REASONING EFFORT AND PROBLEM COMPLEXITY: A SCALING ANALYSIS IN LLMS

Benjamin Estermann
ETH Zürich
Switzerland
estermann@ethz.ch

Roger Wattenhofer
ETH Zürich
Switzerland
wattenhofer@ethz.ch

ABSTRACT

Large Language Models (LLMs) have demonstrated remarkable text generation capabilities, and recent advances in training paradigms have led to breakthroughs in their reasoning performance. In this work, we investigate how the reasoning effort of such models scales with problem complexity. We use the infinitely scalable Tents puzzle, which has a known linear-time solution, to analyze this scaling behavior. Our results show that reasoning effort scales with problem size, but only up to a critical problem complexity. Beyond this threshold, the reasoning effort does not continue to increase, and may even decrease. This observation highlights a critical limitation in the logical coherence of current LLMs as problem complexity increases, and underscores the need for strategies to improve reasoning scalability. Furthermore, our results reveal significant performance differences between current state-of-the-art reasoning models when faced with increasingly complex logical puzzles.

1 INTRODUCTION

Large language models (LLMs) have demonstrated remarkable abilities in a wide range of natural language tasks, from text generation to complex problem-solving. Recent advances, particularly with models trained for enhanced reasoning, have pushed the boundaries of what machines can achieve in tasks requiring logical inference and deduction. A critical factor in the success of these advanced models is the ability to leverage increased computational resources at test time, allowing them to explore more intricate solution spaces. This capability raises a fundamental question: *how does the "reasoning effort" of these models scale as the complexity of the problem increases?*

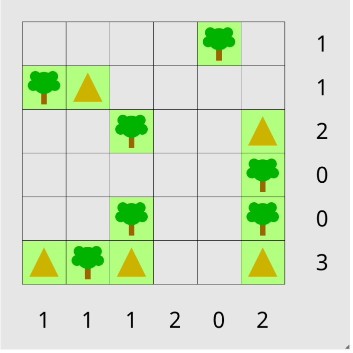


Figure 1: An example instance of a partially solved 6 by 6 tents puzzle. Tents need to be placed next to trees, away from other tents and fulfilling the row and column constraints.

Understanding this scaling relationship is crucial for several reasons. First, it sheds light on the fundamental nature of reasoning within LLMs, moving beyond simply measuring accuracy on isolated tasks. By examining how the computational demands, reflected in token usage, evolve with problem difficulty, we can gain insights into the efficiency and potential bottlenecks of current LLM architectures. Second, characterizing this scaling behavior is essential for designing more effective and resource-efficient reasoning models in the future.

In this work, we address this question by investigating the scaling of reasoning effort in LLMs using a specific, infinitely scalable logic puzzle: the Tents puzzle[1] (see Figure 1). This puzzle offers a controlled environment for studying algorithmic reasoning, as its problem size can be systematically increased, and it possesses a known linear-time

[1] The puzzle is available to play in the browser at https://www.chiark.greenend.org.uk/~sgtatham/puzzles/js/tents.html

solution. Our analysis focuses on how the number of tokens used by state-of-the-art reasoning LLMs changes as the puzzle grid size grows. In addition to reasoning effort, we also evaluate the success rate across different puzzle sizes to provide a comprehensive view of their performance.

2 Related Work

The exploration of reasoning abilities in large language models (LLMs) is a rapidly evolving field with significant implications for artificial intelligence. Several benchmarks have been developed to evaluate the reasoning capabilities of LLMs across various domains. These benchmarks provide standardized tasks and evaluation metrics to assess and compare different models. Notable benchmarks include GSM8K (Cobbe et al., 2021), ARC-AGI (Chollet, 2019), GPQA (Rein et al., 2023), MMLU (Hendrycks et al., 2020), SWE-bench (Jimenez et al., 2023) and NPhard-eval (Fan et al., 2023). These benchmarks cover topics from mathematics to commonsense reasoning and coding. More recently, also math competitions such as AIME2024 (of America, 2024) have been used to evaluate the newest models. Estermann et al. (2024) have introduced PUZZLES, a benchmark focusing on algorithmic and logical reasoning for reinforcement learning. While PUZZLES does not focus on LLMs, except for a short ablation in the appendix, we argue that the scalability provided by the underlying puzzles is an ideal testbed for testing extrapolative reasoning capabilities in LLMs.

