ESANN 2026 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 22-24 April 2026, i6doc.com publ., ISBN 9782875870964. Available from http://www.i6doc.com/en/.

AsymPuzl: A minimal puzzle testbed for LLM-based two agent communication with information asymmetry

Xavier Cadet, Edward Koh and Peter Chin*

Dartmouth College - Thayer School of Engineering
Hanover, New Hampshire - USA

Abstract. Large Language Model (LLM) agents are increasingly studied in multi-turn, multi-agent scenarios, yet most existing setups emphasize open-ended roleplay rather than controlled evaluation. We introduce AsymPuzl, a minimal but expressive two-agent puzzle environment isolating communication under information asymmetry. Each agent observes complementary but incomplete views of a puzzle and must exchange messages to solve it. Using contemporary LLMs, we show that (i) models such as GPT-5 and Claude-4.0 reliably solve puzzles of different sizes by sharing complete information in few turns, (ii) feedback design in multi-agent LLM systems is non-trivial, more information can degrade performance.

1 Introduction

Creating intelligent and autonomous agents to tackle real-world problems or assist humans has been a key goal of Artificial Intelligence [1]. Handling real-world problems often requires synthesizing information from different sources, reasoning through content, and effectively communicating the results. Recent advances in Large Language Models (LLMs) [2, 3, 4], through their capacity for complex reasoning and content synthesis, have positioned them as promising "brains" for autonomous agents [5, 6].

Puzzle-based benchmarks have become popular for evaluating LLM reasoning, spanning rule-based (Sudoku, Crosswords, Minesweeper) and rule-less puzzles (Riddles, common-sense) [7]. Nonetheless, complex real-world tasks often require cooperation and specialization, motivating research into multi-agent LLM systems. Prior work on single-agent puzzle solving found that errors often stem from poor instruction following [8], and that scaling model size yields limited improvement as puzzle complexity grows [9]. Multi-agent LLM collaboration has been studied for shared decision-making [1], role-play with common goals [10], and large-scale social QA under information asymmetry [11].

In this work, we focus on multi-agent coordination under information asymmetry: settings where no single agent holds complete information and communication is necessary for success. We introduce AsymPuzl, a minimal yet expressive two-agent puzzle environment with fine-grained control over task difficulty and feedback granularity.

*This research was funded by the Defense Advanced Research Projects Agency (DARPA), under contract W912CG23C0031.

ESANN 2026 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 22-24 April 2026, i6doc.com publ., ISBN 9782875870964. Available from http://www.i6doc.com/en/.

Our **contributions** are 1) the AsymPuzl test bed for two-agent puzzle solving under information asymmetry with controlled difficulty level, 2) an empirical analysis of feedback granularity on multi-agent collaboration, showing that even in this minimal setting feedback needs to be carefully designed.

2 The AsymPuzl Environment

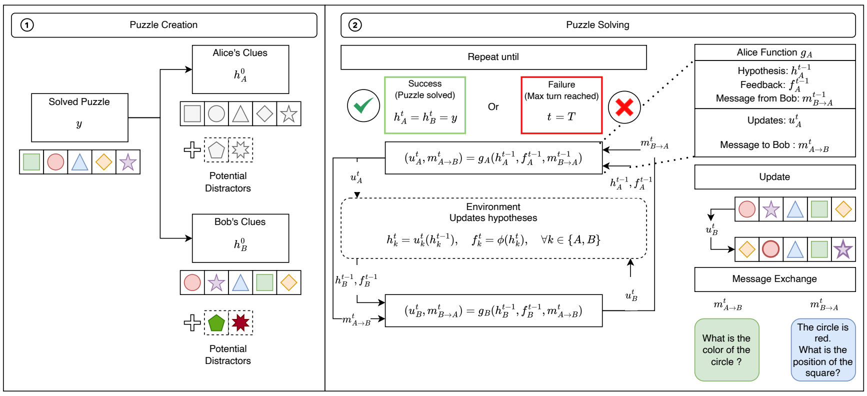

Fig. 1: Overview of the puzzle creation (1) and puzzle solving (2). (1) Given a solved puzzle we generate each agents' clues, (2) the agents generate updates to modify their hypothesis and send messages to each other until the puzzle is solved or they run out of turns.

