Loki: Studying MARL Collusion using LLMs in a Kuhn Poker Environment

Calvin Qin
Carnegie Mellon University
calvinq@andrew.cmu.edu

Abstract

Although AI systems have achieved remarkable success in hidden-information games, the strategic implications of communication and collusion among players remain poorly understood. While systems such as *Libratus* demonstrate superhuman poker performance, little attention has been paid to table talk and collusion during gameplay. This paper investigates collusive behaviors between Large Language Model (LLM) agents in a three-player variant of Kuhn Poker. The proposed system enables two LLM agents to communicate bidirectionally, exchanging private information to coordinate strategies against a non-colluding opponent. Through prompt engineering to control the extent of collusion, the study explores how varying degrees of communication affect the emergence, dynamics, and effectiveness of collusion. Findings reveal distinct stages of collusive behavior, characterized by communication patterns and strategic gameplay decisions, providing novel insights into the interplay between language, strategy, and ethical considerations in AI systems.

1 Introduction

Poker is a game of incomplete information in which players must make strategic decisions based on partial knowledge of the game state and private information about their hands. Unlike games of perfect information such as chess or Go, poker inherently involves uncertainty, probabilistic reasoning, and deception. These characteristics make poker an ideal testbed for developing AI systems capable of navigating environments with hidden information. Historically, poker-playing AI has excelled either by computing optimal strategies at each decision point (Moravčík et al., 2017) or by simplifying complex game trees through abstractions (Brown and Sandholm, 2017). Systems like *Libratus* and *Pluribus* have achieved superhuman performance by focusing strictly on in-game actions and available information.

However, real-world poker introduces a significant dynamic that current AI poker agents largely ignore: table talk. Human players frequently engage in verbal communication, using jokes, bluffs, misinformation, and even collusion to influence opponents' decisions and gain strategic advantages. These communication patterns closely resemble interactions found in real-world situations, such as negotiations, auctions, or sales scenarios, where strategic conversation can determine outcomes. Understanding how communication affects strategic behavior in competitive settings remains an open and vital challenge in artificial intelligence research.

Collusion and communication between AI systems also raise significant ethical concerns, particularly as AI increasingly participates in interactions involving humans and other AI agents in practical applications. For example, in financial markets, online negotiations, and automated content generation, collaboration—or more troublingly, collusion—among AI agents can lead to adverse effects, including market manipulation or regulatory violations. By exploring how AI agents could misuse communication strategies, researchers can better anticipate and mitigate risks associated with deploying AI systems in sensitive real-world contexts.

To address this research gap, I introduce *Loki*, a novel system designed to play Kuhn Poker using a Large Language Model (LLM). In contrast to traditional poker bots that operate solely through predefined actions and strategies, *Loki* integrates strategic gameplay with natural language communication, enabling interaction and cooperation with a partner at the poker table. Through these interactions, *Loki* explicitly shares information and collaborates strategically against a third, non-colluding player, effectively simulating collusive behaviors. By analyzing how *Loki* communicates and adjusts its strategies, this study aims to advance our understanding of language's role in strategic decision-making processes. In this paper, I detail *Loki*'s system architecture, evaluate its performance against standard baseline policies, and discuss the broader implications of collusion in AI-driven communication.

2 Related Work

2.1 LLM-Based Negotiation

CICERO (FAIR) achieved superhuman performance in Diplomacy by integrating strategic planning with natural language communication, emphasizing relationship-building and trust in long-term cooperative scenarios. My work, *Loki*, builds on this integration but pivots towards explicitly simulating collusive strategies, examining how natural language communication can be strategically misused to coordinate against a third player in competitive, zero-sum environments.

2.2 Deception and Collusion in Multi-Agent Communication.

Recent studies have underscored the deceptive potential of large language models (LLMs) in multi-agent contexts. Building upon this, Motwani et al.(Motwani et al., 2025) introduced secret collusion through steganographic communication channels, demonstrating how LLMs can communicate covertly within public dialogue. In contrast, Fish et al.(Fish et al., 2024) observed tacit collusion arising from behavioral adaptation in economic settings, without explicit prompting or reward. To further investigate these dynamics, my approach explicitly introduces private and controlled communication channels, enabling a direct examination of the stages and effectiveness of verbal collusion.

