When Wrong Answers Look Right: Multi-Agent Debate for High-Precision Verification of Hard Competition Mathematics

Anonymous ACL submission

Abstract

High-quality mathematical reasoning data are central to both training and evaluation, but such data are only useful when their answers and reasoning traces are reliable. Candidate solutions may be web-scraped, human-written, adapted from existing competition problems, or generated by LLMs; in all cases, a fluent but subtly wrong solution can silently contaminate downstream use. We present a high-precision multi-agent debate pipeline for blind verification of competition-level mathematics, in which five heterogeneous agents independently analyze a candidate answer, exchange structured arguments, and reach a verdict via an *assessment gate* requiring positive confirming evidence rather than mere non-refutation. On 195 hard-to-verify variants from 58 IMO/USAMO/Putnam-level problems, a single-judge baseline achieves 55.1% precision; our pipeline achieves **92.5%** precision, an $8.3\times$ false-positive reduction. A 2×2 factorial ablation with Shapley-fair attribution shows that 59% of the false-positive reduction stems from debate architecture—not merely additional inference calls—split between assessment gating and role heterogeneity. Anonymous code and data are available at: https://anonymous.4open.science/r/When-Wrong-Answers-Look-Right-5A57.

1 Introduction

High-quality mathematical reasoning data are central to both training and evaluation, but such data are useful only when their answers and reasoning traces are reliable. Candidate solutions may be web-scraped, human-written, adapted from competition problems, or LLM-generated; in all cases, a fluent but subtly wrong solution can silently contaminate downstream use. This makes verification a data filtering problem. For benchmark construction, training-data curation, reward labeling, and RLAIF, false positives are more damaging than false negatives: accepting an incorrect trace introduces a mislabeled example, whereas rejecting a correct trace merely reduces yield.

We study blind answer verification for competition-level mathematics: given a problem, a candidate answer, and a solution trace, decide whether the candidate is safe to accept without access to a reference answer. Single LLM judges fail because they are susceptible to the same surface-coherence cues that make wrong solutions look convincing; on our hard-to-verify variants, a single judge achieves only 55.1% precision.

Our study asks three questions. **Q1**: does verification provide useful signal beyond solving from scratch? **Q2**: does structured exchange reduce false positives beyond repeated independent verification? **Q3**: do heterogeneous roles and an assessment gate reduce correlated errors beyond homogeneous debate and flat voting? We answer Q2–Q3 through controlled baselines and a 2×2 factorial ablation, and treat Q1 as a diagnostic task-structure comparison rather than a fully compute-matched scaling law.

Our pipeline achieves **92.5%** precision, reducing false positives by $8.3\times$ over a single-judge baseline. Moreover, 59% of the false-positive reduction is attributable to debate architecture rather than merely repeated calls. These results support precision-first multi-agent debate as a filtering primitive for mathematical reasoning data, including benchmark curation, reward labeling, and LLM-generated data pipelines.

2 Task and Dataset

Task. Given a competition-level mathematics problem, a candidate numerical answer, and a solution trace, output CORRECT or WRONG, without access to the reference answer. The task is source-agnostic: the trace may be human-written,

web-scraped, adapted from existing problems, or LLM-generated. The deployment goal is not to identify the source, but to decide whether a candidate is safe to accept into a training, evaluation, or reward-labeling pool.

Our primary metric is **precision**: the fraction of accepted variants that are truly correct. A false positive admits an incorrect example into downstream data; a false negative merely reduces yield. We also report recall, FP count, and problem-level accuracy. A problem is correctly handled only when the system accepts the correct variant and rejects all wrong variants in a run.

Dataset. We evaluate on **195 hard-to-verify variants** from **58 IMO/USAMO/Putnam-level problems** spanning combinatorics, number theory, geometry, and algebra. The dataset is a verification stress test, not a generation benchmark: each item pairs a candidate answer with a plausible reasoning trace. Problems come from parameter-perturbed AMO-Bench instances (An et al., 2025), parameter-perturbed IMO/USAMO archive problems, and expert-contributed items. Each problem yields 3–4 variants, exactly one correct.

Correct answers are established by three-model triangulation: two reference-equipped models must agree, while an unguided zero-shot solver must fail; all 58 correct answers were manually reviewed. Wrong answers primarily come from natural LLM failures, supplemented by parameter reuse and targeted error injection. All wrong variants pass plausibility filtering: a simple judge must fail to immediately reject them. Using LLM failures provides a challenging proxy for plausible-but-wrong reasoning that can arise in any large-scale data curation pipeline. Full construction details appear in Appendix A.

3 Pipeline Architecture

Five heterogeneous agents. The pipeline employs five verification personas: (1) **Formalist Verifier**, checking algebraic consistency, arithmetic, and bounds; (2) **Theorem Auditor**, checking applicability of cited theorems and lemmas; (3) **Independent Resolver**, attempting a short independent re-derivation; (4) **Optimization Skeptic**, checking boundary conditions and constraint violations; and (5) **Pragmatic Cross-Checker**, performing numerical spot checks and small-case sanity tests. All five agents use the same base model, `gpt-5.2`, with different system instructions. Heterogeneity

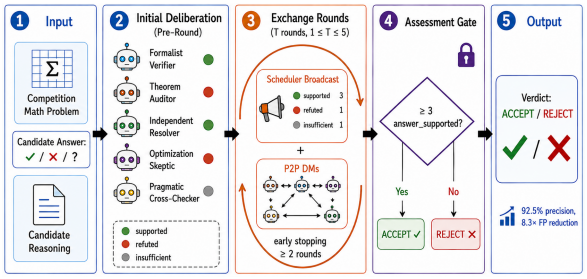

Figure 1: **Precision-first multi-agent debate pipeline.** Five heterogeneous agents **(2)** independently deliberate on a candidate answer, then exchange arguments for up to five rounds via scheduler broadcasts and peer-to-peer direct messages **(3)**. A verdict is accepted only when the **assessment gate (4)** is satisfied: ≥ 3 agents hold `answer_supported`—positive confirming evidence, not mere non-refutation—reducing false positives $8.3\times$ over a single-judge baseline.

targets correlated errors: two agents with different verification lenses are less likely to share the same blind spot than agents with identical prompts (Du et al., 2024; Li et al., 2026b).

