Proceedings of the 4th International Conference on Mathematical Physics and Computational Simulation
DOI: 10.54254/2753-8818/2026.33148

Mathematical Reasoning Ability of Large Models Based on Process Indicators and Result Indicators

Shiyu Sun

School of Economics and Management, Beijing Jiaotong University (Weihai Campus), Weihai, China

23711104@bjtu.edu.cn

Abstract. Large language models perform exceptionally well in mathematical reasoning tasks, generating reasoning chains spontaneously upon receiving result prompts. However, the differences in quality between these spontaneous reasoning chains and those guided by structured process constraints remain unquantified systematically. This study selects 240 mathematical problems from GSM8K and MATH, using the self-consistency sampling strategy to invoke the large language model three times for each problem with the temperature parameter set to 0.7. The final answer is determined through majority voting. An evaluation index is designed to quantify the number of formulas and step markers in the reasoning chains and the stability of multiple generations is measured. The results show that process constraints increase the process quality score of the reasoning chains by 0.14 to 0.20, and the answer accuracy is close on GSM8K, but differs by 11% on MATH. The consistency rates are both 0.95 on GSM8K and 0.94 and 0.90 on MATH respectively. In complex problem types, the improvement in process quality brought by process constraints is greater, but the stability decline is more significant. An inherent trade-off between quality of the process and accuracy of the answer is present, and it provides empirical evidence of enhancing timely approaches in future research.

Keywords: Large language model, Mathematical reasoning, Result-oriented prompts, Process-constrained prompts, Self-consistency sampling

1. Introduction

In recent years, large language models have succeeded in mathematical reasoning tasks, and can produce reasoning steps by use of chain-of-thought methods [1]. The multi-step reasoning has gained importance as a research topic in this area, and the current knowledge in the field discusses prompt strategies, methods of training, and evaluation criteria in different perspectives [2]. Mathematical reasoning is the mental process of producing new conclusions based on known conditions by applying logical rules and mathematical principles, and is defined by rigor (stepwise) and logical consistency (rational) and the verifiability of the result [3]. This reasoning capacity is worth investigating to enhance the interpretability and reliability of its models, and has potential uses, including intelligent education, code generation, and scientific discovery [4].

DOI: 10.54254/2753-8818/2026.33148

Chain-of-thought prompting can be demonstrated to promote the performance of large language models on complex reasoning tasks [5]. In mathematical reasoning, Uesato et al. compared process-oriented and outcome-oriented guidance and found that process information has a significant impact on the quality of reasoning [6], which can be used in subsequent studies in prompt strategies. Zhang et al. also investigated the ability of reasoning patterns to enhance chain-of-thought prompts thus enhancing the strength of the prompting strategies [7]. Rizvi et al. described SPARE framework, which automates supervision of the processes and reward modeling using single-pass annotation and reference guided evaluation [8]. In spite of these developments, there is still no systematic comparison between reasoning chain quality of result-oriented and process-constrained prompts. This discontinuity constrains the elegant planning of timely strategies and their reconfiguration to various application contexts.

In this paper, the controlled experiments were used to quantitatively compare result-oriented and process-constrained prompts in three aspects: (1) accuracy of the answer, (2) verifiability of the reasoning process, and (3) stability of the generation (consistency and variance). The results are likely to provide empirical data on the best approaches to adopt timely strategies in the future work.

2. Method

2.1. Data set

This paper selected two benchmark data sets to reason mathematically. The GSM8K dataset is constituted of elementary-level math word problems [9] and it is used to evaluate the basic reasoning skill of the model, out of which 160 questions are randomly chosen as the test sample. The MATH dataset comprises algebra, geometry and probability and other problems at the level of the high school competitions [10]; it is utilized to analyze the performance of the models on more difficult problems, where 80 questions are randomly selected among this dataset. The questions in both datasets are stored in the form of a JSON, including the text of questions and standard answers.

2.2. Experimental design

The research takes a comparative experimental design to critically examine the effects of two prompting techniques on the mathematical reasoning product of the model. Figure 1 shows the general workflow of the experiment.

