International Journal of Surgery and Anesthesia Research

PRIME OPEN ACCESS

Volume 1, Issue 1

Research article

Date of Submission: 20 May, 2025
Date of Acceptance: 11 June, 2025
Date of Publication: 25 June, 2025

Number of Players, Zero: Using Multi-Agent Collaboration to Evaluate Unsolvable Collaborative Items

Yu Wang* and Yoav Bergner

Department of Administration, Leadership & Technology, New York University Steinhardt School of Culture, Education & Human Development, USA

***Corresponding Author:**
Yu Wang, Department of Administration, Leadership & Technology, New York University Steinhardt School of Culture, Education & Human Development, USA.

Citation: Wang, Y., Bergner, Y. (2025). Number of Players, Zero: Using Multi-Agent Collaboration to Evaluate Unsolvable Collaborative Items. *Int J Surg Anesth Res*, 1(1), 01-06.

Abstract

While the importance of collaborative problem-solving (CPS) is widely acknowledged, developing and evaluating CPS items remains challenging. This study extends to multi-agent systems (MASs) prior work on CPS item templates and the use of large language models for evaluation of collaborative interdependence in math problems designed for two solvers working together. Specifically, we test the hypothesis that MASs more accurately handle evaluation of flawed items that are unsolvable even if both students collaborate with respect to the information and/or answer affordances provided to them.

Introduction

Collaborative problem-solving (CPS) involves individuals working together to solve complex problems through effective communication, shared reasoning, and negotiation. It is increasingly recognized as vital for both educational success and workforce readiness [1-3]. CPS learning and assessment tasks are designed to cultivate and/or assess relevant sub-skills by presenting challenges that require participants to integrate diverse perspectives, share information, and build upon each other's contributions to reach a solution [4,5]. Designing and evaluating CPS assessment tasks (or test items) is challenging due to their inherent complexity [6]. Whether AI automation can help address the challenges of generating, evaluating and ultimately providing solution feedback to CPS items remains an active frontier of research.

Large language models (LLMs), which have shown promise in various domains, represent one potential avenue for automating CPS item evaluation. LLMs have demonstrated impressive performance in mathematical problem solving with "standard" item types; however, they may fall short when it comes to the interdependent nature of CPS [7,8]. The low evaluation accuracy in the latter case, even with enhancements like few-shot learning or chain-of-thought prompting, underscores the need for more sophisticated approaches to effectively tackle CPS tasks. In this study, we extend prior work along these lines to multi-agent systems (MASs) of LLMs. The study was motivated by a hypothesis that, by simulating the solution process between students, MASs could identify poorly constructed items of a particular variety: those that remain unsolvable even when students pool their resources. Evaluating intentionally unsolvable CPS tasks might initially seem odd at first, but it helps to reveal what AI language models can and cannot do, especially when it comes to spotting hidden flaws in (automatically generated) items that appear superficially correct but contain hidden logical contradictions.

Background

Several studies have shown that LLMs struggle with complex reasoning and logical problem-solving. For instance, early transformer models failed to maintain consistency and accuracy across multiple steps in tasks from the MATH dataset [9]. More recent studies with state-of-the-art models revealed systematic decline in performance as task complexity increases

as well as token bias effects, where performance on functionally identical math problems degrades considerably when superficial details are altered, such as changing horses to rabbits [10,11]. While LLMs are often capable of generating correct answers, they do not display deep understanding required for complex, iterative reasoning.

Single-agent systems fall short partly because they rely heavily on static knowledge and predefined prompts, which are insufficient for tasks requiring adaptive problem-solving and interdependent reasoning. However, LLMs have also shown potential for interfacing effectively with other agents or systems, suggesting that collaborative frameworks could unlock their capabilities [12]. Multiagent systems allow for distributed reasoning, role differentiation, and interaction [13]. These capabilities make MASs particularly well-suited for evaluating CPS tasks that involve interdependence and communication [14]. An innovative feature of this study is the application of MASs to unsolvable CPS tasks. Trying to solve these tasks would reveal irreconcilable conflicts or missing information through interaction. Single-agent AI systems seem to lack the ability to anticipate these outcomes, whereas MASs provide a framework for modeling these processes [15,16].