Initial deliberation. Before any exchange, each agent independently: **(A)** restates the claim precisely; **(B)** enumerates 2–4 domain-specific failure modes and checks each, priming adversarial scrutiny before positive evidence is weighed; **(C)** performs one independent check, such as arithmetic, theorem applicability, or a small-case test; **(D)** steelmans the opposition, classifying the strongest counterargument as **FATAL**, **UNCERTAIN**, or **REFUTED**; and **(E)** commits to structured JSON output. The key field is `assessment_type`, which distinguishes `answer_supported` (positive confirming evidence), `answer_refuted` (positive disconfirming evidence), and `reasoning_insufficient` (uncertainty without a specific flaw). This three-way distinction is the basis for the assessment gate.

Exchange rounds. The system runs up to $R{=}5$ exchange rounds, with early stopping only after at least two full rounds. Each round delivers a deterministic scheduler broadcast: vote tally, `assessment_type` breakdown, strongest support and oppose claims, and a disagreement digest of unresolved counterarguments that every agent must address. Agents may also send up to two peer-to-peer direct messages to named recipients. Scheduler-generated challenges pressure weak supporters to produce a concrete check or

switch stance. Agents carry forward only a ≤400-character summary rather than the full chain, reducing context growth and long-history dependence (Yang et al., 2026). The scheduler itself makes no LLM calls.

Assessment gate. The verdict is accepted if and only if at least three of five agents carry `assessment_type = answer_supported`. Thus acceptance requires positive confirming evidence, not merely failure to find a flaw. A 5–0 support majority is still rejected if all agents assign `reasoning_insufficient`; in such cases, the exchange round forces agents to perform concrete checks. Appendix I.2 gives a case study where this gate prevents a false positive that flat majority voting would accept.

4 Experiments

Setup. All systems use `gpt-5.2` with `effort="low"`. We run three independent trials for main systems; ablations are single-run. Seven systems form a controlled ladder: **Single-Judge** (1 call/variant); **MajVote-Solve** (N=11 solve attempts/problem, majority-voted answer); **MajVote-Verify** (N=11 calls/variant, threshold $\lceil 0.6N \rceil = 7$); **MajVote-Precise** (same as MajVote-Verify with a precision-first prompt); **Debate-NoGate** (full debate, stance-only verdict); **Debate-Uniform** (full debate plus gate, five identical agents); and **Debate (Ours)** (heterogeneous agents, exchange, and assessment gate).

MajVote-Solve and MajVote-Verify allocate calls differently: solving produces one answer per problem, whereas verification evaluates candidate variants individually. We therefore treat solve-vs.-verify as a diagnostic comparison, not as a fully compute-matched scaling study. Within the verification ladder, MajVote-Verify and debate variants provide the main comparison for repeated verification versus debate architecture.

Main results. Table 1 shows that Debate reduces FP from 24.7 to 3.0 (8.3×) while raising precision from 55.1% to 92.5%; precision is stable across runs (90.2–95.0%). Repeated verification helps but is insufficient: MajVote-Verify reaches 75.1% precision and 15.7 FP/run, while Debate reaches 92.5% and 3.0 FP/run. MajVote-Precise improves only slightly over MajVote-Verify, indicating that prompt sharpening alone does not explain the gain.

Table 1: Results on 195 competition-math variants from 58 problems. Positive class = `correct`. †3-run average; ‡single run.

System	Prec.	Recall	FP	Prob. Acc.
Single-Judge†	55.1%	52.3%	24.7	33.3%
MajVote-Solve†	61.8%	42.5%	15.3	42.5%
MajVote-Verify†	75.1%	81.6%	15.7	60.9%
MajVote-Precise‡	75.9%	70.7%	13.0	56.9%
Debate-NoGate‡	84.8%	67.2%	7.0	60.3%
Debate-Uniform‡	88.1%	63.8%	5.0	58.6%
Debate (Ours)†	**92.5%**	63.8%	**3.0**	59.8%

Problem-level accuracy is similar for Debate and MajVote-Verify (59.8% vs. 60.9%), but their error profiles differ. MajVote-Verify accepts more true variants but also admits many wrong ones; Debate accepts fewer true variants but reduces contamination to 3.0 FP/run. For high-precision data filtering, this tradeoff is preferable.

Verify vs. solve. We include a diagnostic solve-vs.-verify comparison. MajVote-Solve majority-votes 11 solve attempts per problem and labels each variant correct if its candidate answer matches the majority answer. MajVote-Verify instead examines each candidate answer with its reasoning trace. MajVote-Verify achieves higher precision (75.1% vs. 61.8%) and recall (81.6% vs. 42.5%) with nearly identical FP counts (15.7 vs. 15.3), suggesting that a candidate trace provides useful signal when direct solving is unreliable. Because the procedures allocate computation differently, a fully compute-matched solve-vs.-verify scaling study remains future work.

5 Ablation: Decomposing the FP Reduction

The total FP reduction from Single-Judge to Debate is 21.7. MajVote-Verify, a repeated-verification baseline without debate, accounts for −9.0 FP (41%); the remaining **59% is architectural**. To isolate the structural core, we use a 2×2 factorial design over assessment gating and role heterogeneity:

	No Heterog.	Heterog.
No Gating	MajVote-Precise (13.0)	Debate-NoGate (7.0)
Gating	Debate-Uniform (5.0)	**Debate** (3.0)

Averaging over both ablation orderings gives Shapley-fair values: **assessment gating** −6.0 **FP**

(60%) and **role heterogeneity** -4.0 **FP (40%)**. Gating moves beyond flat voting: MajVote-Precise and Debate-Uniform use homogeneous agents, but requiring `answer_supported` rather than `support` drops FP from 13.0 to 5.0. This changes the flat-vote setting analyzed by Choi et al. (2025) by separating positive evidence from uncertainty. Heterogeneity further reduces correlated errors: role-differentiated agents check algebra, theorem applicability, independent derivations, boundary cases, and numerical sanity checks, making consensus less vulnerable to a shared blind spot.

6 Error Analysis

False positives. The remaining 3.0 FP/run share a consistent structure: the error is subtle (an off-by-one, modular slip, or hidden boundary issue), embedded in an otherwise valid template, and the final answer passes superficial sanity checks. No single agent's local check covers the flawed sub-step. This suggests that the residual false positives require tool augmentation, cross-provider model diversity, or formal verification support (Han et al., 2025).