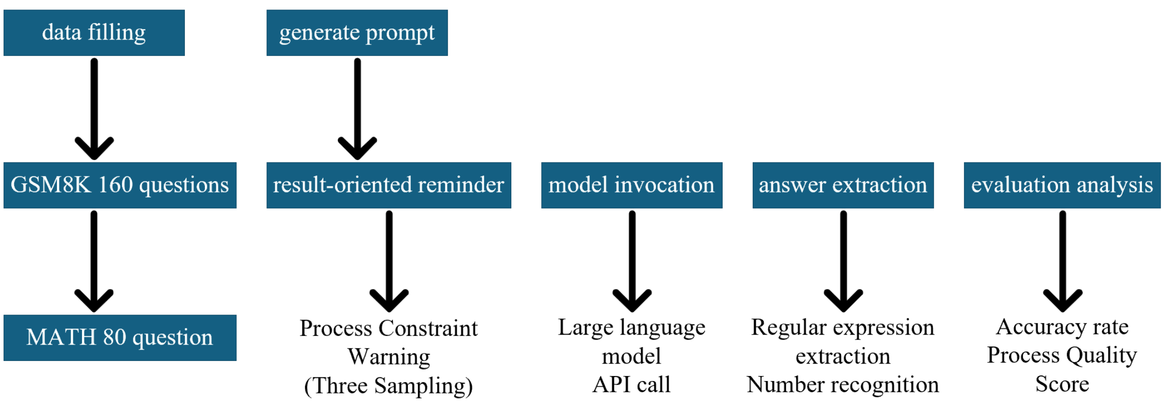

Figure 1. Experimental flowchart (photo credit: original)

DOI: 10.54254/2753-8818/2026.33148

The experimental process consists of five stages. The first stage is data preparation, where questions are randomly selected from the GSM8K and MATH datasets and saved as JSON format files. The second stage is prompt generation, where two types of prompts are constructed for each question: the result-oriented prompt requires the model to directly output the numerical answer, while the process-constrained prompt requires the model to output the reasoning process step by step before providing the answer. The third stage is model invocation, where the constructed prompts are sent to the large language model. To evaluate the stability of the generated results, the self-consistency sampling strategy [11] is adopted to invoke the model three times for each question. The fourth stage is answers extraction, where the numerical answers are extracted from the original responses returned by the model. If extraction fails, it is marked as empty. The fifth stage is evaluation analysis, where various evaluation metrics are calculated based on the extracted answers and the original responses.

2.3. Evaluation index

This study has designed three evaluation indicators (answer accuracy rate, process quality score, stability index) to quantitatively measure the quality of reasoning chain from different perspectives.

The answer accuracy rate is the fundamental indicator for measuring the accuracy of the model's reasoning, and its calculation formula is:

$$\text{Accuracy} = \frac{N_{\text{correct}}}{N_{\text{total}}} \times 100\% \tag{1}$$

Here, N_{correct} represents the number of questions where the voting answer matches the standard answer, and N_{total} represents the total number of questions.

The process quality score is used to quantify the verifiability of the reasoning chain, and the calculation formula is as follows:

$$R_{\text{process}} = R_{\text{outcome}} + 0.2 \times n_{\text{expr}} + 0.1 \times n_{\text{step}} + 0.1 \times I_{\text{lang}} \tag{2}$$

Here, R_{outcome} represents the score for the correct answer (correct = 1, incorrect = 0), n_{expr} represents the number of mathematical expressions included in the reasoning chain. n_{step} represents the number of included step markers (matching "Step 1", "Step 2", "Step 3", "Step 4") , I_{lang} is the language purity indicator (pure Chinese or pure English = 1, mixed Chinese and English = 0)

This score takes into account both the correctness of the answer and the standardization of the process. The higher the score, the better complete and verifiable the reasoning chain is.