Ground Truth and Information Partition. A puzzle instance is a sequence $y = [(s_1, c_1), \ldots, (s_N, c_N)]$ where $s_i \in \mathcal{S}$ (shapes) and $c_i \in \mathcal{C}$ (colors) with $|\mathcal{S}| = |\mathcal{C}| = N$, such that each shape-color pair appears exactly once. Agent A (Alice) receives clues $h_A^0 = [s_1, \ldots, s_N]$: correct positions and shapes, unknown colors. Agent B (Bob) receives clues $h_B^0 = \pi(\{(s_i, c_i)\}_{i=1}^N)$: correct shape-color mappings, shuffled positions (where π is a random permutation). We provide the code at https://github.com/xcadet/asympuzl.

Game Dynamics. At turn $t \in [1, T]$, each agent $k \in \{A, B\}$ receives an instruction about its task i_k, its original clues h_k^0, r_k^t the last message exchange, a working hypothesis h_k^{t-1} that represents its current guess of the puzzle, feedback about its previous hypothesis f_k^t, and the most recent message $m_{j\to k}^{t^*}$ from the other agent (j), where $t^* = t$ if $k = B$, and $t^* = t - 1$ otherwise. To simplify the notation we omit i_k, h_k^0 as they are constant across turns and r_k^t can be seen as an internal component of the agent keeping track of the last messages of each agent. The agent is then tasked to output $m_{k\to j}^t$ - its outgoing message and u_k^t

ESANN 2026 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 22-24 April 2026, i6doc.com publ., ISBN 9782875870964. Available from http://www.i6doc.com/en/.

a set of updates to apply to its working hypothesis:

$$g_k : (h_k^{t-1}, f_k^{t-1}, m_{-k \to k}^{t^*}) \mapsto (m_k^t, u_k^t)$$

Termination. The game ends when $h_A^{(t)} = h_B^{(t)} = y$ (Success, the puzzle is solved) or $t = T$ (Failure due to timeout).

Feedback Modes. We vary feedback granularity: *None* (no signal), *Own* (whether its hypothesis matches y), *Own-detailed* (exactly which positions are incorrect), *Joint* (whether the puzzle is globally solved), *Both* (whether either side of the puzzle is solved), *Both-detailed* (both hypotheses' error positions).

Distractors. To increase complexity we inject *distractors* in the agents' clues such that their clues are of size $N + D$ where D is the number of distractors. The agents are given the true size of the puzzle N as part of their prompt. This requires that agents identify the distractors and focus on the pieces relevant to the puzzle.

3 Experiments

Unless otherwise specified, experiments are conducted on puzzles of size $N = 5$, no distractors $D = 0$ and with individual detailed feedback, as this feedback mode most consistently leads to the highest performance.

For each of our experiments, we set the maximum number of turns $T = 3 \times N$, giving a significant margin for errors and corrections. For each puzzle $p \in \{1, \ldots, P\}$ with P the number of puzzles, let $\tau_p \in \{1, \ldots, T, \infty\}$ be the turn at which the puzzle is solved, with ∞ used if the puzzle isn't solved within T turns, and let $\Gamma = [\tau_1, \cdots, \tau_P]^T$ be the vector of puzzle completions.

Models We evaluated the AsymPuzl environment on LLMs from various providers: OpenAI (GPT-3.5-turbo, GPT-4o [3], GPT-5, OSS-120B, OSS-20B [4]), Meta (Llama 3 8B [12] and Llama 3.3 70B), and Anthropic (Claude-3.5, Claude-4.0), capturing a range of reasoning abilities, response tendencies, and costs.

Completion rate. Given Γ we can define the completion rate for a given pair of agents as:

$$C = \frac{1}{P} \sum_{\tau \in \Gamma} \mathbb{I}_{[\tau \leq T]}$$

3.1 Results

Providing detailed feedback about each agent's own view consistently improved completion percentage. We organize our multiple feedback modes into two groups: 1) **Own feedback**: the agents receive information about their own view only 2) **Joint feedback**: the agents receive information about both views. Detailed

ESANN 2026 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 22-24 April 2026, i6doc.com publ., ISBN 9782875870964. Available from http://www.i6doc.com/en/.

feedback about each agent's own view improved completion percentage for most model (7/9). For example, the GPT-4o model performance rose from 40.0% to 60.3% when given detailed feedback about its current working hypothesis (Table 1). Providing detailed information about the other agent's working hypothesis, however, sometimes hurt performance. We hypothesize that this comes from agents receiving error for hypotheses they cannot observe, leading to ungrounded corrections.

Table 1: (Higher is better) Percentage of 5-piece puzzles solved across 30 seeds. Providing feedback increases the completion percentage, while additionally providing detailed information about the other agent's working hypothesis can hurt performance.

Completion Percentage with 5-pieces (% ↑)						
	Own feedback			**Joint feedback**		
Feedback mode Model	No feedback	Own	Own detailed	Joint	Both	Both detailed
GPT-5	100.0	100.0	100.0	100.0	100.0	100.0
Claude 4.0	100.0	100.0	100.0	100.0	100.0	100.0
Claude 3.5	100.0	100.0	100.0	100.0	100.0	76.7
OSS-120B	96.7	93.3	100.0	83.3	100.0	96.7
Llama 3.3 70B	96.7	93.3	100.0	56.7	73.3	43.3
GPT-4o	40.0	60.0	63.3	26.7	40.0	30.0
OSS-20B	20.0	33.3	56.7	10.0	23.3	23.3
GPT-3.5-Turbo	0.0	0.0	0.0	0.0	0.0	0.0
Llama 3 8B	0.0	0.0	0.0	0.0	0.0	0.0

Larger puzzles are more challenging than smaller ones as weaker models adopt shorter messages. We evaluated the agents on increasing puzzle sizes $N = \{3, 5, 10, 20\}$. GPT-5 solved the puzzles within 2 turns (29 out of 30 seeds for $N = 20$) by exploiting the unlimited message size and sharing its entire view regardless of puzzle size. Both Llama 3.3 70B and Claude 3.5 began failing puzzles starting N=10, as they diverged more often from this strategy. While they both generally also sent their full views for $N = 3, N = 5$, at larger puzzle sizes they began requesting and sending information for only a portion of the puzzle each message. For example, at $N = 20$ using Claude 3.5, one of Alice's first turn messages asks Bob "[...] Could you confirm the color mappings for the first few shapes: octagon, heptagon, trapezoid, nonagon?". These extended exchanges result in nonsensical messages mixing incorrect information into questions, and as errors stack, agents begin partially ignoring each other's messages and requests.

Adding distractor pieces decreased completion percentage. We further analyzed agents on puzzles where we introduce D distractors to one of the agent views. We only introduce distractors in one agent's view, so that the other agent can serve as the anchor. When applied to Alice, we randomly insert the distractors in the clues but maintain the relative order of valid pieces. Claude 3.5, OSS-120B

ESANN 2026 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 22-24 April 2026, i6doc.com publ., ISBN 9782875870964. Available from http://www.i6doc.com/en/.

Table 2: Completion percentage over 30 seeds with different numbers of pieces in the puzzle N and Own detailed feedback. GPT-5, Claude 4.0, and OSS-120B consistently achieve a high completion rate across various puzzle sizes.

	Completion with N-pieces and Own detailed feedback (% ↑)			
Model	$N = 3$	$N = 5$	$N = 10$	$N = 20$
GPT-5	100.0	100.0	100.0	100.0
Claude 4.0	96.7	100.0	100.0	100.0
Claude 3.5	100.0	100.0	96.7	80.0
OSS-120B	100.0	100.0	100.0	100.0
Llama 3.3 70B	100.0	100.0	63.3	0.0

and Llama 3.3 70B models saw their performance decrease when unnecessary information was added in either agent's clues (Table 3), though the observed impact was stronger when they were added to Alice's clues compared to Bob's. Looking at failed examples, we conclude this is because Alice's task requires positional tracking, which is disrupted by inserted distractors. Bob's task requires relational matching (the circle is red), which is invariant to list order. When the distractors are in Alice's clues both GPT-5 and Claude 4.0 usually explicitly attempt to identify which pieces are wrong, for instance GPT-5's Alice asks "Bob, please list the color for each shape below, or says "invalid" if the shape is not part of the puzzle: heptagon, pentagon, circle, hexagon, rectangle, square, triangle, octagon". Meanwhile, Claude 3.5 and Llama 3.3 70B in most cases do not attempt to identify early on the invalid pieces, and assume that their clues are correct.

Table 3: (Higher is better) Percentage of 5-pieces puzzles solved over 30 seeds with different number of distractors pieces D in their clues.

Completion Percentage with 5-pieces and Own detailed feedback (% ↑)							
Ambiguity mode	No extra	Distractors in Alice's view			Distractors in Bob's view		
D	0	3	5	10	3	5	10
GPT-5	100.0	100.0	100.0	100.0	100.0	100.0	100.0
Claude 4.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0
Claude 3.5	100.0	23.3	26.7	3.3	100.0	100.0	100.0
OSS-120B	100.0	86.7	60.0	63.3	86.7	93.3	93.3
Llama 3.3 70B	100.0	33.3	20.0	13.3	96.7	100.0	90.0

Models with the highest completion percentage tend to be more verbose both in their messages exchange and outputs. Models such as GPT-5 and Claude 4.0 tend to use more tokens per messages. These models tend to ask and send information about most of the pieces in their messages and solve the puzzle in fewer messages. Models such as LLama 3 8B and GPT-3.5-Turbo tend to ask and send information about a single piece of the puzzle at a time. Verbose messages like "Position 1: circle, Position 2: triangle, Position 3: square..." encode a

ESANN 2026 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 22-24 April 2026, i6doc.com publ., ISBN 9782875870964. Available from http://www.i6doc.com/en/.

complete mapping that the other agent can directly copy. Short messages like "What color is the circle?" require multi-turn negotiation, compounding error probability.

4 Conclusion

We presented AsymPuzl, an evaluation testbed for multi-turn cooperative play between two LLM-based agents under information asymmetry. We showed that strong models (e.g., GPT-5 and Claude-4.0) can reliably converge with each agent sharing all of its information, as agents ideally would. In contrast, other models struggle with miscommunication. Our results show that feedback matters: simple self-feedback improves performance, but detailed joint feedback can confuse agents. These findings underscore the importance of carefully designing communication and evaluation protocols in multi-agent LLM systems.

References

[1] Weize Chen et al. AgentVerse: Facilitating Multi-Agent Collaboration and Exploring Emergent Behaviors. October 2023.

[2] Hugo Touvron et al. LLaMA: Open and Efficient Foundation Language Models, February 2023. arXiv:2302.13971 [cs].

[3] OpenAI et al. GPT-4o System Card, October 2024. arXiv:2410.21276 [cs].

[4] OpenAI et al. gpt-oss-120b & gpt-oss-20b Model Card, August 2025. arXiv:2508.10925 [cs].

[5] Khanh-Tung Tran et al. Multi-Agent Collaboration Mechanisms: A Survey of LLMs, January 2025. arXiv:2501.06322 [cs].

[6] Shuyan Zhou et al. WebArena: A Realistic Web Environment for Building Autonomous Agents. October 2023.

[7] Panagiotis Giadikiaroglou et al. Puzzle Solving using Reasoning of Large Language Models: A Survey. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 11574–11591, Miami, Florida, USA, November 2024. Association for Computational Linguistics.

[8] Kartikeya Badola et al. Multi-Turn Puzzles: Evaluating Interactive Reasoning and Strategic Dialogue in LLMs, August 2025. arXiv:2508.10142 [cs] version: 3.

[9] Bill Yuchen Lin et al. ZebraLogic: On the scaling limits of LLMs for logical reasoning. In *Forty-Second International Conference on Machine Learning*, 2025.

[10] Guohao Li et al. CAMEL: Communicative Agents for "Mind" Exploration of Large Language Model Society. November 2023.

[11] Wei Liu et al. Autonomous agents for collaborative task under information asymmetry. In *The Thirty-Eighth Annual Conference on Neural Information Processing Systems*, 2024.

[12] Aaron Grattafiori et al. The Llama 3 Herd of Models, November 2024. arXiv:2407.21783 [cs].