Collaborative tasks can vary in their complexity, conceptual vs. procedural requirements, and other affordances [17,18]. In this work, we concentrate on a rather narrow set of two-person collaborative math tasks that are designed to be solved quickly. Three template examples of such tasks were described in, and two of those are relevant to this study: jigsaw and joint-construction items [19]. A jigsaw item is one in which given information that is necessary for problem solution is distributed between the students, such that each has complementary pieces. For example, one student may be given the cross-sectional area of a cylinder, while another has its height, and these must be integrated to determine the surface area. A joint-construction item is one in which the given information may be the same but the solution requires different inputs from each student. For example, one student might need to enter the slope of a line which satisfies a criterion, while their partner enters its y-intercept. Both jigsaw and joint-construction tasks can be included in a simulated collaboration carried out by a MAS. As described below, the item set used for the study includes both solvable and unsolvable versions of these items. The third template type, information-request items, would require a more complex MAS architecture and is not included here.

The second variation in the item set, aside from jigsaw and joint-construction templates concerns solvability. If each of the students in the pair have insufficient information separately but sufficient information when pooled, a jigsaw item is considered solvable. On the other hand, if even when pooled, the information is still insufficient or even self-contradicting, the item may be unsolvable. For joint-construction items, unsolvability implies that each student may be able to provide a component that appears to solve some constraints of the problem, but both students cannot do so at the same time. For example, they would not both be able to provide odd-numbered values for quantities in a word problem which are expected to multiply to an even number. These unsolvable tasks are intentionally designed to test the agents' ability to communicate effectively, identify gaps or conflicts in the information provided, and collaboratively determine that no solution exists.

Commonly, a MAS assigns distinct roles to agents, enabling them to replicate realistic scenarios where success depends on communication and shared reasoning. For example, in tasks involving complementary constraints, one agent's input is essential for another to complete their reasoning, reflecting the interdependent nature of collaboration [20]. A number of software platforms exist for embedding LLMs in multi-agent workflows; this study implemented MASs using CrewAI [21,22].

In sum, MASs bring several advantages to CPS item evaluation: First, MASs excel at navigating tasks involving incomplete, conflicting, or ambiguous information. Agents can iteratively refine their reasoning and collaboratively assess task solvability [16]. Second, by assigning distinct but complementary roles, MASs replicate scenarios where collaboration is necessary for success, allowing for realistic simulations of teamwork [15]. Frameworks like CrewAI offer a scalable framework for evaluating CPS tasks, capturing via simulation intricate details of interaction and reasoning that single-agent systems often overlook [23]. The present study serves as a proof of concept, demonstrating that MASs can outperform single-agent systems in evaluating unsolvable CPS tasks.

Methods

This study employed a multi-agent system (MAS) in CrewAI to evaluate collaborative problem-solving (CPS) tasks. We focused on two types of items—jigsaw and joint-construction tasks—which were designed to be either solvable through collaboration or unsolvable even when two agents work together. For this purpose, we developed 15 solvable (7 jigsaw, 8 joint-construction) and 26 unsolvable items (13 each of jigsaw and joint-construction). Some of the math items have fundamentally different structures, while others represent superficial variations on a theme, for example, matching the number of buses to schoolchildren or the number of apple crates to apples. The quantity of unsolvable items was deliberately larger because that is where we hypothesized to find different performance between single- and multi-agent systems. Both single- and multi-agent models were given a CPS item and prompted to evaluate the item as pass/fail on the main criteria of interdependence and solvability. In other words, items were supposed to be classified as "pass" if it could not be solved by an individual alone but could be solved via collaboration. A problem that could be solved individually or remained unsolvable, even though collaboration, was to be classified as "fail".

