanna Gajda, Tomasz Lehmann, Hubert Niewiadomski, Piotr Nyczyk, and 1 others. 2024. Graph of thoughts: Solving elaborate problems with large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 17682–17690.

Qiguang Chen, Libo Qin, Jinhao Liu, Dengyun Peng, Jiannan Guan, Peng Wang, Mengkang Hu, Yuhang Zhou, Te Gao, and Wanxiang Che. 2025. Towards reasoning era: A survey of long chain-of-thought for reasoning large language models. *arXiv preprint arXiv:2503.09567*.

Xingyu Chen, Jiahao Xu, Tian Liang, Zhiwei He, Jianhui Pang, Dian Yu, Linfeng Song, Qiuzhi Liu, Mengfei Zhou, Zhuosheng Zhang, and 1 others. 2024. Do not think that much for 2+ 3=? on the overthinking of o1-like llms. *arXiv preprint arXiv:2412.21187*.

Jeffrey Cheng and Benjamin Van Durme. 2024. Compressed chain of thought: Efficient reasoning through dense representations. *arXiv preprint arXiv:2412.13171*.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, and 1 others. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Alejandro Cuadron, Dacheng Li, Wenjie Ma, Xingyao Wang, Yichuan Wang, Siyuan Zhuang, Shu Liu, Luis Gaspar Schroeder, Tian Xia, Huanzhi Mao, and 1 others. 2025. The danger of overthinking: Examining the reasoning-action dilemma in agentic tasks. *arXiv preprint arXiv:2502.08235*.

Sicheng Feng, Gongfan Fang, Xinyin Ma, and Xinchao Wang. 2025. Efficient reasoning models: A survey. *arXiv preprint arXiv:2504.10903*.

Thilo Hagendorff and Sarah Fabi. 2025. Beyond chains of thought: Benchmarking latent-space reasoning abilities in large language models. *arXiv preprint arXiv:2504.10615*.

Tingxu Han, Zhenting Wang, Chunrong Fang, Shiyu Zhao, Shiqing Ma, and Zhenyu Chen. 2025. Token-budget-aware LLM reasoning. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 24842–24855.

Shibo Hao, Sainbayar Sukhbaatar, DiJia Su, Xian Li, Zhiting Hu, Jason Weston, and Yuandong Tian. 2024. Training large language models to reason in a continuous latent space. *arXiv preprint arXiv:2412.06769*.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the MATH dataset. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

Mingyu Jin, Qinkai Yu, Dong Shu, Haiyan Zhao, Wenyue Hua, Yanda Meng, Yongfeng Zhang, and Mengnan Du. 2024. The impact of reasoning step length on large language models. *arXiv preprint arXiv:2401.04925*.

Jian Liu, Leyang Cui, Hanmeng Liu, Dandan Huang, Yile Wang, and Yue Zhang. 2020. Logiqa: A challenge dataset for machine reading comprehension with logical reasoning. *arXiv preprint arXiv:2007.08124*.

Ruikang Liu, Yuxuan Sun, Manyi Zhang, Haoli Bai, Xianzhi Yu, Tiezheng Yu, Chun Yuan, and Lu Hou. 2025. Quantization hurts reasoning? an empirical study on quantized reasoning models. *arXiv preprint arXiv:2504.04823*.

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

Xinyin Ma, Guangnian Wan, Runpeng Yu, Gongfan Fang, and Xinchao Wang. 2025. Cot-valve: Length-compressible chain-of-thought tuning. *arXiv preprint arXiv:2502.09601*.

Daniel McFadden. 1963. Constant elasticity of substitution production functions. *The Review of Economic Studies*, 30(2):73–83.

Xuefei Ning, Zinan Lin, Zixuan Zhou, Zifu Wang, Huazhong Yang, and Yu Wang. 2023. Skeleton-of-thought: Prompting llms for efficient parallel generation. *arXiv preprint arXiv:2307.15337*.

Shubham Parashar, Blake Olson, Sambhav Khurana, Eric Li, Hongyi Ling, James Caverlee, and Shuiwang Ji. 2025. Inference-time computations for llm reasoning and planning: A benchmark and insights. *arXiv preprint arXiv:2502.12521*.

Xuan Shen, Yizhou Wang, Xiangxi Shi, Yanzhi Wang, Pu Zhao, and Jiuxiang Gu. 2025a. Efficient reasoning with hidden thinking. *arXiv preprint arXiv:2501.19201*.