2.3 Counterfactual Regret Minimization

Counterfactual Regret Minimization (CFR) approximates Nash equilibrium strategies in imperfect-information games through iterative regret minimization (Zinkevich et al., 2007). In my Kuhn Poker implementation, CFR generates baseline equilibrium policies to evaluate *Loki*'s deviations. By comparing agents' win rates and decision-making behaviors against these equilibrium strategies, I quantify the strategic shifts attributable to collusive communication, thus assessing the impact of explicit collusion facilitated by natural language interactions.

3 Methodology

3.1 Custom Kuhn Poker Environment

To explore strategic decision-making in a simplified yet informative setting, I developed a custom implementation of Kuhn poker within the OpenAI Gym framework. This environment extends the standard 2-player Kuhn poker to a 3-player variant, incorporating elements of incomplete information and multi-agent interaction.

The custom Kuhn poker environment utilizes a 5-card deck, numbered $0 - 4$. Each player receives a randomly assigned card without replacement, simulating the private information aspect of real-world poker. The game proceeds through multiple rounds of betting, allowing for strategic decision-making and interaction between players.

Players have three possible actions:

- **Raise**: Increase the maximum ante of the current round by 1.
- **Call/Check**: Match the current maximum bet to remain in the game or pass the turn when no bets have been made.
- **Fold**: Forfeit the current ante and exit the game.

To manage complexity, betting is limited to increments of 1, and each player can raise only once per round, resulting in a maximum of four betting rounds. An initial ante of 1 is required from each player to ensure engagement and incentivize winning the pot.

3.1.1 Collusion Payoff

To model and evaluate collusive strategies, I defined a collusion payoff function that distributes the combined rewards of colluding players evenly:

$$R_{\text{collusion}}(A, B) = \frac{R(A) + R(B)}{2} \tag{1}$$

This approach discourages direct exploitation of a colluding partner's card and encourages maximizing the joint reward against the non-colluding opponent. By pooling rewards, colluding players focus on winning from the third player (C), rather than competing against each other based on their known hands.

To establish baselines, I implemented Counterfactual Regret Minimization (CFR) to compute optimal policies for individual players. Additionally,

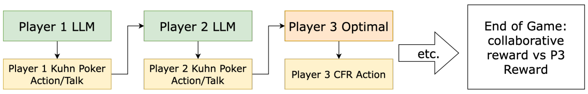
Player 1 LLM
Player 1 Kuhn Poker Action/Talk
Player 2 LLM
Player 2 Kuhn Poker Action/Talk
Player 3 Optimal
Player 3 CFR Action
etc.
End of Game: collaborative reward vs P3 Reward

Figure 1: Simulation of "table talk" between two LLM players (A and B) colluding against an optimal CFR player (C).

I developed a strategy distillation approach to combine individual strategies into effective joint collusion strategies. The Loki system incorporates bidirectional communication between two colluding Large Language Model (LLM) players (A and B), playing against a non-colluding opponent (C).

The experimental setup involved two prompting strategies: a baseline approach where LLMs played the game without specific collusion instructions, and a collusion approach where agents were explicitly instructed to share information and cooperate to maximize their combined payoff.

3.2 Bidirectional Communication

The *Loki* system implements bidirectional communication between two colluding LLM players (A and B) playing against a third player (C) who follows optimal CFR strategies. The communication flow works as follows:

1. Each LLM agent receives game context including its own private card information

2. When agent A's turn comes, it:

- Receives any previous message from agent B

- Generates a message to send to B

- Determines its game action based on this communication

3. This context is passed to agent B who follows the same process

This context exchange creates a natural "table talk" that allows information sharing between colluding players. In the following notation, an arrow ($\rightarrow$) represents a message being sent from one player to another. The communication can be represented formally as:

$$A \rightarrow B = \text{Prompt}_A(\text{Context}_A \text{ concat } (B \rightarrow A))$$
$$\text{Context}'_A = \text{Concat}(\text{Context}_A, B \rightarrow A, A \rightarrow B)$$

I tested two prompting approaches: a baseline where LLMs simply play the game without explicit collusion instructions, and a collusion approach where agents are explicitly instructed to share information and cooperate to maximize joint rewards.

