JOINT CONSISTENCY: A UNIFIED TEST-TIME AGGREGATION FRAMEWORK WITH PAIRWISE COMPARISONS

Anonymous authors
Paper under double-blind review

ABSTRACT

This paper proposes *Joint Consistency* (JC), a unified framework for test-time aggregation that jointly models independent trace-level evaluations and pairwise comparisons. We cast JC as a constrained Ising-type energy minimization problem, which subsumes a broad class of existing aggregation schemes. We instantiate JC with LLM-as-a-judge comparative signals, characterize its theoretical behavior, and develop efficient approximations for practical deployment. Experiments on challenging math reasoning benchmarks show that JC outperforms state-of-the-art baselines across diverse architectures and trace budgets with marginal computational overhead, especially in crowdsourced settings.

1 INTRODUCTION

Test-Time Scaling (TTS) is a widely used strategy for improving the performance of large language models (LLMs) on complex reasoning tasks. In a broad class of TTS methods, candidate solutions are first *fully* generated and then evaluated and aggregated. These methods share a common post-hoc decision problem: *given a fixed set of reasoning traces generated at test time, how should they be aggregated into a single final answer?* This post-hoc problem arises in Self-Consistency (SC) (Wang et al., 2023), Weighted Self-Consistency (WSC) (Guo et al., 2025b), Best-of-N (Jinnai et al., 2025), Self-Certainty (Kang et al., 2025), and related approaches.

We formalize this problem as *Test-Time Aggregation* (TTA). Given N reasoning traces $\{y_i\}_{i=1}^N$, where each trace $y_i = (z_i, a_i)$ consists of a reasoning chain z_i and an answer a_i, the goal of TTA is to infer an *aggregated answer* $\hat{a}$.

While most prior work considers homogeneous settings where traces are sampled from a single model, many practical deployments involve *crowdsourced* traces generated by heterogeneous models or contributors. In such settings, answer frequency used in SC is no longer a reliable proxy for correctness, as it is confounded by varying generator reliability.

A common response is to independently evaluate each trace and aggregate the resulting scores via weighted aggregation (Taubenfeld et al., 2025). However, these approaches rely on well-calibrated *absolute* quality scores, which are difficult to obtain across heterogeneous sources and are misaligned with the inherently *ordinal* nature of aggregation. In contrast, comparative judgments focus on relative quality and are often more reliable, which is a principle well established in LLM alignment (Christiano et al., 2017; Ouyang et al., 2022). Yet, the challenge is that naively incorporating pairwise comparisons incurs a quadratic $O(N^2)$ cost, limiting scalability in test-time aggregation.

Contributions. We propose *Joint Consistency* (JC), a unified framework for Test-Time Aggregation (TTA) that jointly models independent trace-level evaluations and pairwise comparative signals.

- **Unified formulation for test-time aggregation.** JC unifies evaluation-free, independent-evaluation-based, and pairwise-comparison-based aggregation methods within a single objective. By casting TTA as a *constrained Ising-type energy minimization* problem (Section 2.1), JC subsumes a broad class of existing methods (Section B).
- **Interaction modeling via comparative signals.** JC incorporates pairwise comparative signals through an interaction matrix, instantiated using LLM-as-a-judge (Section 2.2). We characterize the resulting formulation under answer-level homogeneity (Theorem A.1), and develop efficient approximations that remain tractable at test time (Section 2.3).

- **Empirical gains on challenging benchmarks.** Across heterogeneous and homogeneous settings, JC outperforms SC and WSC, with over 20% absolute accuracy improvements on the hard problems, while incurring computational cost comparable to WSC (Section 3).

2 JOINT CONSISTENCY

We propose *Joint Consistency* (JC), a test-time aggregation framework that integrates independent trace-level evaluations and pairwise comparative signals via a *constrained Ising-type energy minimization* formulation. Independent evaluations assess traces in isolation and scale linearly, while pairwise comparisons capture trace- or answer-level relational structure but are potentially costly. JC exploits their *complementary* strengths: scalable coarse estimates from independent evaluations and fine-grained relational information from limited pairwise comparisons via efficient approximation. To isolate the effect of pairwise interactions, we also analyze a *pairwise-only* variant of JC.

2.1 JOINT CONSISTENCY AS A CONSTRAINED ISING MODEL

Joint Consistency admits a natural formulation as an *Ising-type* energy-based model, i.e.,

$$\min_{\mathbf{x}\in\{0,1\}^N} H(\mathbf{x}) = -\mu \langle \mathbf{h}, \mathbf{x} \rangle - \mathbf{x}^\top \mathbf{J}\mathbf{x}$$
$$\text{s.t.} \begin{cases} \forall k \in [K], \forall i,j \in \mathcal{I}_k,\ x_i = x_j, \\ \sum_{k=1}^{K} \frac{1}{|\mathcal{I}_k|} \sum_{i\in\mathcal{I}_k} x_i = 1. \end{cases} \tag{1}$$

where h_i encodes independent trace-level evaluations, J_{ij} (known as *interactions* in physics) encodes pairwise comparative signals, $\{\mathcal{I}_k\}_{k=1}^K$ is a partition of the trace index set $[N]$ with $\mathcal{I}_k$ containing all trace indices whose answer equals $a^{(k)}$, and $\mu > 0$ controls the relative strength between the two terms. We impose that all traces producing the same answer share the same indicator, and exactly one answer is selected. Under these constraints, the feasible set reduces to K answer-level configurations, and thus, the global minimizer is obtained by evaluating K candidates, making the global minimizer tractable to identify at inference time.

2.2 INTERACTION MODELING VIA LLM-AS-A-JUDGE (PAIRWISE-ONLY JC)

Setting $\mathbf{h} \equiv \mathbf{0}$ yields a pairwise-only variant driven solely by comparative signals from an LLM-as-a-judge. In general, an instantiation of $\mathbf{J}$ comprises two components: (i) a prompt strategy for eliciting comparative signals, and (ii) a mapping that transforms these generated signals into $\mathbf{J}$.

