EPO: Explicit Policy Optimization for Strategic Reasoning in LLMs via Reinforcement Learning

Anonymous ACL submission

Abstract

Large Language Models (LLMs) have shown impressive reasoning capabilities in well-defined problems with clear solutions, such as mathematics and coding. However, they still struggle with complex real-world scenarios like business negotiations, which require strategic reasoning—an ability to navigate dynamic environments and align long-term goals amidst uncertainty. Existing methods for strategic reasoning face challenges in adaptability, scalability, and transferring strategies to new contexts. To address these issues, we propose explicit policy optimization (*EPO*) for strategic reasoning, featuring an LLM that provides strategies in open-ended action space and can be plugged into arbitrary LLM agents to motivate goal-directed behavior. To improve adaptability and policy transferability, we train the strategic reasoning model via multi-turn reinforcement learning (RL) using process rewards and iterative self-play, without supervised fine-tuning (SFT) as a preliminary step. Experiments across social and physical domains demonstrate *EPO*'s ability of long-term goal alignment through enhanced strategic reasoning, achieving state-of-the-art performance on social dialogue and web navigation tasks. Our findings reveal various collaborative reasoning mechanisms emergent in *EPO* and its effectiveness in generating novel strategies, underscoring its potential for strategic reasoning in real-world applications[1].

1 Introduction

Recent advances in LLMs have significantly enhanced their reasoning capabilities on static problem-solving with well-defined solutions, such as mathematics, coding, and logical reasoning (Anthropic, 2024; OpenAI, 2024b; Qwen, 2024; Google, 2024; Guo et al., 2025). However, these tasks, governed by fixed rules and deterministic outcomes, fail to capture the complexity of real-world scenarios such as business negotiations or policy design, where success hinges on navigating dynamic environments with no predefined solutions. Such scenarios demand **strategic reasoning**[2] (Zhang et al., 2024b): the ability to align long-term goals, manage uncertainty, and adapt to changing conditions. Despite progress in narrow domains, current LLMs have difficulty in integrating these capabilities, exposing a critical gap in human-like behaviors in interactive contexts.

Prior work improving strategic reasoning in LLMs falls into three categories: iterative prompting that decomposes long-term goals into stepwise plans, such as recursive self-improvement (Madaan et al., 2023; Shinn et al., 2024) or extended chain-of-thought (CoT) reasoning (Wei et al., 2022; Yao et al., 2023b); post-training LLMs through imitation learning (IL) or RL (Chen et al., 2023; Song et al., 2024); inference scaling that searches multiple reasoning paths toward goals, such as Best-of-N or Monte Carlo Tree Search (MCTS) (Yu et al., 2023; He et al., 2024; Putta et al., 2024). While these methods show promise, they face critical limitations: (1) prompting methods are limited by the inherent reasoning abilities of LLMs and struggle with real-time adaptation; (2) IL or RL approaches face challenges in generalizing reasoning skills to unseen domains (Patil, 2025), as fine-tuning on specific reasoning datasets may hinder transferability due to over-optimization (Bakhtin et al., 2022; Li et al., 2024); and (3) search algorithms suffer from computational inefficiency and poor scalability to open-ended action space in real-world scenarios.

To tackle these challenges, we introduce a method concerning **explicit policy optimization** for strategic reasoning and leverage RL to learn to reason for optimial strategies. Our method *EPO*, features an LLM dedicated to strategic reasoning, providing real-time strategies and can be seam-

[1]We will release our models, code and data soon.

[2]See Appendix A for the comparison between strategic reasoning and static problem-solving.