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

Zhenyi Shen, Hanqi Yan, Linhai Zhang, Zhanghao Hu, Yali Du, and Yulan He. 2025c. Codi: Compressing chain-of-thought into continuous space via self-distillation. *arXiv preprint arXiv:2502.21074*.

Yang Sui, Yu-Neng Chuang, Guanchu Wang, Jiamu Zhang, Tianyi Zhang, Jiayi Yuan, Hongyi Liu, Andrew Wen, Hanjie Chen, Xia Hu, and 1 others. 2025. Stop overthinking: A survey on efficient reasoning for large language models. *arXiv preprint arXiv:2503.16419*.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2018. Commonsenseqa: A question answering challenge targeting commonsense knowledge. *arXiv preprint arXiv:1811.00937*.

Kimi Team, Angang Du, Bofei Gao, Bowei Xing, Changjiu Jiang, Cheng Chen, Cheng Li, Chenjun Xiao, Chenzhuang Du, Chonghua Liao, and 1 others. 2025. Kimi k1. 5: Scaling reinforcement learning with llms. *arXiv preprint arXiv:2501.12599*.

Rui Wang, Hongru Wang, Boyang Xue, Jianhui Pang, Shudong Liu, Yi Chen, Jiahao Qiu, Derek Fai Wong, Heng Ji, and Kam-Fai Wong. 2025a. Harnessing the reasoning economy: A survey of efficient reasoning for large language models. *arXiv preprint arXiv:2503.24377*.

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

Yiming Wang, Pei Zhang, Siyuan Huang, Baosong Yang, Zhuosheng Zhang, Fei Huang, and Rui Wang. 2025b. Sampling-efficient test-time scaling: Self-estimating the best-of-n sampling in early decoding. *arXiv preprint arXiv:2503.01422*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, and 1 others. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Heming Xia, Yongqi Li, Chak Tou Leong, Wenjie Wang, and Wenjie Li. 2025. Tokenskip: Controllable chain-of-thought compression in llms. *arXiv preprint arXiv:2502.12067*.

Silei Xu, Wenhao Xie, Lingxiao Zhao, and Pengcheng He. 2025a. Chain of draft: Thinking faster by writing less. *arXiv preprint arXiv:2502.18600*.

Yige Xu, Xu Guo, Zhiwei Zeng, and Chunyan Miao. 2025b. Softcot: Soft chain-of-thought for efficient reasoning with llms. *arXiv preprint arXiv:2502.12134*.

Yibo Yan, Jiamin Su, Jianxiang He, Fangteng Fu, Xu Zheng, Yuanhuiyi Lyu, Kun Wang, Shen Wang, Qingsong Wen, and Xuming Hu. 2024. A survey of mathematical reasoning in the era of multimodal large language model: Benchmark, method & challenges. *arXiv preprint arXiv:2412.11936*.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. 2023. Tree of thoughts: Deliberate problem solving with large language models. *Advances in neural information processing systems*, 36:11809–11822.

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

Linan Yue, Yichao Du, Yizhi Wang, Weibo Gao, Fangzhou Yao, Li Wang, Ye Liu, Ziyu Xu, Qi Liu, Shimin Di, and 1 others. 2025. Don't overthink it: A survey of efficient r1-style large reasoning models. *arXiv preprint arXiv:2508.02120*.

Jintian Zhang, Yuqi Zhu, Mengshu Sun, Yujie Luo, Shuofei Qiao, Lun Du, Da Zheng, Huajun Chen, and Ningyu Zhang. 2025a. Lightthinker: Thinking step-by-step compression. *arXiv preprint arXiv:2502.15589*.

Nan Zhang, Yusen Zhang, Prasenjit Mitra, and Rui Zhang. 2025b. When reasoning meets compression: Benchmarking compressed large reasoning models on complex reasoning tasks. *arXiv preprint arXiv:2504.02010*.

Zhen Zhang, Xuehai He, Weixiang Yan, Ao Shen, Chenyang Zhao, Shuohang Wang, Yelong Shen, and Xin Eric Wang. 2025c. Soft thinking: Unlocking the reasoning potential of llms in continuous concept space. *arXiv preprint arXiv:2505.15778*.

A   Noise Injection in Few-shot Prompts.

(1) Reasoning-step noise. Applied to the reasoning chains of exemplars, including: (a) random deletion of reasoning steps, (b) random permutation of step order, (c) random repetition of certain steps.