Example Interaction Matrix J. Here we detail a concrete construction. Let the prompt for eliciting comparative signals ask for the probability that reasoning trace y_i is better than y_j; see Appendix C.1.2. Given an LLM-as-a-judge π_θ, for two reasoning traces y_i, y_j, let $P_\theta(y_i \succeq y_j)$ denote the output probability generated by π_θ that y_i is better than y_j, and

$$p_\theta(y_i \succeq y_j) := \mathbb{E}\left[P_\theta(y_i \succeq y_j)\right] \tag{2}$$

denote its expectation. We construct $\mathbf{J}$ as a positive semi-definite matrix:

$$\mathbf{J}^\theta = \mathbf{C}\mathbf{C}^\top \tag{3}$$

where $\mathbf{C} \in \mathbb{R}^{N\times N}$ is defined entrywise by

$$C_{ij} := \sqrt{\frac{p_\theta(y_i \succeq y_j)}{n_i^2 n_j}}.$$

Here, n_i denotes the number of traces yielding the same answer as y_i. Namely, for $i \in \mathcal{I}_k$, $n_i = |\mathcal{I}_k|$.

Under an *answer-level homogeneity* assumption, the JC objective for an answer reduces to a sum of its pairwise win probabilities against other answers. Consequently, pairwise-only JC selects the answer with the largest aggregate preference score. Details of theoretical characterization (Theorem A.1) are provided in Section A.

Remark 2.1. *The specific construction of $\mathbf{J}^\theta$ in (3) is one illustrative choice and is not claimed to be optimal. A systematic study of optimal interaction matrix design is left for future work.*

Answer	Model	SC(↑)	WSC(↑)	JC-H (↓)
128	DeepSeek-v3.2-Exp (Think)	1	0.9	-0.4683
254	DeepSeek-v3.2-Speciale	1	1.0	-0.6733
2	GPT-5-nano (high)	1	0.4	-0.6583
2	Kimi K2 Thinking	1	0.3	
2	GPT-5.1 (high)	1	0.2	
2	GPT OSS 120B (high)	1	0.3	

Figure 1: Final aggregated answers of three TTS methods are highlighted in yellow on crowdsourced traces on Question 28 from HMMT 2025 (Nov) dataset, with six traces sampled from six frontier models, respectively. The ground-truth answer is 254. The table compares the aggregation results for SC and WSC [2]against $H(\mathbf{x})$ scores of our proposed JC method. For JC, we set $\mathbf{h} \equiv \mathbf{0}$ (pairwise-only variant) to isolate the effect of the interaction matrix and highlight its divergence from WSC.

2.3 Efficient Approximation of Joint Consistency

Naively constructing all pairwise interactions costs $O(N^2)$. Under answer-level homogeneity, only answer-level preferences are required (Theorem A.1), reducing complexity to $O(K^2)$. In practice, difficult problems can still exhibit a large K (e.g., see Table 2). We thus enforce scalability by considering only the *top-κ* most frequent answers. Introducing this *consistency budget* $\kappa > 0$ reduces the cost to $O(\kappa^2)$. Empirically, the optimal κ remains constant as N grows, and a small κ suffices (Section 3.1).

2.4 Joint Consistency over Crowdsourced Traces

Crowdsourced reasoning traces are inherently heterogeneous, rendering intrinsic-value-based WSC and PRM-based TTS methods inapplicable. Consequently, LLM-as-a-judge provides a natural and flexible source of extrinsic evaluation. We evaluate test-time aggregation methods on crowdsourced traces from MathArena Balunović et al. (2025). As shown in Fig. 6, judge-based scoring combined with simple filtering universally improves Pass@1, showing the effectiveness of the judge model.

Independent evaluation fails; pairwise comparison succeeds. Fig. 1 shows a representative failure case. Among six traces, four yield the same incorrect answer '2' while only one yields the correct answer '254'. Although the judge assigns the highest absolute score to the correct trace, WSC still selects '2', illustrating the difficulty of reliable absolute scoring. In contrast, consistent with Theorem A.1, our pairwise-only JC exploits answer-level preferences and correctly identifies '254'.

3 Experiments

3.1 Role of Interaction in Challenging Problems

The effect of modeling inter-response interactions $\mathbf{J}$ becomes most evident in *high-difficulty* crowdsourced settings, as aggregation methods tend to perform comparably on easy questions and thus offer limited discriminative power. We therefore focus on a subset of challenging problems from MathArena, characterized by (i) low Pass@1 accuracy and (ii) high answer diversity with a small average number of traces per answer (see Table 2). To isolate the role of interactions $\mathbf{J}$, we restrict attention to the pairwise-only JC by setting $\mathbf{h} \equiv \mathbf{0}$ *throughout this subsection*. Detailed setups are deferred to Section C.4.

Interaction modeling significantly improves TTA accuracy in challenging settings. As illustrated in Table. 1, the accuracies of SC and WSC plateau at approximately 36% and 62.5%, respectively, showing their marginal returns despite increasing trace budget. This stagnation stands in sharp contrast to homogeneous settings, where increasing the trace budget typically leads to consistent performance gains (Fu et al., 2025). Such behavior suggests that without modeling interactions, simply aggregating additional low-quality traces fails to refine the consensus toward the correct answer in the crowdsourced settings. In contrast, our pairwise-only JC (with $\kappa = 5$) does not exhibit

[2]Here SC/WSC denote aggregation over a set of candidate traces, which need not be generated by a single LLM.

Table 1: **Accuracy (%) and cost (USD) across varying consistency budgets (κ) and voting ratios (N) on challenging crowdsourced problems.** Each cell displays the format "Accuracy(%)/$Cost". Red text denotes the highest accuracy achieved for a specific N. Cells highlighted with a green background indicate configurations that outperform the best overall WSC accuracy (65%). Note that while WSC is theoretically independent of κ, minor fluctuations in its cost and performance are due to the sampling stochasticity of the traces.