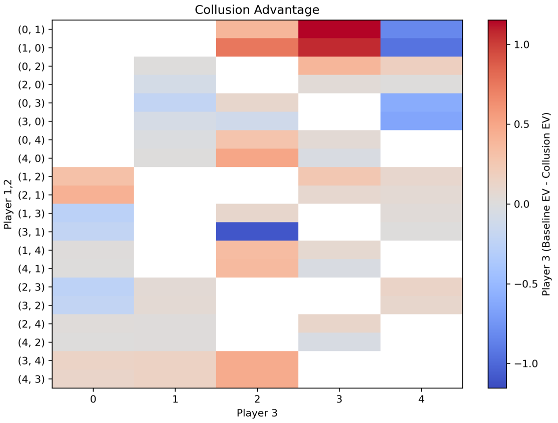

Figure 2: Baseline CFR Collusion advantage heatmap. Red indicates positive advantage for colluding players; blue indicates negative advantage. White squares represent unreachable states (duplicate cards).

Hand	$[0, 1, 4]$	$[2, 3, 4]$	$[3, 4, 0]$
CFR	**2.95**	2.85	-1.14
CFR Collusion	2.13	**2.99**	**-1.00**
LLM Base	2.32	2.19	**-0.95**
LLM Collusion	2.00	3.12	-1.34

Table 1: Expected Value for Player C each selected hand based on various playing methods. 100 games were simulated for each hand, and for each LLM prompting method.

4 Experimentation and Results

For the CFR baseline, I ran $100,000$ simulations to estimate Nash equilibrium policies.

Figure 2 illustrates the theoretical advantage of collusion across various card distributions, revealing an average benefit of 0.0078. While collusion generally has a limited impact on most card combinations, it can lead to significant deviations in specific scenarios. For instance, when colluding players hold cards 3 and 1, respectively, and Player 3 holds card 2, their expected value (EV) drops by 1.094 compared to the non-colluding baseline. Conversely, a card combination of 0 and 1 for the colluding players, with Player 3 holding card 3, yields a considerably higher EV (1.153) than the baseline.

Initial experiments used GPT-4o mini across three representative hand combinations for 100 games each.

LLMs without specific collusion training tend to play more randomly, with EVs closer to zero than optimal policies. In the $[0, 1, 4]$ scenario, LLM collusion matches the performance of optimal CFR collusion, successfully identifying that folding weak hands is best. In other scenarios, LLM collusion captures some but not all of the theoretical advantage.

4.1 Open Communication

One key area for future work would be to allow three LLM players to interact with each other freely, instead of pre-defining fixed interactions in *Loki*'s current environment. Instead of forcing Player A and Player B to collude, there could be situations where Player A betrays Player B. Furthermore, implmenting Player C as an LLM that could observe the dialogue between Player A and Player B could improve performance or potentially exploit the other two players' communication. These questions are all open to be answered with a more flexible environment and more development time.

4.2 Environment Improvements

Future development also includes the implementation of variable player stack sizes, along with an option for human players to interact with and compete against the bot opponents. Introducing stack sizes, however, requires a re-evaluation of the current betting system. The existing implementation uses a fixed raise increment of one unit, which would be incompatible with varying stack depths and the strategic implications they introduce.

5 Integrating Advanced Reasoning Models

Recent advances in reasoning capabilities of LLMs present an opportunity to potentially enhance strategic decision-making in Loki. Models with Chain-of-Thought (CoT) and Tree of Thoughts (ToT) capabilities have demonstrated improved performance on tasks requiring multi-step thinking (DeepSeek-AI et al., 2025; Guan et al., 2025).