Performance of the MASs was compared to two single-agent conditions. The conditions were as follows:
Single-agent evaluation without simulation prompt: In this condition, item evaluations were generated using an optimized, zero-shot prompt developed in prior research. The single LLM (GPT-4o) classified each item (pass/fail), based on consideration of its individual and collaborative solvability.

Single-agent evaluation with simulation prompt: In addition to the zero-shot prompt, the single-agent LLM was prompted to simulate a collaboration between two students and to evaluate the items based on this simulation. The prompt included the same instructions given to student agents in the MAS.

The multi-agent evaluation: In this condition, a MAS was instantiated with four distinct agent roles:

- **Student A and Student B (identical):** Simulated learners who engage in collaborative dialogue to address complementary or conflicting task components.
- **Intermediary:** A mediator responsible for passing messages between the two studentagents, recording their exchanges, and determining when the collaboration had concluded.
- **Evaluator:** An agent that receives the collaboration transcript from the intermediary and is tasked with assessing whether the item receives a pass or fail evaluation for CPS use.

The multi-agent system is shown schematically in Figure 1. The intermediary (Agent 3) may seem superfluous. However, we found that when two agents (representing Students A and B) are allowed to dialogue directly, they may enter into an endless loop of repetitive agreement. The intermediary agent resolves this issue by only passing messages when the exchange is moving constructively forward, and it terminates the dialogue when it becomes unproductive. It makes sure that the collaborative transcript will be summarized once the student agents reach an agreement or identify a conflict.

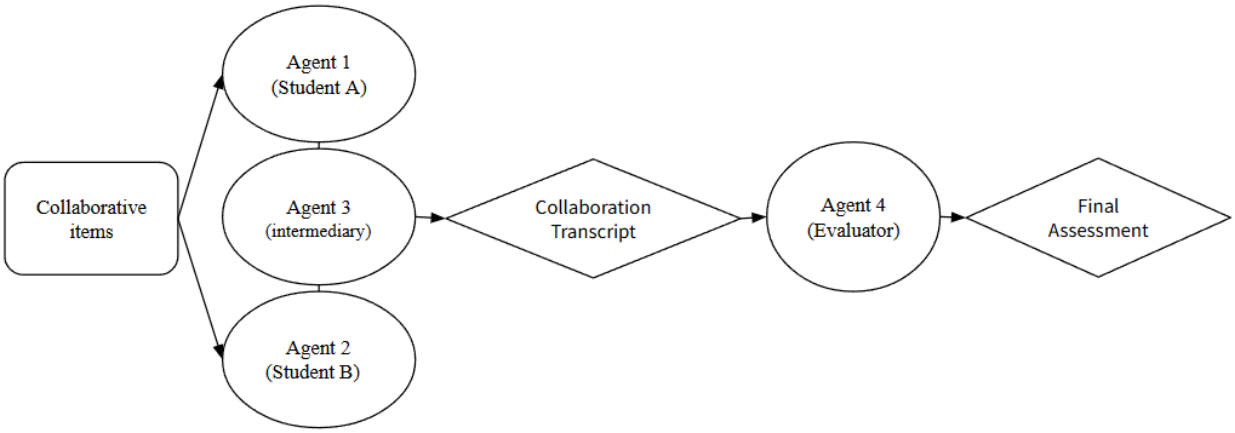

Figure 1: The Multi-Agent System Framework.

Collaborative items are distributed between Student A and Student B (Agent 1 and Agent 2); the intermediary (Agent 3) documents the complete collaboration transcript and forwards it to the evaluator (Agent 4); the evaluator provides the final assessment.

Both single- and multi-agent language models are known to have considerable output variability; agents may generate quite different outcomes across separate runs, even when the input remains the same. This variability is amplified further in the multi-round interactions of MASs, where one agent's output affects subsequent reasoning and responses. To account for this variability, each evaluation task was run five times, with results aggregated to provide a clearer comparison across conditions.

Results

Table 1 presents the aggregated findings from five evaluation runs across the three experimental conditions: the single-agent with/without simulation prompt and the MAS. Below, a focused analysis is presented for the 13 questions with the lowest classification accuracy in the single-agent condition, providing insights into the performance differences among the approaches.