N	0.10		0.20		0.40		0.60		0.80	
κ	JC ($\mathbf{h} \equiv \mathbf{0}$)	WSC ($\mathbf{J} \equiv \mathbf{0}$)	JC ($\mathbf{h} \equiv \mathbf{0}$)	WSC ($\mathbf{J} \equiv \mathbf{0}$)	JC ($\mathbf{h} \equiv \mathbf{0}$)	WSC ($\mathbf{J} \equiv \mathbf{0}$)	JC ($\mathbf{h} \equiv \mathbf{0}$)	WSC ($\mathbf{J} \equiv \mathbf{0}$)	JC ($\mathbf{h} \equiv \mathbf{0}$)	WSC ($\mathbf{J} \equiv \mathbf{0}$)
2	36.5%/$0.05	45.6%/$0.10	44.0%/$0.09	54.4%/$0.20	51.9%/$0.17	62.8%/$0.41	56.9%/$0.25	65.0%/$0.63	60.6%/$0.34	65.0%/$0.83
3	35.8%/$0.12	45.7%/$0.11	42.9%/$0.25	57.2%/$0.22	52.5%/$0.48	63.7%/$0.41	58.8%/$0.74	63.7%/$0.63	60.0%/$1.00	62.5%/$0.83
4	46.9%/$0.24	46.2%/$0.11	66.2%/$0.47	63.1%/$0.22	75.6%/$0.94	62.8%/$0.42	78.8%/$1.39	64.4%/$0.63	85.6%/$1.85	62.5%/$0.83
5	42.5%/$0.37	49.1%/$0.10	68.1%/$0.72	59.1%/$0.21	78.1%/$1.49	62.5%/$0.42	80.0%/$2.21	64.4%/$0.64	85.6%/$2.95	63.1%/$0.83
6	50.0%/$0.55	46.1%/$0.10	60.6%/$1.11	57.5%/$0.21	75.0%/$2.23	64.4%/$0.42	80.0%/$3.35	63.1%/$0.62	79.4%/$4.47	65.0%/$0.83
8	54.7%/$1.04	50.0%/$0.11	62.5%/$2.15	57.5%/$0.22	80.0%/$4.23	62.8%/$0.41	81.2%/$6.37	65.0%/$0.64	86.2%/$8.43	63.7%/$0.83
10	48.1%/$1.69	54.7%/$0.11	70.0%/$3.36	58.3%/$0.21	75.6%/$6.71	63.4%/$0.42	83.1%/$10.12	64.4%/$0.61	85.6%/$13.44	63.1%/$0.84
12	50.6%/$2.31	48.1%/$0.11	61.9%/$4.66	59.4%/$0.20	76.9%/$9.36	63.1%/$0.41	80.0%/$14.19	63.7%/$0.61	80.6%/$18.87	63.1%/$0.83
14	46.9%/$2.89	51.1%/$0.11	58.8%/$5.76	59.4%/$0.21	74.4%/$11.52	64.7%/$0.42	71.2%/$17.58	63.7%/$0.63	67.5%/$23.35	63.7%/$0.83
16	52.5%/$3.41	52.8%/$0.10	61.3%/$6.85	60.9%/$0.20	71.2%/$13.73	63.7%/$0.41	65.0%/$20.96	64.4%/$0.62	64.4%/$27.79	62.8%/$0.84
18	52.2%/$4.04	45.0%/$0.10	63.1%/$8.23	61.6%/$0.21	68.1%/$16.43	63.7%/$0.43	70.6%/$25.35	63.7%/$0.63	65.6%/$33.53	64.4%/$0.83
20	46.2%/$4.65	51.6%/$0.10	53.8%/$9.50	59.1%/$0.21	70.0%/$18.93	60.6%/$0.42	66.9%/$29.29	63.7%/$0.63	63.1%/$38.75	62.5%/$0.83

the same saturation behavior, instead maintaining a positive scaling law as N increases, ultimately reaching an accuracy of 85.6%. This represents a substantial 22.5% absolute improvement over the WSC baseline.

Interaction modeling is cost-efficient. By scaling the number of pairwise comparisons proportionally with N per answer pair, the cost of evaluating interactions $\mathbf{J}$ remains a constant factor of that of independent evaluations $\mathbf{h}$, provided that κ is not too large. For instance, at $\kappa = 5$, modeling $\mathbf{J}$ cost approximately $3.5\times$ of modeling $\mathbf{h}$ across all trace budgets. We observe that (i) **Pairwise-only JC is cost-efficient**: with a subset ratio of 0.2 and $\kappa = 4$, it achieves 66.2% accuracy at a cost of $\$0.47$, outperforming the best WSC baseline, which reaches 65% accuracy at a higher cost of $\$0.63$ using a larger trace budget ratio of 0.6; (ii) **WSC exhibits a performance ceiling**: increasing the budget fails to improve WSC beyond 65% accuracy, highlighting the limitations of independent-evaluation–based aggregation for complex reasoning tasks.

3.2 Large Scale Experiments on Crowdsourced Traces

In this section, we evaluate Joint Consistency with a non-zero linear term $\mathbf{h}$ to show that JC exploits the *complementary* strengths of independent evaluations and pairwise comparisons. We denote the linear coefficient as $\mathbf{h}^\theta$ when it is derived from the verbal confidence scores of a judge model π_θ. We use GPT-OSS-20B to score both individual traces (to determine $\mathbf{h}^\theta$) and pairwise interactions (to determine $\mathbf{J}^\theta$), using the prompts detailed in Appendix C.1.1 and Appendix C.1.2. Due to page limit, we defer the experimental results of the costs (Table 5), accuracy comparisons in crowdsourced settings (Table 7 and Table 8) to Appendix D.3.

Experiment 1: Evaluation of TTS methods across available crowdsourced traces. We evaluate our framework using traces from four datasets: AIME, BRUMO, HMMT 2025 Feb, and HMMT 2025 Nov, submitted to the MathArena Leaderboard. We compare our Joint Consistency method ($\mathbf{h}^\theta$) against several baselines: a pairwise-only variant ($\mathbf{h} \equiv \mathbf{0}$), SC ($\mathbf{h} \equiv \mathbf{1}$ and $\mathbf{J} \equiv \mathbf{0}$), and WSC ($\mathbf{J} \equiv \mathbf{0}$). We use three different trace budget ratios $\{10\%, 20\%, 50\%\}$ of the available traces. Across all experiments, we maintain a fixed sampling budget for $\mathbf{J}$ as specified in Table 5. To mitigate the effects of stochasticity, we report the average results over 200 independent trials. As shown in Table 7, our Joint Consistency method with $\mathbf{h}^\theta$ consistently outperforms all baselines across all datasets and trace budgets, demonstrating its robustness and effectiveness.