False negatives. False negatives remain the main cost of the precision-first design. Most arise from capability saturation: agents recognize that a derivation is hard but fail to obtain enough positive evidence to pass the assessment gate. A smaller fraction comes from active false refutation, where an agent incorrectly identifies a flaw, or from mixed evidence, where support and uncertainty fail to clear the gate. These errors point to higher-effort inference, stronger steelman enforcement, and adaptive tool use as natural extensions.

7 Related Work

Judging and verification. LLM-as-a-Judge is widely used for scalable evaluation (Zheng et al., 2023), with variants improving alignment via chain-of-thought evaluation, rubrics, or judge tuning (Liu et al., 2023; Kim et al., 2024; Zhu et al., 2025). However, single-pass judges remain vulnerable to shallow reasoning and surface-form bias. Recent Agent-as-a-Judge discussions argue for more agentic evaluators using planning, tools, memory, and collaboration (You et al., 2026). Preference leakage (Li et al., 2026a), LLM-as-a-Verifier-style repeated checking (Kwok et al., 2026), and process supervision (Lightman et al., 2024) further motivate verifiers robust to fluent but wrong reasoning.

Multi-agent debate. Prior work uses debate and multi-agent discussion to improve factuality, evaluation, and reasoning (Irving et al., 2018; Chan et al., 2024; Du et al., 2024; Liang et al., 2024; Li et al., 2024). These systems typically rely on flat voting or general-purpose discussion; Choi et al. (2025) show that homogeneous flat-vote debate need not improve expected correctness. Our pipeline instead combines heterogeneous roles, structured exchange, and an assessment gate requiring positive evidence, aligning with dynamic debate systems that emphasize diverse reasoning trajectories (Li et al., 2026b).

Mathematical reasoning data. Tool-augmented verifiers ground judgments through execution, search, or formal checking (Han et al., 2025; Ospanov et al., 2025), while agentic reward modeling uses verifiable correctness signals (Peng et al., 2025). FrontierMath, Humanity's Last Exam, AMO-Bench, and VeRA highlight the need for difficult mathematical evaluation data and verified augmentation (Glazer et al., 2024; Center for AI Safety et al., 2026; An et al., 2025; Cheng et al., 2026). Test-time scaling and confidence-estimation work show that multiple reasoning paths contain useful signal (Wang et al., 2023; Snell et al., 2024; Zhang et al., 2025); we use structured disagreement and heterogeneous roles to reduce false positives in blind competition-math verification.

8 Conclusion

We present a high-precision multi-agent debate pipeline for blind verification of hard competition mathematics. The system reduces false positives by $8.3\times$ over a single-judge baseline and reaches 92.5% precision on 195 hard-to-verify variants. A factorial ablation shows that assessment gating and role heterogeneity independently reduce false positives, supporting multi-agent debate as a practical mechanism for reliable mathematical data filtering. More broadly, the verifier is source-agnostic: it can filter web-scraped solutions, human-written derivations, adapted competition problems, benchmark items, reward-labeling traces, and LLM-generated candidates. This framing separates verification from synthetic generation; the central requirement is not where a trace comes from, but whether it is safe to accept. Such a verifier can therefore support both ordinary benchmark curation and self-improving mathematical AI systems that rely on generate–verify–train loops.

Limitations

First, recall remains limited at 63.8%. This is a deliberate consequence of the precision-first design: the verifier rejects candidates when agents cannot obtain positive confirming evidence. For data filtering this is often acceptable, but it reduces yield. Improving recall without admitting false positives will likely require stronger models, higher-effort inference, cross-provider ensembles, or tool-augmented verification.

Second, our solve-vs.-verify comparison is diagnostic rather than a complete compute-scaling study. Solving and verification allocate computation differently: solving produces one answer per problem, while verification evaluates candidate variants individually. A fully compute-matched study that equalizes total calls across problem sizes, variant counts, and inference budgets remains future work.

Third, our evaluation is limited to 195 variants from 58 competition-mathematics problems. Larger and more diverse benchmarks are needed to support stronger significance testing and to characterize performance across mathematical domains, difficulty levels, and error types.

Finally, the current system is deliberately tool-free and uses fixed agent roles. This isolates the effect of debate architecture, but it limits coverage on problems that require exact symbolic computation, diagrammatic reasoning, or formal proof checking. Future systems could route difficult cases to Python, symbolic algebra systems, theorem provers, web search, or domain-specific tools while preserving the precision-first objective.

Ethical Considerations

This work aims to improve the reliability of mathematical reasoning data used for training and evaluation. A high-precision verifier may reduce the amount of incorrect reasoning admitted into benchmarks, reward-labeling pools, or training data. However, automated verification systems can also create a false sense of certainty if their accepted outputs are treated as infallible. Our precision-first design mitigates this risk by preferring rejection over accepting weakly supported examples, but human auditing remains important in high-stakes deployments.

References

Shengnan An, Xunliang Cai, Xuezhi Cao, Xiaoyu Li, Yehao Lin, Junlin Liu, Xinxuan Lv, Dan Ma, Xuanlin Wang, Ziwen Wang, and Shuang Zhou. 2025. Amobench: Large language models still struggle in high school math competitions. *CoRR*.

Center for AI Safety, Scale AI, and HLE Contributors Consortium. 2026. A benchmark of expert-level academic questions to assess AI capabilities. *Nature*, 649:1139–1146.

Chi-Min Chan, Weize Chen, Yusheng Su, Jianxuan Yu, Wei Xue, Shanghang Zhang, Jie Fu, and Zhiyuan Liu. 2024. Chateval: Towards better llm-based evaluators through multi-agent debate. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*.

Zerui Cheng, Jiashuo Liu, Chunjie Wu, Jianzhu Yao, Pramod Viswanath, Ge Zhang, and Wenhao Huang. 2026. Vera: Verified reasoning data augmentation at scale. *CoRR*, abs/2602.13217.

Hyeong Kyu Choi, Xiaojin Zhu, and Yixuan Li. 2025. Debate or vote: Which yields better decisions in multi-agent large language models? *CoRR*, abs/2508.17536.

