

---

# Topology of Reasoning: Understanding Large Reasoning Models through Reasoning Graph Properties

---

Gouki Minegishi<sup>1</sup> Hiroki Furuta<sup>2†</sup> Takeshi Kojima<sup>1</sup> Yusuke Iwasawa<sup>1</sup> Yutaka Matsuo<sup>1</sup>

<sup>1</sup>The University of Tokyo, <sup>2</sup>Google DeepMind

{minegishi,furuta,t.kojima,iwasawa,matsuo}@weblab.t.u-tokyo.ac.jp

## Abstract

Recent large-scale reasoning models have achieved state-of-the-art performance on challenging mathematical benchmarks, yet the internal mechanisms underlying their success remain poorly understood. In this work, we introduce the notion of a reasoning graph, extracted by clustering hidden-state representations at each reasoning step, and systematically analyze three key graph-theoretic properties: cyclicity, diameter, and small-world index, across multiple tasks (GSM8K, MATH500, AIME 2024). Our findings reveal that distilled reasoning models (e.g., DeepSeek-R1-Distill-Qwen-32B) exhibit significantly more recurrent cycles (about 5 per sample), substantially larger graph diameters, and pronounced small-world characteristics (about 6x) compared to their base counterparts. Notably, these structural advantages grow with task difficulty and model capacity, with cycle detection peaking at the 14B scale and exploration diameter maximized in the 32B variant, correlating positively with accuracy. Furthermore, we show that supervised fine-tuning on an improved dataset systematically expands reasoning graph diameters in tandem with performance gains, offering concrete guidelines for dataset design aimed at boosting reasoning capabilities. By bridging theoretical insights into reasoning graph structures with practical recommendations for data construction, our work advances both the interpretability and the efficacy of large reasoning models. Implementation available here: [https://github.com/gouki510/Topology\\_of\\_Reasoning](https://github.com/gouki510/Topology_of_Reasoning)

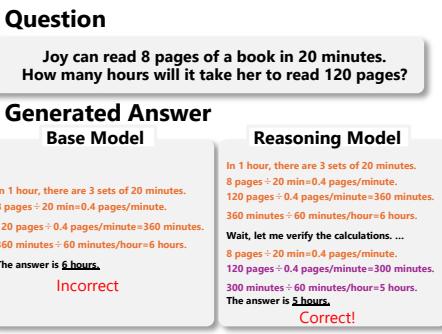
## 1 Introduction

Recent advances in large reasoning models, such as OpenAI-o1 families [47], extended thinking mode in Gemini [29], Claude [2], Grok [66], and DeepSeek-R1 [10], have achieved striking performance gains pushing the frontier across expert-level coding, competitive math, and PhD-level science questions. These recent reasoning models are characterized with to think and reason for longer before responding. Inspired by the breakthroughs in reasoning capabilities, novel methods to imitate reasoning abilities with smaller models have been developed such as supervised fine-tuning (SFT) techniques [44, 69] and distillation [10]. However, despite these notable successes, the internal mechanisms enabling their remarkable reasoning capabilities remain unclear, particularly in comparison to traditional, non-reasoning models.

To understand the key factors behind the success of recent reasoning models, we introduce the concept of a *reasoning graph* (Figure 1). In mathematical tasks, for example, a reasoning graph can be defined as the path through simple computational states (e.g., addition or subtraction) toward the final answer, where each state corresponds to a node in the graph. Prior to recent breakthroughs in reasoning models, some previous works [48, 63] have empirically and theoretically shown that LLMs

---

<sup>†</sup>Work done as an advisory role only.



- **Insights for Effective SFT Data Construction** (Section 5): By examining how SFT datasets influence reasoning graph characteristics, we reveal that refined datasets lead to larger graph diameters and better reasoning performance. These insights provide actionable guidelines for designing training data to explicitly enhance reasoning capabilities.

## 2 Related Works

**Approaches to Enhanced Reasoning** Recent methods for enhancing reasoning in language models include: (1) search-based methods that utilize additional inference-time computation or iterative self-improvement, and (2) reinforcement learning (RL)-based fine-tuning, which has notably driven significant performance breakthroughs. Search-based methods involve external computation of inference time such as parallel sampling [34, 5], sophisticated verifier-based searches [38, 70, 62], or internal in-context refinement and self-correction strategies [17, 18, 71, 33, 27]. Despite their effectiveness, these methods often require careful design or redundant computation. RL methods autonomously discover effective reasoning strategies, encompassing both off-policy [74, 24] and on-policy techniques [75, 30, 9]. Recent advances like DeepSeek-R1 [10], leveraging algorithms such as PPO [50] and GRPO [51], demonstrate significant improvements through structured reasoning traces [35, 73]. A notable phenomenon is the emergence of the “*aha moment*”, characterized by models spontaneously revising their reasoning strategies, inspiring new training approaches such as s1 [44] and Think-DPO [69].

Motivated by these findings, this research elucidates reasoning improvements by analyzing underlying *reasoning graph* structures.

**Analytical Studies on Reasoning Capabilities** Earlier studies prior to recent breakthroughs in RL-based reasoning emphasized reasoning graphs to explain LLM capabilities. Previous work [48] theoretically and empirically demonstrated that reasoning capabilities emerge due to the locality property inherent in natural language data; specifically, models achieve better accuracy by traversing intermediate variables frequently co-occurring during training. Additionally, Wang et al. [63] proposed extracting reasoning graphs by clustering internal model states using  $K$ -means, hypothesizing that pre-training data enables a random walk over these graph nodes. Other studies [13, 67, 6, 59] have created simple toy tasks with explicit reasoning graphs to better understand the mechanisms underlying reasoning abilities in language models. More recently, DeepSeek-R1 [10] have demonstrated significant improvements through RL without explicit reasoning supervision. Analyses following this advancement have explored steering vectors [58], cognitive behaviors [19], and anthropomorphic expressions [68].

However, while recent reasoning models are featured with enhanced reasoning traces, no studies have analyzed them through reasoning graph perspective. Addressing this gap is essential for understanding current advancements.

## 3 Graph Properties of LLM’s Reasoning Process

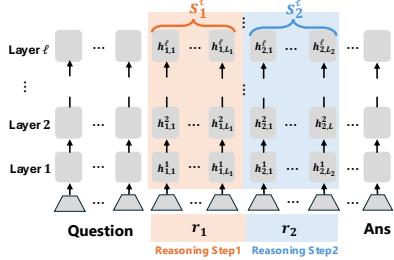
### 3.1 Extracting Reasoning Graph from LLM’s Representations

Let  $\mathcal{D} = \{x_n\}_{n=1}^N$  be our evaluation set of  $N$  questions. For each question  $x \in \mathcal{D}$ , we prompt the model to generate a sequence of intermediate reasoning steps  $R = (r_1, r_2, \dots, r_T)$ , where each  $r_t$  is delimited by a newline character ('\n') and thus represents one reasoning step. We denote the total number of reasoning steps per question as  $T$ , and the token length of a segment as  $L_t$ . Let  $h_{t,\mu}^\ell \in \mathbb{R}^d$  be the hidden state at Transformer layer  $\ell$  for the  $\mu$ -th token of segment  $r_t$  (illustrated in Figure 2-(a)), and define the segment representation as the mean:  $s_t^\ell = \frac{1}{L_t} \sum_{\mu=1}^{L_t} h_{t,\mu}^\ell$ .

**Node Definition** Following the previous work [63], we aggregate all segment representations  $\mathcal{S} = \{s_t^\ell \mid 1 \leq t \leq T, x \in \mathcal{D}\}$  and run  $K$ -means (default  $K = 200$ ) to obtain clusters  $\{C_k\}_{k=1}^K$  with centroids  $\{c_k\}$ . Each centroid  $c_k$  corresponds to a node  $v_k$  in the reasoning graph:

$$V = \{v_1, \dots, v_K\}, \quad d(v_i, v_j) = \|c_i - c_j\|_2.$$

The distance  $d(v_i, v_j)$  between nodes  $v_i$  and  $v_j$  is defined as the Euclidean distance between their corresponding centroids  $c_i$  and  $c_j$ . Figure 2-(b) presents representative nodes obtained from clustering



(a) Extract reasoning graphs from LLMs

| Node                           | Generated Reasoning Step  |
|--------------------------------|---|
| <b>Multiplicative</b> (Node 4) | First leg: $10 \times 30 = 300$ .<br>Now multiply that by $120 \times 1.157625$ . |
| <b>Additive</b> (Node 21)      | Total chairs: $22 + 66 = 88$ .<br>Total sofas: $20 + 64 = 84$ .                   |
| <b>Wait</b> (Node 15)          | Wait, but we need to add this ...<br>Wait, no. Let me clarify ...                 |

(b) Representative Nodes from Reasoning Graph

Figure 2: (a) Illustration of the methodology used to extract reasoning graphs from LLMs. (b) Representative nodes obtained from clustering the DeepSeek-R1-Distill-Qwen-32B using GSM8K dataset.

the DeepSeek-R1-Distill-Qwen-32B [10] using GSM8K dataset [7]. Each clustered node corresponds to simple computations encountered within tasks. As characteristic of reasoning models, some nodes include the term “wait”, indicative of an “aha moment” where the model rechecks its outputs. A more fine-grained analysis, leveraging an LLM-as-judge to characterize the semantics of these clusters, is provided in [Appendix B](#).

**Edge Construction** Informally, the edges in the reasoning graph represent the sequential path of nodes visited by the model for each question during inference. Formally, for each question  $x$ , let  $\pi = (i_1, i_2, \dots, i_T)$ , where  $i_t = \arg \min_k \|s_t^\ell - c_k\|_2$  assigns segment  $r_t$  to its nearest centroid. We then define the directed-edge set

$$E = \{(v_{i_t} \rightarrow v_{i_{t+1}}) \mid t = 1, \dots, T-1\},$$

yielding the *reasoning graph*  $G = (V, E)$  for that question. The reasoning graph properties (cycle density, diameter, small-world index) are then computed over  $G$ .

### 3.2 Measuring Graph Properties

Having extracted reasoning graphs from LLM representations, we evaluate their structural properties from three perspectives: (1) Cycles, (2) Diameter, and (3) Small-World index. Simple implementations of each method are provided in [Appendix C](#).

**Cycles** We detect cycles in the reasoning graph, defined as repeated visits to the same node, excluding self-loops and adjacent duplicates. This is because repetitive behaviors frequently observed in large reasoning models do not represent meaningful cycles [68]. We define the *cycle detection ratio* as the proportion of reasoning graphs containing at least one cycle across all samples. Additionally, we measure the *cycle count* of each reasoning graph as the maximum number of repeated visits to any single node (excluding self-loops, which repeat the same sentence).

**Diameter** To compute the reasoning graph diameter, defined as the maximum shortest path distance between any two reachable nodes, we run Dijkstra’s algorithm [12] from each node  $u$ . We record its distance map  $d(u, v)$ , and define: diameter =  $\max_u \max_{v \neq u} d(u, v)$ . A large diameter indicates that the reasoning graph explores a wider range of potential reasoning nodes during inference.