Experiment 2: Evaluation on lower-tier model subsets. To assess the impact of model capability and reduce reliance on high-performing solution generators, we specifically evaluate aggregation performance on a subset of the lowest-performing models. Specifically, we rank models by their Pass@1 accuracy on the AIME 2025 dataset and retain only the R least performing models, using 4 traces from each. We vary $R \in \{3, 5, 10, 15, 20, 30\}$ while keeping all other settings identical to Experiment 1. The results in Table 8 indicate that pairwise-only JC ($\mathbf{h} \equiv \mathbf{0}$) still maintains superior performance.

REFERENCES

Art of Problem Solving. 2025 AIME I. Art of Problem Solving Wiki, 2025a. URL https://artofproblemsolving.com/wiki/index.php/2025_AIME_I. Accessed: 2025.

Art of Problem Solving. 2025 AIME II. Art of Problem Solving Wiki, 2025b. URL https://artofproblemsolving.com/wiki/index.php/2025_AIME_II. Accessed: 2025.

Mislav Balunović, Jasper Dekoninck, Ivo Petrov, Nikola Jovanović, and Martin Vechev. Matharena: Evaluating llms on uncontaminated math competitions. *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmark*, 2025.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 2017.

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.

Yichao Fu, Xuewei Wang, Yuandong Tian, and Jiawei Zhao. Deep think with confidence. In *NeurIPS 2025 Workshop on Efficient Reasoning*, 2025.

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*, 2025a.

Jiaxin Guo, Zewen Chi, Li Dong, Qingxiu Dong, Xun Wu, Shaohan Huang, and Furu Wei. Reward reasoning models. In *Neural Information Processing Systems*, 2025b.

Yuu Jinnai, Tetsuro Morimura, Kaito Ariu, and Kenshi Abe. Regularized best-of-n sampling with minimum bayes risk objective for language model alignment. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, 2025.

Zhewei Kang, Xuandong Zhao, and Dawn Song. Scalable best-of-n selection for large language models via self-certainty. *arXiv preprint arXiv:2502.18581*, 2025.

OpenAI. gpt-oss-120b & gpt-oss-20b model card, 2025. URL https://arxiv.org/abs/2508.10925.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 2022.

Amir Taubenfeld, Tom Sheffer, Eran Ofek, Amir Feder, Ariel Goldstein, Zorik Gekhman, and Gal Yona. Confidence improves self-consistency in llms. *arXiv preprint arXiv:2502.06233*, 2025.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V Le, Ed H Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*, 2023.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

A THEORETICAL CHARACTERIZATION OF PAIRWISE-ONLY JC

Under an *answer-level homogeneity* assumption (4), pairwise-only Joint Consistency with (3) admits the following closed-form characterization.

Theorem A.1. *Assume that for any $k, k' \in [K]$, there is a constant $p_\theta(a^{(k)} \succeq a^{(k')})$ such that for all $i \in \mathcal{I}_k$ and $j \in \mathcal{I}_{k'}$,*

$$p_\theta(y_i \succeq y_j) = p_\theta(a^{(k)} \succeq a^{(k')})). \tag{4}$$

Then, for any feasible $\mathbf{x}$ of (1), i.e., $\mathbf{x} = \mathbb{1}_{\mathcal{I}_k}$ for some k,

$$\mathbf{x}^\top \mathbf{J}^\theta \mathbf{x} = \sum_{k'=1}^{K} p_\theta(a^{(k)} \succeq a^{(k')}). \tag{5}$$

Since $\mathbf{h} \equiv \mathbf{0}$, an optimizer of (1) satisfies

$$\hat{k} = \arg\max_{k \in [K]} \sum_{k'=1}^{K} p_\theta(a^{(k)} \succeq a^{(k')}), \tag{6}$$

where $p_\theta(y_i \succeq y_j)$ is defined in (2).

Theorem A.1 states that the JC objective for an answer reduces to a sum of its pairwise win probabilities against other answers under answer-level homogeneity. Consequently, pairwise-only JC selects the answer with the largest aggregate preference score.

Proof. Fix any feasible $\mathbf{x}$. By the constraints on $\mathbf{x}$, there exists a unique $k \in [K]$ such that $\mathbf{x} = \mathbb{1}_{\mathcal{I}_k}$, and hence

$$\mathbf{x}^\top \mathbf{J}^\theta \mathbf{x} = \sum_{i=1}^{N} \sum_{j=1}^{N} J_{ij}^\theta x_i x_j = \sum_{i,j \in \mathcal{I}_k} J_{ij}^\theta.$$

For any $i, j \in \mathcal{I}_k$, we have $n_i = n_j = |\mathcal{I}_k|$. Then,

$$\begin{aligned} J_{ij}^\theta &= \sum_{\ell=1}^{N} C_{i\ell} C_{j\ell} \\ &= \sum_{\ell=1}^{N} \frac{1}{|\mathcal{I}_k|^2 n_\ell} \sqrt{p_\theta(y_i \succeq y_\ell)\, p_\theta(y_j \succeq y_\ell)} \end{aligned}$$

By the homogeneity assumption of the theorem, for any $\ell \in \mathcal{I}_{k'}$, $p_\theta(y_i \succeq y_\ell) = p_\theta(y_j \succeq y_\ell) = p_\theta(a^{(k)} \succeq a^{(k')})$, and

$$\sqrt{p_\theta(y_i \succeq y_\ell)\, p_\theta(y_j \succeq y_\ell)} = p_\theta(a^{(k)} \succeq a^{(k')}).$$

It follows that

$$\begin{aligned} J_{ij}^\theta &= \sum_{\ell=1}^{N} \frac{1}{|\mathcal{I}_k|^2 n_\ell}\, p_\theta(a^{(k)} \succeq a^{(k')}) \\ &= \frac{1}{|\mathcal{I}_k|^2} \sum_{k'=1}^{K} \sum_{\ell \in \mathcal{I}_{k'}} \frac{1}{|\mathcal{I}_{k'}|}\, p_\theta(a^{(k)} \succeq a^{(k')}) \\ &= \frac{1}{|\mathcal{I}_k|^2} \sum_{k'=1}^{K} p_\theta(a^{(k)} \succeq a^{(k')}) \sum_{\ell \in \mathcal{I}_{k'}} \frac{1}{|\mathcal{I}_{k'}|} \\ &= \frac{1}{|\mathcal{I}_k|^2} \sum_{k'=1}^{K} p_\theta(a^{(k)} \succeq a^{(k')}), \end{aligned}$$

where we used $\sum_{\ell \in \mathcal{I}_{k'}} 1/|\mathcal{I}_{k'}| = 1$.