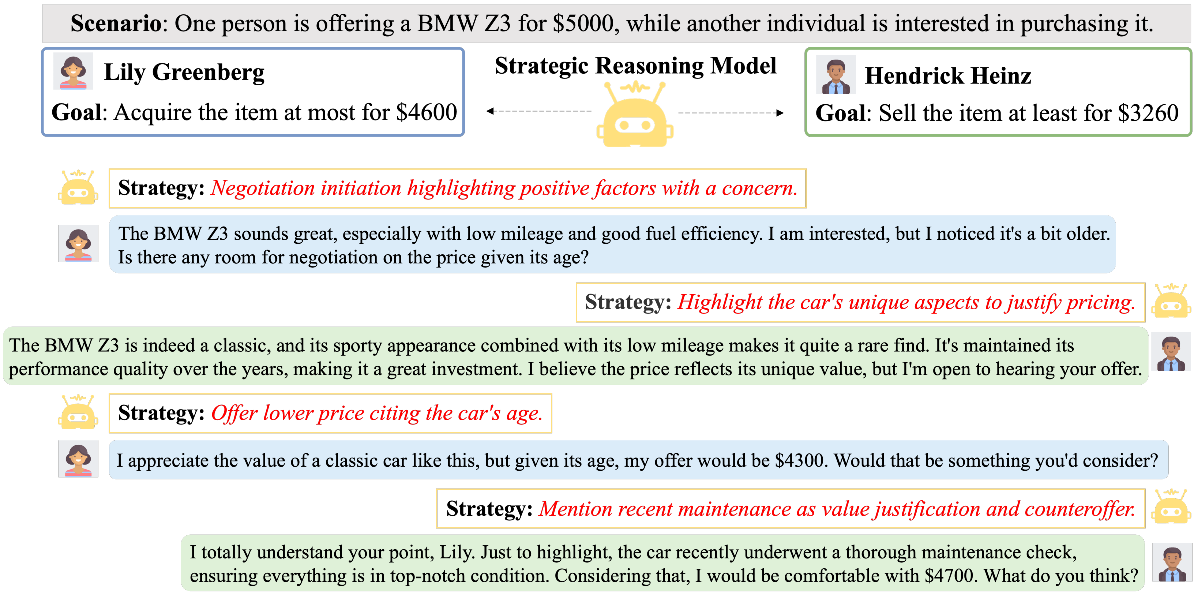


Figure 1: ***EPO* incentivizes goal-directed behavior from LLM agents in interactive scenarios.** In such scenarios, each participant's goals and strategies remain private to themselves. Notably, our strategic reasoning model can assist all involved parties, enabling to increase the overall average payoff for all participants.

lessly integrated with LLM agents to incentivize goal-directed behavior. As shown in Figure 1, our reasoning model can assist arbitrary LLM agents in achieving long-term goals across multiple interaction turns. To enhance adaptability and policy transferability, we optimize the reasoning model purely through multi-turn RL without SFT as a preliminary step. For simplicity and ease of implementation, we employ a REINFORCE (Sutton et al., 1999) policy gradient RL objective for training, enabling the reasoning model to learn optimal strategies. Additionally, we utilize a process reward model (PRM) to assess the effectiveness of generated strategies and incorporate iterative self-play to scale up RL training.

Our method, leveraging the features of *EPO*, addresses the limitations of prior work from several perspectives. Firstly, unlike previous multi-agent frameworks that are limited to prompting engineering (Gandhi et al., 2023; Duan et al., 2024), *EPO* is capable to enhance strategic reasoning capabilities via RL. This allows our reasoning model to adapt to real-time environment feedback. Moreover, *EPO* allows LLM agents interacting with environments to remain unchanged, preserving their ability to generalize to new domains and avoids over-optimization issues common in IL or RL methods. Meanwhile, our reasoning model can be seamlessly plugged into these LLM agents, regardless of their openness or inherent capabilities. Finally, the reasoning model, being an LLM, can formulate strategies in the vast, open-ended language space without high computational costs.