**Small-World index** Small-world organisation is a robust network feature that has been observed in diverse domains—social networks [57, 65], biological and neural systems [28, 53], ecological webs [43], and technological networks such as the World-Wide Web [1]. While the graph diameter represents the maximum geodesic length, it says nothing about local connectivity. We therefore evaluate the *small-world index*. Following Humphries and Gurney [26], we first symmetrise the directed reasoning graph to obtain an undirected neighbour set  $\mathcal{N}(i)$  for each node  $i$ . With  $n_i = |\mathcal{N}(i)|$ ,  $N_C = |\{i : n_i \geq 2\}|$ , and  $N_L = \sum_u \sum_{v \neq u} \mathbf{1}_{\{v \text{ reachable from } u\}}$ , we define

$$C_i = \frac{\#\{\text{edges among neighbors of } i\}}{n_i(n_i - 1)/2}, \quad C = \frac{1}{N_C} \sum_{i: n_i \geq 2} C_i, \quad L = \frac{\sum_u \sum_{v \neq u} d(u, v)}{N_L},$$

where  $d(u, v)$  is the shortest-path distance from node  $u$  to node  $v$ . Letting  $N$  be the total number of nodes and  $K$  the mean degree of the undirected graph, we approximate the corresponding

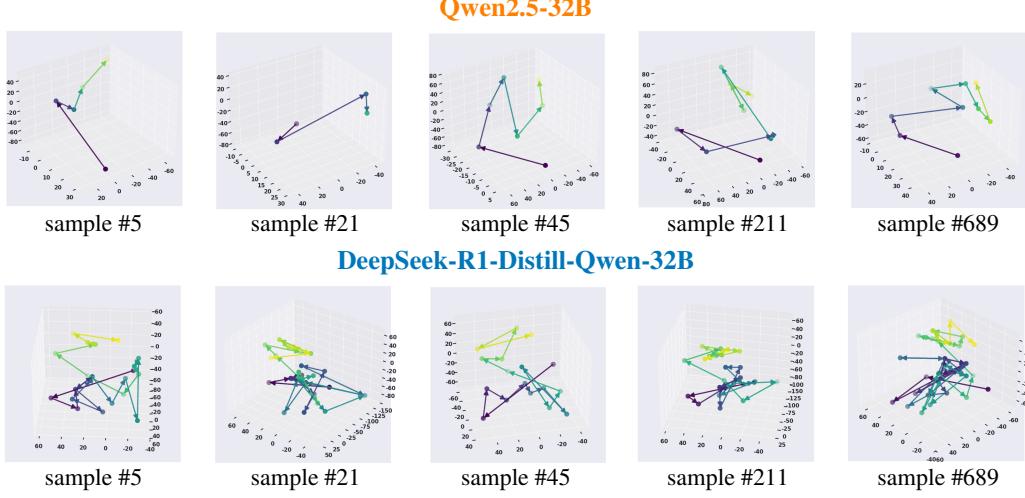


Figure 3: Visualization of reasoning graphs on GSM8K dataset using t-SNE embeddings. The upper row shows graphs from **base model** (Qwen2.5-32B), while the lower row represents those from the **large reasoning model** (DeepSeek-R1-Distill-Qwen-32B). Compared to the base model, the reasoning model exhibits qualitatively broader exploration with notably more cycles in its reasoning graphs.

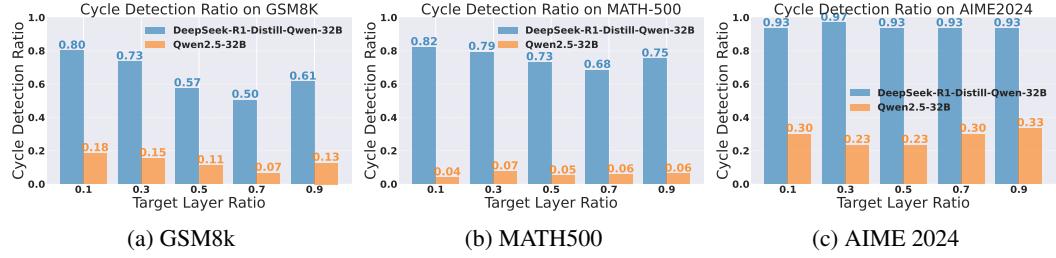


Figure 4: Comparison of cycle detection ratios across different layers in the **large reasoning model** (DeepSeek-R1-Distill-Qwen-32B) and the **base model** (Qwen2.5-32B), evaluated on three tasks: (a) GSM8K, (b) MATH500, and (c) AIME 2024. Results consistently show that the large reasoning model exhibits significantly higher cycle detection ratios than the base model at all layer ratios and tasks. Additionally, a trend emerges, indicating that the cycle detection ratio increases as task difficulty escalates from GSM8K through MATH500 to AIME 2024.

Erdős–Rényi random-graph baseline values [4] as  $C_{\text{rand}} = \frac{K}{N-1}$ ,  $L_{\text{rand}} = \frac{\ln N}{\ln K}$ , and define the small-world index by  $S = \frac{C/C_{\text{rand}}}{L/L_{\text{rand}}}$ . The clustering coefficient describes the tendency of nodes to form tightly interconnected groups, while the average path length indicates how efficiently information propagates through the network. The small-world index combines these characteristics, highlighting a graph’s ability to maintain local cohesion while supporting rapid global connectivity.

## 4 Analyzing Enhanced Reasoning through Graph-Theoretic Properties

We utilize the Qwen2.5 family distilled from DeepSeek-R1 [10] as our large reasoning models, available in sizes of 1.5B, 7B, 14B, and 32B parameters. The corresponding base models and their details are provided in [Appendix D](#). By default, we use the highest-performing 32B variant. Unless otherwise specified, we extract reasoning graphs from the hidden layer positioned at 90% depth (e.g., layer 58 in the 64-layer 32B model). We employ the GSM8K [7], MATH500 [23], and AIME 2024 [11] datasets for constructing the reasoning graphs. Additional analyses on non-mathematical tasks, including StrategyQA [22] and LogicalDeduction from BIG-Bench [54], are provided in [Appendix E](#).

### 4.1 Visualization and Quantification of Cycles in Reasoning Graphs

To intuitively capture the characteristics of reasoning graphs in large reasoning models, we first visualize reasoning graphs for some samples from the GSM8K dataset using 3-dimensional t-SNE

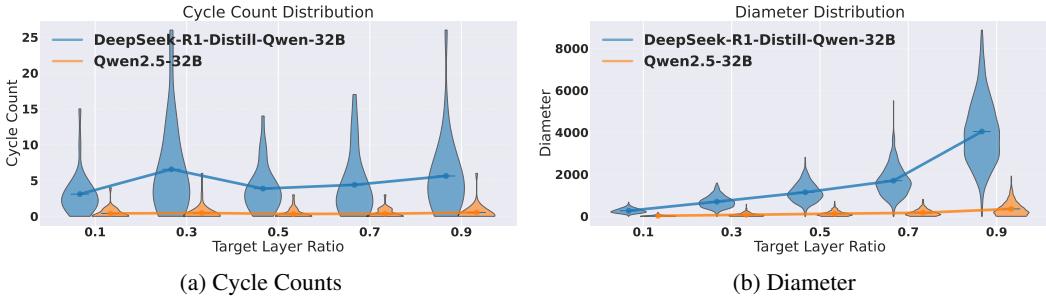


Figure 5: (a) Distribution of cycle counts for the [large reasoning model](#) (DeepSeek-R1-Distill-Qwen-32B) and the [base model](#) (Qwen2.5-32B) across various hidden layer depths. The reasoning model exhibits significantly higher cycle counts. (b) Distribution of reasoning graph diameters across various hidden layer depths. The diameter of reasoning graphs increases progressively with deeper layers. The reasoning model demonstrates significantly larger graph diameters, indicating a broader exploration space compared to the base model.

embedding, as depicted in [Figure 3](#). In the visualization, reasoning graphs are represented as directed arrows connecting nodes visited during inference. The base model demonstrates relatively simple and predominantly *acyclic* reasoning graphs. In contrast, the large reasoning model exhibits more complex structures, characterized by frequent *cyclic* patterns and broader node coverage.

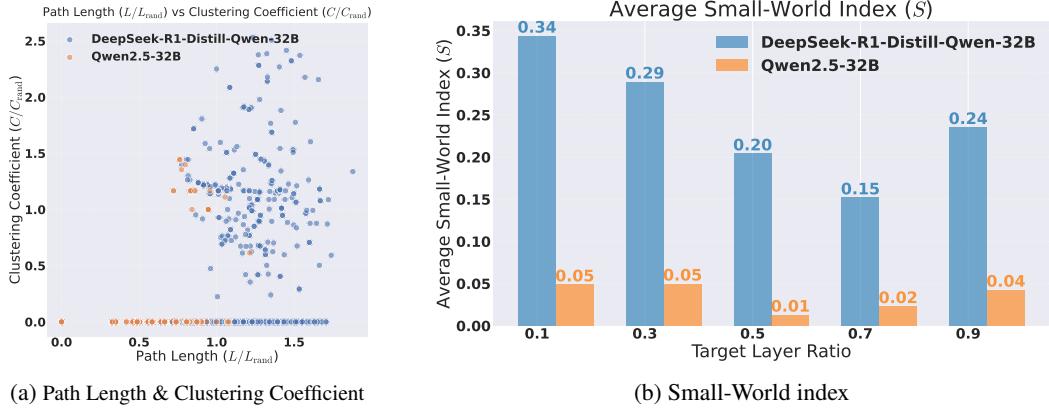
To quantitatively validate these qualitative observations, we employed the cycle detection method introduced in [Section 3.2](#). [Figure 4](#) shows cycle detection rates for GSM8K, MATH500, and AIME 2024 datasets, comparing the large reasoning model (blue) with the base model (orange). The horizontal axis denotes different relative depths of hidden layers (0.1, 0.3, 0.5, 0.7, and 0.9), corresponding respectively to layers 6, 19, 32, 45, and 58 in the 64-layer Qwen2.5-32B. Across all layers, the large reasoning model consistently exhibited a notably higher frequency of cyclic reasoning graphs compared to the base model. Additionally, we observed higher cycle detection rates at the earlier and later layers, with lower detection rates in intermediate layers. This pattern suggests that intermediate layers compress token representations, making less cycle detection difficult, whereas layers closer to input or output exhibit clearer cyclic behaviors. The results for varying the hyperparameter  $k$  of the  $K$ -means clustering are provided in [Appendix F](#), showing consistent trends across all tested values of  $k$ . Furthermore, another consistent trend emerges in which cycle detection ratios increase with the increasing complexity of tasks, progressing from GSM8K through MATH500 to AIME 2024. These findings reinforce the hypothesis that cycles within reasoning graphs contribute to the enhanced reasoning capabilities observed in large reasoning models.

[Figure 5](#)-(a) illustrates the distribution of cycle counts per sample. The large reasoning model consistently exhibits higher cycle counts. It indicates that the large reasoning model not only exhibits a higher proportion of samples containing cycles but also features a higher average number of cycles per sample, approximately five cycles on average. These findings also emphasize the importance of cyclic structures in reasoning graphs as a critical characteristic that improves reasoning performance.

## 4.2 Analyzing Reasoning Exploration through Graph Diameter

To better understand exploratory behaviors within reasoning graphs, we analyzed the distribution of graph diameters for both the large reasoning model and the base model using the GSM8K dataset. As shown in [Figure 5](#)-(b), the large reasoning model consistently demonstrates larger graph diameters across all examined layers compared to the base model. This indicates that the large reasoning model explores a wider range of reasoning nodes during inference, likely contributing to its superior reasoning performance.

Moreover, we observed a clear trend of progressively increasing graph diameters in deeper hidden layers, suggesting that richer contextual representations at deeper layers correspond to broader exploration scopes. These observations imply similarities between expanding the number of output tokens (thus enlarging the exploration scope) and increasing model depth from the perspective of reasoning graph diameters. Our findings suggest a unified explanation based on reasoning graph diameters, which aligns closely with recent studies that emphasize improved reasoning through iterative deep-layer processing [\[42, 21\]](#). Additional results for the MATH500 and AIME 2024 datasets can be found in [Appendix G](#), demonstrating the same trends across all tasks.



(a) Path Length & Clustering Coefficient

(b) Small-World index

Figure 6: (a) Distribution of average path lengths and clustering coefficients in the **large reasoning model** (DeepSeek-R1-Distill-Qwen-32B) and the **base model** (Qwen2.5-32B). Reasoning models exhibit larger clustering coefficients and longer path lengths, indicating densely clustered yet widely separated reasoning nodes. (b) Comparison of small-world index computed from clustering coefficients and average path lengths. Across all layers, the reasoning model consistently exhibits higher small-world characteristics compared to the base model.

### 4.3 Small-World Structure and Enhanced Reasoning Efficiency

To gain deeper insights into the graph characteristics underlying reasoning graphs, we examine their small-world properties. Specifically, using the AIME 2024 dataset, we examine clustering coefficients ( $C$ ), average path lengths ( $L$ ), and their relationship through the Small-World Index ( $S$ ).