Finally, summing over $i, j \in \mathcal{I}_k$ yields

$$\begin{aligned} \mathbf{x}^\top \mathbf{J}^\theta \mathbf{x} &= \sum_{i,j \in \mathcal{I}_k} \frac{1}{|\mathcal{I}_k|^2} \sum_{k'=1}^{K} p_\theta(a^{(k)} \succeq a^{(k')}) \\ &= \sum_{k'=1}^{K} p_\theta(a^{(k)} \succeq a^{(k')}), \end{aligned}$$

which completes the proof.

□

B Existing Aggregation Methods as Special Cases of JC

The constrained Ising formulation in (1) provides a unifying view of a broad class of test-time aggregation methods, by choosing the linear term $\mathbf{h}$ and the interaction matrix $\mathbf{J}$ appropriately.

Methods that evaluate reasoning traces purely in isolation correspond to setting $\mathbf{J} \equiv \mathbf{0}$. In this case, the objective in (1) reduces to a linear form

$$H(\mathbf{x}) = -\langle \mathbf{h}, \mathbf{x} \rangle.$$

Self-Consistency (SC) (Wang et al., 2023) is recovered by choosing $\mathbf{J} \equiv \mathbf{0}$ and $h_i \equiv 1$ for all $i \in [N]$. Similarly, Weighted Self-Consistency (WSC) arises by setting $h_i = \omega(q, z_i)$, where $\omega(q, z_i)$ is the weight of y_i derived from intrinsic signals, such as average log-probabilities, or extrinsic signals, such as verbalized confidence scores generated by a judge LLM. Although Self-Certainty (Kang et al., 2025) aggregates traces according to their relative ranks, the underlying confidence scores used for ranking are evaluated independently for each trace, without reference to other candidates. Thus, Self-Certainty also falls into the category of $\mathbf{J} \equiv \mathbf{0}$, with h_i given by a Borda-style rank-based score $h_i = (N - r_i + 1)^q$, where r_i denotes the rank induced by the average log-probability of trace z_i, and $q > 0$ is a hyperparameter.

C Detailed Experimental Settings

C.1 Prompts

C.1.1 Ising-h Prompt

Prompt C.1: Ising-h Prompt

user

Please reason step by step, and put your final answer within `\\boxed{}`.
{question}

assistant

{response (cot+answer)}

user

Please evaluate the above answer based on the following criteria:

1. Is the answer correct?
2. Is the reasoning process correct?

Please choose an evaluation score among 0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0.
Please only output only the evaluation score.

C.1.2 ISING-J PROMPT

Prompt C.2: Ising-J Prompt

user

Suppose there are two responses to the same question. Please output the probability that Response 1 is a better answer than Response 2.

Question
{question}

Response 1
{response1 (cot+answer)}

Response 2
{response2 (cot+answer)}

Instruction
Now, please output the probability (a real number between 0 and 1) that Response 1 is a better answer than Response 2. Please only output the number.

C.1.3 GENERATION PROMPT

Prompt C.3: Generation Prompt

user

{question}
Please reason step by step, and put your final answer within `\\boxed{}`.

C.1.4 SCORING PROMPT

The scoring prompt is identical to the Ising-h prompt described in Subsection C.1.1.

C.2 STATISTICS FOR THE CHALLENGING SUBSET OF QUESTIONS

Table 2 presents the statistics for the challenging subset of questions that we experiment on in Section 3.1.

C.3 HYPERPARAMETERS

C.3.1 GENERATION HYPERPARAMETERS

Table 3 summarizes the decoding hyperparameters used during generation for all models. For each model, the temperature, top-p, top-k, and maximum generation length are fixed across experiments, and we use the model's native tokenizer. A dash (—) denotes that the corresponding decoding option is disabled. The Reasoning budget (API param) column specifies how each model's reasoning behavior is controlled during generation. For models that expose explicit API-level controls, we report the corresponding parameter setting (e.g., `reasoning_effort` for GPT-OSS). For models without a dedicated reasoning parameter, reasoning is enabled either by default or via prompt-level instructions.

Table 2: Statistics for the challenging subset of questions. Higher values (↑) for #Traces/#Ans and Pass@1 typically indicate *lower* problem difficulty.

Dataset	Q Idx	#Traces/#Ans (↑)	Pass@1 (↑)
AIME	14	228/75≈3.0	13.6%
	15	228/78≈2.9	3.5%
	All	Avg≈7.0	69.8%
BRUMO	13	152/20≈7.6	46.1%
	30	152/48≈3.2	21.1%
	All	Avg≈14.7	86.1%
HMMT Feb	19	228/54≈4.2	10.5%
	20	228/103≈2.2	12.7%
	All	Avg≈5.8	60.0%
HMMT Nov	10	60/11≈5.5	6.7%
	28	60/20=3.0	11.7%
	All	Avg≈14.4	87.8%

Table 3: Generation hyperparameters for different models.

Model	Temperature	Top-p	Top-k	Max seq len	Reasoning budget (API param)
Qwen3-8B	0.6	0.95	20	40k	True (`enable_thinking`)
Qwen3-32B	0.6	0.95	20	40k	True (`enable_thinking`)
DeepSeek-8B	0.6	0.95	—	130k	—

C.3.2 Scoring Hyperparameters

Table 4 summarizes the decoding hyperparameters used during scoring for all models. For each model, the temperature, top-p, top-k, and maximum generation length are fixed across experiments, and we use the model's native tokenizer. The Reasoning budget (API param) column specifies how each model's reasoning behavior is controlled during generation. For models that expose explicit API-level controls, we report the corresponding parameter setting (e.g., `reasoning_effort` for GPT-OSS). For models without a dedicated reasoning parameter, reasoning is enabled either by default or via prompt-level instructions.