Experiments across social and physical domains demonstrate that *EPO* is able to align long-term goals with enhanced strategic reasoning via RL, achieving state-of-the-art performance on social dialogue and web navigation tasks. Results and analysis show that our strategic reasoning model learns to reason for optimal strategies through multi-turn RL, transferring its policy to diverse scenarios. We also discover various collaborative mechanisms between the reasoning model and LLM agents for interacting, and uncover creative strategies produced by the model. Our contributions are threefold:

- We propose explicit policy optimization for strategic reasoning, featuring a strategic reasoning model (LLM) providing real-time strategies for arbitrary LLM agents to incentivize goal-directed behavior in dynamic interactive environments.
- We develop a lightweight multi-turn RL pipeline for training the reasoning model with process rewards and iterative self-play, improving its policy transferability to unseen scenarios without SFT as a preliminary step.
- Results and analysis in diverse domains demonstrate the superior performance of *EPO* over baselines in long-term goal alignment. A

variety of collaborative reasoning mechanisms emerge in *EPO* as well as novel strategies devised by the reasoning model.

2 Related Work

Strategic Reasoning in LLMs. Motivated by in-context learning and reasoning abilities of LLMs, recent work utilizes LLMs for strategic reasoning in dynamic environment (Zhang et al., 2024b). While some work employs prompting techniques for opponent analysis (Duan et al., 2024), theory of mind (Gandhi et al., 2023; Wilf et al., 2024), or in-context demonstrations (Fu et al., 2023; Wang et al., 2024b), their effectiveness is limited by inherit abilities and issues like unfaithful explanations (Turpin et al., 2024). Fine-tuning LLMs through IL or RL (Zeng et al., 2023; He et al., 2024; Putta et al., 2024) can help but struggles to generalize reasoning skills across domains. Beyond prompting and fine-tuning, modular enhancements such as memory modules, external tools (Zhu et al., 2023; Ge et al., 2023; Hao et al., 2023; Sun et al., 2024), and multi-agent systems (Bakhtin et al., 2022, 2023; Xu et al., 2023; Ma et al., 2024) have been explored. Some of these work develop dialogue action planners via RL (Deng et al., 2023; Li et al., 2024) or game-theoretic algorithms (Gemp et al., 2024), but rely on finite, predefined action sets or lack interpretability. In contrast, our approach supports open-ended action space and improves the interpretability of strategic reasoning by generating strategies in natural language. Concurrent to our work, CoPlanner (Wang et al., 2024a) focuses on static reasoning with limited planning rounds, whereas our method enhances strategic reasoning for dynamic long-horizon planning.

Reinforcement learning for LLMs. Prior RL applications for LLMs often focus on static problem-solving like question answering (Liu et al., 2021; Suzgun et al., 2023), math problems (Lightman et al., 2024; Kumar et al., 2024), and preference alignment (Christiano et al., 2017; Guan et al., 2024). For example, methods in reinforcement learning from human feedback (RLHF) (Ouyang et al., 2022; Gulcehre et al., 2023; Rafailov et al., 2024) treat reward maximization as a one-step bandit problem, which prioritizes human-like responses rather than interactive engagement. Instead, numerous agent tasks require interactions and complex strategies, prompting the development of multi-turn RL algorithms (Zhou et al., 2024b; Shani et al., 2024) for LLMs. Our work considers strategic reasoning as an RL challenge, demonstrating the feasibility of training LLMs for strategic reasoning through a pure RL process, independent of the specific RL algorithm used.

3 Method

We first provide an overview of our method *EPO*, accompanied by a formulation of strategic reasoning process. We then introduce a multi-turn RL pipeline to explicitly optimize the policy of the strategic reasoning model in *EPO*.

3.1 Overview

Assume an interactive scenario involving an LLM agent LLM_d aiming to achieve a long-term goal G through sequential interactions. At each turn t, LLM_d receives an observation x_t (e.g., adversary messages or environment states) and generates a response y_t that balances immediate context with progress toward the goal G. Traditional approaches model this process as $P(y_t|G, h_{1:t-1}, x_t)$, where $h_{1:t-1} = \{x_1, y_1, ..., x_{t-1}, y_{t-1}\}$ is the interaction history between LLM_d and an external environment or other agents. However, this formulation does not explicitly consider the strategic reasoning process in long-term goal alignment.