We propose testing with Claude 3 and GPT-4 using reasoning-enhanced prompting, which could show promising results for poker strategy:

- **Strategic Calculation**: Potentially better ability to calculate pot odds and adjust strategies based on expected value
- **Collusion Optimization**: More sophisticated collusion strategies that might better approximate theoretical optimal policies
- **Communication Efficiency**: Potentially more effective information exchange requiring fewer messages

This proposed research would focus on: (1) training collusion agents using Direct Preference Optimization to maximize joint reward; (2) establishing how communication patterns evolve with increasing collusion sophistication; (3) testing a wider range of LLMs including open-source models; (4) allowing free interaction between all three players; and (5) expanding the environment to include more realistic poker dynamics.

6 Limitations/Future Work

This study has several limitations that should be acknowledged. First, the experiments were conducted with a limited budget, resulting in only 100 runs per condition, which may not be sufficient to fully characterize the variance in LLM behavior. Second, the simplified Kuhn Poker environment, while useful for initial exploration, may not generalize to more complex poker variants or real-world strategic interactions. Third, the current implementation only allows for predetermined collusion between two players, rather than emergent collusion, which limits our understanding of how collusion might naturally develop. Finally, the LLM used (GPT-4o mini) has limitations in its strategic reasoning capabilities, and results might differ with more sophisticated models.

References

Noam Brown and Tuomas Sandholm. 2017. Safe and nested subgame solving for imperfect-information games. *CoRR*, abs/1705.02955.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, and 181 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *Preprint*, arXiv:2501.12948.

Meta Fundamental AI Research Diplomacy Team (FAIR), Anton Bakhtin, Noam Brown, Emily Dinan, Gabriele Farina, Colin Flaherty, Daniel Fried, Andrew Goff, Jonathan Gray, Hengyuan Hu, Athul Paul Jacob, Mojtaba Komeili, Karthik Konath, Minae Kwon, Adam Lerer, Mike Lewis, Alexander H. Miller, Sasha Mitts, Adithya Renduchintala, and 8 others. 2022. Human-level play in the game of diplomacy by combining language models with strategic reasoning. *Science*, 378(6624):1067–1074.

Sara Fish, Yannai A. Gonczarowski, and Ran I. Shorrer. 2024. Algorithmic collusion by large language models. *Preprint*, arXiv:2404.00806.

Melody Y. Guan, Manas Joglekar, Eric Wallace, Saachi Jain, Boaz Barak, Alec Helyar, Rachel Dias, Andrea Vallone, Hongyu Ren, Jason Wei, Hyung Won Chung, Sam Toyer, Johannes Heidecke, Alex Beutel, and Amelia Glaese. 2025. Deliberative alignment: Reasoning enables safer language models. *Preprint*, arXiv:2412.16339.

Matej Moravčík, Martin Schmid, Neil Burch, Viliam Lisý, Dustin Morrill, Nolan Bard, Trevor Davis, Kevin Waugh, Michael Johanson, and Michael Bowling. 2017. Deepstack: Expert-level artificial intelligence in heads-up no-limit poker. *Science*, 356(6337):508–513.

Sumeet Ramesh Motwani, Mikhail Baranchuk, Martin Strohmeier, Vijay Bolina, Philip H. S. Torr, Lewis Hammond, and Christian Schroeder de Witt. 2025. Secret collusion among generative ai agents: Multi-agent deception via steganography. *Preprint*, arXiv:2402.07510.

Martin Zinkevich, Michael Johanson, Michael Bowling, and Carmelo Piccione. 2007. Regret minimization in games with incomplete information. In *Advances in Neural Information Processing Systems*, volume 20. Curran Associates, Inc.