The reasoning capabilities of traditional LLMs without specific prompting strategies are quite limited (Huang & Chang, 2022). Using prompt techniques such as chain-of-thought (Wei et al., 2022), least-to-most (Zhou et al., 2022) and tree-of-thought (Yao et al., 2023), the reasoning capabilities of traditional LLMs can be greatly improved. Lee et al. (2024) have introduced the Language of Thought Hypothesis, based on the idea that human reasoning is rooted in language. Lee et al. propose to see the reasoning capabilities through three different properties: Logical coherence, compositionality and productivity. In this work we will mostly focus on the aspect of algorithmic reasoning, which falls into logical coherence. Specifically, we analyze the limits of logical coherence.

With the release of OpenAI's o1 model, it became apparent that new training strategies based on reinforcement learning are able to boost the reasoning performance even further. Since o1, there now exist a number of different models capable of enhanced reasoning (Guo et al., 2025; DeepMind, 2025; Qwen, 2024; OpenAI, 2025). Key to the success of these models is the scaling of test-time compute. Instead of directly producing an answer, or thinking for a few steps using chain-of-thought, the models are now trained to think using several thousands of tokens before coming up with an answer.

3 Methods

3.1 The Tents Puzzle Problem

In this work, we employ the Tents puzzle, a logic problem that is both infinitely scalable and solvable in linear time[2], making it an ideal testbed for studying algorithmic reasoning in LLMs. The Tents puzzle, popularized by Simon Tatham's Portable Puzzle Collection (Tatham), requires deductive reasoning to solve. The puzzle is played on a rectangular grid, where some cells are pre-filled with trees. The objective is to place tents in the remaining empty cells according to the following rules:

- no two tents are adjacent, even diagonally
- there are exactly as many tents as trees and the number of tents in each row and column matches the numbers around the edge of the grid
- it is possible to match all tents to trees so that each tent is orthogonally adjacent to its own tree (a tree may also be adjacent to other tents).

An example instance of the Tents puzzle is visualized in Figure 1 in the Introduction. The scalability of the puzzle is achieved by varying the grid dimensions, allowing for systematic exploration of problem complexity. Where not otherwise specified, we used the ”easy” difficulty preset available

[2] See a description of the algorithm of the solver as part of the PUZZLES benchmark here: https://github.com/ETH-DISCO/rlp/blob/main/puzzles/tents.c#L206C3-L206C67

in the Tents puzzle generator, with ”tricky” being evaluated in Appendix A.2.1. Crucially, the Tents puzzle is designed to test extrapolative reasoning as puzzle instances, especially larger ones, are unlikely to be present in the training data of LLMs. We utilized a text-based interface for the Tents puzzle, extending the code base provided by Estermann et al. (2024) to generate puzzle instances and represent the puzzle state in a format suitable for LLMs.

Models were presented with the same prompt (detailed in Appendix A.1) for all puzzle sizes and models tested. The prompt included the puzzle rules and the initial puzzle state in a textual format. Models were tasked with directly outputting the solved puzzle grid in JSON format. This one-shot approach contrasts with interactive or cursor-based approaches previously used in (Estermann et al., 2024), isolating the reasoning process from potential planning or action selection complexities.