To address this, we propose *EPO* that introduces an LLM LLM_s dedicated to strategic reasoning and providing strategies a to motivate goal-directed behavior from LLM_d. As shown in Figure 2, LLM_s synthesizes the goal G, interaction history $h_{1:t-1}$, prior strategies $a_{1:t-1}$, and the latest observation x_t to propose a strategy in open-ended action space:

$$a_t = LLM_s(s_{sys}, G, h_{1:t-1}, a_{1:t-1}, x_t). \quad (1)$$

This strategy then encourages LLM_d to produce goal-directed behavior:

$$y_t = LLM_d(d_{sys}, G, h_{1:t-1}, a_{1:t}, x_t). \quad (2)$$

Here, s_{sys} and d_{sys} are role-specific system prompts, which can be combined with various prompting techniques, such as CoT or Tree-of-Thought (Yao et al., 2023a). Crucially, LLM_d selectively adopts a_t when generating its behavior, allowing it to override suboptimal strategies while retaining domain-general linguistic skills. The external environment provides feedback afterward (e.g., adversary reactions or goal progress), updating h_t

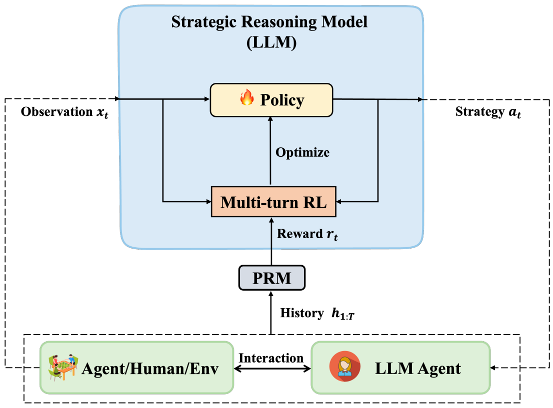


Figure 2: **Overview of *EPO*.** The solid line shows the RL training process of the strategic reasoning model, while the dotted line indicates how our reasoning model motivates goal-directed behavior from LLM agents.

for subsequent turns. With explicit policy optimization for strategic reasoning, *EPO* enables continuous strategy refinement in LLM_s while maintaing the generalization ability of LLM_d, addressing the over-optimization issues in prior work.

3.2 Learning to Reason for Optimal Strategies

To equip LLM_s with adaptive strategic reasoning, we design a lightweight multi-turn RL pipeline that optimizes its policy through iterative self-play, prioritizing process rewards over terminal outcomes. This approach enables LLM_s to learn to refine strategies based on real-time environmental feedback rather than align to predefined solutions or fixed reward models as in prior RLHF methods.

Specifically, we define the policy optimization of strategic reasoning model LLM_s as a partially observable Markov decision process (POMDP). The optimization objective is to maximize the expected return:

$$J(\theta) = \mathbb{E}_{\tau\sim\pi_\theta}[R(\tau)] = \mathbb{E}_{\pi_\theta}[\sum_{t'=t}^{T}\gamma^{T-t'}r_{t'}], \quad (3)$$

where $R(\tau)$ represents the long-term value of a trajectory $\tau = (h_1, a_1, r_1, ..., h_T, a_T, r_T)$ generated by the policy $\pi_\theta(a_t|h_t)$ until the terminal turn T. $\gamma \in [0, 1)$ is the discount factor that controls the model's temporal bias and $r_{t'}$ is the immediate reward at each interaction turn.