C.4 Detailed Setups for Section 3.1

We compare pairwise-only JC with SC and WSC. To simulate varying resource constraints, aggregation is performed on trace subsets $\{10\%, 20\%, 40\%, 60\%, 80\%\}$ of the total available traces. For pairwise-only JC, we adopt the efficient approximation in Section 2.3, varying consistency budget $\kappa \in \{2, 3, \ldots, 20\}$. For each κ, only the top-κ most frequent answers are retained. $\mathbf{J}$ is estimated by sampling M trace pairs for each answer pair and querying an LLM-as-a-judge for comparative signals, yielding an $O(M\kappa^2)$ evaluation cost. To align cost with other baselines, we set M *proportional* to N, resulting in an overall cost of $O(N\kappa^2)$. Table 1 provides detailed results averaged over 20 random trials. We next present our main observations.

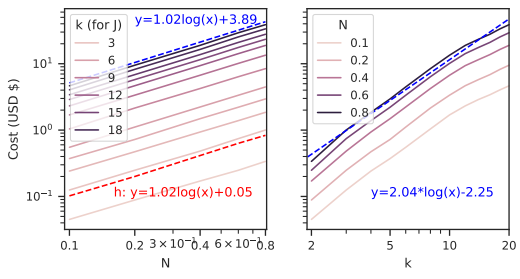

Figure 2: Cost scaling with respect to voting ratio N and consistency budget κ.

Table 4: Scoring hyperparameters for different models.

Model	Temperature	Top-p	Top-k	Max seq len	Reasoning budget (API param)
DeepSeek-8B	0.7	0.8	20	40k	Add <think>...</think> behind prompt
GPT-OSS-20B	1.0	1.0	0	40k	low (reasoning_effort)

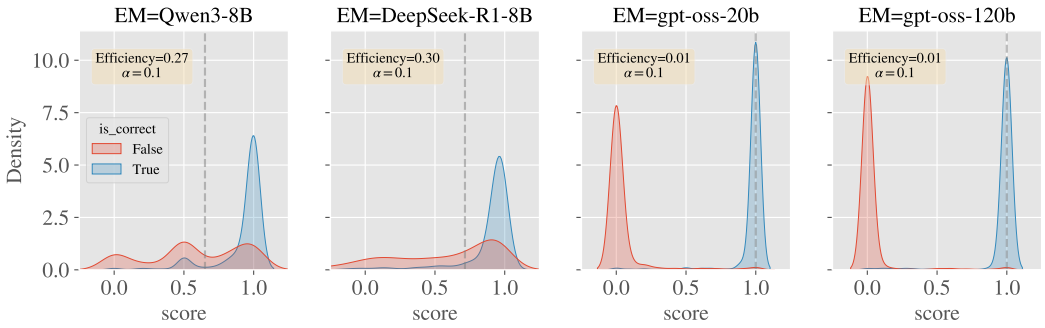

Figure 3: Note that the thinking mode of Qwen3 and DeepSeek models are both disabled while GPT-OSS models using a reasoning effort of low.

D Additional Experiments

D.1 Large Scale Experiments on Homogeneous Traces

To compare Joint Consistency (JC) with intrinsic-evaluation-based aggregation (WSC) baselines, we conduct large-scale experiments in a homogeneous setting on the AIME 2025 dataset (Art of Problem Solving, 2025a;b). We evaluate our approach against three representative baselines based on intrinsic evaluation: Self-Consistency (SC) Wang et al. (2023), Self-Certainty Kang et al. (2025), and DeepConf Fu et al. (2025). For trace generation, we use three Large Language Models (LLMs): Qwen3-8B, Qwen3-32B (Yang et al., 2025), and DeepSeek-R1-8B (Guo et al., 2025a). For the evaluation, DeepSeek-R1-8B, GPT-OSS-20B (OpenAI, 2025) are used as judge models. For each problem, we generate an initial pool of 256 reasoning traces, and uniformly subsample 64 traces for aggregation. To ensure statistical robustness, all experiments are repeated over 64 independent trials. Detailed prompts and hyperparameters are provided in Appendix C.1 and C.3, respectively.

Joint Consistency consistently outperforms baselines based on intrinsic evaluation. As shown in Table 6, JC achieves the best performance across all intrinsic test-time scaling baselines. We further observe that performance improves with the capability of both the trace generator and the judge. In particular, using GPT-OSS 20B as the judge consistently outperforms DeepSeek-R1 8B, suggesting that more powerful models provide more reliable evaluations. These results highlight the importance of judge model capacity in intrinsic-evaluation-based aggregation.

D.2 Empirical Trace Evaluation Results

Figure 6 shows Pass@1 accuracies of reasoning traces from 57 models on the HMMT-2025 (Feb). Confidence scores for these traces are evaluated using a GPT-OSS-20B model with low reasoning effort. The inset plot illustrates the score density functions for correct versus incorrect reasoning traces across the full set of 6,840 traces.

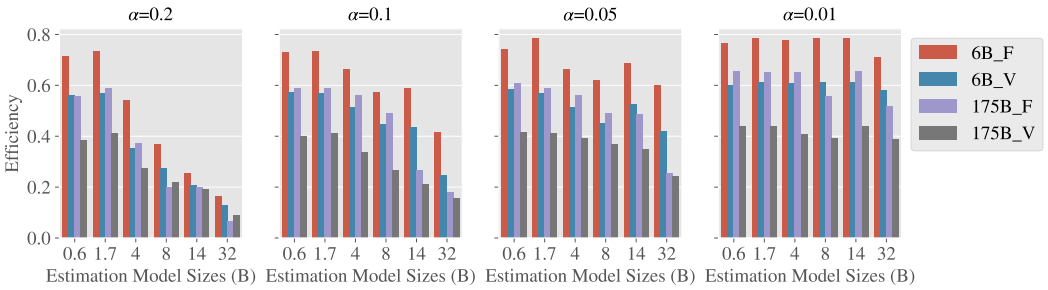

Figure 4: Efficiency evaluation of estimation models on the GSM-8K dataset (Cobbe et al., 2021). We assess the Qwen3 series (0.6B to 32B parameters) across given responses generated from 4 models 6B_F, 6B_V, 175B_F, and 175B_V. Model distinguishability is compared using efficiency scores across a range of hyperparameters ($\alpha \in \{0.2, 0.1, 0.05, 0.01\}$), where higher α values represent more conservative threshold selections. Detailed distributions are deferred to Fig. 5.