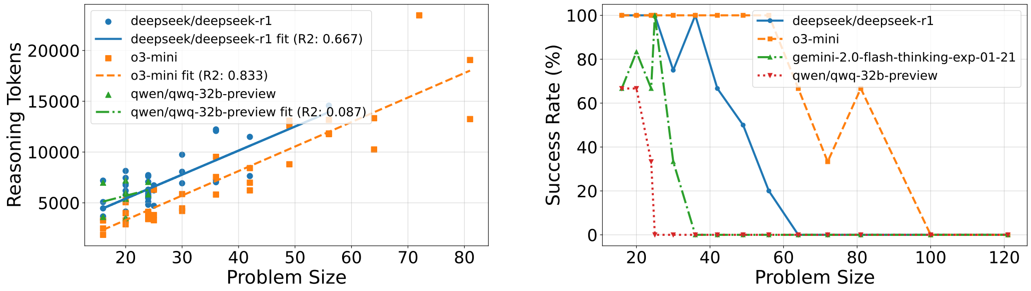
3.2 Evaluation Criteria

Our evaluation focuses on two key metrics: *success rate* and *reasoning effort*. Success is assessed as a binary measure: whether the LLM successfully outputs a valid solution to the Tents puzzle instance, adhering to all puzzle rules and constraints. We quantify problem complexity by the *problem size*, defined as the product of the grid dimensions (rows × columns). To analyze the scaling of reasoning effort, we measure the *total number of tokens* generated by the LLMs to produce the final answer, including all thinking tokens. We hypothesize a linear scaling relationship between problem size and reasoning effort, and evaluate this hypothesis by fitting a linear model to the observed token counts as a function of problem size. The goodness of fit is quantified using the R^2 metric, where scores closer to 1 indicate that a larger proportion of the variance in reasoning effort is explained by a linear relationship with problem size. Furthermore, we analyze the percentage of correctly solved puzzles across different problem sizes to assess the performance limits of each model.

3.3 Considered Models

We evaluated the reasoning performance of the following large language models known for their enhanced reasoning capabilities: Gemini 2.0 Flash Thinking (DeepMind, 2025), OpenAI o3-mini (OpenAI, 2025), DeepSeek R1 Guo et al. (2025), and Qwen/QwQ-32B-Preview Qwen (2024).

4 Results



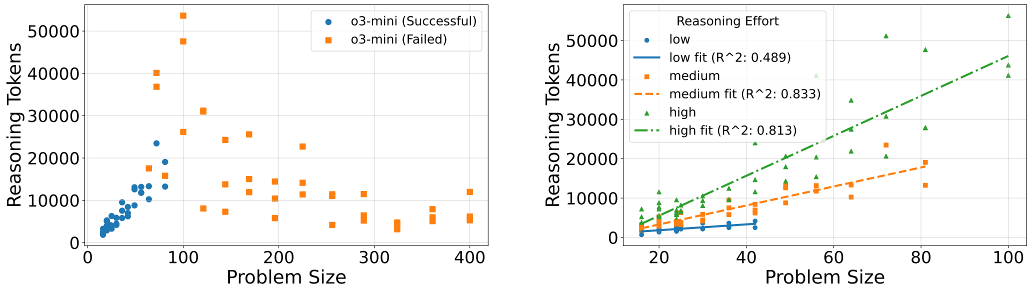
(a)

(b)

Figure 2: (a) Reasoning effort in number of reasoning tokens versus problem size for DeepSeek R1, o3-mini, and Qwen/QwQ-32B-Preview. Successful attempts only. Linear fits are added for each model. Gemini 2.0 Flash Thinking is excluded due to unknown number of thinking tokens.
(b) Solved percentage versus problem size for all models. No model solved problems larger than size 100. o3-mini achieves the highest success rate, followed by DeepSeek R1 and Gemini 2.0 Flash Thinking. Qwen/QwQ-32B-Preview struggles with problem instances larger than size 20.

The relationship between reasoning effort and problem size reveals interesting scaling behaviors across the evaluated models. Figure 2a illustrates the scaling of reasoning effort, measured by the number of reasoning tokens, as the problem size increases for successfully solved puzzles. For DeepSeek R1 and o3-mini, we observe a roughly linear increase in reasoning effort with problem size. Notably, the slopes of the linear fits for R1 and o3-mini are very similar, suggesting comparable scaling behavior in reasoning effort for these models, although DeepSeek R1 consistently uses more tokens than o3-mini across problem sizes. Qwen/QwQ-32B-Preview shows a weaker linear correlation, likely due to the limited number of larger puzzles it could solve successfully.

The problem-solving capability of the models, shown in Figure 2b, reveals performance limits as problem size increases. None of the models solved puzzles with a problem size exceeding 100. o3-mini demonstrates the highest overall solvability, managing to solve the largest problem instances, followed by DeepSeek R1 and Gemini 2.0 Flash Thinking. Qwen/QwQ-32B-Preview's performance significantly degrades with increasing problem size, struggling to solve instances larger than 25.



(a)

(b)

Figure 3: (a) Reasoning effort in number of reasoning tokens versus problem size for o3-mini. A peak in reasoning effort is observed around problem size 100, followed by a decline for larger problem sizes. (b) Reasoning effort in number of reasoning tokens versus problem size for o3-mini, categorized by low, medium, and high reasoning effort strategies. Steeper slopes are observed for higher reasoning effort strategies. High reasoning effort enables solving larger instances but also increases token usage for smaller, already solvable problems.