Policy Optimization. From the POMDP formulation, we derive a REINFORCE RL objective for training LLM_s:

$$\mathcal{L}(\theta) = \quad (4)$$

$$-\mathbb{E}_{\pi_\theta}\left[\frac{1}{T}\sum_{t=1}^{T}R_t\frac{1}{|k_t|}\sum_{i=0}^{k_t}\log\pi_\theta(a_{t,i}|h_{1:t-1},a_{t,1:i-1},x_t)\right],$$

where $a_{t,i}$ denotes the i-th token in strategy a_t, $a_{t,1:i-1}$ represents previously generated tokens within a_t, and k_t is the total number of tokens of strategy a_t. R_t is the cumulative discounted rewards from turn t to T, ensuring strategies align with long-term goals. For each training sample, we use max-absolute normalization for R_t to stabilize policy gradients across trajectories of varying lengths and reward scales.

Due to its simplicity and compatibility with offline settings (e.g., pre-collected interaction data), this RL objective can integrate seamlessly with multi-task and cross-domain training (as demonstrated in our experiments) and avoids the computational overhead of on-policy sampling. However, our framework *EPO* is algorithm-agnostic. Online RL methods such as PPO (Schulman et al., 2017) or other offline approaches like DPO (Rafailov et al., 2024) could also optimize π_θ. Furthermore, we could improve the algorithmic stability by introducing baselines to reduce variance (Ahmadian et al., 2024), and combine search techniques by sampling serveral candidate strategies from LLM_s and reranking them via learned value functions.

Process Rewards. For RL training, we assign each strategy a_t produced by LLM_s with a process (or immediate) reward r_t, reflecting its criticality in advancing LLM_d toward its goal. Specifically, $r_t = 1$ if a_t is deemed critical and 0 otherwise. To identify critical strategies, we employ an LLM[3] LLM_p as the PRM to output a list containing key strategy indexes in a training sample:

$$[S_{idx}] = LLM_p(p_{sys}, G, h_{1:T}, score, a_{1:T}), \quad (5)$$

where S_{idx} denotes the indexes of strategy $a_{1:T}$ critical in achieving the goal G. p_{sys} refers to the system prompt for PRM (refer to Appendix C.2), and $h_{1:T}$ is the full interation history until the terminal turn T. $score$ represents the final goal completion rate for corresponding sample. By integrating process rewards into policy optimization, our method enables the strategic reasoning model to generate strategies that are both tactically sound in the short term and coherent over extended horizons.

Iterative Self-Play. To scale up RL, we train LLM_s using iterative self-play, where two *EPO* instances take turns interacting as different partners. During self-play, each *EPO* instance uses its LLM_s to devise strategies, encouraging their

[3]We use GPT-4o.

paired LLM_d agents to behave toward their specific goals. We record the entire trajectory data, including the interaction history between agents LLM_d and the strategies proposed by LLM_s. We then evaluate each strategy with the PRM and obtain process rewards to retrain LLM_s in the next RL iteration based on Eq. 4, until LLM_s's policy stabilizes. Throughout this process, LLM_d's weights remain unchanged to ensure it can generalize to various domains without overfitting to particular behavior patterns.

Transferability. The transferability of our method is mainly due to two factors: (1) the explicit policy optimization for strategic reasoning where our reaosning model LLM_s can be flexibly plugged into any LLM_d, while LLM_d maintains its general-domain capabilities without the need of additional training; (2) the open-ended action space where LLM_s produces strategies. Therefore, the reasoning model can undergo RL training across different domains, improving its policy transferability to novel scenarios.

4 Experiment

The goal of our experiments is to demonstrate the efficacy and justify the design of *EPO* in enhancing strategic reasoning via RL. Our experiments address the following questions: **(1)** Is it necessary to explicitly optimize policy for strategic reasoning and what are the unique advantages? **(2)** How iterative self-play can scale RL training in *EPO*? **(3)** How the reasoning model in *EPO* affects agent's behavior and under what circumstances can optimal reasoning be achieved? **(4)** Which components of *EPO* are critical for effective strategic reasoning? To this end, we evaluate *EPO* against baselines, perform analysis and run ablations.