Table 5: Costs for MathArena Reasoning Tasks. This includes generation (G) costs and trace-scoring (Ising **h**/**J**) expenses. Generation figures are sourced from MathArena, while scoring costs are calculated via API usage. All costs are reported in $USD. The estimation model is GPT-OSS-20B with low reasoning effort.

			Trace (G)		Ising (**h**)			Ising (**J**)		
Dataset	#Q	#M	Size	Cost	Size	Cost	**h**/G	Size	Cost	**J**/G
AIME'25 [1]	30	58	6960	499.84	6960	3.25	0.65%	586	1.36	0.09%
BRUMO'25 [2]	30	38	4560	345.29	4560	1.60	0.46%	386	1.58	0.15%
HMMT'25-Feb [3]	30	57	6840	574.77	6840	2.78	0.48%	600	1.76	0.10%
HMMT'25-Nov [4]	30	15	1800	139.98	1800	0.74	0.53%	220	1.01	0.24%

[1] https://huggingface.co/datasets/MathArena/aime_2025_outputs
[2] https://huggingface.co/datasets/MathArena/brumo_2025_outputs
[3] https://huggingface.co/datasets/MathArena/hmmt_feb_2025_outputs
[4] https://huggingface.co/datasets/MathArena/hmmt_nov_2025_outputs

D.3 Tables for "Large Scale Experiments on Crowdsourced Traces"

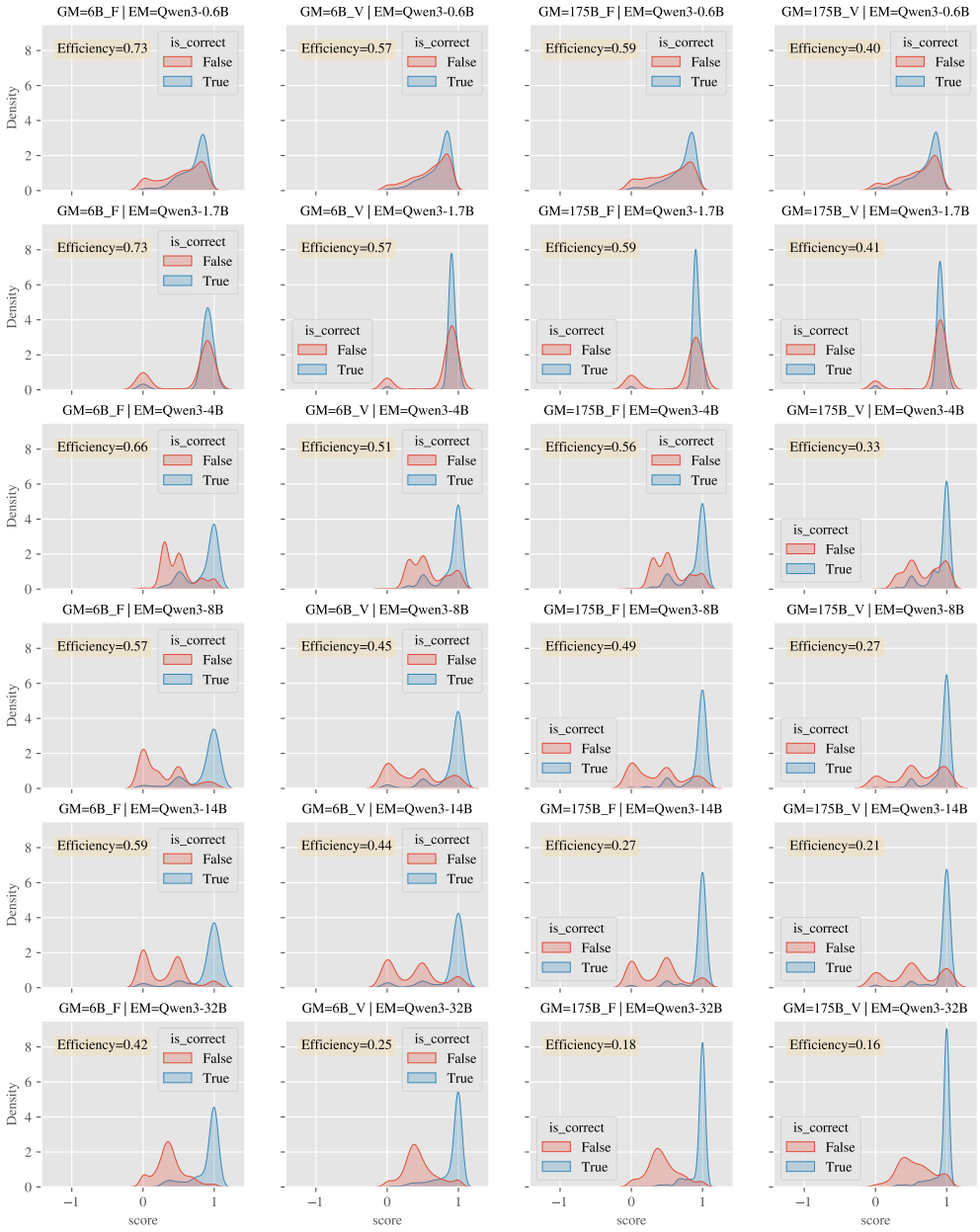

Figure 5: Kernel Density Estimation (KDE) of estimation scores across various Generative Models (GMs) and Estimation Models (EMs) on the GSM-8K dataset (Cobbe et al., 2021). The evaluation utilizes four GMs sourced from the GSM-8K benchmark and a suite of EMs from the Qwen3 series (Yang et al., 2025), with parameter scales ranging from 0.6B to 32B. Efficiency scores are derived using an adjustment parameter of $\alpha = 0.1$.

Table 6: **Test-time scaling accuracy on AIME 2025.** For Joint Consistency (JC), μ is fixed to 10. All experiments are repeated for 64 times.