A more detailed analysis of o3-mini's reasoning effort (Figure 3a) reveals a non-monotonic trend. While generally increasing with problem size initially, reasoning effort peaks around a problem size of 100. Beyond this point, the reasoning effort decreases, suggesting a potential ”frustration” effect where increased complexity no longer leads to proportionally increased reasoning in the model. The same behavior could not be observed for other models, see Appendix A.2.2. It would be interesting to see the effect of recent works trying to optimize reasoning length would have on these results (Luo et al., 2025).

Figure 3b further explores o3-mini's behavior by categorizing reasoning effort into low, medium, and high strategies. The steepness of the scaling slope increases with reasoning effort, indicating that higher effort strategies lead to a more pronounced increase in token usage as problem size grows. While high reasoning effort enables solving larger puzzles (up to 10x10), it also results in a higher token count even for smaller problems that were already solvable with lower effort strategies. This suggests a trade-off where increased reasoning effort can extend the solvable problem range but may also introduce inefficiencies for simpler instances.

5 Conclusion

This study examined how reasoning effort scales in LLMs using the Tents puzzle. We found that reasoning effort generally scales linearly with problem size for solvable instances. Model performance varied, with o3-mini and DeepSeek R1 showing better performance than Qwen/QwQ-32B-Preview and Gemini 2.0 Flash Thinking. These results suggest that while LLMs can adapt reasoning effort to problem complexity, their logical coherence has limits, especially for larger problems. Future work should extend this analysis to a wider variety of puzzles contained in the PUZZLES benchmark to include puzzles with different algorithmic complexity. These insights could lead the way to find strategies to improve reasoning scalability and efficiency, potentially by optimizing reasoning length or refining prompting techniques. Understanding these limitations is crucial for advancing LLMs in complex problem-solving.

References

François Chollet. On the measure of intelligence. *arXiv preprint arXiv:1911.01547*, 2019.

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

DeepMind. Gemini flash thinking. `https://deepmind.google/technologies/gemini/flash-thinking/`, 2025. Accessed: February 6, 2025.

Benjamin Estermann, Luca A Lanzendörfer, Yannick Niedermayr, and Roger Wattenhofer. Puzzles: A benchmark for neural algorithmic reasoning. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024.

Lizhou Fan, Wenyue Hua, Lingyao Li, Haoyang Ling, and Yongfeng Zhang. Nphardeval: Dynamic benchmark on reasoning ability of large language models via complexity classes. *arXiv preprint arXiv:2312.14890*, 2023.

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

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*, 2020.

Jie Huang and Kevin Chen-Chuan Chang. Towards reasoning in large language models: A survey. *arXiv preprint arXiv:2212.10403*, 2022.

Carlos E Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik Narasimhan. Swe-bench: Can language models resolve real-world github issues? *arXiv preprint arXiv:2310.06770*, 2023.

Seungpil Lee, Woochang Sim, Donghyeon Shin, Wongyu Seo, Jiwon Park, Seokki Lee, Sanha Hwang, Sejin Kim, and Sundong Kim. Reasoning abilities of large language models: In-depth analysis on the abstraction and reasoning corpus. *ACM Transactions on Intelligent Systems and Technology*, 2024.

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

Mathematical Association of America. 2024 aime i problems. `https://artofproblemsolving.com/wiki/index.php/2024_AIME_I`, 2024. Accessed: February 6, 2025.

OpenAI. Openai o3 mini. `https://openai.com/index/openai-o3-mini/`, 2025. Accessed: February 6, 2025.

Qwen. Qwq: Reflect deeply on the boundaries of the unknown, November 2024. URL `https://qwenlm.github.io/blog/qwq-32b-preview/`.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R Bowman. Gpqa: A graduate-level google-proof q&a benchmark. *arXiv preprint arXiv:2311.12022*, 2023.

Simon Tatham. Simon tatham's portable puzzle collection. `https://www.chiark.greenend.org.uk/~sgtatham/puzzles/`. Accessed: 2025-02-06.

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

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models, may 2023. *arXiv preprint arXiv:2305.10601*, 14, 2023.

Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Claire Cui, Olivier Bousquet, Quoc Le, et al. Least-to-most prompting enables complex reasoning in large language models. *arXiv preprint arXiv:2205.10625*, 2022.

A APPENDIX

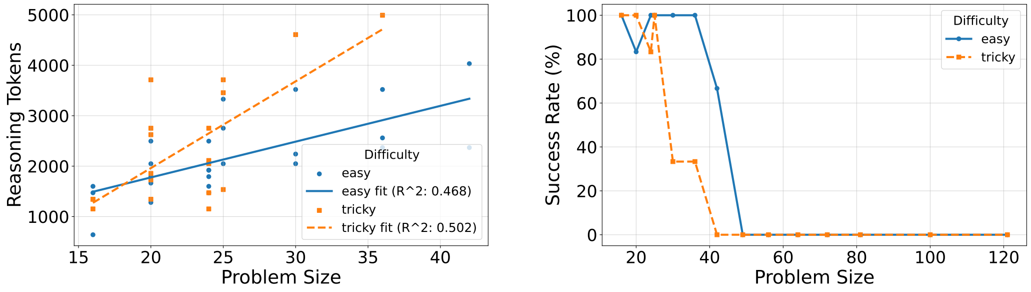
A.1 FULL PROMPT

The full prompt used in the experiments is the following, on the example of a 4x4 puzzle:

```
You are a logic puzzle expert. You will be given a logic puzzle to
 solve. Here is a description of the puzzle:
You have a grid of squares, some of which contain trees. Your aim
is to place tents in some of the remaining squares, in such a way
that the following conditions are met:
There are exactly as many tents as trees.
The tents and trees can be matched up in such a way that each tent
 is directly adjacent (horizontally or vertically, but not
diagonally) to its own tree. However, a tent may be adjacent to
other trees as well as its own.
No two tents are adjacent horizontally, vertically or diagonally.
The number of tents in each row, and in each column, matches the
numbers given in the row or column constraints.
Grass indicates that there cannot be a tent in that position.
You receive an array representation of the puzzle state as a grid.
 Your task is to solve the puzzle by filling out the grid with the
 correct values. You need to solve the puzzle on your own, you
cannot use any external resources or run any code. Once you have
solved the puzzle, tell me the final answer without explanation.
Return the final answer as a JSON array of arrays.
Here is the current state of the puzzle as a string of the
internal state representation:
A 0 represents an empty cell, a 1 represents a tree, a 2
represents a tent, and a 3 represents a grass patch.
Tents puzzle state:
Current grid:
[[0 0 1 0]
 [0 1 0 0]
 [1 0 0 0]
 [0 0 0 0]]
The column constraints are the following:
[1 1 0 1]
The row constraints are the following:
[2 0 0 1]
```

A.2 ADDITIONAL FIGURES

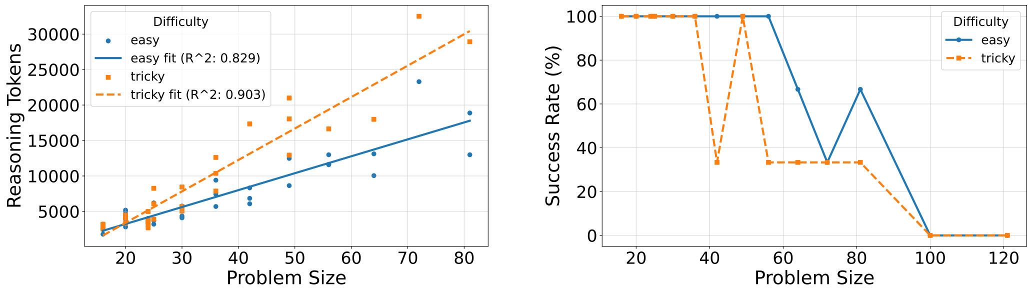
A.2.1 EASY VS. TRICKY PUZZLES



(a)

(b)