Figure 6-(a) depicts the distributions of clustering coefficients and average path lengths for the large reasoning model and the base model at a layer ratio of 0.9. The large reasoning model clearly demonstrates notably higher clustering coefficients alongside longer average path lengths. This combination indicates that reasoning graphs in large reasoning models form densely interconnected local clusters while also having some nodes connected by relatively long-range paths. Such a structure allows quick access via short paths to arbitrary nodes within local clusters, facilitating easier recovery from incorrect reasoning pathways and potentially enhancing reasoning performance. This structural pattern aligns with recent theoretical findings [31] that model reasoning processes as Markov chains, comprising densely connected nodes (representing simple reasoning steps) and sparsely connected critical transitions (representing complex reasoning steps). Additional results for other layer ratios are provided in Appendix H, consistently showing similar trends across all layers.

In Figure 6-(b), we further present the Small-World Index ( $S$ ) computed across different hidden layer depths. The large reasoning model consistently shows higher  $S$  values compared to the base model across all layers. Interestingly, we observe a declining trend in  $S$  near intermediate layers, reflecting the previously discussed cyclic reasoning behaviors. Collectively, these findings underscore the essential contribution of small-world graph characteristics to advanced reasoning performance and suggest valuable avenues for future theoretical and empirical exploration.

### 4.4 Impact of Model Size on Reasoning Graph Properties

To clarify how model size influences reasoning graph properties, we analyzed the relationship between cycle detection ratios, reasoning graph diameters, cycle counts, and task accuracy on the AIME 2024 dataset. The results of the small-world index relative to model size are detailed in Appendix I.

Figure 7-(a) demonstrates that the cycle detection ratio generally increases with model size, peaking at a 100% cycle detection rate in the 14B model. Interestingly, our largest 32B model, which achieves the highest task accuracy, exhibits a lower cycle detection ratio than the 14B model. To better understand the reason for this trend, we compare the outputs generated by the 14B and 32B models in Figure 8. We find that the 14B model experiences *language mixing*, a phenomenon recently reported [68] where language model switches and repeat different languages during the reasoning process. Such undesirable cyclicity likely explains why the 14B model, despite its higher cycle detection ratio, underperforms relative to the 32B model.

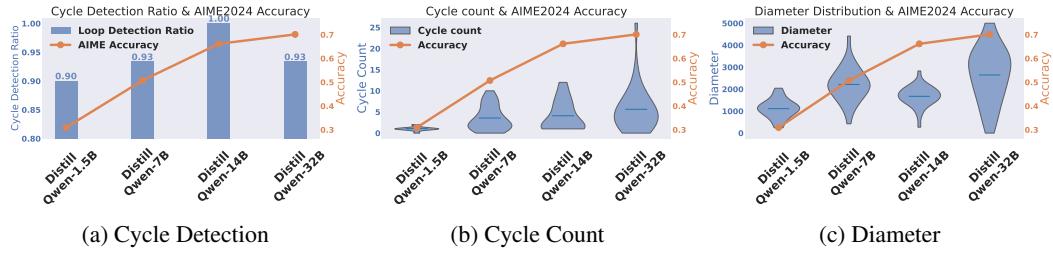


Figure 7: (a) Relationship between cycle detection ratio and AIME 2024 accuracy across different model sizes. The cycle detection ratio generally increases with model size up to 14B, which achieves a 100% cycle detection ratio. (b) Relationship between cycle count and accuracy across different model sizes. Larger models demonstrate increased cycle counts, with the 32B model, which achieves the highest accuracy, exhibiting the greatest number of cycles. (c) Relationship between reasoning graph diameter and accuracy across different model sizes. The 32B model, achieving the highest accuracy, also exhibits the largest graph diameter.

These findings indicate that although cyclic reasoning generally enhances reasoning effectiveness, certain types of cycles, such as language mixing, do not positively contribute to performance.

Figure 7-(b) illustrates that cycle counts progressively rise with model size, reaching the maximum in the 32B model. The observed positive correlation suggests that iterative revisit of reasoning nodes fosters deeper refinement, thereby substantially enhancing performance in complex math tasks. Figure 7-(c) highlights that the reasoning graph diameter expands with model size, with the largest (32B) model consistently showing the most significant diameters alongside the highest accuracy. This suggests that broader exploration and complex reasoning paths are crucial for superior reasoning performance.

Collectively, these results imply that larger model capacities enable greater cyclicity and wider exploration within reasoning graphs, aligning with prior analyses showing smaller models struggle more in reasoning tasks [36].

## 5 Evolution of Reasoning Graph Properties during Supervised Fine-Tuning

To bridge the graph-theoretic properties of large reasoning models observed in Section 4 with practical improvements in reasoning performance, we conducted SFT using the s1 dataset [44], which significantly enhances the reasoning capabilities of the Qwen2.5-32B-Instruct. We analyzed the evolution of reasoning graph properties across training steps to elucidate how these characteristics emerge through training. Detailed training parameters are provided in Appendix J. Our experiments utilized two versions of the dataset: the original version (s1-v1.0<sup>1</sup>) and an updated version (s1-v1.1<sup>2</sup>), each consisting of 1000 training samples. Performance metrics on benchmark datasets indicated higher efficacy for the updated dataset, with v1.1 achieving 94.4% accuracy compared to 92.6% for v1.0 on MATH500, and 56.7% versus 50.0% accuracy on AIME 2024, respectively [44].

Figure 9-(a) illustrates the differences in reasoning graph diameters between s1-v1.0 and s1-v1.1 at 200 training steps, and Figure 9-(b) at 400 training steps across various layers on AIME 2024. It can be observed that, on average, s1-v1.1 consistently produces larger diameters across all layers. Furthermore, there are more samples exhibiting larger diameters at 400 steps compared to 200 steps. These findings suggest that the diameter of reasoning graphs is amplified by SFT, and notably, superior SFT data such as s1-v1.1 enhances the reasoning graph diameter, effectively expanding the exploration space. The results for other checkpoints are provided in Appendix K.



Figure 8: Comparison of reasoning outputs from the 14B and 32B models. The 14B model exhibits *language mixing* [68], switching languages through the response, while the 32B model maintains a consistent language.

<sup>1</sup><https://huggingface.co/datasets/simplescaling/s1K>

<sup>2</sup><https://huggingface.co/datasets/simplescaling/s1K-1.1>

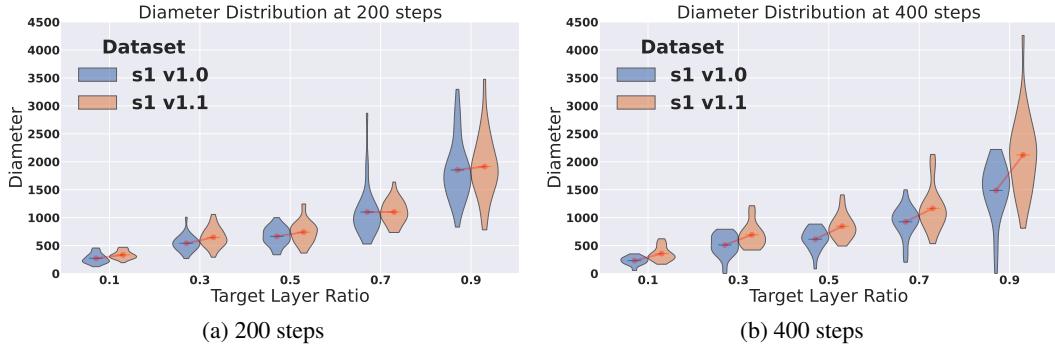


Figure 9: Comparison of reasoning graph diameter distributions between datasets s1-v1.0 and s1-v1.1 across different hidden layers at (a) 200 training steps and (b) 400 training steps. Dataset s1-v1.1 consistently yields larger graph diameters compared to s1-v1.0, and graph diameters increase as training progresses from 200 to 400 steps, indicating enhanced exploration capacity facilitated by superior SFT data.

## 6 Discussion

In this work, we conducted an extensive analysis of reasoning graphs derived from large reasoning models, uncovering key structural properties that correlate with their enhanced performance. Our main findings highlight that large reasoning models consistently exhibit (1) greater cyclicity, (2) broader exploratory behaviors (larger diameters), and (3) pronounced small-world characteristics compared to base models. These insights suggest sophisticated structures in reasoning graphs as a critical factor driving reasoning improvements. Our results connect several observed behaviors in large reasoning models and offer implications for constructing more effective training datasets.

**Aha Moment** Models trained via RL have been reported to exhibit an intriguing phenomenon known as the “*aha moment*,” where the model reconsidered its intermediate answers during reasoning [10, 68]. From the perspective of our reasoning graph analysis, this phenomenon aligns consistently with the observed *cyclic* structures (as illustrated in Figure 1). Although the “*aha moment*” was initially identified as a phenomenon at the generated token level, our study quantitatively measures this behavior through the *cycle* properties of reasoning graphs, thereby contributing to a deeper mechanistic understanding of the “*aha moment*” from the internal states of LLMs.

**Overthinking and Underthinking** Recent studies have highlighted specific reasoning inefficiencies in large reasoning models. *Overthinking*, characterized by redundant or excessively long reasoning processes, has been frequently observed, particularly in agent-based tasks [32, 55, 8, 14]. Conversely, models in the o1 family display *underthinking*, rapidly switching thoughts without adequately exploring potentially valuable reasoning paths [64]. These phenomena align closely with the graph properties we have analyzed: redundant cyclic structures (discussed in Section 4.4) explain overthinking, while overly extensive exploratory behaviors (reflected in larger graph diameters, discussed in Section 4.2) may account for underthinking. Thus, our research clarifies these unique behaviors of large reasoning models through the lens of reasoning graph characteristics.

**Implications for Reasoning SFT-Data Construction** Some studies have significantly improved reasoning performance through SFT or DPO [49, 15] with limited data [72, 44, 69]. While these studies typically create datasets based on qualitative criteria such as difficulty, quality, and diversity, which are inherently challenging to quantify, our proposed metric based on reasoning graph characteristics extracted from hidden states provide novel insights for dataset construction. For example, the high-quality dataset s1-v1.1 demonstrated notably larger graph diameters, suggesting its structural properties are indicative of superior reasoning potential. Furthermore, as shown in Appendix L, even when comparing the s1-v1.0 against LIMO, the s1 dataset consistently yields reasoning graphs with larger diameters and more cycles across all layer depths. This suggests that higher-quality SFT data induces more exploratory and reflective reasoning behavior, further supporting the use of graph-theoretic metrics as indicators of data effectiveness.

## Acknowledgements

We thank Heiga Zen and Yinlam Chow for their support and reviewing the draft of this paper. We also appreciate the funding support from Google Japan.

## References

- [1] Réka Albert, Hawoong Jeong, and Albert-László Barabási. Diameter of the world-wide web. *Nature*, 401(6749):130–131, September 1999. ISSN 1476-4687. doi: 10.1038/43601. URL <http://dx.doi.org/10.1038/43601>.
- [2] Anthropic. Claude’s extended thinking, February 2025. URL <https://www.anthropic.com/news/visible-extended-thinking>. Accessed: 2025-05-11.
- [3] Leonard Bereska and Stratis Gavves. Mechanistic interpretability for AI safety - a review. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL <https://openreview.net/forum?id=ePUVetPKu6>. Survey Certification, Expert Certification.
- [4] Béla Bollobás. *Random Graphs*, volume 73 of *Cambridge Studies in Advanced Mathematics*. Cambridge University Press, Cambridge, 2 edition, 2001. ISBN 9780521797221.
- [5] Bradley Brown, Jordan Juravsky, Ryan Ehrlich, Ronald Clark, Quoc V. Le, Christopher Ré, and Azalia Mirhoseini. Large language monkeys: Scaling inference compute with repeated sampling, 2024. URL <https://arxiv.org/abs/2407.21787>.
- [6] Vivien Cabannes, Charles Arnal, Wassim Bouaziz, Xingyu Alice Yang, Francois Charton, and Julia Kempe. Iteration head: A mechanistic study of chain-of-thought. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL <https://openreview.net/forum?id=QBCxWp0t5w>.
- [7] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.
- [8] Alejandro Cuadron, Dacheng Li, Wenjie Ma, Xingyao Wang, Yichuan Wang, Siyuan Zhuang, Shu Liu, Luis Gaspar Schroeder, Tian Xia, Huanzhi Mao, Nicholas Thumiger, Aditya Desai, Ion Stoica, Ana Klimovic, Graham Neubig, and Joseph E. Gonzalez. The danger of overthinking: Examining the reasoning-action dilemma in agentic tasks, 2025. URL <https://arxiv.org/abs/2502.08235>.
- [9] Ganqu Cui, Lifan Yuan, Zefan Wang, Hanbin Wang, Wendi Li, Bingxiang He, Yuchen Fan, Tianyu Yu, Qixin Xu, Weize Chen, et al. Process reinforcement through implicit rewards. *arXiv preprint arXiv:2502.01456*, 2025.
- [10] DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jiawei Wang, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuan Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Shengfeng Ye, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wanjia Zhao, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu,

Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanhong Xu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning, 2025. URL <https://arxiv.org/abs/2501.12948>.