Yilun Du, Shuang Li, Antonio Torralba, Joshua B. Tenenbaum, and Igor Mordatch. 2024. Improving factuality and reasoning in language models through multiagent debate. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, pages 11733–11763.

Elliot Glazer, Ege Erdil, Tamay Besiroglu, Diego Chicharro, Evan Chen, Alex Gunning, Caroline Falkman Olsson, Jean-Stanislas Denain, Anson Ho, Emily de Oliveira Santos, Olli Järviniemi, Matthew Barnett, Robert Sandler, Matej Vrzala, Jaime Sevilla, Qiuyu Ren, Elizabeth Pratt, Lionel Levine, Grant Barkley, and 5 others. 2024. Frontiermath: A benchmark for evaluating advanced mathematical reasoning in AI. *CoRR*, abs/2411.04872.

Jiuzhou Han, Wray L. Buntine, and Ehsan Shareghi. 2025. Verifiagent: a unified verification agent in language model reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2025, Suzhou, China, November 4-9, 2025*, pages 16410–16431.

Geoffrey Irving, Paul F. Christiano, and Dario Amodei. 2018. AI safety via debate. *CoRR*, abs/1805.00899.

Seungone Kim, Jamin Shin, Yejin Choi, Joel Jang, Shayne Longpre, Hwaran Lee, Sangdoo Yun, Seongjin Shin, Sungdong Kim, James Thorne, and Minjoon Seo. 2024. Prometheus: Inducing fine-grained evaluation capability in language models. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*.

Jacky Kwok, Shulu Li, Pranav Atreya, Yuejiang Liu, Marco Pavone, Ion Stoica, and Azalia Mirhoseini. 2026. LLM-as-a-verifier: A general-purpose verification framework. Notion Blog. https://llm-as-a-verifier.notion.site/.

Dawei Li, Renliang Sun, Yue Huang, Ming Zhong, Bohan Jiang, Jiawei Han, Xiangliang Zhang, Wei Wang, and huan liu. 2026a. Preference leakage: A contamination problem in LLM-as-a-judge. In *The Fourteenth International Conference on Learning Representations*.

Yu Li, Shenyu Zhang, Rui Wu, Xiutian Huang, Yongrui Chen, Wenhao Xu, Guilin Qi, and Dehai Min. 2024. Mateval: A multi-agent discussion framework for advancing open-ended text evaluation. In *Database Systems for Advanced Applications - 29th International Conference, DASFAA 2024, Gifu, Japan, July 2-5, 2024, Proceedings, Part VII*, pages 415–426.

Zhenghao Li, Zhi Zheng, Wei Chen, Jielun Zhao, Yong Chen, Tong Xu, and Enhong Chen. 2026b. Dynadebate: Breaking homogeneity in multi-agent debate with dynamic path generation. *CoRR*, abs/2601.05746.

Tian Liang, Zhiwei He, Wenxiang Jiao, Xing Wang, Yan Wang, Rui Wang, Yujiu Yang, Shuming Shi, and Zhaopeng Tu. 2024. Encouraging divergent thinking in large language models through multi-agent debate. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, EMNLP 2024, Miami, FL, USA, November 12-16, 2024*, pages 17889–17904.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2024. Let's verify step by step. In *International Conference on Learning Representations*, volume 2024, pages 39578–39601.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-eval: NLG evaluation using gpt-4 with better human alignment. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 2511–2522.

Azim Ospanov, Zijin Feng, Jiacheng Sun, Haoli Bai, Shen Xin, and Farzan Farnia. 2025. HERMES: towards efficient and verifiable mathematical reasoning in llms. *CoRR*, abs/2511.18760.

Hao Peng, Yunjia Qi, Xiaozhi Wang, Zijun Yao, Bin Xu, Lei Hou, and Juanzi Li. 2025. Agentic reward modeling: Integrating human preferences with verifiable correctness signals for reliable reward systems. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pages 15934–15949.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. 2024. Scaling LLM test-time compute optimally can be more effective than scaling model parameters. *CoRR*, abs/2408.03314.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*.

Zhicheng Yang, Zhijiang Guo, Yinya Huang, Yongxin Wang, Wenlei Shi, Yiwei Wang, Xiaodan Liang, and Jing Tang. 2026. Accordion-thinking: Self-regulated step summaries for efficient and readable LLM reasoning. *CoRR*, abs/2602.03249.

Runyang You, Hongru Cai, Caiqi Zhang, Qiancheng Xu, Meng Liu, Tiezheng Yu, Yongqi Li, and Wenjie Li. 2026. Agent-as-a-judge. *Preprint*, arXiv:2601.05111.

Caiqi Zhang, Chang Shu, Ehsan Shareghi, and Nigel Collier. 2025. All roads lead to rome: Graph-based confidence estimation for large language model reasoning. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, EMNLP 2025, Suzhou, China, November 4-9, 2025*, pages 31814–31824.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Lianghui Zhu, Xinggang Wang, and Xinlong Wang. 2025. Judgelm: Fine-tuned large language models are scalable judges. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*.

A Dataset Construction Details

Problem sources. The 58 source problems are drawn from three channels: (i) parameter-perturbed AMO-Bench instances, (ii) parameter-perturbed IMO/USAMO archive problems, and (iii) expert-contributed problems with hand-verified answers. Each problem yields 3–4 variants, exactly one of which is labeled `correct`, for a total of 195 variants.

Correct-answer construction. Correct answers are established by three-model triangulation. Two reference-equipped solvers are given the original competition problem's official answer and solution as guidance for validating the perturbed instance; they are not given the perturbed instance's answer. An unguided solver receives only the perturbed problem statement. A candidate answer is accepted only when the two reference-equipped solvers agree and the unguided solver fails. Manual review checks both the final numerical answer and the decisive derivation steps. All 58 correct answers passed manual review.

Wrong-answer construction. The dataset contains 137 wrong variants across 58 problems. Wrong answers come primarily from natural LLM failures: the model solves each problem from scratch, and genuinely incorrect outputs are retained together with their original reasoning traces. We additionally include a small number of parameter-reuse and targeted error-injection variants to cover underrepresented failure modes. We do not use random numerical distractors; every wrong candidate is paired with a plausible solution trace.

Plausibility filtering. Wrong variants are retained only if a simple single-judge verifier fails to immediately reject them. This removes trivially detectable arithmetic or formatting errors and ensures that retained wrong variants are fluent, coherent, and nontrivial to refute. Using LLM failures does not restrict the deployment setting to LLM-generated data; rather, it provides a challenging proxy for plausible-but-wrong reasoning that can arise in any large-scale data curation pipeline.

Quality gates. Each retained item passes three gates: (1) **correctness**, via multi-provider agreement and manual review; (2) **difficulty**, via zero-shot solve failure; and (3) **plausibility**, via simple-judge filtering.

B System Comparison Table

Our work is the only system in this comparison that simultaneously targets **Olympiad-level** competition mathematics (IMO/USAMO/Putnam), operates without ground-truth access, pursues a precision-first objective, and achieves high precision (92.5%) without external tools. Prior math-focused systems (DynaDebate, VerifiAgent) evaluate on standard benchmarks up to AMC/AIME difficulty; scaling to the Olympiad regime introduces qualitatively harder proof-based derivations where single-judge precision collapses to near chance (55.1%). The assessment gate and the 2×2 factorial attribution study are unique to our design.

C Full Results Table

D Pipeline Protocol Details

D.1 Initial Deliberation JSON Schema

Each agent's initial output includes the following fields: `verdict` (binary support/oppose), `assessment_type` (`answer_supported` / `answer_refuted` / `assessment_type = reasoning_insufficient_but_answer_not_refuted`, `evidence_grade` (`strong`/`medium`/`weak`), `rationale` (full chain-of-thought), `summary` (≤400 chars, propagated to subsequent rounds following the Accordion principle (Yang et al., 2026)), `swing_issue` (single most decisive factor, one sentence), `key_checks` (list of specific checks performed), `confidence` ($[0, 1]$ scalar), `outbound_broadcast` (one claim for all agents to address), `outbound_dm` (list of 0–2 targeted messages to specific agents).

D.2 Scheduler Logic

The scheduler is deterministic and generates no LLM calls. It assembles the broadcast and challenge messages programmatically from the previous round's `swing_issue` and `summary` fields. Three challenge categories are generated: (i) cross-challenge between the highest-priority support and oppose agents; (ii) weak-supporter pressure demanding a concrete check or stance switch; (iii) undecided-resolution directing `reasoning_insufficient` agents to compare competing claims.

Table 2: Comparison of representative LLM evaluation and verification systems across seven key dimensions. *Domain*: competition-level math (Comp. Math), general NLP, long-horizon agent benchmarks (Agents). *Prob. Level*: difficulty of math problems, where applicable ("—" = non-math domain; Easy–Med. = GSM8K / MATH-dataset level; AMC/AIME = competition level up to and including AIME; **Olympiad** = IMO/USAMO/Putnam). *Agents*: H = heterogeneous roles; K = K i.i.d. re-calls. *Verdict*: flat vote, debate + flat vote, or debate + assessment gate. *Tools*: external Python/solver tools. *Blind*: no ground-truth access. *Prec.-first*: optimized to minimize false positives.

System	Domain	Prob. Level	Agents	Verdict	Tools	Blind	Prec.-first
LLM-as-a-Judge (Zheng et al., 2023)	General NLP	—	1	Single-pass	—	✓	×
LLM-as-a-Verifier (Kwok et al., 2026)	Agents	—	$1\times K$	MajVote	—	✓	×
ChatEval (Chan et al., 2024)	General NLP	—	N (H)	Debate + vote	—	✓	×
DynaDebate (Li et al., 2026b)	Reasoning	AMC/AIME	N (H)	Debate + vote	Code, Search	✓	×
VerifiAgent (Han et al., 2025)	Math / Code	Easy–Med.	1 (loop)	Agentic chain	Python, Z3	×	×
Debate (Ours)	**Comp. Math**	**Olympiad**	**5 (H)**	**Debate + gate**	**—**	✓	✓

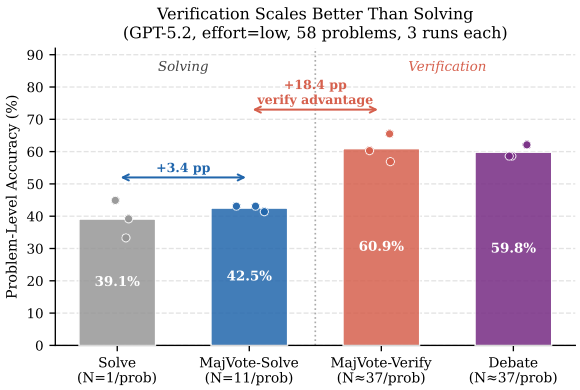

Figure 2: Diagnostic comparison of solving vs. verification. The two procedures allocate calls differently: solving produces one answer per problem, whereas verification evaluates candidate variants individually. At the variant level (Table 1), both systems produce similar false-positive counts (15.3 vs. 15.7), while verification achieves higher recall (81.6% vs. 42.5%).

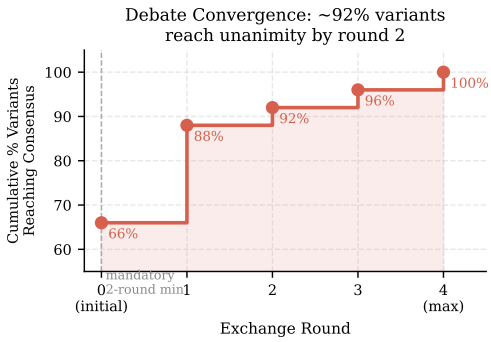

Figure 3: Cumulative debate convergence by exchange round. 92% of variants reach unanimous consensus by round 2, keeping average cost at ≈11 calls/variant despite a 5-round maximum. Only 8% require the full five rounds.

D.3 Problem Difficulty Distribution

Of the 58 problems, GPT-5.2 (effort=low) solves from scratch: ≥2/3 runs: 25 problems; exactly 1/3 runs: 9 problems; 0/3 runs: 24 problems. The 24 unsolvable-from-scratch problems represent the hardest tier, where candidate-trace verification appears most useful relative to direct solving.

D.4 Debate Convergence and FP Ladder

E Baseline and Evaluation Protocol

Variant-level labels for MajVote-Solve. MajVote-Solve operates at the problem level. It performs 11 independent solve attempts and majority-votes the numerical answer. At the variant level, a variant is labeled CORRECT if and only if its candidate answer exactly matches this majority-voted answer. Thus a successful solve labels all variants of the same problem simultaneously, whereas verification systems classify each variant independently. For this reason, the solve-vs.-verify comparison is diagnostic rather than a complete compute-scaling study.

Call-budget accounting. Table 4 summarizes per-problem call budgets.

F Additional Results

G Why Debate Breaks the Martingale

Choi et al. (2025) prove that for homogeneous agents in iterated argument exchange, the expected vote tally is a martingale: debate adds no information beyond a single majority vote over the same number of calls. Our ablation (Debate-Uniform FP=5.0 vs. MajVote-Verify FP=15.7) shows this prediction fails when three mechanisms are simultaneously present:

(1) Assessment gating. The verdict rule rejects answers even when a *majority* of agents

Table 3: Per-run results for all systems. 195 variants, 58 problems. All systems: `gpt-5.2`, `effort="low"`. †MajVote-Verify precision/recall: per-run values; avg computed from pooled TP/FP. ‡Single-run result (ablation variant).

System	Run	Acc	Prec	Recall	F1	FP	FN	Prob. Acc
Single-Judge	r1	72.8%	54.5%	51.7%	53.1%	25	28	18/58
	r2	72.8%	54.9%	48.3%	51.4%	23	30	18/58
	r3	73.9%	55.9%	56.9%	56.4%	26	25	22/58
	avg	**73.2%**	**55.1%**	**52.3%**	**53.6%**	**24.7**	**27.7**	**19.3/58**
MajVote-Verify ($N = 11$)†	r1	87.7%	76.6%	84.5%	80.3%	15	9	38/58
	r2	85.6%	73.4%	81.0%	77.1%	17	11	33/58
	r3	86.2%	75.4%	79.3%	77.3%	15	12	35/58
	avg	**86.5%**	**75.1%**	**81.6%**	**78.2%**	**15.7**	**10.7**	**35.3/58**
MajVote-Precise‡	–	84.6%	75.9%	70.7%	73.2%	13	17	33/58
Debate-NoGate‡	–	86.7%	84.8%	67.2%	75.0%	7	19	35/58
Debate-Uniform‡	–	86.7%	88.1%	63.8%	74.0%	5	21	34/58
Debate (Ours)	r1	87.2%	90.2%	63.8%	74.7%	4	21	34/58
	r2	87.2%	92.3%	62.1%	74.2%	3	22	34/58
	r3	88.7%	95.0%	65.5%	77.6%	2	20	36/58
	avg	**87.7%**	**92.5%**	**63.8%**	**75.5%**	**3.0**	**21.0**	**34.7/58**

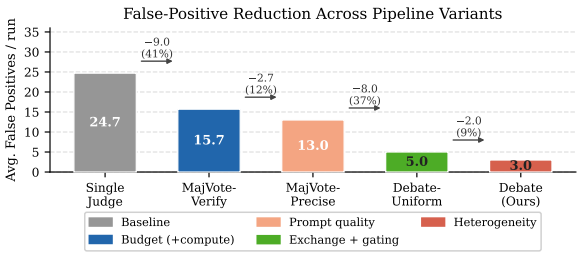

Figure 4: Step-by-step false-positive reduction across pipeline variants, coloured by mechanism type. Arrows show the FP delta and its share of the total 21.7-unit reduction.

Table 4: Approximate call budgets. Debate and repeated-verification baselines operate at comparable per-variant budgets, while MajVote-Solve allocates calls at the problem level.

System	Calls/variant	Calls/problem
Single-Judge	1	3–4
MajVote-Solve	—	11
MajVote-Verify	11	33–44
MajVote-Precise	11	33–44
Debate-NoGate	≈11	≈33–44
Debate-Uniform	≈11	≈33–44
Debate (Ours)	≈11	≈33–44

voted `support`, if fewer than three produced `answer_supported` (positive evidence) rather than `reasoning_insufficient` (uncertainty). This precision-biased filter has no analogue in majority voting and contributes approximately **31%** of the structural FP reduction over MajVote-Verify: removing it raises FP from 3.0 to 7.0 (Debate-NoGate, Table 1).

(2) Adversarial framing. Wrong variants carry the model's own fluent reasoning, so a naive majority vote ("does this look correct?") reliably misjudges them. The debate protocol forces agents to steelman the opposition and find a FATAL objection—a materially stronger epistemic standard. Even *homogeneous* agents surface genuine disagreement on contested variants (61 of 195 variants per run undergo at least one stance change), generating signal that a flat vote cannot capture.

(3) Role heterogeneity. Each persona analyzes the problem through a distinct lens: algebraic consistency, theorem applicability, independent re-derivation, boundary conditions, numerical spot-checking. Two agents with different lenses who independently agree have a lower probability of correlated error than two agents sharing a lens. The information value of heterogeneous consensus is strictly higher, and accounts for an additional **16%** of structural FP reduction (Debate-Uniform → Debate: FP=5.0→3.0). This is consistent with the pre-

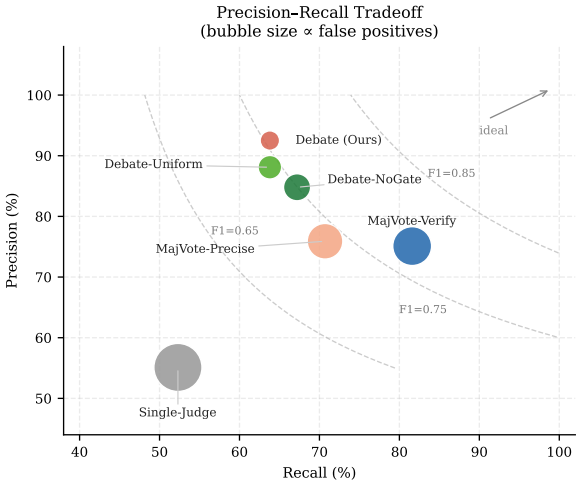

Figure 5: Precision–recall tradeoff across all pipeline variants. Bubble size is proportional to FP/run. Debate achieves the highest precision and lowest false-positive count.

Table 5: Per-problem correct count (0–3) for MajVote-Verify and Debate. A problem is "correct" only when the correct variant is accepted *and* all wrong variants are rejected in that run. Selected problems; full data in Figure 7. Abbrev.: Dbt = Debate, MV = MajVote-Verify.

Prob	MV	Dbt	Prob	MV	Dbt	Prob	MV	Dbt
P1	3	3	P22	3	3	P43	3	3
P2	3	3	P23	0	0	P44	3	3
P3	3	3	P24	0	0	P45	3	3
P4	3	3	P25	0	0	P46	1	0
P5	3	3	P26	0	3	P47	0	0
P6	3	3	P27	0	2	P48	2	3
P7	0	0	P28	0	2	P49	2	0
P8	0	0	P29	0	0	P50	3	3
P12	3	0	P30	0	0	P51	0	3
P13	0	0	P31	0	0	P52	1	1
P20	3	0	P32	0	0	P55	2	3
P21	3	3	P38	3	0	P57	3	3

diction of Li et al. (2026b) and the graph-theoretic view of self-consistency in Zhang et al. (2025).

H Per-Problem Analysis

Figure 7 visualizes the per-problem outcome across three runs for MajVote-Verify and Debate side by side. Table 5 lists per-problem correct counts for a representative subset. A problem is counted as correct in a given run only when the pipeline accepts the correct variant *and* rejects all wrong variants; 0–3 indicates how many of the three independent runs achieved this.

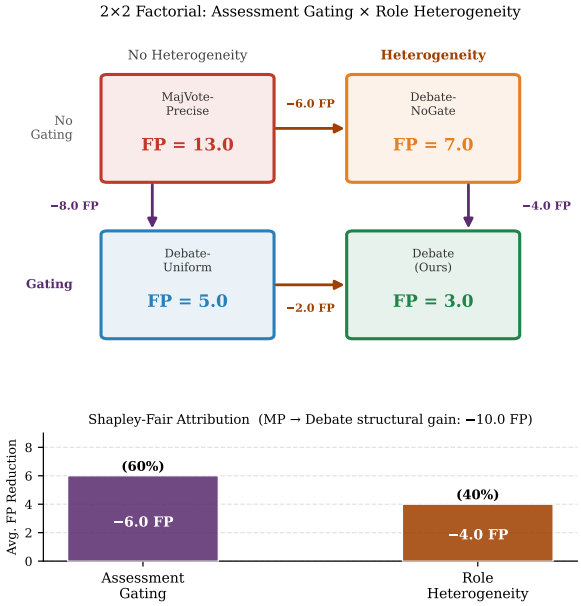

Figure 6: *Top*: 2×2 factorial design; each cell shows FP/run. *Bottom*: Shapley-fair attribution of the structural FP reduction, with assessment gating contributing 60% and role heterogeneity 40%.

H.1 MajVote-Verify and Debate Problem-Level Correct Counts (out of 3 runs)

The table reveals five interpretable patterns related to the four difficulty tiers described in Section 6:

Debate-only success (Tier 2). P26 and P51 are the clearest examples: MajVote-Verify achieves 0/3 despite accepting the correct variant in each run, because it also accepts one or more wrong variants (FP contamination disqualifies the problem). Debate's adversarial exchange correctly rejects these wrong variants through cascading counterexamples that emerge from multi-round bilateral challenges. These are exactly the cases where structured disagreement adds value that repetition cannot: the same simple prompt, applied 11 times, consistently misjudges the same plausible wrong argument.

MajVote-only success (Tier 3). P38 and P49 show the opposite: Debate achieves 0/3 despite no FP contamination, because all five agents independently output `reasoning_insufficient` for the *correct* variant, failing the assessment gate. P12 and P20 exhibit the same pattern. These are coordinate-geometry and trigonometric problems where agents correctly identify that they cannot fully re-derive the computation through text-only spot checks, but incorrectly treat this uncertainty as grounds for rejection. The assessment gate, designed to enforce positive evidence, here overfires:

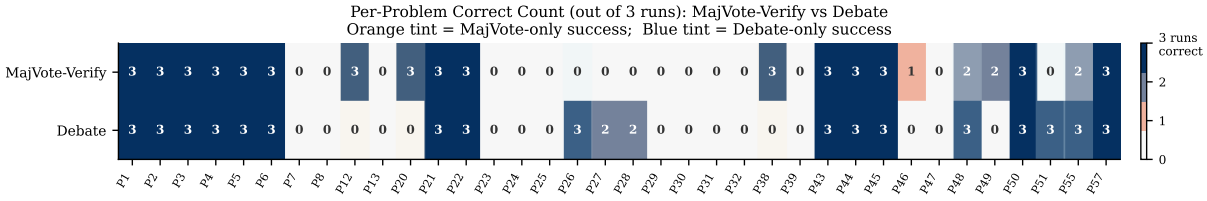

Figure 7: Per-problem correct count (0–3 runs) for MajVote-Verify (left) and Debate (right) on all 58 problems. **Orange** columns: MajVote-Verify succeeds but Debate fails (geometry problems where all agents produce `reasoning_insufficient`, triggering overcautious rejection). **Blue** columns: Debate succeeds but MajVote-Verify fails (problems with plausible wrong variants that require multi-round adversarial argument to reject; majority voting accepts the wrong variant alongside the correct one). **White/light** cells: neither system succeeds consistently (large-derivation problems that hit a raw capability ceiling regardless of pipeline structure).

"I cannot independently verify this computation" becomes a false refusal.

Large-derivation failures (Tier 4). P30–32 (Domino series) and P29 (grid-game) fail both systems in every run; P27–28 are boundary cases discussed below as partial successes. These hit a raw capability ceiling: at `effort="low"`, the model cannot reproduce the derivation independently, so every agent produces `reasoning_insufficient`. Raising `effort` or adding symbolic tools (Han et al., 2025) are the most direct remedies.

Partial success cases. P27–28 show 0/3 for MajVote-Verify but 2/3 for Debate. Even at the boundary of capability, Debate's structured exchange allows recovery in some runs: when one agent finds a checkable sub-step in the correct variant's reasoning, the disagreement digest propagates it to the others, occasionally enabling the correct verdict. MajVote-Verify never recovers on these because its majority vote is too sensitive to FP contamination from correlated errors.

Consistent successes (Tier 1). Problems P1–6, P21–22, P43–45, P50, P57 achieve 3/3 for both systems. These are algebraically or arithmetically checkable problems where any capable verifier—single-judge or debate—can directly validate intermediate steps without re-deriving the full solution. They constitute the "easy" stratum where pipeline investment above simple majority voting provides minimal marginal return.

I Dataset Example and Debate Case Study

I.1 Dataset Entry: Problem 25 (Olympiad-Level Combinatorics)

Problem 25 is representative of the dataset's difficulty tier: the base model *never* solves it from scratch across three attempts, and its wrong variants exploit common misapplications of advanced combinatorial theory.

Problem statement. Consider a 1849×1849 grid of unit squares. Rectangular tiles with sides parallel to the grid lines may be placed on the grid (non-overlapping, integer side lengths). Determine the *minimum* number of tiles required so that exactly one unit square in every row and exactly one unit square in every column remains uncovered.

Correct variant (answer $= 1932$). The key reduction: a set of exactly one uncovered square per row and per column is a permutation matrix; the problem minimizes the number of axis-parallel rectangles tiling the complement. A sharp result gives the minimum as $N + \lceil 2\sqrt{N} \rceil - 3$ for an $N \times N$ board. Since $N = 1849 = 43^2$ and $\lceil 2\sqrt{1849} \rceil = 86$, the answer is $1849 + 86 - 3 = 1932$. The reasoning constructs an explicit 43-block staircase permutation achieving this bound and proves no fewer tiles can suffice.

Wrong variant 1 (answer $= 3696$). The argument claims the minimum is $2n - 2$ for any $n \times n$ board and supplies an explicit construction using $2n - 2$ tiles (tiling the upper triangle and lower triangle separately). The construction is correct; the error is that the claimed minimum is *not* a lower bound—the permutation can be chosen to allow far fewer tiles when n is a perfect square. The reasoning is coherent and the upper-bound construction is valid, making this variant superficially convincing.

Wrong variant 2 (answer $= 2012$). The candidate leverages $1849 = 43 \times 43$ to invoke an alleged "standard block-extremal formula":

$$R(N) = N + 2a + 2b - 9 \quad \text{when } N = ab,$$

applying it with $a = b = 43$ to obtain $1849 +$

$86 + 86 - 9 = 2012$. The derivation is presented as authoritative, with no proof of the formula and no explicit tiling construction, yet the argument reads fluently and correctly identifies the block-factorization structure. This is the hardest variant: the answer is plausible, the approach (43×43 factorization) is mathematically meaningful, and the error—an unverified formula—is not surfaced by surface-level checks.

I.2 Case Study: Assessment Gating Catches a 5–0 Initial Majority

We trace the Debate (Ours) pipeline on Problem 25, Variant 2 (answer = 2012, true label: *wrong*). This case illustrates how the assessment gate and exchange-round counterexamples together prevent a false positive that a majority-vote pipeline would have produced.

Initial deliberation. All five agents independently evaluate the candidate's reasoning. Every agent flags the same structural weakness—the formula $R(N) = N + 2a + 2b - 9$ is asserted without proof—but cannot supply a concrete refutation. Consequently, all five vote `support` (vote tally: 5–0), yet all five assign `assessment_type = reasoning_insufficient_but_answer_not_refuted`.

Representative initial swing issues (paraphrased from agent JSON):

- Agent 1 (Formalist Verifier): "No rigorous lower bound and no explicit construction; the claimed formula $R(N) = N + 2a + 2b - 9$ is unsupported and appears dubious under small-case sanity checks."
- Agent 2 (Independent Resolver): "Whether there is a genuine known theorem giving the exact minimum for the complement of a permutation matrix when N is a perfect square; the candidate's formula reference is unattributed."

A naive majority-vote pipeline (**Debate-NoGate**) would *accept* this variant here: 5–0 satisfies any supermajority threshold. The assessment gate rejects it: fewer than three agents hold `answer_supported`, so the pipeline moves to the exchange round.

Scheduler broadcast (after initial deliberation). The scheduler detects all five as "weak-support" agents and issues the following directive (verbatim excerpt):

> *"assessment mix: `reasoning_insufficient=` 5 — weak-support agents (must provide a concrete check or switch to oppose): 1, 2, 3, 4, 5."*

All five are required to either find a concrete check confirming the formula or switch to oppose.

Exchange round (Round 1 → Round 2). Each agent independently performs a small-N sanity check on the formula, arriving at distinct counterexamples:

- **Agent 1** (Formalist Verifier): $N = 2 = 1 \times 2 \Rightarrow R(2) = 2 + 2 + 4 - 9 = -1$, impossible.
- **Agent 2** (Independent Resolver): $N = 4 = 2 \times 2 \Rightarrow R(4) = 4 + 4 + 4 - 9 = 3$, but the diagonal-hole complement of a 4×4 board requires ≥ 4 rectangles (explicit construction).
- **Agent 3** (Theorem Auditor): $N = 1 = 1 \times 1 \Rightarrow R(1) = 1 + 2 + 2 - 9 = -4$, impossible.
- **Agent 4** (Optimization Skeptic): $N = 3$ explicit tiling analysis shows formula unreliable; no valid lower bound in candidate solution.
- **Agent 5** (Pragmatic Cross-Checker): $N = 4$ structural argument: in the diagonal-uncovered arrangement, any valid rectangle must lie wholly on one side of the diagonal, forcing ≥ 4 tiles.

Every agent independently falsifies the formula via a different small case. All five flip to `oppose`. Final vote: **0–5** (`wrong`). **Correct prediction.**

Why this case matters. Three distinct mechanisms are active simultaneously: *(i)* **Assessment gating**: the initial 5–0 majority is not accepted because no agent produced positive confirming evidence; Debate-NoGate would have output a false positive here. *(ii)* **Exchange-driven falsification**: the scheduler correctly identifies weak-support agents and requires concrete checks, triggering the falsification round that was never initiated in the initial deliberation. *(iii)* **Role heterogeneity**: the five agents approach the small-N sanity check differently ($N = 1$, $N = 2$, $N = 3$, $N = 4$ algebraic, $N = 4$ structural), producing five independently valid counterexamples—any one of which is sufficient, but their diversity provides mutual reinforcement and reduces the chance that all five make a correlated error on the same check.