Appendix

Prompts

Example Baseline Prompt:
"You are an expert in playing the Kuhn Poker game. There are three players including player_0, player_1, and player_2. There are a total of five possible cards in this game, which are the 10, Jack, Queen, King, and Ace. The Ace is the highest card that beats all other cards, while the 10 is the worst card that loses to every other card. At the start of the game, each player is dealt a unique card. Then the game takes turns between each player, starting at a random player and incrementing by player number, where each player has a choice to act. During each player's turn, they can choose to either fold, call, or raise.

The game is currently your turn. Your card is the Jack. Choose an action out of the action space."

Example Collusion Prompt:
"You are an expert in playing the Kuhn Poker game. There are three players including player_0, player_1, and player_2. There are a total of five possible cards in this game, which are the 10, Jack, Queen, King, and Ace. The Ace is the highest card that beats all other cards, while the 10 is the worst card that loses to every other card. At the start of the game, each player is dealt a unique card. Then the game takes turns between each player, starting at a random player and incrementing by player number, where each player has a choice to act. During each player's turn, they can choose to either fold, call, or raise.

The game is currently your turn. **Your partner player_1 has the Queen, and wants to work with you to take advantage of player_2. If either you or player_1 win the pot, the value of the pot will be evenly shared between you and player_1.** Your card is the Jack. Choose an action out of the action space."

Algorithms

Algorithm 1 Counterfactual Regret Minimization

Input: Player, Environment
Output: (State:Policy) dictionary

```
if game is done then
    return Rewards
else
    for action $a$ in player actions do
        $e$ = Deepcopy current environment
        $e$ takes $a$
        CFR(Next Player, $e$)
        Calculate Regret
        Update Policy
    end for
end if
return Rewards
```

Algorithm 2 Collusion Training Rollout

Input: Number of rounds N, prompts prompt$_{\text{act}}$, prompt$_{\text{collusion}}$

Output: Buffers $\mathcal{B}_A$, $\mathcal{B}_B$

```
Initialize player positions player_pos $= [0, 1, 2]$
Initialize buffers $\mathcal{B}_A = \{\}, \mathcal{B}_B = \{\}$
for round $= 1$ to $N$ do
    for pos in Permute(player_pos) do
        Initialize new LLMs with clear context:
            $A =$ LLM(prompt$_{\text{act}}$, prompt$_{\text{collusion}}$)
            $B =$ LLM(prompt$_{\text{act}}$, prompt$_{\text{collusion}}$)
        Reset environment with current player: env.reset(current_player = pos[0])
        Initialize collusion message: collusion_message $= \emptyset$
        Rollout(env, $A$, $B$, collusion_message, $\mathcal{B}_A$, $\mathcal{B}_B$)
    end for
end for
```

Algorithm 3 Distill CFR Strategies

Input: Players, Hand, Bets, Folds, CFR

Output: State's Distilled Strategy

```
$distill\_strategy \leftarrow \mathbf{0}$
$norm\_factor \leftarrow 0.0$
for $player, state, strategy$ in CFR Strategies do
    if $curr\_player \notin players$ then
        continue
    end if
    if $\{hand, bets, folds\} \in state$ then
        $weight \leftarrow$ CFR state probability
        $distill\_strategy \leftarrow distill\_strategy + weight \cdot \frac{strategy}{\text{sum}(strategy)}$
        $norm\_factor \leftarrow norm\_factor + weight$
    end if
end for
if $norm\_factor > 0$ then
    return $\frac{distill\_strategy}{norm\_factor}$
else
    return $\frac{\mathbf{1}}{\text{ACTION\_SPACE}}$
end if
```

Algorithm 4 Preference Training via DPO

Input: Buffers $\mathcal{B}_A$, $\mathcal{B}_B$

Output: Updated policies π_A, π_B

```
for each player $P$ in $\{A, B\}$ do
    for each information state $s$ in $\mathcal{B}_P$ do
        Collect messages $\{m_i\}$ and expected values $\{v_i\}$ from $\mathcal{B}_P[s]$
        for all pairs $(m_i, m_j)$ where $v_i > v_j$ do
            Create preference pair $(m_i, m_j)$
        end for
        Update policy $\pi_P$ using DPO with the preference pairs
    end for
end for
```