	Solvable (Pass) (*N* = 15)	**Unsolvable (Fail) (*N* = 26)**	**Overall (*N* = 41)**
Single agent w/o sim prompt	0.93 (0.07)	0.52 (0.03)	0.67 (0.02)
Single agent with sim prompt	0.95 (0.03)	0.48 (0.04)	0.65 (0.03)
Multi-agent system	0.83 (0.10)	0.88 (0.08)	0.86 (0.07)

Table 1: A Comparison of Classification Accuracy Between Conditions. Mean (Standard Deviation) Values are Shown Averaged Across Items and Across Five Replication Runs

For solvable items, the MAS achieved a slightly lower accuracy than the two single-agent models, but none of the pairwise differences in the Pass column of Table 1 are statistically significant at a typical level (alpha = 0.05). In contrast, the MAS excelled on unsolvable questions (middle column of Table 1), achieving an accuracy of 0.88, where both single-agent conditions did no better than chance (50% for a binary classification task). The MAS's ability to dynamically recognize and articulate irreconcilable constraints likely contributed to its superior performance on this challenging subset of questions, underscoring the strength of multi-agent interaction for tasks requiring reasoning and collaboration. The high accuracy of the single-agent LLMs in solvable tasks clearly did not generalize; the models may too easily pass items that "look" collaborative.

How consistent are the single- and multi-agent models? Looking at the classification accuracy for the 13 most challenging items across the five replication runs revealed distinctive distributions. All of these items, shown in Table 2, came from the unsolvable group. The single-agent models seldom or never classified them correctly. The "base" prompt achieved 0/5 accuracy for 7 items and 1/5 accuracy for 6 more. With the addition of the simulation prompt, the single-agent model performed similarly. In stark contrast, the MAS demonstrated relatively consistent, but not perfect, improvement on these challenging questions (5/5 on 7 items, 4/5 on four more, etc.). In this experiment, single-agent LLMs were sometimes lucky and MASs were sometimes unlucky.

	0%	20%	40%	60%	80%	100%
Single agent w/o sim prompt	7	6	0	0	0	0
Single agent with sim prompt	8	4	1	0	0	0
Multi-agent system	0	0	1	1	4	7

Table 2: Frequency Table of the Accurate Classification of 13 Challenging CPS Items Over Five Runs. 0%, 20%, ... 100% Corresponds to Correctness on 0, 1, ... 5 Runs. Rows Correspond to Conditions.

Discussion and Conclusions

The results reveal important distinctions in how MAS and single-agent models process CPS tasks. The MAS's superior performance on unsolvable questions points to a distinct advantage in recognizing irreconcilable conflicts. The advantage can be attributed to distributed reasoning and iterative dialogue, which explicitly articulates conflicts and contradictions that remain implicit in single-agent approaches [15,16]. As shown in our results, the MAS's iterative interactions consistently identified and resolved constraints through negotiation, directly addressing the limitations inherent in static, prompt-based evaluations by single-agent models. Although single-agent models had appeared to perform competently on solvable items, their poor record on unsolvable variants suggests that these models may have simply passed most items without genuinely recognizing the conditions for a successful CPS task.

Despite its strengths, the MAS also exhibited variability in its outcomes. On the same item, agents sometimes identified an irreconcilable conflict, concluding that the task was unsolvable. In other runs, however, the agents failed to identify the conflict or reached different conclusions. This variability points to a limitation in the current approach—inconsistent MAS outputs during dynamic, iterative collaboration—and the need for further refinements to enhance output consistency. At the same time, it also illustrates the intrinsic complexity and stochasticity involved in replicating human collaborative interactions.

