Evaluating the Safety and Skill Reasoning of Large Reasoning Models Under Compute Constraints

Adarsha Balaji
Mathematics and Computer Science Division
Argonne National Laboratory
Lemont, IL 60439
abalaji@anl.gov

Le Chen
Data Science and Learning Division
Argonne National Laboratory
Lemont, IL 60439
lechen@anl.gov

Rajeev Thakur
Data Science and Learning Division
Argonne National Laboratory
Lemont, IL 60439
thakur@anl.gov

Franck Cappello
Mathematics and Computer Science Division
Argonne National Laboratory
Lemont, IL 60439
cappello@mcs.anl.gov

Sandeep Madireddy
Mathematics and Computer Science Division
Argonne National Laboratory
Lemont, IL 60439
smadireddy@anl.gov

Abstract

Test-time compute scaling has demonstrated the ability to improve the performance of reasoning language models by generating longer chain-of-thought (CoT) sequences. However, this increase in performance comes with an increase in computational cost. In this work, we investigate two compute constraint strategies: (1) reasoning length constraint and (2) model quantization, as methods to reduce the compute demand of reasoning models and study their impact on their safety performance. Specifically, we explore two approaches to apply compute constraints to reasoning models: (1) fine-tuning reasoning models using a length-controlled policy optimization (LCPO) based reinforcement learning method to satisfy a user-defined CoT reasoning length, and (2) applying quantization to maximize the generation of CoT sequences within a user-defined compute constraint. Furthermore, we study the trade-off between the computational efficiency and the safety of the model. We demonstrate that under a fixed compute budget, quantized reasoning models, that reason for longer (more reasoning tokens), perform at par with full-precision reasoning models.

1 Introduction

Existing benchmarks and evaluation protocols fall short of capturing the trade-offs between compute at test-time i.e. reasoning length and performance i.e. accuracy and safety of large reasoning models (LRMs). First, they often conflate raw performance with computational cost Xu et al. (2025), either by increasing model size or by increasing their reasoning budget at inference time Snell et al. (2024). For example, a leader-board for math problem solving might rank models solely by final answer accuracy, implicitly rewarding those that might use extremely long chain-of-thought or multiple-sample voting to get a higher score. Such improvements typically come with higher computational costs, often scaling with the size (parameters) of LRMs.

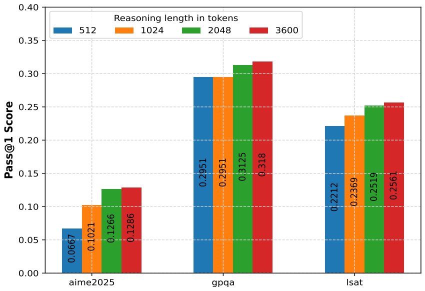

Figure 1: Performance of the baseline L1 model for science and math reasoning skills for increasing reasoning length. We observe that the performance of the reasoning model improves with an increase in the number of reasoning tokens for math and science skills.

In practice, such performance differences matter at test-time, yet evaluations rarely report the average token usage or compute time per question. Only recently have reasoning models such as S1 Muennighoff et al. (2025) and L1 Aggarwal & Welleck (2025) begun to emphasize efficiency metrics such as accuracy as a function of the compute budget (tokens). Without these, researchers risk pursuing methods that yield marginal accuracy gains at disproportionate compute costs. Moreover, performance is often tied to the number of steps a model takes to complete a task without considering its size (parameters) or the number of operations per step (FLOPs). For instance, GPT-4, a 1.76 trillion parameters model, might solve a puzzle in 50 steps whereas a much smaller model might need 500; the compute budget for the two models completing the same task are vastly different. In addition to model size, quantization can often balance efficiency and performance by reducing the compute and memory footprint of a model with minimal loss in performance Li et al. (2025). Therefore, it is necessary to evaluate the performance of reasoning models as a function of both accuracy and compute efficiency, rather than accuracy alone. While compute constraints such as reasoning budgets and quantization have been studied, separately, for general, science, and math reasoning, their combined impact on model skill and safety performance remains unexplored.