The stability index is used to measure the consistency of the model's multiple generation results, and it includes two dimensions: consistency rate and variance. The calculation processes of the consistency rate and variance are respectively shown in formulas (3) and formula (4).

$$\text{Consistency} = \frac{1}{N} \sum_{i=1}^{N} \frac{n_{\text{same}}^{(i)}}{3} \tag{3}$$

DOI: 10.54254/2753-8818/2026.33148

Here, N represents the total number of questions in this dataset, and $n_{\text{same}}^{(i)}$ indicates the number of times the answers to this question are the same in three samples (3 if all three are the same, 2 if two are the same, 1 if one is the same, and 0 if all are different). The consistency rates for a single question range from 0 to 1.

$$\text{Variance} = \frac{1}{3}\sum_{i=1}^{3}\left(R_{\text{process}}^{(i)} - \bar{R}_{\text{process}}\right)^2 \tag{4}$$

Here, $R_{\text{process}}^{(i)}$ represents the process quality score of the i_{th} sampling, and $\bar{R}_{\text{process}}$ represents the average process quality score of the three samplings. The smaller the variance, the more stable the process quality generated in the three samplings is, and the more reliable the model's answer to this problem is.

3. Outcome

3.1. Overall experimental results

The experimental results of the two prompt methods on the two datasets are shown in Table 1. This table comprehensively presents the data of three dimensions: the accuracy rate of answers, the quality score of the process, and the stability index, providing a basis for subsequent analysis.

Table 1. Comparison of experimental results of two prompting methods

Method	Data Set	Average result score	Average process score	Answer accuracy rate	Stability index (Consensus rate/variance)
Result-oriented	GSM8K	0.96	1.26	95.62%	0.95 / 0.0383
Result-oriented	MATH	0.65	0.79	65.00%	0.94 / 0.0280
Process-constrained	GSM8K	0.94	1.40	94.38%	0.95 / 0.0384
Process-constrained	MATH	0.54	0.99	53.75%	0.90 / 0.0552

3.2. Comparison of answer accuracy rate and process quality

As can be seen from Table 1, on the GSM8K dataset, the accuracy rate of result-oriented prompts is 95.62%, while that of process constraint prompts is 94.38%, and the difference is only 1.24 percentage points. On the MATH dataset, the accuracy rate of result-oriented prompts is 65.00%, and that of process constraint prompts is 53.75%, with a difference of 11.25 percentage points. This result indicates that in simple question types, the two prompt methods have comparable effects, while in complex question types, result-oriented prompts have a slight advantage.

In terms of process quality scoring, process constraint prompts are higher than result-oriented prompts on both datasets. In GSM8K, the process score increases from 1.26 to 1.40, an increase of 0.14; in MATH, it increases from 0.79 to 0.99, an increase of 0.20. This indicates that the mandatory step-by-step guidance does indeed enable the model to generate more complete and verifiable reasoning steps, including more formulas and structured markers. The comparison of process quality scores for the two datasets is shown in Figures 2 and 3.

DOI: 10.54254/2753-8818/2026.33148

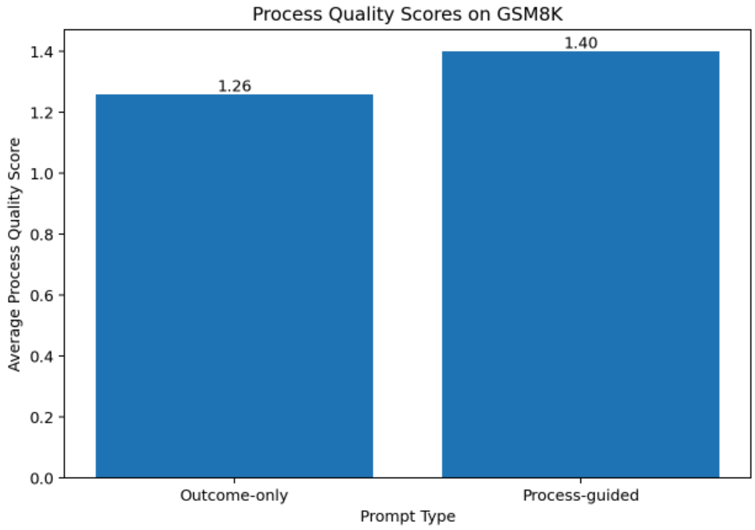

Figure 2. Comparison of process quality scores for the two methods on GSM8K (photo credit: original)