The demonstrated ability of MASs to correctly evaluate unsolvable CPS tasks suggests promising avenues for future research in collaborative item evaluation. Future work might focus on extending the scope of CPS tasks to include more diverse scenarios. Simply, one may incorporate not only unsolvable questions but also tasks that, inappropriately, can be solved by one of the students independently. More ambitious collaborative task designs for evaluating might include cases where agents make choices that actually change the state of the problem space, for example selecting certain paths to solution. Enhancing consistency and reducing variability in MAS classification processes may also be explored. In the current study, we averaged over repeated tests to mitigate the effect of discrepancies. However, it may be possible to learn iterative refinements to the agent prompts that would make the simulation-based evaluation more consistent. Ultimately, the most interesting use-cases for MASs may emerge from studying the reasoning failures of single-agent AI models [24].

References

1. Johnson, D. W., & Johnson, R. T. (2009). An educational psychology success story: Social interdependence theory and cooperative learning. Educational Researcher, 38(5), 365-379.
2. OECD. (2017). PISA 2015 results (Volume V): Collaborative problem solving. OECD Publishing.
3. Rios, J. A., Ling, G., Pugh, R., Becker, D., & Bacall, A. (2020). Identifying critical 21st-century skills for workplace success: A content analysis of job advertisements. Educational Researcher, 49(2), 80-89.
4. Griffin, P., & Care, E. (Eds.). (2014). Assessment and teaching of 21st century skills: Methods and approach. Springer.
5. Graesser, A. C., Fiore, S. M., Greiff, S., Andrews-Todd, J., Foltz, P. W., & Hesse, F. W. (2018). Advancing the science of collaborative problem solving. Psychological Science in the Public Interest, 19(2), 59–92.

6. Graesser, A. C., Foltz, P. W., Rosen, Y., Shaffer, D. W., & Forsyth, C. (2018). Challenges of assessing collaborative problem-solving. Assessment and Teaching of 21st Century Skills: Research and Applications, 75–91.
7. Zhang, R., Jiang, D., Zhang, Y., Lin, H., Guo, Z., Qiu, P., ... & Li, H. (2024, September). Mathverse: Does your multi-modal llm truly see the diagrams in visual math problems? In European Conference on Computer Vision (pp. 169-186). Cham: Springer Nature Switzerland.
8. Anghel, E., Wang, Y., Gopalakrishnan, M., Mansukhani, P., & Bergner, Y. (2024). Can LLMs evaluate items measuring problem-solving? CEUR Workshop Proceedings, 3772. CEUR-WS.
9. Hendrycks, D., Burns, C., Kadavath, S., Arora, A., Basart, S., Tang, E., ... & Steinhardt, J. (2021). Measuring Mathematical Problem Solving with the MATH Dataset. In Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2).
10. Mirzadeh, I., Alizadeh, K., Shahrokhi, H., Tuzel, O., Bengio, S., & Farajtabar, M. (2024). Gsm-symbolic: Understanding the limitations of mathematical reasoning in large language models. arXiv preprint arXiv:2410.05229.
11. Jiang, B., Xie, Y., Hao, Z., Wang, X., Mallick, T., Su, W. J., ... & Roth, D. (2024). A Peek into token bias: large language models are not yet genuine reasoners. arXiv preprint arXiv:2406.11050.
12. Shinn, N., Cassano, F., Gopinath, A., Narasimhan, K., & Yao, S. (2023). Reflexion: Language agents with verbal reinforcement learning. Advances in Neural Information Processing Systems, 36, 8634-8652.
13. Wooldridge, M. (2009). An Introduction to Multiagent Systems. John Wiley & Sons.
14. Hong, S., Zheng, X., Chen, J., Cheng, Y., Wang, J., Zhang, C., ... & Wu, C. (2023). Metagpt: Meta programming for multi-agent collaborative framework. arXiv preprint arXiv:2308.00352, 3(4), 6.
15. Sreedhar, K., & Chilton, L. (2024). Simulating human strategic behavior: Comparing single and multi-agent LLMs. arXiv preprint arXiv:2402.08189.
16. Talebirad, Y., & Nadiri, A. (2023). Multi-agent collaboration: Harnessing the power of intelligent LLM agents. arXiv preprint arXiv:2306.03314.
17. Mullins, D., Rummel, N., & Spada, H. (2011). Are two heads always better than one? Differential effects of collaboration on students' computer-supported learning in mathematics. International Journal of Computer-Supported Collaborative Learning, 6(3), 421–443.
18. Janssen, J., & Kirschner, P. A. (2020). Applying collaborative cognitive load theory to computer-supported collaborative learning: Towards a research agenda. Educational Technology Research and Development, 68(2), 783-805.
19. Bergner, Y., & Wang, Y. (2023). Mathchops: A platform for developing collaborative higher order problem solving in mathematics. Proceedings of the 16th International Conference on Computer-Supported Collaborative Learning (CSCL 2023) (pp. 51–58). International Society of the Learning Sciences.
20. Cardoso, R. C., & Ferrando, A. (2021). A review of agent-based programming for multi-agent systems. Computers, 10(2), 16.
21. Crawford, N., Duffy, E. B., Evazzade, I., Foehr, T., Robbins, G., Saha, D. K., ... & Ziolkowski, M. (2024). BMW Agents-A Framework for Task Automation Through Multi-Agent Collaboration. arXiv preprint arXiv:2406.20041.
22. João, M. (2024). CrewAI.
23. Qian, C., Xie, Z., Wang, Y., Liu, W., Dang, Y., Du, Z., ... & Sun, M. (2024). Scaling large-language-model- based multi-agent collaboration. arXiv preprint arXiv:2406.07155.
24. Canese, L., Cardarilli, G. C., Di Nunzio, L., Fazzolari, R., Giardino, D., Re, M., & Spanò, S. (2021). Multi - agent reinforcement learning: A review of challenges and applications. Applied Sciences, 11(11), 4948.