- [11] Di Zhang. Aime\_1983\_2024, 2025. URL [https://huggingface.co/datasets/di-zhang-fdu/AIME\\_1983\\_2024](https://huggingface.co/datasets/di-zhang-fdu/AIME_1983_2024).
- [12] Edsger W Dijkstra. A note on two problems in connexion with graphs. *Numerische mathematik*, 1(1):269–271, 1959.
- [13] Subhabrata Dutta, Joykirat Singh, Soumen Chakrabarti, and Tanmoy Chakraborty. How to think step-by-step: A mechanistic understanding of chain-of-thought reasoning. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL <https://openreview.net/forum?id=uHLDkQVtyC>.
- [14] Chenrui Fan, Ming Li, Lichao Sun, and Tianyi Zhou. Missing premise exacerbates overthinking: Are reasoning models losing critical thinking skill?, 2025. URL <https://arxiv.org/abs/2504.06514>.
- [15] Hiroki Furuta, Kuang-Huei Lee, Shixiang Shane Gu, Yutaka Matsuo, Aleksandra Faust, Heiga Zen, and Izzeddin Gur. Geometric-averaged preference optimization for soft preference labels. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.
- [16] Hiroki Furuta, Gouki Minegishi, Yusuke Iwasawa, and Yutaka Matsuo. Towards empirical interpretation of internal circuits and properties in grokked transformers on modular polynomials. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL <https://openreview.net/forum?id=MzSf70uXJO>.
- [17] Kanishk Gandhi, Dorsa Sadigh, and Noah D Goodman. Strategic reasoning with language models. *arXiv preprint arXiv:2305.19165*, 2023.
- [18] Kanishk Gandhi, Denise HJ Lee, Gabriel Grand, Muxin Liu, Winson Cheng, Archit Sharma, and Noah Goodman. Stream of search (sos): Learning to search in language. In *First Conference on Language Modeling*, 2024.
- [19] Kanishk Gandhi, Ayush Chakravarthy, Anikait Singh, Nathan Lile, and Noah D. Goodman. Cognitive behaviors that enable self-improving reasoners, or, four habits of highly effective stars, 2025. URL <https://arxiv.org/abs/2503.01307>.
- [20] Leo Gao, Tom Dupre la Tour, Henk Tillman, Gabriel Goh, Rajan Troll, Alec Radford, Ilya Sutskever, Jan Leike, and Jeffrey Wu. Scaling and evaluating sparse autoencoders. In *The Thirteenth International Conference on Learning Representations*, 2025. URL <https://openreview.net/forum?id=tcsZt9ZNKD>.
- [21] Jonas Geiping, Sean McLeish, Neel Jain, John Kirchenbauer, Siddharth Singh, Brian R. Bartoldson, Bhavya Kailkhura, Abhinav Bhatele, and Tom Goldstein. Scaling up test-time compute with latent reasoning: A recurrent depth approach, 2025. URL <https://arxiv.org/abs/2502.05171>.
- [22] Mor Geva, Daniel Khashabi, Elad Segal, Tushar Khot, Dan Roth, and Jonathan Berant. Did Aristotle Use a Laptop? A Question Answering Benchmark with Implicit Reasoning Strategies. *Transactions of the Association for Computational Linguistics (TACL)*, 2021.

[23] Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. *NeurIPS*, 2021.

[24] Matthew Douglas Hoffman, Du Phan, David Dohan, Sholto Douglas, Tuan Anh Le, Aaron Parisi, Pavel Sountsov, Charles Sutton, Sharad Vikram, and Rif A Saurous. Training chain-of-thought via latent-variable inference. In *NeurIPS*, 2023.

[25] Robert Huben, Hoagy Cunningham, Logan Riggs Smith, Aidan Ewart, and Lee Sharkey. Sparse autoencoders find highly interpretable features in language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL <https://openreview.net/forum?id=F76bwRSLeK>.

[26] Mark Humphries and Kevin Gurney. Network ‘small-world-ness’: A quantitative method for determining canonical network equivalence. *PloS one*, 3:e0002051, 02 2008.

[27] Hyeonbin Hwang, Doyoung Kim, Seungone Kim, Seonghyeon Ye, and Minjoon Seo. Self-explore: Enhancing mathematical reasoning in language models with fine-grained rewards. *arXiv preprint arXiv:2404.10346*, 2024.

[28] H. Jeong, B. Tombor, R. Albert, Z. N. Oltvai, and A.-L. Barabási. The large-scale organization of metabolic networks. *Nature*, 407(6804):651–654, October 2000. ISSN 1476-4687. doi: 10.1038/35036627. URL <http://dx.doi.org/10.1038/35036627>.

[29] Koray Kavukcuoglu. Gemini 2.5: Our most intelligent ai model, March 2025. URL <https://blog.google/technology/google-deepmind/gemini-model-thinking-updates-march-2025/>. Accessed: 2025-05-11.

[30] Amirhossein Kazemnejad, Milad Aghajohari, Eva Portelance, Alessandro Sordoni, Siva Reddy, Aaron Courville, and Nicolas Le Roux. Vineppo: Unlocking rl potential for llm reasoning through refined credit assignment. *arXiv preprint arXiv:2410.01679*, 2024.

[31] Juno Kim, Denny Wu, Jason Lee, and Taiji Suzuki. Metastable dynamics of chain-of-thought reasoning: Provable benefits of search, rl and distillation, 2025. URL <https://arxiv.org/abs/2502.01694>.

[32] Abhinav Kumar, Jaechul Roh, Ali Naseh, Marzena Karpinska, Mohit Iyyer, Amir Houmansadr, and Eugene Bagdasarian. Overthink: Slowdown attacks on reasoning llms, 2025. URL <https://arxiv.org/abs/2502.02542>.

[33] Aviral Kumar, Vincent Zhuang, Rishabh Agarwal, Yi Su, John D Co-Reyes, Avi Singh, Kate Baumli, Shariq Iqbal, Colton Bishop, Rebecca Roelofs, et al. Training language models to self-correct via reinforcement learning. *arXiv preprint arXiv:2409.12917*, 2024.

[34] Chen Li, Weiqi Wang, Jingcheng Hu, Yixuan Wei, Nanning Zheng, Han Hu, Zheng Zhang, and Houwen Peng. Common 7b language models already possess strong math capabilities. *arXiv preprint arXiv:2403.04706*, 2024.

[35] Dacheng Li, Shiyi Cao, Tyler Griggs, Shu Liu, Xiangxi Mo, Shishir G Patil, Matei Zaharia, Joseph E Gonzalez, and Ion Stoica. Llms can easily learn to reason from demonstrations structure, not content, is what matters! *arXiv preprint arXiv:2502.07374*, 2025.

[36] Yuetai Li, Xiang Yue, Zhangchen Xu, Fengqing Jiang, Luyao Niu, Bill Yuchen Lin, Bhaskar Ramasubramanian, and Radha Poovendran. Small models struggle to learn from strong reasoners, 2025. URL <https://arxiv.org/abs/2502.12143>.

[37] Tom Lieberum, Senthooran Rajamanoharan, Arthur Conmy, Lewis Smith, Nicolas Sonnerat, Vikrant Varma, János Kramár, Anca Dragan, Rohin Shah, and Neel Nanda. Gemma scope: Open sparse autoencoders everywhere all at once on gemma 2, 2024. URL <https://arxiv.org/abs/2408.05147>.

[38] Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let’s verify step by step. In *The Twelfth International Conference on Learning Representations*, 2023.

[39] Gouki Minegishi, Hiroki Furuta, Yusuke Iwasawa, and Yutaka Matsuo. Rethinking evaluation of sparse autoencoders through the representation of polysemous words. In *The Thirteenth International Conference on Learning Representations*, 2025. URL <https://openreview.net/forum?id=HpUs2EXj01>.

[40] Gouki Minegishi, Hiroki Furuta, Shohei Taniguchi, Yusuke Iwasawa, and Yutaka Matsuo. Beyond induction heads: In-context meta learning induces multi-phase circuit emergence. In *Forty-second International Conference on Machine Learning*, 2025. URL <https://openreview.net/forum?id=Xw01vF13aV>.

[41] Gouki Minegishi, Yusuke Iwasawa, and Yutaka Matsuo. Bridging lottery ticket and grokking: Understanding grokking from inner structure of networks. *Transactions on Machine Learning Research*, 2025. ISSN 2835-8856. URL <https://openreview.net/forum?id=eQeYyup1tm>.

[42] Takeru Miyato, Sindy Löwe, Andreas Geiger, and Max Welling. Artificial kuramoto oscillatory neurons. In *The Thirteenth International Conference on Learning Representations*, 2025. URL <https://openreview.net/forum?id=nwDRD4AMoN>.

[43] JOSE M. MONTOYA and RICARD V. SOLÉ. Small world patterns in food webs. *Journal of Theoretical Biology*, 214(3):405–412, 2002. ISSN 0022-5193. doi: <https://doi.org/10.1006/jtbi.2001.2460>. URL <https://www.sciencedirect.com/science/article/pii/S0022519301924609>.

[44] Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candes, and Tatsunori Hashimoto. s1: Simple test-time scaling. In *Workshop on Reasoning and Planning for Large Language Models*, 2025. URL <https://openreview.net/forum?id=LdH0vrgAHm>.

[45] Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Scott Johnston, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. In-context learning and induction heads. *Transformer Circuits Thread*, 2022. <https://transformer-circuits.pub/2022/in-context-learning-and-induction-heads/index.html>.

[46] OpenAI, :, Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, Aleksander Mądry, Alex Baker-Whitcomb, Alex Beutel, Alex Borzunov, Alex Carney, Alex Chow, Alex Kirillov, Alex Nichol, Alex Paino, Alex Renzin, Alex Tachard Passos, Alexander Kirillov, Alexi Christakis, Alexis Conneau, Ali Kamali, Allan Jabri, Allison Moyer, Allison Tam, Amadou Crookes, Amin Tootoochian, Amin Tootoonchian, Ananya Kumar, Andrea Vallone, Andrej Karpathy, Andrew Braunstein, Andrew Cann, Andrew Codispoti, Andrew Galu, Andrew Kondrich, Andrew Tulloch, Andrey Mishchenko, Angela Baek, Angela Jiang, Antoine Pelisse, Antonia Woodford, Anuj Gosalia, Arka Dhar, Ashley Pantuliano, Avi Nayak, Avital Oliver, Barret Zoph, Behrooz Ghorbani, Ben Leimberger, Ben Rossen, Ben Sokolowsky, Ben Wang, Benjamin Zweig, Beth Hoover, Blake Samic, Bob McGrew, Bobby Spero, Bogo Giertler, Bowen Cheng, Brad Lightcap, Brandon Walkin, Brendan Quinn, Brian Guarraci, Brian Hsu, Bright Kellogg, Brydon Eastman, Camillo Lugaresi, Carroll Wainwright, Cary Bassin, Cary Hudson, Casey Chu, Chad Nelson, Chak Li, Chan Jun Shern, Channing Conger, Charlotte Barette, Chelsea Voss, Chen Ding, Cheng Lu, Chong Zhang, Chris Beaumont, Chris Hallacy, Chris Koch, Christian Gibson, Christina Kim, Christine Choi, Christine McLeavey, Christopher Hesse, Claudia Fischer, Clemens Winter, Coley Czarnecki, Colin Jarvis, Colin Wei, Constantin Koumouzelis, Dane Sherburn, Daniel Kappler, Daniel Levin, Daniel Levy, David Carr, David Farhi, David Mely, David Robinson, David Sasaki, Denny Jin, Dev Valladares, Dimitris Tsipras, Doug Li, Duc Phong Nguyen, Duncan Findlay, Edede Oiwoh, Edmund Wong, Ehsan Asdar, Elizabeth Proehl, Elizabeth Yang, Eric Antonow, Eric Kramer, Eric Peterson, Eric Sigler, Eric Wallace, Eugene Brevdo, Evan Mays, Farzad Khorasani, Felipe Petroski Such, Filippo Raso, Francis Zhang, Fred von Lohmann, Freddie Sulit, Gabriel Goh, Gene Oden, Geoff Salmon, Giulio Starace, Greg Brockman, Hadi Salman, Haiming Bao, Haitang Hu, Hannah Wong, Haoyu Wang, Heather Schmidt, Heather

Whitney, Heewoo Jun, Hendrik Kirchner, Henrique Ponde de Oliveira Pinto, Hongyu Ren, Hui-wen Chang, Hyung Won Chung, Ian Kivlichan, Ian O’Connell, Ian O’Connell, Ian Osband, Ian Silber, Ian Sohl, Ibrahim Okuyucu, Ikai Lan, Ilya Kostrikov, Ilya Sutskever, Ingmar Kanitscheider, Ishaan Gulrajani, Jacob Coxon, Jacob Menick, Jakub Pachocki, James Aung, James Betker, James Crooks, James Lennon, Jamie Kiros, Jan Leike, Jane Park, Jason Kwon, Jason Phang, Jason Teplitz, Jason Wei, Jason Wolfe, Jay Chen, Jeff Harris, Jenia Varavva, Jessica Gan Lee, Jessica Shieh, Ji Lin, Jiahui Yu, Jiayi Weng, Jie Tang, Jieqi Yu, Joanne Jang, Joaquin Quinonero Candela, Joe Beutler, Joe Landers, Joel Parish, Johannes Heidecke, John Schulman, Jonathan Lachman, Jonathan McKay, Jonathan Uesato, Jonathan Ward, Jong Wook Kim, Joost Huizinga, Jordan Sitkin, Jos Kraaijeweld, Josh Gross, Josh Kaplan, Josh Snyder, Joshua Achiam, Joy Jiao, Joyce Lee, Juntang Zhuang, Justyn Harriman, Kai Fricke, Kai Hayashi, Karan Singh, Katy Shi, Kavin Karthik, Kayla Wood, Kendra Rimbach, Kenny Hsu, Kenny Nguyen, Keren Gu-Lemberg, Kevin Button, Kevin Liu, Kiel Howe, Krithika Muthukumar, Kyle Luther, Lama Ahmad, Larry Kai, Lauren Itow, Lauren Workman, Leher Pathak, Leo Chen, Li Jing, Lia Guy, Liam Fedus, Liang Zhou, Lien Mamitsuka, Lilian Weng, Lindsay McCallum, Lindsey Held, Long Ouyang, Louis Feuvrier, Lu Zhang, Lukas Kondraciuk, Lukasz Kaiser, Luke Hewitt, Luke Metz, Lyric Doshi, Mada Aflak, Maddie Simens, Madelaine Boyd, Madeleine Thompson, Marat Dukhan, Mark Chen, Mark Gray, Mark Hudnall, Marvin Zhang, Marwan Aljubeh, Mateusz Litwin, Matthew Zeng, Max Johnson, Maya Shetty, Mayank Gupta, Meghan Shah, Mehmet Yatbaz, Meng Jia Yang, Mengchao Zhong, Mia Glaese, Mianna Chen, Michael Janner, Michael Lampe, Michael Petrov, Michael Wu, Michele Wang, Michelle Fradin, Michelle Pokrass, Miguel Castro, Miguel Oom Temudo de Castro, Mikhail Pavlov, Miles Brundage, Miles Wang, Minal Khan, Mira Murati, Mo Bavarian, Molly Lin, Murat Yesildal, Nacho Soto, Natalia Gimelshein, Natalie Cone, Natalie Staudacher, Natalie Summers, Natan LaFontaine, Neil Chowdhury, Nick Ryder, Nick Stathas, Nick Turley, Nik Tezak, Niko Felix, Nithanth Kudige, Nitish Keskar, Noah Deutsch, Noel Bundick, Nora Puckett, Ofir Nachum, Ola Okelola, Oleg Boiko, Oleg Murk, Oliver Jaffe, Olivia Watkins, Olivier Godement, Owen Campbell-Moore, Patrick Chao, Paul McMillan, Pavel Belov, Peng Su, Peter Bak, Peter Bakkum, Peter Deng, Peter Dolan, Peter Hoeschele, Peter Welinder, Phil Tillet, Philip Pronin, Philippe Tillet, Prafulla Dhariwal, Qiming Yuan, Rachel Dias, Rachel Lim, Rahul Arora, Rajan Troll, Randall Lin, Rapha Gontijo Lopes, Raul Puri, Reah Miyara, Reimar Leike, Renaud Gaubert, Reza Zamani, Ricky Wang, Rob Donnelly, Rob Honsby, Rocky Smith, Rohan Sahai, Rohit Ramchandani, Romain Huet, Rory Carmichael, Rowan Zellers, Roy Chen, Ruby Chen, Ruslan Nigmatullin, Ryan Cheu, Saachi Jain, Sam Altman, Sam Schoenholz, Sam Toizer, Samuel Miserendino, Sandhini Agarwal, Sara Culver, Scott Ethersmith, Scott Gray, Sean Grove, Sean Metzger, Shamez Hermani, Shantanu Jain, Shengjia Zhao, Sherwin Wu, Shino Jomoto, Shirong Wu, Shuaiqi Xia, Sonia Phene, Spencer Papay, Srinivas Narayanan, Steve Coffey, Steve Lee, Stewart Hall, Suchir Balaji, Tal Broda, Tal Stramer, Tao Xu, Tarun Gogineni, Taya Christianson, Ted Sanders, Tejal Patwardhan, Thomas Cunningham, Thomas Degry, Thomas Dimson, Thomas Raoux, Thomas Shadwell, Tianhao Zheng, Todd Underwood, Todor Markov, Toki Sherbakov, Tom Rubin, Tom Stasi, Tomer Kaftan, Tristan Heywood, Troy Peterson, Tyce Walters, Tyna Eloundou, Valerie Qi, Veit Moeller, Vinnie Monaco, Vishal Kuo, Vlad Fomenko, Wayne Chang, Weiyi Zheng, Wenda Zhou, Wesam Manassra, Will Sheu, Wojciech Zaremba, Yash Patil, Yilei Qian, Yongjik Kim, Youlong Cheng, Yu Zhang, Yuchen He, Yuchen Zhang, Yujia Jin, Yunxing Dai, and Yury Malkov. Gpt-4o system card, 2024. URL <https://arxiv.org/abs/2410.21276>.

[47] OpenAI, :, Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, Alex Iftimie, Alex Karpenko, Alex Tachard Passos, Alexander Neitz, Alexander Prokofiev, Alexander Wei, Allison Tam, Ally Bennett, Ananya Kumar, Andre Saraiva, Andrea Vallone, Andrew Duberstein, Andrew Kondrich, Andrey Mishchenko, Andy Applebaum, Angela Jiang, Ashvin Nair, Barret Zoph, Behrooz Ghorbani, Ben Rossen, Benjamin Sokolowsky, Boaz Barak, Bob McGrew, Borys Minaiev, Botao Hao, Bowen Baker, Brandon Houghton, Brandon McKinzie, Brydon Eastman, Camillo Lugaresi, Cary Bassin, Cary Hudson, Chak Ming Li, Charles de Bourcy, Chelsea Voss, Chen Shen, Chong Zhang, Chris Koch, Chris Orsinger, Christopher Hesse, Claudia Fischer, Clive Chan, Dan Roberts, Daniel Kappler, Daniel Levy, Daniel Selsam, David Dohan, David Farhi, David Mely, David Robinson, Dimitris Tsipras, Doug Li, Dragos Oprica, Eben Freeman, Eddie Zhang, Edmund Wong, Elizabeth Proehl, Enoch Cheung, Eric Mitchell, Eric Wallace, Erik Ritter, Evan Mays, Fan Wang, Felipe Petroski Such, Filippo Raso, Florencia

Leoni, Foivos Tsimpourlas, Francis Song, Fred von Lohmann, Freddie Sulit, Geoff Salmon, Giambattista Parascandolo, Gildas Chabot, Grace Zhao, Greg Brockman, Guillaume Leclerc, Hadi Salman, Haiming Bao, Hao Sheng, Hart Andrin, Hessam Bagherinezhad, Hongyu Ren, Hunter Lightman, Hyung Won Chung, Ian Kivlichan, Ian O’Connell, Ian Osband, Ignasi Clavera Gilaberte, Ilge Akkaya, Ilya Kostrikov, Ilya Sutskever, Irina Kofman, Jakub Pachocki, James Lennon, Jason Wei, Jean Harb, Jerry Twore, Jiacheng Feng, Jiahui Yu, Jiayi Weng, Jie Tang, Jieqi Yu, Joaquin Quiñonero Candela, Joe Palermo, Joel Parish, Johannes Heidecke, John Hallman, John Rizzo, Jonathan Gordon, Jonathan Uesato, Jonathan Ward, Joost Huizinga, Julie Wang, Kai Chen, Kai Xiao, Karan Singh, Karina Nguyen, Karl Cobbe, Katy Shi, Kayla Wood, Kendra Rimbach, Keren Gu-Lemberg, Kevin Liu, Kevin Lu, Kevin Stone, Kevin Yu, Lama Ahmad, Lauren Yang, Leo Liu, Leon Maksin, Leyton Ho, Liam Fedus, Lilian Weng, Linden Li, Lindsay McCallum, Lindsey Held, Lorenz Kuhn, Lukas Kondraciuk, Lukasz Kaiser, Luke Metz, Madelaine Boyd, Maja Trebacz, Manas Joglekar, Mark Chen, Marko Tintor, Mason Meyer, Matt Jones, Matt Kaufer, Max Schwarzer, Meghan Shah, Mehmet Yatbaz, Melody Y. Guan, Mengyuan Xu, Mengyuan Yan, Mia Glaese, Mianna Chen, Michael Lampe, Michael Malek, Michele Wang, Michelle Fradin, Mike McClay, Mikhail Pavlov, Miles Wang, Mingxuan Wang, Mira Murati, Mo Bavarian, Mostafa Rohaninejad, Nat McAleese, Neil Chowdhury, Neil Chowdhury, Nick Ryder, Nikolas Tezak, Noam Brown, Ofir Nachum, Oleg Boiko, Oleg Murk, Olivia Watkins, Patrick Chao, Paul Ashbourne, Pavel Izmailov, Peter Zhokhov, Rachel Dias, Rahul Arora, Randall Lin, Rapha Gontijo Lopes, Raz Gaon, Reah Miyara, Reimar Leike, Renny Hwang, Rhythm Garg, Robin Brown, Roshan James, Rui Shu, Ryan Cheu, Ryan Greene, Saachi Jain, Sam Altman, Sam Toizer, Sam Toyer, Samuel Miserendino, Sandhini Agarwal, Santiago Hernandez, Sasha Baker, Scott McKinney, Scottie Yan, Shengjia Zhao, Shengli Hu, Shibani Santurkar, Shraman Ray Chaudhuri, Shuyuan Zhang, Siyuan Fu, Spencer Papay, Steph Lin, Suchir Balaji, Suvansh Sanjeev, Szymon Sidor, Tal Broda, Aidan Clark, Tao Wang, Taylor Gordon, Ted Sanders, Tejal Patwardhan, Thibault Sottiaux, Thomas Degry, Thomas Dimson, Tianhao Zheng, Timur Garipov, Tom Stasi, Trapit Bansal, Trevor Creech, Troy Peterson, Tyna Eloundou, Valerie Qi, Vineet Kosaraju, Vinnie Monaco, Vitchyr Pong, Vlad Fomenko, Weiyi Zheng, Wenda Zhou, Wes McCabe, Wojciech Zaremba, Yann Dubois, Yinghai Lu, Yining Chen, Young Cha, Yu Bai, Yuchen He, Yuchen Zhang, Yunyun Wang, Zheng Shao, and Zhuohan Li. Openai o1 system card, 2024. URL <https://arxiv.org/abs/2412.16720>.

- [48] Ben Prystawski, Michael Y. Li, and Noah Goodman. Why think step by step? reasoning emerges from the locality of experience. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL <https://openreview.net/forum?id=rcXXNFV1En>.
- [49] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.
- [50] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017. URL <https://arxiv.org/abs/1707.06347>.
- [51] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models, 2024. URL <https://arxiv.org/abs/2402.03300>.
- [52] Lee Sharkey, Bilal Chughtai, Joshua Batson, Jack Lindsey, Jeffrey Wu, Lucius Bushnaq, Nicholas Goldowsky-Dill, Stefan Heimersheim, Alejandro Ortega, Joseph Isaac Bloom, Stella Biderman, Adrià Garriga-Alonso, Arthur Conmy, Neel Nanda, Jessica Mary Rumbelow, Martin Wattenberg, Nandi Schoots, Joseph Miller, William Saunders, Eric J Michaud, Stephen Casper, Max Tegmark, David Bau, Eric Todd, Atticus Geiger, Mor Geva, Jesse Hoogland, Daniel Murfet, and Thomas McGrath. Open problems in mechanistic interpretability. *Transactions on Machine Learning Research*, 2025. ISSN 2835-8856. URL <https://openreview.net/forum?id=91H76m9Z94>. Survey Certification.
- [53] Olaf Sporns and Jonathan Zwi. The small world of the cerebral cortex. *Neuroinformatics*, 2: 145–62, 02 2004. doi: 10.1385/NI:2:2:145.
- [54] Aarohi Srivastava et al. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models, 2022. URL <https://arxiv.org/abs/2206.04615>.

[55] Yang Sui, Yu-Neng Chuang, Guanchu Wang, Jiamu Zhang, Tianyi Zhang, Jiayi Yuan, Hongyi Liu, Andrew Wen, Shaochen Zhong, Hanjie Chen, and Xia Hu. Stop overthinking: A survey on efficient reasoning for large language models, 2025. URL <https://arxiv.org/abs/2503.16419>.

[56] Adly Templeton, Tom Conerly, Jonathan Marcus, Jack Lindsey, Trenton Bricken, Brian Chen, Adam Pearce, Craig Citro, Emmanuel Ameisen, Andy Jones, Hoagy Cunningham, Nicholas L Turner, Callum McDougall, Monte MacDiarmid, C. Daniel Freeman, Theodore R. Sumers, Edward Rees, Joshua Batson, Adam Jermyn, Shan Carter, Chris Olah, and Tom Henighan. Scaling monosemanticity: Extracting interpretable features from claude 3 sonnet. *Transformer Circuits Thread*, 2024. URL <https://transformer-circuits.pub/2024/scaling-monosemanticity/index.html>.

[57] JEFFREY TRAVERS and STANLEY MILGRAM. An experimental study of the small world problem\*\*the study was carried out while both authors were at harvard university, and was financed by grants from the milton fund and from the harvard laboratory of social relations. mr. joseph gerver provided invaluable assistance in summarizing and criticizing the mathematical work discussed in this paper. In Samuel Leinhardt, editor, *Social Networks*, pages 179–197. Academic Press, 1977. ISBN 978-0-12-442450-0. doi: <https://doi.org/10.1016/B978-0-12-442450-0.50018-3>. URL <https://www.sciencedirect.com/science/article/pii/B9780124424500500183>.

[58] Constantin Venhoff, Iván Arcuschin, Philip Torr, Arthur Conmy, and Neel Nanda. Understanding reasoning in thinking language models via steering vectors. In *Workshop on Reasoning and Planning for Large Language Models*, 2025. URL <https://openreview.net/forum?id=OwhVWN0Bcz>.

[59] Boshi Wang, Xiang Yue, Yu Su, and Huan Sun. Grokking of implicit reasoning in transformers: A mechanistic journey to the edge of generalization. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, editors, *Advances in Neural Information Processing Systems*, volume 37, pages 95238–95265. Curran Associates, Inc., 2024. URL [https://proceedings.neurips.cc/paper\\_files/paper/2024/file/ad217e0c7fecc71bdf48660ad6714b07-Paper-Conference.pdf](https://proceedings.neurips.cc/paper_files/paper/2024/file/ad217e0c7fecc71bdf48660ad6714b07-Paper-Conference.pdf).

[60] Boshi Wang, Xiang Yue, Yu Su, and Huan Sun. Grokking of implicit reasoning in transformers: A mechanistic journey to the edge of generalization. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL <https://openreview.net/forum?id=D4QgSWxi0b>.

[61] Kevin Ro Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. Interpretability in the wild: a circuit for indirect object identification in GPT-2 small. In *The Eleventh International Conference on Learning Representations*, 2023. URL <https://openreview.net/forum?id=NpsVSN6o4ul>.

[62] Peiyi Wang, Lei Li, Zhihong Shao, RX Xu, Damai Dai, Yifei Li, Deli Chen, Yu Wu, and Zhifang Sui. Math-shepherd: Verify and reinforce llms step-by-step without human annotations. *arXiv preprint arXiv:2312.08935*, 2023.

[63] Xinyi Wang, Alfonso Amayuelas, Kexun Zhang, Liangming Pan, Wenhui Chen, and William Yang Wang. Understanding the reasoning ability of language models from the perspective of reasoning paths aggregation. In *ICLR 2024 Workshop: How Far Are We From AGI*, 2024. URL <https://openreview.net/forum?id=r0vnM4Wh1H>.

[64] Yue Wang, Qiuzhi Liu, Jiahao Xu, Tian Liang, Xingyu Chen, Zhiwei He, Linfeng Song, Dian Yu, Juntao Li, Zhusong Zhang, Rui Wang, Zhaopeng Tu, Haitao Mi, and Dong Yu. Thoughts are all over the place: On the underthinking of o1-like llms, 2025. URL <https://arxiv.org/abs/2501.18585>.

[65] Duncan J. Watts and Steven H. Strogatz. Collective dynamics of ‘small-world’ networks. *Nature*, 393:440–442, 1998. URL <https://api.semanticscholar.org/CorpusID:3034643>.

[66] xAI. Grok 3 beta — the age of reasoning agents, February 2025. URL <https://x.ai/news/grok-3>. Accessed: 2025-05-11.

- [67] Violet Xiang, Charlie Snell, Kanishk Gandhi, Alon Albalak, Anikait Singh, Chase Blagden, Duy Phung, Rafael Rafailov, Nathan Lile, Dakota Mahan, et al. Towards system 2 reasoning in llms: Learning how to think with meta chain-of-thought. *arXiv preprint arXiv:2501.04682*, 2025.
- [68] Shu Yang, Junchao Wu, Xin Chen, Yunze Xiao, Xinyi Yang, Derek F. Wong, and Di Wang. Understanding aha moments: from external observations to internal mechanisms, 2025. URL <https://arxiv.org/abs/2504.02956>.
- [69] Wang Yang, Hongye Jin, Jingfeng Yang, Vipin Chaudhary, and Xiaotian Han. Thinking preference optimization, 2025. URL <https://arxiv.org/abs/2502.13173>.
- [70] Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. *Advances in neural information processing systems*, 36:11809–11822, 2023.
- [71] Tian Ye, Zicheng Xu, Yuanzhi Li, and Zeyuan Allen-Zhu. Physics of language models: Part 2.2, how to learn from mistakes on grade-school math problems. *arXiv preprint arXiv:2408.16293*, 2024.
- [72] Yixin Ye, Zhen Huang, Yang Xiao, Ethan Chern, Shijie Xia, and Pengfei Liu. Limo: Less is more for reasoning, 2025. URL <https://arxiv.org/abs/2502.03387>.
- [73] Edward Yeo, Yuxuan Tong, Morry Niu, Graham Neubig, and Xiang Yue. Demystifying long chain-of-thought reasoning in llms, 2025. URL <https://arxiv.org/abs/2502.03373>.
- [74] Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah Goodman. Star: Bootstrapping reasoning with reasoning. *Advances in Neural Information Processing Systems*, 35:15476–15488, 2022.
- [75] Eric Zelikman, Georges Raif Harik, Yijia Shao, Varuna Jayasiri, Nick Haber, and Noah Goodman. Quiet-star: Language models can teach themselves to think before speaking. In *First Conference on Language Modeling*, 2024.

## A Broader Impacts

This research sheds light on the underlying mechanisms responsible for improved reasoning performance in LLMs, potentially impacting various fields within artificial intelligence and machine learning. From an interpretability perspective, our findings offer explanations for performance gains in previously opaque (black-box) large reasoning models and provide insights toward the construction of better reasoning architectures.

## B Automatic Node Labeling with LLMs

To better understand what kinds of reasoning patterns are represented by the centroids when  $k = 200$ , we conducted an automatic labeling experiment using a large language model. Specifically, we used the GPT-4o-mini [46] API to assign a theme to each centroid, based on the reasoning steps associated with it in the reasoning graph of DeepSeek-R1-Distill-Qwen-32B on the GSM8K dataset.

We provided the following system prompt to GPT, and then input multiple reasoning steps corresponding to each centroid as the user prompt:

```
You are a data analyst. The following is an output from a
LLM.
Your task is to carefully read the text and summarize its main
theme in 1-3 English words.
```

**Table 1** presents the assigned **theme** for each centroid, together with **example** reasoning steps that were mapped to that centroid. This analysis revealed that many centroids align with interpretable reasoning patterns. In addition to previously reported in **Figure 2** such as *Add*, *Multiply*, and *Wait*, we identified centroids associated with higher-level computations (e.g., *Calculations Totals*, *Averages*), semantics-bound reasoning (e.g., *Age Calculations*, *Cost Calculations*), and structural elements (e.g., *Answer Formatting*, *Placeholder Tags*).

We also observed centroids linked to reasoning behaviors, such as *Planning*, which reflects the model’s initial steps when approaching a math problem. Moreover, instances of *Wait* appear to split into multiple subtypes. For example, centroids like *Calculation Correctness* and *Reevaluation* capture the model’s tendency to reassess or double-check its output. A distinct centroid labeled *Inconsistencies* highlights cases where the model detects contradictions in its reasoning and attempts to revise its calculations.

Overall, these results indicate that the centroids discovered by the reasoning graph clustering procedure correspond to a wide variety of meaningful and interpretable reasoning techniques.