(2) Semantic noise. For mathematical reasoning tasks (GSM8K, MATH500), we randomly perturb numeric values by small offsets. For commonsense and logical reasoning tasks, we reverse logical connectives according to a fixed mapping:

```
reversal_map = {
    'therefore': 'however',
    'thus': 'but',
    'so': 'although',
    'because': 'despite',
    'since': 'although',
    'then': 'otherwise',
    'implies': 'contradicts',
    'leads to': 'prevents',
}
```

This modification preserves fluency but alters semantic validity, making it a strong stress test. Noise is injected with equal probability across exemplars unless otherwise specified. By introducing these perturbations, we evaluate not only the efficiency–effectiveness trade-off but also the robustness of efficient reasoning methods under imperfect supervision.

B   Proof of Theorem 1

Proof of Theorem 1. Write the weighted power mean

$$M_\rho(a,b) = \left(w\, a^\rho + (1-w)\, b^\rho \right)^{1/\rho}, \quad a, b > 0, \; w \in (0,1), \tag{5}$$

so that $E_3(A, T; \rho) = M_\rho(a,b)$. Let $x_1 = \log a$, $x_2 = \log b$, $\Delta = |x_1 - x_2|$, and define

$$S(\rho) = w\, e^{\rho x_1} + (1-w)\, e^{\rho x_2},$$
$$L(\rho) = \log M_\rho(a,b) = \frac{1}{\rho} \log S(\rho) \tag{6}$$

for $\rho \neq 0$; set $M_0(a,b) = a^w b^{1-w}$ (the continuous extension at $\rho = 0$), whence $L(0) = wx_1 + (1-w)x_2$.

Let $f(\rho) = \log S(\rho)$. A direct differentiation gives

$$f'(\rho) = \frac{we^{\rho x_1}x_1 + (1-w)e^{\rho x_2}x_2}{S(\rho)} = \mathbb{E}_{\pi_\rho}[X],$$
$$f''(\rho) = \mathrm{Var}_{\pi_\rho}[X] \geq 0, \tag{7}$$

where π_ρ is the softmax distribution on $\{x_1, x_2\}$ with weights proportional to $we^{\rho x_1}$ and $(1-w)e^{\rho x_2}$. Since $X \in [\min\{x_1, x_2\}, \max\{x_1, x_2\}]$ with range Δ, we have the uniform variance bound

$$0 \leq f''(\rho) = \mathrm{Var}_{\pi_\rho}[X] \leq \frac{\Delta^2}{4} \qquad (\forall\, \rho \in \mathbb{R}). \tag{8}$$

For $\rho \neq 0$, differentiate $L(\rho) = f(\rho)/\rho$ and set $\phi(\rho) = \rho f'(\rho) - f(\rho)$:

$$L'(\rho) = \frac{f'(\rho)}{\rho} - \frac{f(\rho)}{\rho^2} = \frac{\phi(\rho)}{\rho^2},$$
$$\phi'(\rho) = \rho f''(\rho),$$
$$\phi(0) = 0 \; (\text{since } f(0) = \log(w + 1 - w) = 0). \tag{9}$$

Hence $\phi(\rho) = \int_0^\rho t\, f''(t)\, dt$ and therefore

$$L'(\rho) = \frac{1}{\rho^2} \int_0^\rho t\, f''(t)\, dt. \tag{10}$$

Using $f''(t) \leq \Delta^2/4$ and changing variables if $\rho < 0$,

$$|L'(\rho)| \leq \frac{1}{\rho^2} \int_0^{|\rho|} t\, \frac{\Delta^2}{4}\, dt = \frac{\Delta^2}{8} \qquad (\rho \neq 0). \tag{11}$$

At $\rho = 0$, taking the limit in the integral representation yields

$$L'(0) = \tfrac{1}{2} f''(0) = \tfrac{1}{2} \mathrm{Var}_{\pi_0}[X] \leq \frac{\Delta^2}{8}. \tag{12}$$

Thus $|L'(\rho)| \leq \Delta^2/8$ for all $\rho \in \mathbb{R}$. By the mean-value theorem,

$$|L(\rho_1) - L(\rho_2)| \leq \frac{\Delta^2}{8} |\rho_1 - \rho_2|, \tag{13}$$

which is the claimed Lipschitz bound on $\log E_3$. Exponentiating gives the multiplicative bounds.

Tightness. With $w = \tfrac{1}{2}$ and $x_1 = -x_2 = \Delta/2$, we have $L'(0) = \Delta^2/8$, so the constant cannot be improved in general.