In summary, our contributions are:

- We study the impact of test-time compute-constraint methods, such as constrained reasoning length and quantization, on the skill and safety performance of LRMs;
- We adopt the Length Controlled Policy Optimization reinforcement learning method, presented in Aggarwal & Welleck (2025), to safety fine-tune a reasoning model with a precise user-defined length control using the SafeChain dataset Jiang et al. (2025);
- We apply weight quantization (GPTQ) methods on the baseline and safety fine-tuned models and study its impact on skill and safety performance for increasing reasoning chain-of-thought lengths;
- We analyze the trade-off between the two compute-constraint methods under a compute budget.
- We demonstrate that under a fixed compute budget, quantized reasoning models, that reason for longer (more reasoning tokens), perform at par with full-precision reasoning models.

2 Related Works

Recent research on scaling laws has established that model performance follows predictable power-law relationships with respect to the number of parameters, the size of the training dataset, and the available compute Kaplan et al. (2020).

Test time compute performance scales with increased compute - the more reasoning tokens a model can generate, the more accurate its response

Reasoning Models. The emergence of Large Reasoning Models (LRMs) reflects a shift from treating reasoning as an incidental property of LLMs to deliberately training models to "think before they answer." Early methods such as Chain-of-Thought (CoT) prompting Wei et al. (2022), self-consistency, and more structured approaches like Tree- and Graph-of-Thought Yao et al. (2023); Besta et al. (2024) demonstrated that extending intermediate reasoning can improve accuracy but at increasing computational cost Chen et al. (2024). Building on this, LRMs such as OpenAI's o1 Jaech et al. (2024) and DeepSeek-R1 Guo et al. (2025) employ large-scale supervised fine-tuning and reinforcement learning to explicitly incentivize the generation of long, structured reasoning traces. This paradigm enables models to decompose complex tasks, explore alternatives, and self-correct, but also tightly couples performance to test-time computation. The reliance of LRMs on extended reasoning motivates our study of how compute constraints through length control or quantization impact their skill and safety trade-offs in practice.

Quantization Quantization is an approach that reduces the precision of model parameters like weights and activations. It is common for neural networks to be implemented in bfloat16, a format supported by most neural accelerators. Post-training quantization (PTQ) approaches such as GPTQ Frantar et al. (2022), GGUF Lin et al. (2016), and ZeroQuant Yao et al. (2022) enable efficient compression without retraining, while quantization-aware training (QAT) methods such as LR-QAT Bondarenko et al. (2024) preserve task-specific performance through learnable scaling factors. Recent work extends these techniques to LLMs, though primarily evaluated on language understanding rather than reasoning tasks.

3 Method

3.1 Safety Reasoning Length Control

In this work, we fine-tune the baseline - L1-Exact-1.5B reasoning model to improve its safety performance while training the model on a user-defined reasoning token length. For brevity, we refer to this baseline model as L1-1.5B. We extend the work done in Aggarwal & Welleck (2025) and use the length controlled policy optimization (LCPO) reinforcement learning method presented in Aggarwal & Welleck (2025) to fine-tune the baseline model. We modify the reward function to combine the (1) safety reward (r_s) and (2) the length penalty (r_l). The safety reward is determined by using the Llama-Guard-3 model as a safety judge. The additional safety fine-tuning is requires as the baseline model (L1-Exact-1.5B) is only fine-tuned for length-control using the DeepScaleR-Preview-Dataset dataset Luo et al. (2025). We use a learning rate of 1e-6 and a batch size of 64. We train the model for 300 steps. The dataset used for safety fine-tuning based on the SafeChain dataset Jiang et al. (2025). The methodology used to generate the dataset is described below.

3.1.1 Dataset Creation

We conduct our training on the L1-Exact-1.5B model Aggarwal & Welleck (2025) using a chain-of-thought (CoT) style safety dataset called SafeChain Jiang et al. (2025) that can improve a model's safety performance while preserving its math and coding performance across all benchmarks.

To enable length control, we augment each prompt in the SafeChain dataset with a target length instruction.

$$X_i = x_i + \text{"Think for n tokens"} \tag{1}$$

where, x_i is the prompt from the SafeChain dataset and X_i is the augmented prompt used for training. The value n is randomly picked from 0 to 4000 for each prompt.

3.2 Quantization

We conduct a comprehensive investigation into the effects of quantization techniques, in particular weight-only quantization (GPTQ). GPTQ Frantar et al. (2022) is a one-shot weight quantization method, based on approximate second-order information and error compensation, that is both highly-accurate and highly-efficient. We perform INT8 (8-bit) and INT4 (4-bit) quantization of the weights of the model while leaving the activation operation at the original Bfloat-16. Our initial investigation is limited to weight-only quantization, we will explore weight-activation and KV-cache quantization methods in future work.

4 Evaluation

We evaluate the compute-constrained models along two dimensions. First, we measure the models' performance, in terms of accuracy and safety, for an increase in reasoning token length. In this work, we demonstrate results for the following target reasoning lengths - 512, 1024, 2048, 3600. Second, we measure compute efficiency, in terms of throughput (tokens/sec), and determine the reasoning time (in seconds) a model takes to generate a fixed number of reasoning tokens.

4.1 Evaluated Benchmarks

We evaluate the compute-constrained models using science, math, and safety representative reasoning benchmarks: GPQA Diamond Rein et al. (2024) consists of 198 PhD-level science questions from Biology, Chemistry, and Physics. MATH500 Hendrycks et al. (2021) is a benchmark consisting of competition-level math problems of varying difficulty. AIME2025 Hendrycks et al. (2020) contains 30 problems from the AIME1 and AIME2 math jam. Following previous work Aggarwal & Welleck (2025), we evaluate our model on the same subset selected by OpenAI Achiam et al. (2023). In addition to these three common science and math reasoning benchmarks, we evaluated the safety performance of the reasoning models using the StrongReject dataset Souly et al. (2024), a state-of-the-art safety evaluation dataset with 60 jailbreak queries. For all benchmarks, we generate a sample for each question with a temperature of 0 (greedy) to measure accuracy. Through these benchmarks, we can evaluate the reasoning ability of LLMs from different perspectives.

4.2 Safety Evaluator

Our safety evaluation work builds on the prior work Jaech et al. (2024); Jiang et al. (2025). We consider the LLama-Guard Chi et al. (2024) evaluator to generate a safety score based on the work done in Jiang et al. (2025) to assess the effectiveness of *four* state-of-the-art safety evaluators - Llama-guard Chi et al. (2024), Refusal String Matching (RS-Matching) Zou et al. (2023), OpenAI Moderation API Kivlichan et al. (2024), and HarmBench Mazeika et al. (2024).

4.3 Metric

We evaluate the models in terms of their math and science reasoning skill and safety. We measure the accuracy of the models' math and science skills using the pass@1 score and measure the safety performance of the models' using the safe@1 score. We define these performance metrics as follows:

4.3.1 Skill Evaluation

We use the pass@1 metric to evaluate the skill and safety performance of the reasoning models discussed in this work. We define the pass@1 score as shown in Equation 2:

$$pass@1 = \frac{1}{K} \cdot \sum_{i=1}^{K} p_i \tag{2}$$

where p_i is a binary score that indicates whether a response y_i to a query q_i is correct for skill tasks.

4.3.2 Safety Evaluation

We use the safe@1 metric to evaluate the skill and safety performance of the reasoning models discussed in this work. As described in Guo et al. (2025); Jiang et al. (2025) we define the safe@1 score as shown in Equation 3:

$$safe@1 = \frac{1}{K} \cdot \sum_{i=1}^{K} s_i \tag{3}$$

where s_i is a binary score, generated using a state-of-the-art evaluator, that indicates whether a response y_i to a query q_i is correct for skill tasks and safe (1) or not (0) for safety tasks. We generate the safe score (s) using the Llama-Guard-3-8B Chi et al. (2024) evaluator.

4.4 Evaluation Protocol

We evaluate our compute constraint approaches using the following methods: (1) we evaluate the overall performance (skill and safety reasoning) when generating responses at different target lengths. In our experiments, target lengths are selected from {512, 1024, 2048, 3600} tokens; (2) we evaluate the overall performance (skill and safety reasoning) of the quantized models when generating responses at different target lengths.

4.5 Reasoning Length Controlled Compute-Constraint

4.5.1 Science and Math Skill Evaluation

We start by evaluating the baseline L1 model using science and math skill datasets. We choose the AIME, GPQA, and LSAT reasoning datasets to evaluate L1 as they have not been used in the training of the L1 model. Figure 1 shows the performance of the L1 model for an increasing number of reasoning tokens (512, 1024, 2048, and 3600). We observe that the reasoning performance of the L1 model scales with an increase in reasoning tokens used to generate an answer. While the reasoning performance scales linearly for the GPQA and LSAT datasets, we observe that the evaluation on the AIME dataset scales well for reasoning tokens below 2048 tokens and evens out for larger reasoning lengths. The key trend we want to highlight here is that an *increase in reasoning length increases the performance of these models*.

5 Results

5.1 Baseline

We evaluate our compute-constrained reasoning model against the following baseline models:

- L1-Qwen-1.5B: a LCPO-based fined-tuned version of Agentic-24K with a context length of 4K. The model serves as a fair baseline for a reasoning length controlled model. For brevity, we refer to this model as L1-1.5B.
- L1-Qwen-8B: a LCPO-based fined-tuned version of Agentic-24K with a context length of 4K. The model serves as a fair baseline for a reasoning length constrained model. For brevity, we refer to this model as L1-8B.

5.2 Reasoning Length Controlled Compute-Constraint

5.2.1 Science and Math Skill Evaluation

We start by evaluating the baseline L1 model using science and math skill datasets. We choose the AIME, GPQA, and LSAT reasoning datasets to evaluate L1 as they have not been used in the training of the L1 model.

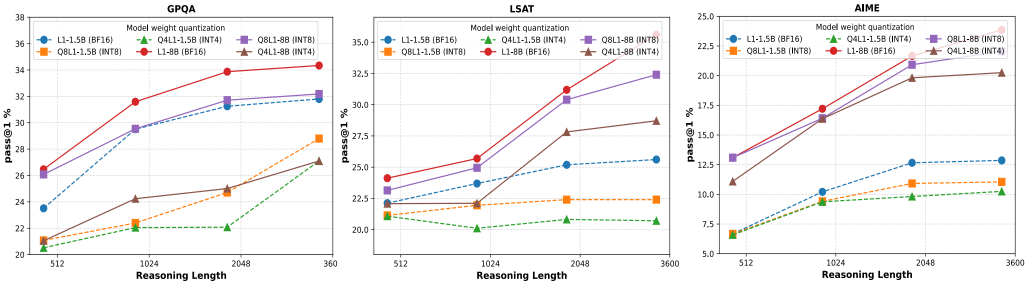

Figure 2: Performance of the baseline L1 and its post-training quantized models for skill based reasoning tasks - (a) GPQA, (b) LSAT and (c) AIME.

Figure 2 shows the performance of the L1 model for an increasing number of reasoning tokens (512, 1024, 2048, and 3600). We observe that the reasoning performance of the L1 model scales with an increase in reasoning tokens used to generate an answer. While the reasoning performance scales linearly for the GPQA and LSAT datasets, we observe that the evaluation on the AIME dataset scales well for reasoning tokens below 2048 tokens and evens out for larger reasoning lengths. The key trend we want to highlight here is that an *increase in reasoning length increases the performance of LRMs*.

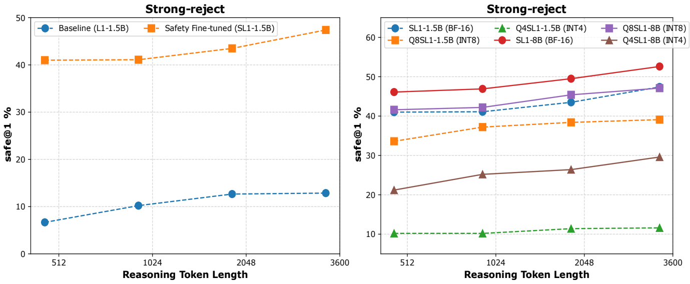

Figure 3: (a) Performance of the baseline L1 and SL1 (Safety fine-tuned L1) under varying reasoning token budgets; (b) performance of SL1-1.5B and SL1-8B models for BF16 (circle), INT8 (square) and INT4 (triangle) weight precision.

5.2.2 Safety Evaluation

In this section, we quantitatively assess the safety reasoning capability of the baseline L1 model. Figure 3 (a) illustrates the safety performance of the baseline L1-1.5B model when evaluated using the StrongReject Souly et al. (2024) dataset for increasing reasoning length. The safe@1 score (defined in Section 4.3.2) indicates the percentage of safe (1) or unsafe (0) responses to jailbreak queries from the StrongReject dataset. This score is generated using the Llama-Guard-3 Chi et al. (2024) safety evaluator.

From Figure 3 (a) we observe that the safety performance of the baseline L1-1.5B model does not match the state-of-the-art safety performance of similar 1.5B models Jiang et al. (2025). This is expected, as the baseline L1 model has been finetuned using only science and math reasoning skill-based datasets. To address this poor safety performance, we fine-tune the baseline L1-1.5B and L1-8B models (S-L1) using the LCPO RL method and the Safechain Jiang et al. (2025) dataset. Training is performed for 300 iterations using the VeRL engine. We train the model using reasoning traces with target lengths between 1 and 4000 tokens. Figure 3 (a) and (b) (circle) illustrate the improved safety performance of the safety fine-tuned L1-1.5B and L1-8B models. For the rest of this work, we refer to safety fine-tuned models as SL1.

5.3 Quantization Compute-Constraint

In this section, we study the impact of quantization-based compute constraints on the performance of reasoning models. We evaluate the reasoning models for skill and safety performance.

5.3.1 Science and Math Skill Evaluation

Figure 2 illustrates the performance of the post-training weight quantized L1 model for LSAT, AIME and GPQA reasoning datasets. We compare the performance of the full-precision L1 model with *two* levels of weight quantization - INT8 and INT4. Using post-training quantization method (GPTQ), we create the Q8L1 (INT8) and Q4L1 (INT4) models.

We make *two* observations: (1) the reasoning performance of the L1, Q8L1, and Q4L1 models improves with an increase in reasoning length, but this effect is less profound as we increase the quantization level from INT8 to INT4, and (2) the performance of the INT4 quantized model drops significantly for all three evaluated skill datasets, irrespective of the reasoning length. Due to observation *two*, we limit our study to INT8 quantized models.

5.3.2 Safety Evaluation

Figure 3 B illustrates the safety performance of the SL1-1.5B and SL1-8B models in comparison to its INT8 and INT4 weight quantized implementations. *First*, observe that the safety performance of the baseline and quantized models improves with an increase in reasoning length. *Second*, we observe a significant drop in performance of the 4-bit quantized model (SL1) for safety. For a reasoning

Model	AIME (tokens/s)	StrongReject (tokens/s)
SL-1.5B	42.11	71.60
Q8-SL-1.5B	69.19	107.33
SL-8B	23.18	31.04
Q8-SL-8B	37.21	45.10

Table 1: Average throughput (tokens/s) of the evaluated models on the AIME and StrongReject datasets.

length of 512, the safe@1 score drops from 40% to 10%, making the quantized model far more susceptible to jailbreak queries. However, the safety performance of the 8-bit quantized model does not deteriorate significantly. We observe a 3-7% drop in safety performance of the 8-bit quantized models when compared to the baseline.

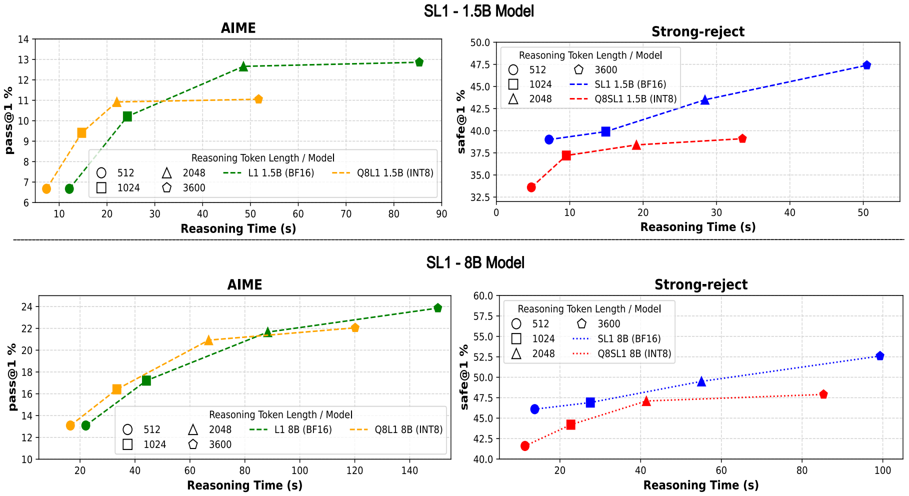

Figure 4: Trade-off between compute constraint methods with a fixed compute budget. (a) Evaluation of the SL1 model (BF16 and INT8) using the AIME dataset; (b) Evaluation of the SL1 model (BF16 and INT8) using the StrongReject dataset.

5.4 Impact of Compute Constraints on Reasoning

In this section, we study the impact of the two chosen compute constraint methods on skill and safety performance. We also detail the trade-off between the two methods, with an aim to demonstrate that full precision and quantized models can show similar performance with-in a fixed compute budget -by varying their reasoning token lengths. To relate the performance of the model to its compute budget, we propose observing the reasoning time, i.e., the inference time (seconds) for the model to generate reasoning tokens. This metric combines the throughput (tokens/s) of a model with the number of reasoning tokens it is afforded within the compute budget.

We report the throughput (tokens/s) of the SL1 and Q8SL1 models in Table 1. The throughput is measured separately (average length of the queries (tokens) varies) for a batch size of 1 on an A100 GPU with a GPU utilization of 0.6. Details on the experimental setup used can be found in Section 4. Figure 4 illustrates the performance of the full precision and quantized versions of SL1 model. The skill and safety reasoning performance of the model is measured as a function of the total compute (tokens) needed to generate an answer. The compute budget is measured as the reasoning time in seconds (x-axis) needed to generate a fixed number of tokens (512, 1024, 2048, and 3600) in this experiment. We calculate the compute budget by multiplying the throughput (tokens/s) of the model for a given dataset by the total number of reasoning tokens.

In Figure 4 (column one) we observe the evaluation of L1 and Q8L1 models using the AIME dataset. For a reasoning length of 512 tokens, the L1 and QL1 models have a similar accuracy. However, the reasoning time (seconds) of the Q8L1-1.5B model of 7.355 seconds is 39.32% lower than the reasoning time of the L1 (BF16) model at 12.13 seconds. This is also true for a reasoning length of 1024 tokens. In the case of a reasoning length of 2048 tokens, the L1 model has a higher pass@1 accuracy of 12.66% when compared to the B8SL1 model with 10.92%. However, we observe that the SL1 model reasons for 48.54 seconds, a 119.98% increase in compute budget when compared to the B8L1 model at 22.06 seconds. In figure 4 (a), we highlight that the L1 model reasoning for 1024 tokens (green square) has a similar performance (1%) to a Q8SL1 model reasoning for 2048 tokens (orange triangle) with a similar compute budget. Similarly, in figure 4 (c) we highlight that the L1 model reasoning for 1024 tokens (green square) has a similar performance (1%) to a Q8SL1 model reasoning for 1024 tokens (orange triangle) with a 16.66% smaller compute budget.

In Figure 4 (a) we see the evaluation of SL1-1.5B and Q8SL1-1.5B models using the StrongReject dataset. We observe that the QSL1 model reasoning for 2048 tokens (red triangle) has a safety performance similar (1.4% drop in safety score) to the SL1 model reasoning for 1024 tokens (blue square). In this scenario, both the models even have a similar compute budget of 19.08 seconds and 14.91 seconds for the Q8SL1 and SL1 models, respectively. This is further highlighted with the SL1-8B and Q8SL1-8B models. We observe that the QSL1-8B model for a reasoning length of 2048 (triangle) has a safety performance similar (2% drop in safety score) to the SL1-8B model reasoning for 2048 tokens (blue square) with a compute budget 16.4% lower.

6 Conclusions

In this work, we study how compute constraints affect the safety performance of reasoning models. We explore two methods to apply compute constraints: (1) a Length Controlled Policy Optimization (LCPO), a simple reinforcement learning-based method that enables user-defined control over reasoning length, and (2) weight quantization, which reduces the compute demands of the reasoning model and ensures their execution within a user-defined compute budget, namely, inference time. We further demonstrate that within a fixed compute budget (reasoning time), a quantized reasoning model can perform at par with a full-precision model. This is because within the fixed compute budget, the quantized model can generate more reasoning tokens and hence compensate for the loss in performance observed due to quantization.

7 Acknowledgment

This material is based upon work supported by Laboratory Directed Research and Development (LDRD) funding from Argonne National Laboratory, provided by the Director, Office of Science, of the U.S. Department of Energy under Contract No. DE-AC02-06CH11357.

References

Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Aggarwal, P. and Welleck, S. L1: Controlling how long a reasoning model thinks with reinforcement learning, 2025. *URL https://arxiv. org/abs/2503.04697*, 2025.

Besta, M., Blach, N., Kubicek, A., Gerstenberger, R., Podstawski, M., Gianinazzi, L., Gajda, J., Lehmann, T., Niewiadomski, H., Nyczyk, P., et al. Graph of thoughts: Solving elaborate problems with large language models. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pp. 17682–17690, 2024.

Bondarenko, Y., Del Chiaro, R., and Nagel, M. Low-rank quantization-aware training for LLMs. *arXiv preprint arXiv:2406.06385*, 2024.

Chen, L., Ahmed, N. K., Dutta, A., Bhattacharjee, A., Yu, S., Mahmud, Q. I., Abebe, W., Phan, H., Sarkar, A., Butler, B., et al. The landscape and challenges of HPC research and LLMs. *arXiv preprint arXiv:2402.02018*, 2024.

Chi, J., Karn, U., Zhan, H., Smith, E., Rando, J., Zhang, Y., Plawiak, K., Coudert, Z. D., Upasani, K., and Pasupuleti, M. Llama guard 3 vision: Safeguarding human-AI image understanding conversations. *arXiv preprint arXiv:2411.10414*, 2024.

Frantar, E., Ashkboos, S., Hoefler, T., and Alistarh, D. Gptq: Accurate post-training quantization for generative pre-trained transformers. *arXiv preprint arXiv:2210.17323*, 2022.

Guo, D., Yang, D., Zhang, H., Song, J., Zhang, R., Xu, R., Zhu, Q., Ma, S., Wang, P., Bi, X., et al. DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Hendrycks, D., Burns, C., Basart, S., Zou, A., Mazeika, M., Song, D., and Steinhardt, J. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*, 2020.

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

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

Jiang, F., Xu, Z., Li, Y., Niu, L., Xiang, Z., Li, B., Lin, B. Y., and Poovendran, R. Safechain: Safety of language models with long chain-of-thought reasoning capabilities. *arXiv preprint arXiv:2502.12025*, 2025.

Kaplan, J., McCandlish, S., Henighan, T., Brown, T. B., Chess, B., Child, R., Gray, S., Radford, A., Wu, J., and Amodei, D. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

Kivlichan, I., Harriman, J., Raymond, C., Shah, M., Chaudhuri, S. R., and Gu-Lemberg, K. Upgrading the moderation API with our new multimodal moderation model. *Website*, 2024.

Li, Z., Su, Y., Yang, R., Xie, C., Wang, Z., Xie, Z., Wong, N., and Yang, H. Quantization meets reasoning: Exploring llm low-bit quantization degradation for mathematical reasoning. *arXiv preprint arXiv:2501.03035*, 2025.

Lin, Z., Courbariaux, M., Memisevic, R., and Bengio, Y. Neural networks with few multiplications. In *ICLR*, 2016.

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

Mazeika, M., Phan, L., Yin, X., Zou, A., Wang, Z., Mu, N., Sakhaee, E., Li, N., Basart, S., Li, B., et al. Harmbench: A standardized evaluation framework for automated red teaming and robust refusal. *arXiv preprint arXiv:2402.04249*, 2024.

Muennighoff, N., Yang, Z., Shi, W., Li, X. L., Fei-Fei, L., Hajishirzi, H., Zettlemoyer, L., Liang, P., Candès, E., and Hashimoto, T. s1: Simple test-time scaling. *arXiv preprint arXiv:2501.19393*, 2025.

Rein, D., Hou, B. L., Stickland, A. C., Petty, J., Pang, R. Y., Dirani, J., Michael, J., and Bowman, S. R. Gpqa: A graduate-level google-proof Q&A benchmark. In *First Conference on Language Modeling*, 2024.

Snell, C., Lee, J., Xu, K., and Kumar, A. Scaling LLM test-time compute optimally can be more effective than scaling model parameters. *arXiv preprint arXiv:2408.03314*, 2024.

Souly, A., Lu, Q., Bowen, D., Trinh, T., Hsieh, E., Pandey, S., Abbeel, P., Svegliato, J., Emmons, S., Watkins, O., et al. A StrongREJECT for empty jailbreaks. *arXiv preprint arXiv:2402.10260*, 2024.

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

Xu, F., Hao, Q., Zong, Z., Wang, J., Zhang, Y., Wang, J., Lan, X., Gong, J., Ouyang, T., Meng, F., et al. Towards large reasoning models: A survey of reinforced reasoning with large language models. *arXiv preprint arXiv:2501.09686*, 2025.

Yao, S., Yu, D., Zhao, J., Shafran, I., Griffiths, T., Cao, Y., and Narasimhan, K. Tree of thoughts: Deliberate problem solving with large language models. *Advances in neural information processing systems*, 36:11809–11822, 2023.

Yao, Z., Yazdani Aminabadi, R., Zhang, M., Wu, X., Li, C., and He, Y. Zeroquant: Efficient and affordable post-training quantization for large-scale transformers. *Advances in neural information processing systems*, 35:27168–27183, 2022.

Zou, A., Wang, Z., Carlini, N., Nasr, M., Kolter, J. Z., and Fredrikson, M. Universal and transferable adversarial attacks on aligned language models. *arXiv preprint arXiv:2307.15043*, 2023.