Table 1: Examples of automatically identified themes and corresponding reasoning steps.

| Theme (Node id)                          | Examples of Reasoning Steps  |
|--|--|
| <b>Calculations — Totals (node 83)</b>   | <ul style="list-style-type: none"> <li><math>5 + 10 = 15</math>. Then, <math>15 + 9 = 24</math>. Finally, <math>24 + 3 = 27</math>.</li> <li>Let me add them step by step. <math>500 + 1500 = 2000</math>. Then, <math>2000 + 125 = 2125</math>.</li> </ul>                        |
| <b>Calculations — Average (node 119)</b> | <ul style="list-style-type: none"> <li>Average = (Sum of all values) / (Number of values).</li> <li>Average Speed = Total Distance / Total Time = 250 miles / 5 hours = 50 mph.</li> </ul>   |
| <b>Calculations — Division (node 41)</b> | <ul style="list-style-type: none"> <li><math>125,000 / 20 = 6,250</math>.</li> <li><math>120 \text{ pieces} / (15 \text{ pieces per pack}) = 8 \text{ packs}</math>.</li> <li><math>80 / 10 = 8 \text{ weeks}</math>.</li> </ul>   |
| <b>Age Calculations (node 147)</b>       | <ul style="list-style-type: none"> <li>Sum of their ages in two years = <math>(B + 2) + (2B + 2) = 28</math>.</li> <li><math>C = 2 \times (\text{James's age in 8 years}) - 5</math>.</li> </ul>   |
| <b>Cost Calculations (node 168)</b>      | <ul style="list-style-type: none"> <li><math>\\$47.00 \times 5 = \\$235.00</math>.</li> <li>Keenan's weekly cost = <math>\\$160 \div 4 = \\$40</math>.</li> <li><math>8 \times 8 = \\$64.00</math>.</li> </ul>   |
| <b>Answer Format (node 137)</b>          | <ul style="list-style-type: none"> <li>The answer is: 50.</li> <li>The answer is: 8.</li> </ul>  |
| <b>Placeholder Tags (node 26)</b>        | <ul style="list-style-type: none"> <li><code>&lt;/think&gt;</code></li> </ul>  |
| <b>Planning (node 163)</b>               | <ul style="list-style-type: none"> <li>Okay, so I need to figure out how much Leah has spent on her new kitten so far. Let me break it down step by step.</li> <li>Hmm, let's break it down step by step.</li> </ul>   |
| <b>Calculation Correctness (node 52)</b> | <ul style="list-style-type: none"> <li>Wait, that seems straightforward, but let me double-check...</li> <li>Wait, let me double-check my calculations to make sure...</li> </ul>  |
| <b>Reevaluation (node 100)</b>           | <ul style="list-style-type: none"> <li>Wait, maybe I made a mistake in the equations.</li> <li>Wait, maybe I made a mistake in the equations. Let me try to model it again...</li> </ul>   |
| <b>Inconsistencies (node 110)</b>        | <ul style="list-style-type: none"> <li>Wait, that's a problem. <math>15 + 8 = 23</math>, which is more than total time 20.</li> <li>Wait, perhaps the shows are part of the 30%, and the other activities are part of the remaining 70%. But that doesn't make sense...</li> </ul> |

## C Measuring the Graph Property Implementation

Here is a sketch of Python code for detecting cycles and computing the diameter of reasoning graphs:

```

from collections import defaultdict, deque
import heapq

def analyze_graph_simple(path, distances):
    adj = defaultdict(list)
    for u, v, w in zip(path, path[1:], distances):
        if u != v:
            adj[u].append((v, w))
    # Cycle detection
    seen, has_loop = set(), False
    loop_count = 0
    entry_node = None
    for i, node in enumerate(path):
        if node in seen:
            has_loop = True
            entry_node = node
            loop_count = path.count(node) - 1
            break
        seen.add(node)

    # Diameter and Avg Path Length
    def dijkstra(u):
        dist = {u: 0}
        heap = [(0, u)]
        while heap:
            d, node = heapq.heappop(heap)
            for neighbor, weight in adj[node]:
                new_dist = d + weight
                if neighbor not in dist or new_dist < dist[neighbor]:
                    dist[neighbor] = new_dist
                    heapq.heappush(heap, (new_dist, neighbor))
        return dist

    all_distances = [dijkstra(node) for node in adj]
    diameter = max(max(dists.values()) for dists in all_distances), default=0)
    avg_path_length = \
        sum(sum(dists.values()) for dists in all_distances) / sum(len(dists)-1 for dists in all_distances)

    # Clustering Coefficient
    undirected = defaultdict(set)
    for u, neighbors in adj.items():
        for v, _ in neighbors:
            undirected[u].add(v)
            undirected[v].add(u)

    clustering_sum, count_cc = 0, 0
    for node, nbrs in undirected.items():
        if len(nbrs) < 2:
            continue
        actual_edges = sum(1 for v in nbrs for w in nbrs if v < w and w in undirected[v])
        clustering_sum += actual_edges / (len(nbrs) * (len(nbrs)-1) / 2)
        count_cc += 1
    avg_clustering = clustering_sum / count_cc if count_cc else 0

    # Small-World Index
    N = len(undirected)
    K = sum(len(nbrs) for nbrs in undirected.values()) / N if N else 0
    C_rand = K / (N - 1) if N > 1 else 0
    L_rand = math.log(N) / math.log(K) if N > 1 and K > 1 else float('inf')

    clustering_norm = avg_clustering / C_rand if C_rand else 0
    path_length_norm = avg_path_length / L_rand if L_rand else 0
    small_world_index = clustering_norm / path_length_norm if path_length_norm else 0

    return has_loop, loop_count, diameter, avg_clustering, avg_path_length, small_world_index

```

## D Details of Large Reasoning Model and Base Model

For the large reasoning models, we utilize distilled models from the DeepSeek-R1 series, which are derived from Qwen-based models. Correspondingly, their base models are the original Qwen and Llama models prior to distillation. A detailed list of these models is provided in [Table 2](#).

Table 2: Comparison of large reasoning models and corresponding base models.

| Base Model                     | Large Reasoning Model                      |
|--------------------------------|--|
| Qwen2.5-Math-1.5B <sup>1</sup> | DeepSeek-R1-Distill-Qwen-1.5B <sup>2</sup> |
| Qwen2.5-Math-7B <sup>3</sup>   | DeepSeek-R1-Distill-Qwen-7B <sup>4</sup>   |
| Llama-3.1-8B <sup>5</sup>      | DeepSeek-R1-Distill-Llama-8B <sup>6</sup>  |
| Qwen2.5-14B <sup>7</sup>       | DeepSeek-R1-Distill-Qwen-14B <sup>8</sup>  |
| Qwen2.5-32B <sup>9</sup>       | DeepSeek-R1-Distill-Qwen-32B <sup>10</sup> |

## E Extension to Non-Math Reasoning Tasks

To examine whether the observed properties of reasoning graphs generalize beyond mathematical reasoning, we extended our evaluation to include two additional tasks inspired by prior work [63]: (i) **StrategyQA** [22], a multi-hop question answering dataset, and (ii) **LogicalDeduction**, a logical reasoning dataset from BIG-Bench [54]. In [Figure 10](#), we compare reasoning graph properties between DeepSeek-R1-Distill-Qwen-32B (reasoning model) and Qwen2.5-32B (base model) on these datasets. Similar to the math tasks reported in the main paper, the reasoning model exhibits markedly different graph characteristics, including higher cycle rates, larger diameters, and stronger small-world properties. These consistent patterns across multiple domains provide additional evidence that the reasoning graph framework captures structural properties in a general manner.

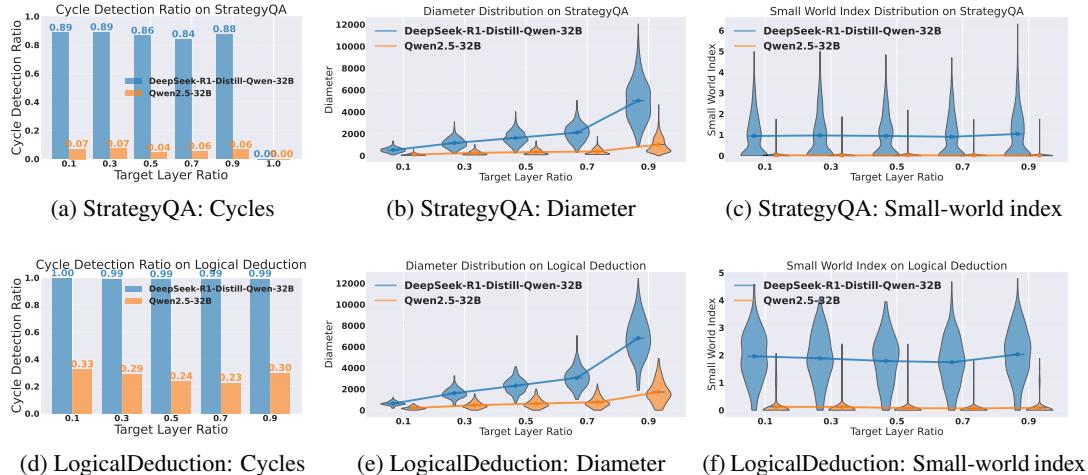


Figure 10: Comparison of reasoning graph properties between Distill-Qwen-32B and Qwen2.5-32B on two non-math reasoning datasets: (top row) StrategyQA and (bottom row) LogicalDeduction. Columns correspond to Cycles, Diameter, and Small-world index.

<sup>1</sup><https://huggingface.co/Qwen/Qwen2.5-Math-1.5B>

<sup>2</sup><https://huggingface.co/deepseek-ai/DeepSeek-R1-Distill-Qwen-1.5B>

<sup>3</sup><https://huggingface.co/Qwen/Qwen2.5-Math-7B>

<sup>4</sup><https://huggingface.co/deepseek-ai/DeepSeek-R1-Distill-Qwen-7B>

<sup>5</sup><https://huggingface.co/meta-llama/Llama-3.1-8B>

<sup>6</sup><https://huggingface.co/deepseek-ai/DeepSeek-R1-Distill-Llama-8B>

<sup>7</sup><https://huggingface.co/Qwen/Qwen2.5-14B>

<sup>8</sup><https://huggingface.co/deepseek-ai/DeepSeek-R1-Distill-Qwen-14B>

<sup>9</sup><https://huggingface.co/Qwen/Qwen2.5-32B>

<sup>10</sup><https://huggingface.co/deepseek-ai/DeepSeek-R1-Distill-Qwen-32B>

## F Experiments with Different $K$ Values in $K$ -Means Clustering

We report differences in cycle detection ratios when varying the number of clusters in  $K$ -means clustering, specifically for the GSM8K dataset, as shown in Figure 11. As expected, decreasing results in fewer clusters and a higher ratio of detected cycles. Across all values, the large reasoning model consistently exhibits a higher cycle ratio compared to the base model.

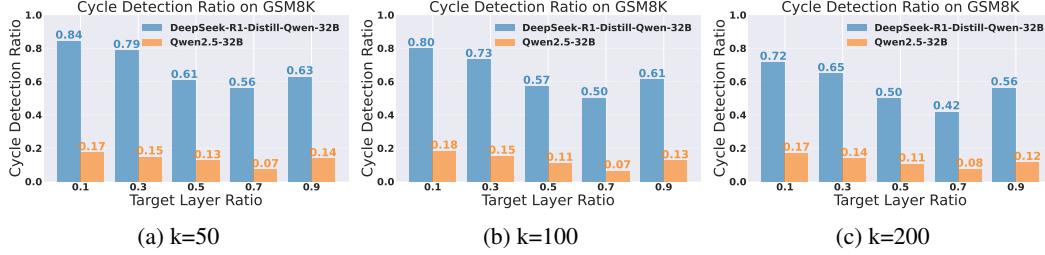


Figure 11: Comparison of cycle detection ratios across different layers for the **large reasoning model (DeepSeek-R1-Distill-Qwen-32B)** and the **base model (Qwen2.5-32B)**, evaluated on three tasks: **(a)**  $k = 50$ , **(b)**  $k = 100$ , and **(c)**  $k = 200$ . Results consistently show that the large reasoning model exhibits significantly higher cycle detection ratios than the base model at all layer ratios and  $k$ .

## G Diameter Analysis on MATH500 and AIME 2024

Figure 12 compares reasoning graph diameters for the MATH500 and AIME 2024 datasets. The large reasoning model consistently exhibits greater diameters than the base model, with a clear trend of increasing diameter in deeper hidden layers, aligning with observations from the GSM8K dataset.

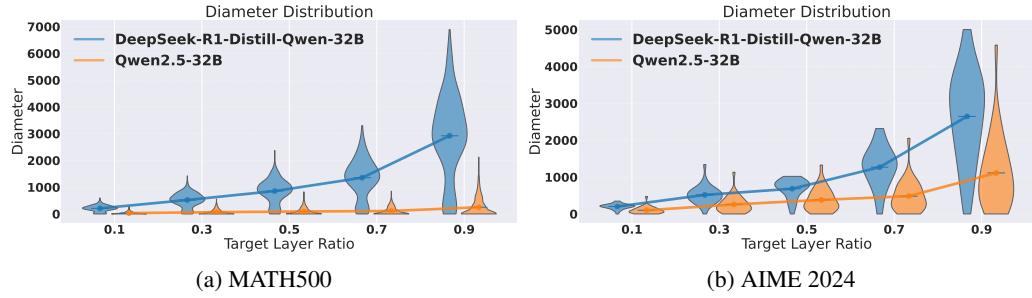


Figure 12: Diameter of Reasoning Graph in DeepSeek-R1-Distill-Qwen-32B and Qwen2.5-32B.

## H Layer-wise Clustering Coefficient and Average Path Length

Figure 13 shows the clustering coefficient and average path length for each layer. The large reasoning model consistently exhibits higher clustering coefficients at all layer ratios, contributing to its enhanced small-world characteristics.

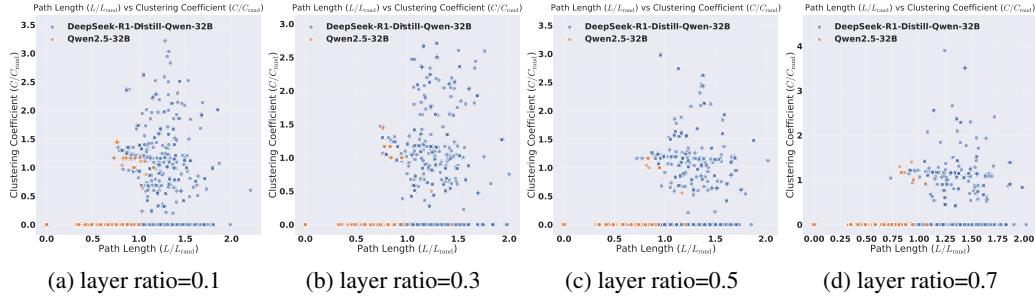


Figure 13: The clustering coefficient and average path length for each layer.

## I Model Size and Small-World Index

Figure 14 presents the relationship between model size and the Small-World Index. The results suggest that the Small-World property becomes more pronounced as the model size increases.

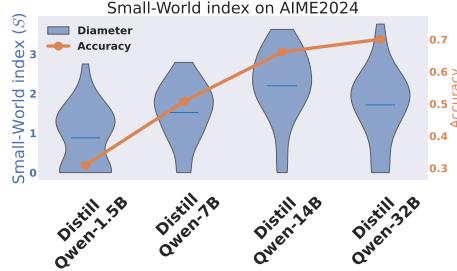


Figure 14: Relationship between model size and the Small-World Index.

## J Training Details

We conducted supervised fine-tuning (SFT) experiments using the Qwen2.5-32B-Instruct model as our base model. The training was executed on a computing node equipped with 8 NVIDIA H200 GPUs for training and a single NVIDIA H200 GPU for inference.

The detailed training configuration is summarized in [Table 3](#).

Table 3: Detailed training configuration for the SFT experiments.

| Parameter                   | Value   |
|-----------------------------|---|
| Base Model                  | Qwen2.5-32B-Instruct                              |
| Dataset                     | simplescaling/s1K or simplescaling/s1K-1.1        |
| Number of Epochs            | 5   |
| Learning Rate               | $1 \times 10^{-5}$                                |
| Learning Rate Scheduler     | Cosine (minimum LR: 0)                            |
| Batch Size                  | 8 (Effective: 8 GPUs $\times$ micro-batch size 1) |
| Gradient Accumulation Steps | 1   |
| Weight Decay                | $1 \times 10^{-4}$                                |
| Optimizer                   | AdamW ( $\beta_1 = 0.9, \beta_2 = 0.95$ )         |
| Warmup Ratio                | 0.05  |
| Precision                   | bf16  |
| Gradient Checkpointing      | Enabled   |
| FSDP Configuration          | Full Shard Data Parallel (auto-wrap)              |
| Block Size                  | 32768 tokens                                      |

Inference was conducted using a single NVIDIA H200 GPU to evaluate trained models and generate results presented in the paper.

## K Diameter of other s1 checkpoints

[Figure 15](#) compares the diameters of reasoning graphs at 100 and 500 training steps using the S1 dataset [44]. In both cases, version s1-v1.1 demonstrates larger diameters compared to s1-v1.0, and the diameters tend to increase with further training steps.

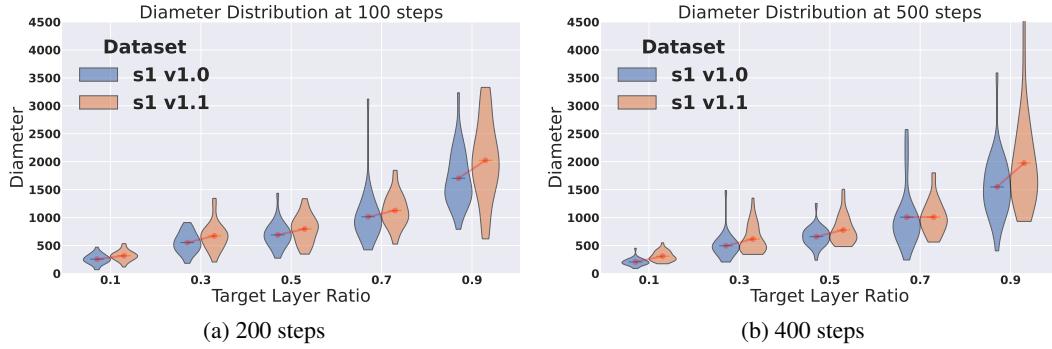


Figure 15: Cycle Graph Detection Ratio in DeepSeek-R1-Distill-Qwen-32B and Qwen2.5-32B.

## L Impact of SFT Data Quality on Reasoning Graph Structure.

To further investigate the relationship between data quality and reasoning-graph properties, we compared two supervised fine-tuning (SFT) datasets: LIMO [72] and s1 v1.0 [44]. The s1 dataset has previously been recognized for its strong performance in enhancing reasoning abilities [44].

We constructed reasoning graphs from the hidden states of Qwen2.5-32B-Instruct, prompted with data from s1 and LIMO (note that we did not fine-tune Qwen2.5-32B-Instruct on these datasets).

As shown in Figure 16, reasoning graphs derived from s1 consistently exhibit larger diameters and higher cycle counts across all examined layer depths. This indicates that the s1 dataset inherently induces exploration of a broader range of latent reasoning states, resulting in more iterative reasoning. In contrast, graphs derived from LIMO show narrower reasoning trajectories with fewer cycles, suggesting more linear and potentially shallow reasoning processes.

These findings suggest that higher-quality SFT data possess richer reasoning-graph characteristics (such as increased cycles and larger diameters), which in turn contribute to improved performance when used for fine-tuning. Thus, reasoning-graph analysis from hidden states offers a novel perspective and practical guidance for creating better SFT datasets.

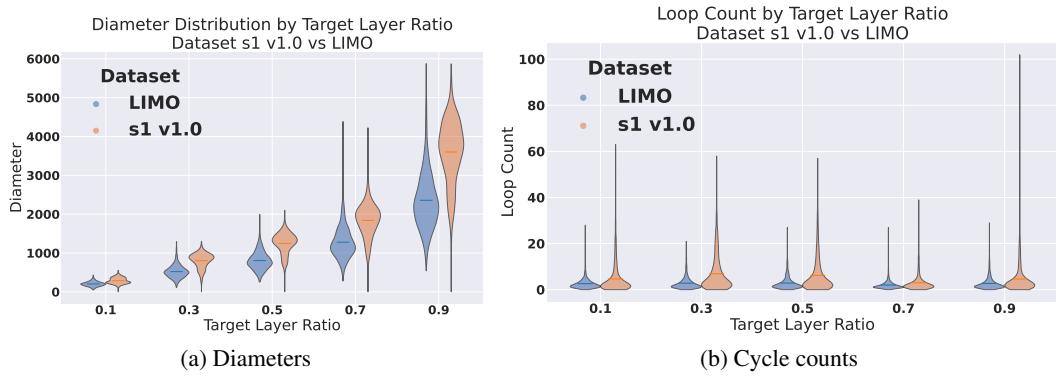


Figure 16: We constructed reasoning graphs from the hidden states of Qwen2.5-32B-Instruct, prompted with data from s1 and LIMO (note that we did not fine-tune Qwen2.5-32B-Instruct on these datasets). The s1 dataset—regarded as higher-quality—consistently yields larger diameters and higher cycle counts, indicating broader exploration and more reflective reasoning behavior.

## M Limitations

This study introduces the concept of reasoning graphs as a tool to identify distinctive graph-theoretic properties that may explain recent breakthroughs in the reasoning performance of large language models (LLMs). While our findings provide a novel explanatory perspective on the reasoning capabilities of advanced models, concrete guidelines on constructing models with superior reasoning performance remain insufficient. Although we experimentally examine the relationship between graph properties and reasoning-SFT in Section 5, using these insights as a first step toward building more effective reasoning models is left for future work.

Our analysis focuses on transitions in the context direction of hidden states, but it does not provide feature-level [25, 56, 20, 37, 39] or circuit-level analyses [45, 61, 40] as commonly studied in mechanistic interpretability [3, 52]. Furthermore, how the distinctive graph properties observed in reasoning models emerge during training dynamics [16, 60, 41] remains an open question, which we leave as an important direction for future work.

## NeurIPS Paper Checklist

### 1. Claims

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

Answer: [\[Yes\]](#)

Justification: The abstract and introduction clearly state that analyzing reasoning graph properties provides insights into the capabilities of large reasoning models, aligning well with the paper's empirical results and scope.

Guidelines:

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

### 2. Limitations

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

Answer: [\[Yes\]](#)

Justification: Limitations are explicitly discussed in [Appendix M](#), covering assumptions and empirical scope.

Guidelines:

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

### 3. Theory assumptions and proofs

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

Answer: [NA]

Justification: This paper is empirical and does not present theoretical results.

Guidelines:

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

#### 4. Experimental result reproducibility

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

Answer: [Yes]

Justification: The appendix provides detailed Python code snippets illustrating methods for measuring graph properties, ensuring reproducibility.

Guidelines:

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

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

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

Answer: [\[Yes\]](#)

Justification: The code is provided in a single submission zip file with supplementary material, facilitating full reproduction.

Guidelines:

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

## 6. Experimental setting/details

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

Answer: [\[Yes\]](#)

Justification: Training details for supervised fine-tuning (SFT) are comprehensively presented in the [Appendix J](#).

Guidelines:

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

## 7. Experiment statistical significance

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

Answer: [\[Yes\]](#)

Justification: The [Appendix J](#) clearly describes training details and includes distribution analyses of sample results, supporting statistical significance.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer “Yes” if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

## 8. Experiments compute resources

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

Answer: [\[Yes\]](#)

Justification: Detailed descriptions of computational resources used for experiments are provided in the [Appendix J](#).

Guidelines:

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

## 9. Code of ethics

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

Answer: [\[Yes\]](#)

Justification: The research adheres fully to the ethical guidelines, involving no violation of ethical standards in methodology or reporting.

Guidelines:

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

## 10. Broader impacts

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

Answer: [\[Yes\]](#)

Justification: Broader societal impacts are explicitly discussed in the ??.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.

- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

## 11. Safeguards

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

Answer: [\[Yes\]](#)

Justification: Safeguards for the responsible release and use of models and data are described in the [Appendix A](#).

Guidelines:

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

## 12. Licenses for existing assets

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

Answer: [\[Yes\]](#)

Justification: All used assets, including datasets and pre-existing models, are properly cited, and their respective licenses and terms of use are explicitly mentioned.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.

- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, [paperswithcode.com/datasets](https://paperswithcode.com/datasets) has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

### 13. New assets

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

Answer: [\[Yes\]](#)

Justification: The [Appendix C](#) includes thorough documentation for all newly introduced code and datasets.

Guidelines:

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

### 14. Crowdsourcing and research with human subjects

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

Answer: [\[NA\]](#)

Justification: This research does not involve crowdsourcing or human subjects.

Guidelines:

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

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

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

Answer: [\[NA\]](#)

Justification: This research does not involve crowdsourcing or human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

#### 16. Declaration of LLM usage

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

Answer: [Yes]

Justification: The [Appendix D](#) clearly documents the large language models used in experiments and mentions LLM usage in writing and formatting.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (<https://neurips.cc/Conferences/2025/LLM>) for what should or should not be described.