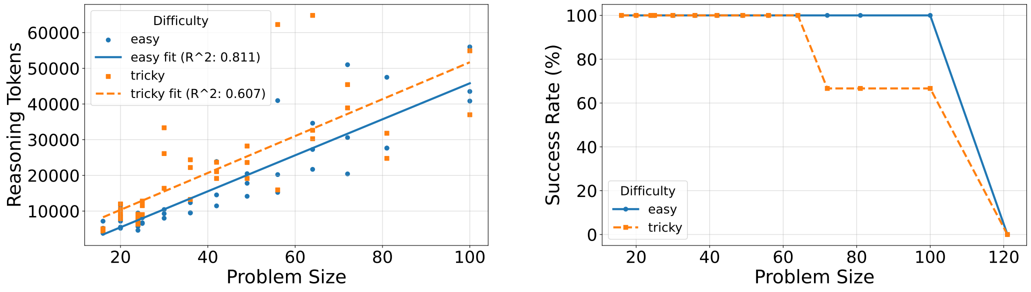
Figure 4: (a) Reasoning effort in number of reasoning tokens versus problem size for o3-mini with reasoning effort **low**. Successful tries only. Linear fits are added for each model. (b) Solved percentage versus problem size for o3-mini with reasoning effort **low**.



(a)

(b)

Figure 5: (a) Reasoning effort in number of reasoning tokens versus problem size for o3-mini with reasoning effort **medium**. Successful tries only. Linear fits are added for each model. (b) Solved percentage versus problem size for o3-mini with reasoning effort **medium**.



(a)

(b)

Figure 6: (a) Reasoning effort in number of reasoning tokens versus problem size for o3-mini with reasoning effort **high**. Successful tries only. Linear fits are added for each model. (b) Solved percentage versus problem size for o3-mini with reasoning effort **high**.

A.2.2 Reasoning Effort for All Models

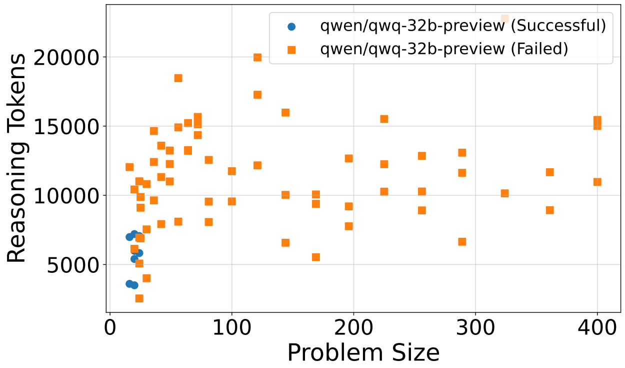


Figure 7: Reasoning effort in tokens for Qwen QwQ.

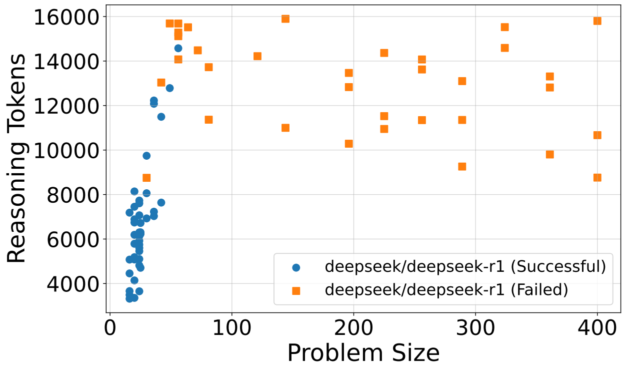


Figure 8: Reasoning effort in tokens for Deepseek R1.

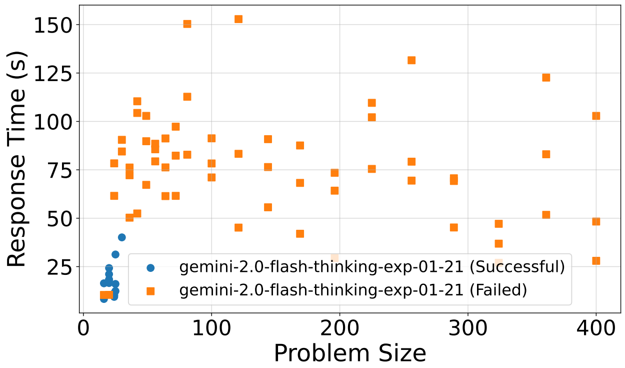


Figure 9: Reasoning effort quantified by response time for Gemini-2.0-flash-thinking.

A.3 Cost

Total cost of these experiments was around 80 USD in API credits.