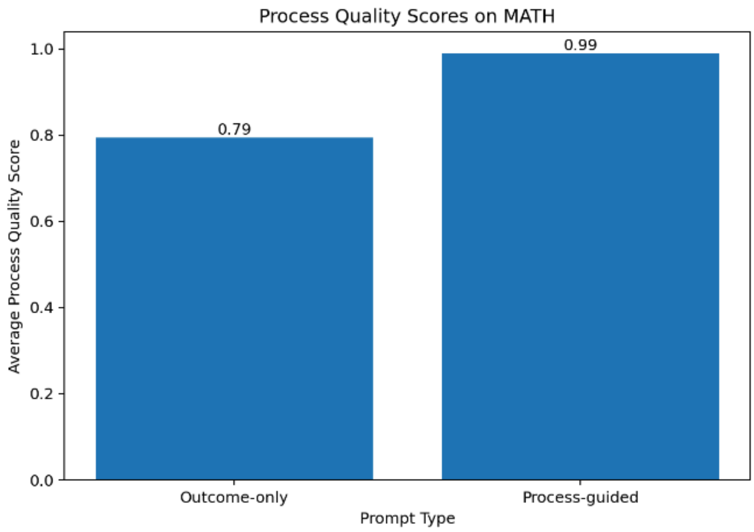

Figure 3. Comparison of process quality scores for the two methods on MATH (photo credit: original)

The above results reveal a core contradiction - although process constraints enhance the verifiability of the reasoning chain, they come at the expense of answer accuracy, especially in complex question types. This finding challenges the intuition that "process constraints are always better", indicating that forcing step-by-step procedures may interfere with the model's free creativity in solving difficult problems.

3.3. Generation stability comparison

From the perspective of the stability index, the consistency rates of the two prompt methods on GSM8K are both 0.95. The variance of the process constraint prompt (0.0384) is slightly higher than that of the result-oriented prompt (0.0383). On MATH, the consistency rate of the result-oriented

DOI: 10.54254/2753-8818/2026.33148

prompt is 0.94, while that of the process constraint prompt is 0.90; in terms of variance, the result-oriented prompt is 0.0280, and the process constraint prompt is 0.0552. This indicates that on complex question types, the generation stability of the process constraint prompt is slightly lower than that of the result-oriented prompt, and mandatory step-by-step instructions may increase the uncertainty in model generation.

The comparison of the stability of the two methods is shown in Figure 4. From the figure, it can be intuitively seen that the stability of the two methods is basically the same on GSM8K, while on MATH, the stability of the process constraint prompt is slightly lower.

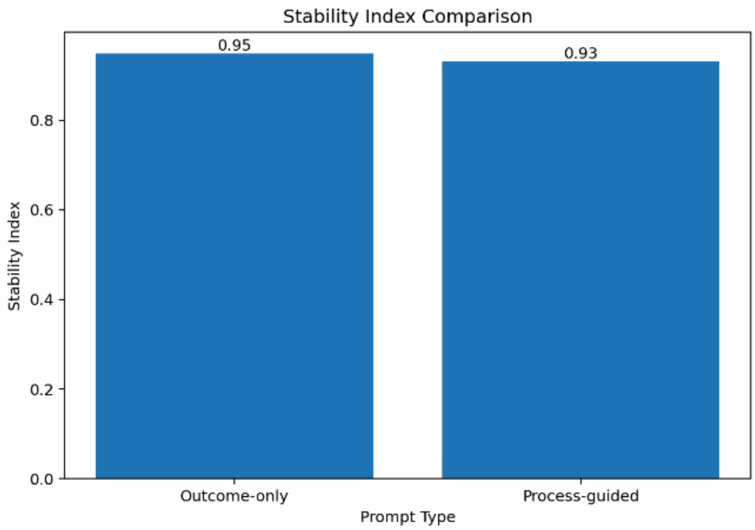

Figure 4. Comparison of stability indices of the two methods (photo credit: original)

The consistent distribution of the three sampling answers under the process constraint method is shown in Figures 5 and 6. In the GSM8K dataset (Figure 5) the share of the same response in all three samplings is 92.5% and the shares of two identical responses and one different response and of three wholly different responses are 6.9% and 0.6%, respectively. In the MATH data (Figure 6), the proportions are 83.8%, 16.2% and 0%.