Judge Models	TTS Methods	Trace Generation Models		
		Qwen3		DS-R1
		8B	32B	8B
-	SC	0.7703±0.0213	0.8021±0.0081	0.8307±0.0207
	Self-Cert.	0.7656±0.0204	0.8000±0.0144	0.8307±0.0238
	DeepConf	0.7677±0.0212	0.8026±0.0089	0.8312±0.0211
DS-R1 8B	WSC	0.7682±0.0224	0.8068±0.0158	0.8547±0.0223
	JC	**0.8411**±0.0218	**0.8490**±0.0176	**0.8708**±0.0217
GPT-OSS 20B	WSC	0.8135±0.0305	0.8339±0.0216	0.8844±0.0243
	JC	**0.8615**±0.0296	**0.9089**±0.0237	**0.9354**±0.0194

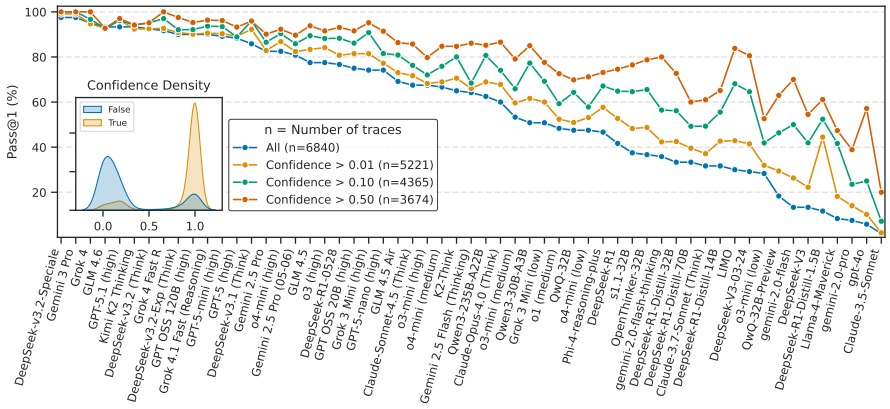

Figure 6: Pass@1 accuracies of reasoning traces from 57 models on the HMMT-2025 (Feb). Confidence scores for these traces are evaluated using a GPT-OSS-20B model with low reasoning effort. The inset plot illustrates the score density functions for correct versus incorrect reasoning traces across the full set of 6,840 traces.

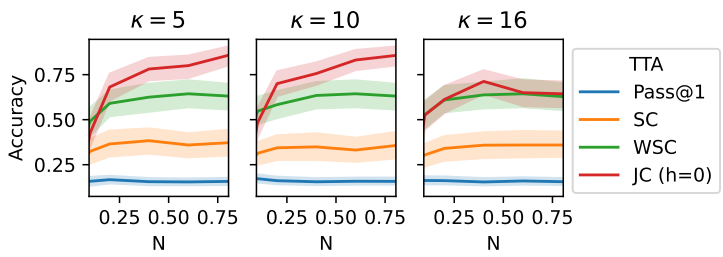

Figure 7: Accuracies on challenging reasoning problems across varying trace budgets N and consistency budget κ.

Table 7: **Accuracies of Joint Consistency methods on crowd-sourced traces from MathArena Platform**. The GPT-OSS-20B model with low reasoning effort is used to score both $\mathbf{h}$ and $\mathbf{J}$ with cost specified in Table 5. The μ is fixed to 10 for all experiments. For each ratio, we repeat subsampling for 200 times to reduce variance.

			$\mathbf{J} \equiv \mathbf{0}$		$\mathbf{J} = \mathbf{J}^\theta$	
Dataset	Ratio	Pass@1	$\mathbf{h} \equiv \mathbf{1}$	$\mathbf{h} = \mathbf{h}^\theta$	$\mathbf{h} \equiv \mathbf{0}$	$\mathbf{h} = \mathbf{h}^\theta$
AIME 2025	0.100	0.6986±0.05	0.9354±0.04	0.9534±0.04	0.9030±0.05	**0.9547**±0.04
	0.200	0.6995±0.04	0.9407±0.04	0.9600±0.04	0.8828±0.06	**0.9612**±0.04
	0.500	0.6984±0.04	0.9432±0.04	0.9658±0.03	0.8670±0.06	**0.9660**±0.03
BRUMO 2025	0.100	0.8612±0.04	0.9726±0.03	0.9900±0.02	0.9098±0.05	**0.9947**±0.01
	0.200	0.8624±0.04	0.9784±0.03	0.9969±0.01	0.8577±0.06	**0.9985**±0.01
	0.500	0.8612±0.03	0.9870±0.02	**1.0000**±0.00	0.7793±0.08	**1.0000**±0.00
HMMT 2025 (Feb)	0.100	0.5788±0.05	0.9219±0.05	0.9491±0.04	0.9367±0.04	**0.9655**±0.03
	0.200	0.5790±0.04	0.9393±0.04	0.9761±0.03	0.9235±0.05	**0.9825**±0.02
	0.500	0.5798±0.04	0.9430±0.04	0.9970±0.01	0.9022±0.05	**0.9977**±0.01
HMMT 2025 (Nov)	0.100	0.8793±0.04	0.9329±0.04	0.9334±0.05	0.8725±0.06	**0.9372**±0.04
	0.200	0.8792±0.04	0.9377±0.04	0.9390±0.04	0.8063±0.07	**0.9412**±0.04
	0.500	0.8788±0.04	0.9365±0.04	0.9387±0.04	0.6883±0.08	**0.9397**±0.04

Table 8: **Performance of TTS methods on traces from lower-tier models**. Utilizing AIME 2025 traces from the MathArena leaderboard, models are ranked by their baseline Pass@1 accuracies. The column R denotes the subset of the R lowest-ranked models whose traces were aggregated.

		$\mathbf{J} \equiv \mathbf{0}$		$\mathbf{J} = \mathbf{J}^\theta$	
R	Pass@1	$\mathbf{h} \equiv \mathbf{1}$	$\mathbf{h} = \mathbf{h}^\theta$	$\mathbf{h} \equiv \mathbf{0}$	$\mathbf{h} = \mathbf{h}^\theta$
3	11.7	20.3	25.5	25.7	**26.1**
5	17.2	29.4	33.7	34.8	**36.0**
10	27.1	43.7	50.6	**58.7**	54.7
15	34.7	56.1	71.2	**77.0**	71.7
20	40.9	70.4	76.3	**80.4**	77.7
30	52.1	83.3	89.0	88.0	**89.7**