Appendix: Single-agent with/without simulation pre-prompt

Below is the prompt used for a single LLM to evaluate a collaborative question. Base instructions shown in sans-serif font; additional instructions for simulation shown in serif font.

You will be asked to evaluate one educational exercise for math students working in pairs. The exercise will be presented to you in two parts, the exercise version shown only to Student A (called Version A) and the exercise version shown only to Student B (Version B). Students A and B are assigned to be partners. Student A cannot see Version B, and Student B cannot see Version A, but they can communicate via chat. The exercise should require both Student A and Student B to submit some answers in an answer field or fields. In the exercise, the students may need to share information that is only available to one of them, communicate to choose what information they need to ask for, or decide which one will answer which part of the problem in order to solve it.

Your criterion for evaluation of the exercise is whether or not the exercise is solvable if and only if it requires both Student A and Student B to collaborate to solve the problem. If so, indicate pass. It is not acceptable if either Student A or Student B can work separately, independently, and without communicating and still get the correct answer, or in cases where the problem cannot be solved even when working together. In such cases, indicate fail. It is not necessary for you to solve the problem. However, you may describe the solution process in explaining your reasons for your evaluation.

You should simulate the collaboration between two students and then evaluate the collaborative question based on the simulated transcript. Here is the instruction for student A and student B:

Using ONLY the information provided to engage in a discussion with your partner about the question. In your final response, refrain from just summarizing your thoughts or limiting it to a single sentence. Ensure that your answer comprehensively includes both your cognitive reasoning and either a specific question or a conclusive statement. Identify and explicitly state any gaps in the information that prevent a complete solution, but avoid making assumptions or hypotheses. Focus on addressing the primary question, and avoid introducing additional considerations or hypothetical constraints unless they are explicitly included in your instructions. If you encounter conflicting information or interpretations from your partner, evaluate the conflict carefully rather than immediately conceding or agreeing. Note that successful collaboration is not required if the conditions for both students cannot be simultaneously satisfied. After agreeing on a conclusion with your partner, double-check that every part of your solution aligns with the constraints contained in the information provided to you, such as numerical requirements, ranges, or other conditions. Moreover, evaluate whether you could have independently solved the problem using only the information you initially received. Clearly state this assessment as part of your final response.

When providing your evaluation, please format it as follows: Verdict: [pass or fail] Reason: [explanation for verdict]
The following is the exercise you need to evaluate: