Teammate-Conditioned Communication in Agentic Multi-Agent Systems: A Controlled Codenames Benchmark with Shared Persona Memory

Anonymous Submission
MEIS@CVPR 2026 (Non-Archival, Work-in-Progress)

Abstract

Collaborative agentic systems must adapt communication not only to task state but also to the teammate who will interpret and act on that communication. We study this question in a controlled Codenames-style benchmark with shared persona memory, where agents receive graded reveals along two axes: self-memory reveal and partner-memory reveal. We use a single-round forced-choice protocol in which a speaker produces one legal clue and a listener returns an ordered Top-5 guess list, allowing us to isolate teammate-conditioned communication without multi-turn confounds. Across API-driven experiments with `o4-mini` on a completed 80-board, two-prompt-mode suite (1440 records total), we observe reveal-sensitive differences that persist after parser stabilization and offline rescoring under an alternative penalty profile. `Hard` prompts show gains along both reveal axes, while `prioritized` prompts retain a strong partner-reveal gradient with weaker self-reveal gains. Overall, the results suggest that shared persona memory is a measurable coordination resource in agentic multi-agent communication, and they are consistent with distinct roles for self- and partner-conditioned context in one-round collaboration.

1 Introduction

Effective coordination in agentic multi-agent systems depends not only on modeling the external task state, but also on modeling the teammate who will interpret and execute a message. This issue is central in embodied and semi-embodied settings such as robotics, autonomous systems, and collaborative assistants, where communication is bandwidth-limited and miscoordination can directly affect task success and safety. Yet many language-agent evaluations still assume a generic teammate, making it difficult to test whether agents are truly partner-conditioned or simply applying static heuristics.

We treat shared persona memory as an operationalized form of common ground [9, 6].

Research Question.

- Given the same board state, does revealing self and partner persona memory change one-round communication outcomes?
- Do self reveal and partner reveal play different roles in cooperative language-agent communication?

We therefore study teammate-conditioned communication in a controlled simulation benchmark rather than an end-to-end embodied deployment. Codenames provides a compact testbed in which collaborator modeling, clue generation, interpretation, and risk can be measured under fixed communication budgets. This makes the setting useful for evaluating a component that matters broadly in agentic multi-agent systems: whether an agent changes its communication policy as teammate information becomes available.

Contributions.

- We introduce a controlled single-round 3×3 benchmark for teammate-conditioned communication under explicit self/partner memory reveal.
- We report completed 80-board, two-prompt-mode evidence (`hard` and `prioritized`) with parser/fallback diagnostics and non-fallback checks.
- We evaluate robustness by rescoring the same trajectories under the original penalty profile, separating policy behavior from evaluator profile choice.

2 Related Work

Our benchmark design is motivated by classic work on common ground and grounding [9, 6, 3, 2], which we operationalize as structured persona memory with controlled reveal levels. Prior multi-agent communication work shows that coordination quality depends strongly on teammate assumptions [5, 8, 11, 14, 13, 7]; our setting isolates this factor in a fixed-budget language game. Recent LLM-agent studies emphasize role decomposition and social reasoning [15, 4, 10, 12, 1]; we contribute a compact evaluation primitive for teammate-conditioned communication that can complement richer embodied or multi-turn settings.

3 Task and Protocol

3.1 Single-round 3x3 reveal grid

Each board is evaluated under 3×3 reveal settings:

$$(\texttt{self_reveal}, \texttt{partner_reveal}) \in \{0, 50, 100\}^2.$$

For each setting, the protocol has three deterministic stages: the speaker receives the board and revealed persona slices and

outputs one legal clue with a number; the listener receives that clue and returns an ordered Top-5 guess list; the environment executes guesses in order under standard stop-on-miss rules.

For each board, we evaluate all nine reveal conditions under the same board instance so that differences are attributable to reveal condition rather than board sampling. Self reveal controls how much of an agent's own persona memory is visible to that agent, and partner reveal controls how much of the teammate persona memory is visible. The 50% setting exposes a fixed partial slice of structured persona fields, while 100% exposes the full persona memory defined for the pair.

Top-5 semantics are explicit: this is a ranked action proposal, not free-form discussion. No `stop` action is available in this mode.

3.2 Reward and diagnostics

Primary metric is first-round reward. For ordered guesses $g_1, \ldots, g_m$:

$$R = \sum_{i=1}^{m} r(g_i),$$

where execution depth m follows protocol constraints and rollout halts early on turn-ending events.

We report first-round reward (`aggressive` profile in-run), `hit@5`, first own-hit rank in Top-5, assassin rate, and parser/fallback diagnostics. The in-run aggressive reward profile is $(+1.0, 0.0, 0.0, -5.0)$ for (OWN, NEUTRAL, OPPONENT, ASSASSIN), while offline rescoring uses the original profile $(+1.0, -0.5, -1.0, -5.0)$. This separation allows evaluator-profile robustness checks on the same generated trajectories.

4 Method

4.1 Persona-grounded communication setup

Each agent has structured persona memory (occupation, expertise, seniority, recurring interests, jargon/examples). Reveal controls what portion of this memory is shown per condition.

4.2 Constrained clue/guess protocol

We use strict JSON contracts for speaker/listener outputs and finite board-word actions. Key controls in the current stack include a minimum clue-number floor (≥ 3 in the main runs), optional prompt pressure toward larger clue numbers when supported, memory-sensitive board filtering with anchor constraints, and parser retries with explicit fallback tracking. The `prioritized` and `hard` modes share the same evaluator and output contracts but differ in memory instruction strength: `prioritized` treats memory as a strong secondary signal after board evidence, whereas `hard` adds explicit memory-priority instructions and enforces reveal-dependent Top-5 memory alignment minima when aligned words are available (at least 1 at reveal 50, at least 2 at reveal 100).

Table 1: Run quality and parser stability (Top-5).

Arm	Records	Speaker parse	Speaker fb	Listener fb
Hard	720	0.994	0.006	0.000
Prioritized	720	0.988	0.013	0.000

4.3 Prompt policy note

By default, the model prompt does **not** receive explicit numeric reward coefficients. Reward variants are evaluator-side settings. This separates policy behavior from metric interpretation. We now also expose an optional reward-disclosure flag for controlled A/B tests (`--disclose-reward-function`); this control is reported separately from the default setting.

5 Experiments

5.1 Setup

We evaluated 10 directed persona pairs with 8 boards per pair under the full 3×3 reveal grid and two prompt modes (`hard` and `prioritized`), yielding 1440 total records. The model was `o4-mini` accessed through an OpenAI-compatible API, and the listener produced ordered Top-5 guesses in the main runs.

5.2 Run quality progression

Parser stability remained high in both arms: speaker parse/fallback was $0.994/0.006$ in `hard` and $0.988/0.013$ in `prioritized`, while listener fallback remained 0% in both.

5.3 Main result slices

Across the completed 80-board suite, reveal-sensitive differences are visible in both prompt modes, but the shape of the effect differs by mode. In the `hard` arm, reward generally increases as either reveal axis increases, with strongest cells under high reveal. In the `prioritized` arm, partner reveal shows the clearest and most consistent gradient, while self reveal produces smaller and less uniform gains. This pattern suggests that teammate-conditioned context matters in one-round coordination, but that its effect depends on prompt policy.

Risk trends are also reveal-sensitive. In `hard`, assassin rate decreases as reveal increases on either axis. In `prioritized`, higher partner reveal still lowers assassin exposure relative to low-partner conditions.

5.4 Reward-profile robustness

On the completed 80-board suite, trajectory-rescoring under original penalties shifts absolute means but preserves directional patterns. For `hard`, mean reward changes from 2.113 (aggressive) to 1.535 (original). For `prioritized`, mean reward changes from 1.936 to 1.335. In both modes, higher partner-reveal cells remain stronger under rescoring, supporting a reward-robust claim rather than a single-profile artifact.

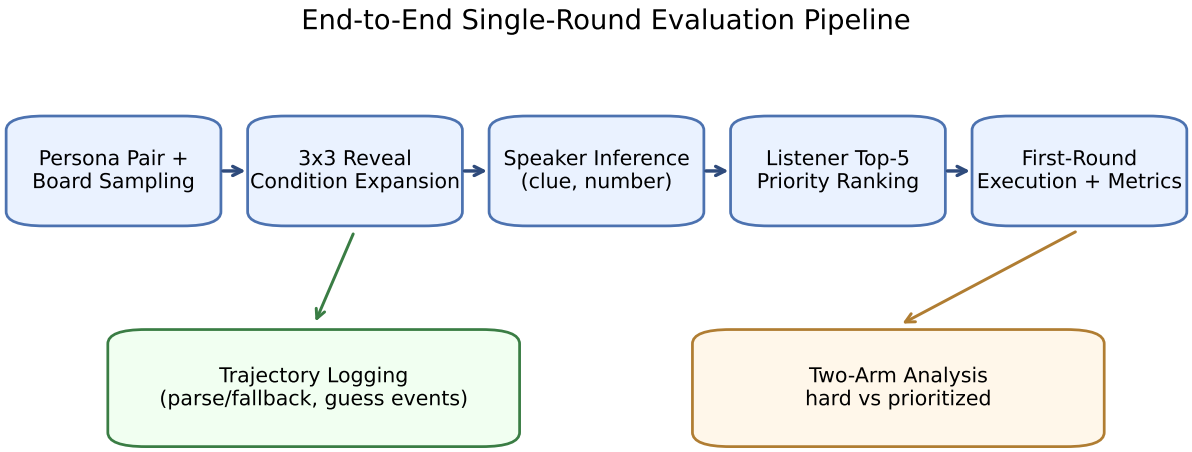

Figure 1: End-to-end single-round evaluation pipeline. The run keeps policy generation and evaluator scoring decoupled, then reports both aggressive-profile reward and original-profile trajectory rescoring.

Table 2: Collapsed marginal effects on completed 80-board arms ($n = 720$ each).

Axis	Low	High	Δ reward	Δ assassin (pp)
Hard self reveal (0 → 100)	1.775	2.542	+0.767	-3.75
Hard partner reveal (0 → 100)	1.817	2.471	+0.654	-6.67
Prioritized self reveal (0 → 100)	1.879	1.996	+0.117	+2.50
Prioritized partner reveal (0 → 100)	1.675	2.342	+0.667	-1.67

5.5 Self vs partner marginal effects

To make asymmetry explicit, we collapse each 3×3 arm by one axis at a time. Table 2 shows a consistent partner-reveal gain in both arms, while self-reveal gain is larger in `hard` than in `prioritized`. These marginals are consistent with an asymmetry in how self- and partner-memory reveal affect one-round coordination.

5.6 Fallback-robustness check

The 80-board conclusions are not driven by fallback artifacts. Fallback is rare in both arms (speaker 0.6% hard, 1.25% prioritized; listener 0.0% both), and recomputing collapsed trends on non-fallback-only records ($n = 716$ hard, $n = 711$ prioritized) yields near-identical marginals.

6 Discussion

The collapsed marginals are consistent with an asymmetry between self- and partner-memory reveal. In particular, partner reveal improves reward in both prompt modes, while self reveal has a stronger effect in the `hard` arm than in the `prioritized` arm. We interpret this as suggestive evidence that self-conditioned and partner-conditioned context may contribute differently to one-round coordination, though the present study is intended as a controlled benchmark rather than a definitive mechanistic account.

Our main takeaway is not that this benchmark fully captures real-world multi-agent interaction, but that it isolates a measurable coordination primitive: communication conditioned on teammate information. This is relevant to agentic multi-agent systems in which agents must communicate under limited bandwidth, partial shared context, and safety-sensitive execution. In that sense, the benchmark functions as a controlled evaluation layer for collaborator modeling, complementing richer embodied or multi-turn settings rather than replacing them.

7 Limitations and Next Steps

This study uses a synthetic Codenames-style environment rather than a human or embodied deployment, and the present evidence is strongest for single-round behavior under a fixed communication protocol. The benchmark therefore trades ecological realism for measurement control. In addition, the conclusions should be interpreted at the level of observed behavioral patterns under the current prompts, parser constraints, and reward definitions, rather than as a complete account of teammate modeling in general multi-agent systems. Small reward-disclosure controls were run separately but are omitted from the main analysis because their estimates remain more fallback-sensitive than the primary 80-board results.

8 Conclusion

We present a controlled benchmark for teammate-conditioned communication in agentic multi-agent systems. In a single-round Codenames-style protocol, graded reveal of shared persona memory produces measurable differences in reward and risk, with especially consistent gains from partner-memory reveal across prompt modes. These results support the use of controlled communication benchmarks as a useful evaluation

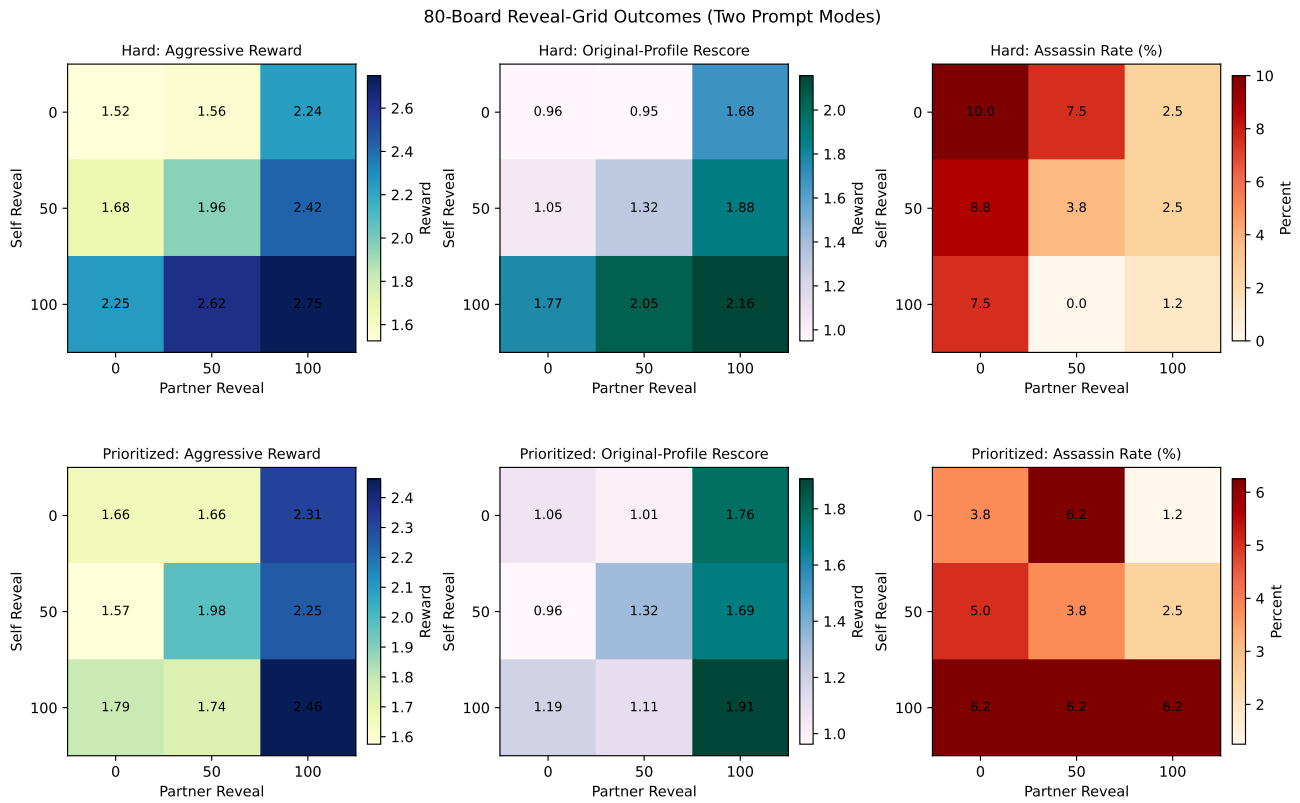

Figure 2: 80-board two-arm heatmap diagnostics. Rows are prompt modes (`hard`, `prioritized`); columns show aggressive in-run reward, original-profile trajectory rescore, and assassin rate.

layer for collaborator modeling in broader multi-agent and embodied-agent research.

References

[1] A. Bakhtin, N. Brown, E. Dinan, G. Farina, C. Flaherty, D. Fried, A. Goff, J. Gray, H. Hu, E. Jacob, et al. Human-level play in the game of diplomacy by combining language models with strategic reasoning. *Science*, 378(6624):1067–1074, 2022.

[2] H. H. Clark. *Using Language*. Cambridge University Press, 1996.

[3] H. H. Clark and S. E. Brennan. Grounding in communication. In L. B. Resnick, J. M. Levine, and S. D. Teasley, editors, *Perspectives on Socially Shared Cognition*, pages 127–149. American Psychological Association, 1991.

[4] Y. Du, S. Li, A. Torralba, J. Tenenbaum, and I. Mordatch. Improving factuality and reasoning in language models through multiagent debate. *arXiv preprint arXiv:2305.14325*, 2023.

[5] J. Foerster, Y. Assael, N. de Freitas, and S. Whiteson. Learning to communicate with deep multi-agent reinforcement learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2016.

[6] H. P. Grice. Logic and conversation. In *Syntax and Semantics: Speech Acts*, volume 3, pages 41–58. Academic Press, 1975.

[7] N. Jaques, A. Lazaridou, E. Hughes, C. Gulcehre, P. Ortega, D. Strouse, J. Leibo, and N. de Freitas. Social influence as intrinsic motivation for multi-agent deep reinforcement learning. In *International Conference on Machine Learning (ICML)*, 2019.

[8] A. Lazaridou, A. Peysakhovich, and M. Baroni. Multi-agent cooperation and the emergence of (natural) language. In *International Conference on Learning Representations (ICLR)*, 2017.

[9] D. Lewis. *Convention: A Philosophical Study*. Harvard University Press, 1969.

[10] G. Li, H. Hammoud, H. Itani, D. Khizbullin, and B. Ghanem. Camel: Communicative agents for "mind" exploration of large language model society. *arXiv preprint arXiv:2303.17760*, 2023.

[11] R. Lowe, Y. Wu, A. Tamar, J. Harb, P. Abbeel, and I. Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[12] J. S. Park, J. O'Brien, C. Cai, M. Morris, P. Liang, and M. Bernstein. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology (UIST)*, 2023.

[13] T. Rashid, M. Samvelyan, C. De Witt, G. Farquhar, J. Foerster, and S. Whiteson. Qmix: Monotonic value function factorisation for deep multi-agent reinforcement learning. In *International Conference on Machine Learning (ICML)*, 2018.

[14] P. Sunehag, G. Lever, A. Gruslys, W. Czarnecki, V. Zambaldi, M. Jaderberg, M. Lanctot, N. Sonnerat, J. Leibo, K. Tuyls, and T. Graepel. Value-decomposition networks for cooperative multi-agent learning. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS)*, 2018.

[15] S. Yao, J. Zhao, D. Yu, N. Du, I. Shafran, K. Narasimhan, and Y. Cao. React: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations (ICLR)*, 2023.