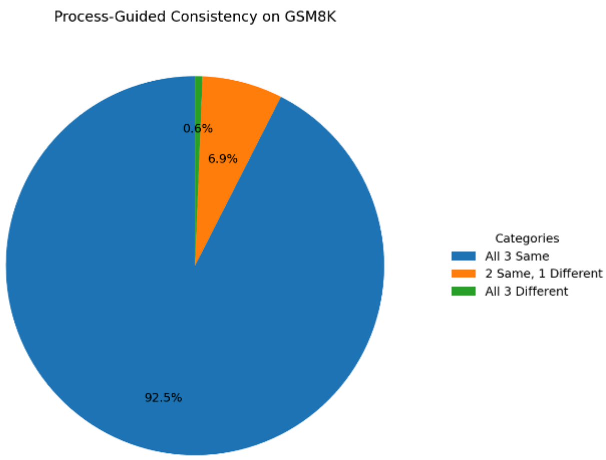

Figure 5. Consistency distribution of GSM8K under process constraint mode (photo credit: original)

DOI: 10.54254/2753-8818/2026.33148

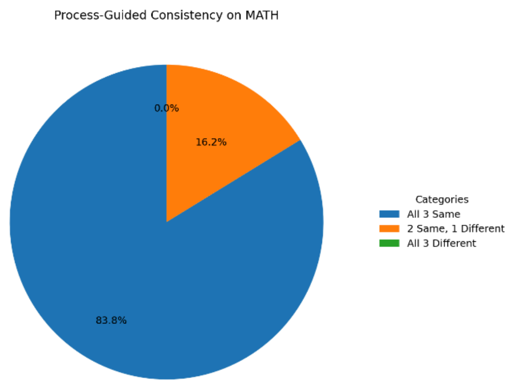

Figure 6. Consistency distribution of MATH under process constraint mode (photo credit: original)

These findings also reveal the pattern of constraints on processes. When using complex tasks, the stepwise reasoning will increase the uncertainty of the model outputs and thus decrease the consistency. This result indicates a trade-off between the quality of processes and stability of the generation: improvements in verifiability can come at the price of reliability.

3.4. The difficulty of the dataset has an impact

The accuracy of answers and the quality score of the process are worse in the MATH dataset than in GSM8K, which proves the influence of the difficulty of the problems on the model performance. MATH demonstrates a lower accuracy (30.62 percentage points) and lower process score (0.47 points) compared to GSM8K when using result-oriented prompts. At process constrained prompts, MATH would score 40.63 percent points less accurate and 0.41 less process.

It is important to note that the difference in the improvement of the process score due to process constraints is higher on MATH (0.20) than on GSM8K (0.14), indicating that complex types of problems are more benefited by process constraints with regard to reasoning chain verifiability. But, in terms of stability, process constraints provide a lower consistency rate on MATH (0.90) than on GSM8K (0.95) and the variance on MATH (0.0552) is considerably greater than that on GSM8K (0.0384). The additional analysis of Figure 6 shows that in case of process constraints, the percentage of two same, one different case is the highest, in MATH, 16.2 vs. 6.9 percent in GSM8K. These findings suggest that though step by step instruction is beneficial in enhancing the quality of reasoning process in solving complex problems, it also significantly increases the uncertainty of the outputs.

When combined, these results indicate an interplay between the difficulty of the datasets and the prompting strategy. The two prompting methods are similar in terms of simple types of problems. In complex types of problems, process constraints improve the quality of reasoning processes at the expense of accuracy and stability. This finding provides a basis for the design of adaptive prompt strategies - simple questions can be prompted simply, while difficult questions need to balance process quality and accuracy.

4. Conclusion

This paper systematically evaluates the impact of result-oriented prompts and process-constrained prompts on the mathematical reasoning output of the model by comparing their performance on 240

DOI: 10.54254/2753-8818/2026.33148

math problems. The findings indicate that on the dataset of GSM8K, both prompting approaches are equally accurate, with process-constrained prompts demonstrating a 0.14 boost in process quality score. Result-oriented prompts are more accurate compared to process-constrained prompts on the MATH dataset and the latter still raise the process quality score by 0.20. In terms of stability of generation, process-constrained prompts have a lower rate of consistency on MATH than result-oriented prompts do. The results reveal one of the underlying trade-offs between process quality, accuracy in answers, and stability in generation. Even though process-constrained prompts increase the verifiability of reasoning chain, it comes at the price of lower accuracy and stability, especially with complex problems.

This study has three significant implications. In theory, it exposes the trade-off where process constraints are traded-off to enhance accuracy and stability to enhance verifiability as a new perspective of empiricist research on timely strategy. The study methodologically develops a multi-dimensional assessment system which offers a measurable instrument to be used later in studies on reasoning ability. Practically, it brings out the dual-edged sword effect of having to adhere to step-by-step instructions, which may imply that the ratio between process quality and accuracy must be varied depending on the difficulty of the task in real-life application.

There are two ways the future work can go. One is to develop adjustive prompt strategy, which flexibly changes the degree of guidance depending on problem difficulty. The other is to consider the jointness of process constraints and self-consistency sampling, in which multiple sampling and voting mechanisms are used to ensure quality of reasoning, and enhance generation stability.

References

[1] Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E. H., Le, Q. V., & Zhou, D. (2022). Chain-of-thought prompting elicits reasoning in large language models. Advances in Neural Information Processing Systems, 35, 24824–24837.

[2] Plaat, A., Wong, A., Böhm, C., van Stein, N., Bäck, T., & Sandholm, T. (2025). Multi-step reasoning with large language models: A survey. ACM Computing Surveys, 58(6), Article 160, 1–35.

[3] Wang, P.-Y., Liu, T.-S., Wang, C., Li, J., & Zhang, M. (2026). A survey on large language models for mathematical reasoning. ACM Computing Surveys, 58(8), 1–35.

[4] Saha, S., Chaturvedi, A., Saha, S., Garain, U., & Asher, N. (2025). KisMATH: Do LLMs Have Knowledge of Implicit Structures in Mathematical Reasoning?. arXiv preprint arXiv: 2507.11408.

[5] Wang, B., Min, S., Deng, X., Shen, J., Wu, Y., Zettlemoyer, L., & Sun, H. (2023, July). Towards understanding chain-of-thought prompting: An empirical study of what matters. In Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (pp. 2717-2739).

[6] Uesato, J., Kushman, N., Kumar, R., Song, H. F., Siegel, N. Y., Wang, L., ... & Higgins, I. (2022). Solving math word problems with process-based and outcome-based feedback.

[7] Zhang, Y., Wang, X., Wu, L., & Wang, J. (2025, April). Enhancing chain of thought prompting in large language models via reasoning patterns. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 39, No. 24, pp. 25985-25993).

[8] Rizvi, M. I. H., Zhu, X., & Gurevych, I. (2026, March). Spare: Single-pass annotation with reference-guided evaluation for automatic process supervision and reward modelling. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 40, No. 39, pp. 32808-32816).

[9] Cobbe, K., Kosaraju, V., Bavarian, M., Chen, M., Jun, H., Kaiser, L., ... & Schulman, J. (2021). Training verifiers to solve math word problems. arXiv preprint arXiv: 2110.14168.

[10] Hendrycks, D., Burns, C., Kadavath, S., Arora, A., Basart, S., Tang, E., ... & Steinhardt, J. (2021). Measuring mathematical problem solving with the math dataset. arXiv preprint arXiv: 2103.03874.

[11] Wang, X., Wei, J., Schuurmans, D., Le, Q., Chi, E., Narang, S., ... & Zhou, D. (2022). Self-consistency improves chain of thought reasoning in language models. arXiv preprint arXiv: 2203.11171.