4.1 Experimental Settings

Environments and Datasets. We evaluate *EPO* in three distinct social and physical environments requiring strategic reasoning: (1) SOTOPIA (Zhou et al., 2024a) for social interaction with goals. A challenging subset of scenarios (SOTOPIA-hard) demands advanced strategic reasoning; (2) WebShop (Yao et al., 2022) for web navigation with dense final rewards (0–1); (3) ALFWorld (Shridhar et al., 2021) for the embodied household, comprising out-of-distribution (OOD) task variations and binary rewards indicating task success.

Dataset statistics are shown in Table 5. For SOTOPIA, we collect training data for the reasoning model from SOTOPIA-π (Wang et al., 2024c), where training scenarios are completely non-overlapping with test ones to ensure generalization. We use GPT-4-Turbo to generate strategies and dialogue histories with CoT prompting. For WebShop and ALFWorld, the training data follows (Song et al., 2024) where each trajectory contains CoT rationales that we assume as the strategy for each action step. See Appendix B for more details of the environments and data collection.

Evaluation Prompts and Metrics. We evaluate *EPO* using zero-shot prompting for SOTOPIA and one-shot prompting for WebShop and ALFWorld following (Song et al., 2024). For SOTOPIA, we measure goal completion (0-10) and overall score using GPT-4o (OpenAI, 2024a) as a proxy for human judgment following (Zhou et al., 2024a). For WebShop and ALFWorld, we use average reward as the metric. Detailed prompts are provided in Appendix C.1.

Implementation Details. We mainly use Llama3-8B-Instruct (Dubey et al., 2024) as the base model for RL training. The batch size is 32 and the learning rate is 1e-6 with 3% warm-up and a cosine scheduler. We set the learning epochs to 3 and the discount factor to 0.99. The total training steps are 19k in SOTOPIA and 20k in cross-domain training on WebShop and ALFWorld. During evaluation, we use GPT-4o or Llama3.1-70B-Instruct (Meta, 2024) as the self-play dialogue agent in SOTOPIA. For WebShop and ALFWorld, we use GPT-4o or Llama3-8B-Instruct as the agent for interating with environments. By default, our reasoning model is plugged into all interacting agents within an environment. On SOTOPIA, the temperature of our reasoning model and LLMs responsible for self-chat is set to be 0.7. For WebShop and ALFWorld, LLMs responsible for interacting with the environment adopt greedy decoding (tempature 0). Due to computational costs, we only report results in a single run. All experiments are conducted on 6 NVIDIA A100 80G GPUs. Refer to Appendix D for more details.

Baselines and Comparisons. We evaluate our method against several baselines: (1) **Vanilla**, a standard prompting method; (2) **ReAct** (Yao et al., 2023b), which employs CoT reasoning within a single LLM to generate rationales before actions;

Method	GPT-4o				Llama3.1-70B-Instruct			
	Hard		All		Hard		All	
	Goal	Overall	Goal	Overall	Goal	Overall	Goal	Overall
Vanilla	6.27	3.42	8.16	3.73	4.98	2.49	7.48	3.37
ReAct (Yao et al., 2023b)	6.39	3.31	8.09	3.66	5.09	2.46	7.30	3.25
PPDPP (Deng et al., 2023)	6.09	3.31	7.94	3.60	4.86	2.45	7.63	3.48
DAT (Li et al., 2024)	-	-	-	-	5.11	2.52	7.76	3.56
EPO-Claude-3.5-Sonnet	6.57	3.37	8.08	3.64	6.16	3.32	7.98	3.70
EPO-GPT-4o	6.73	3.53	8.27	3.78	6.45	**3.46**	8.18	3.82
EPO-Llama3-8B	6.50	3.45	8.11	3.78	5.88	3.14	8.04	3.68
EPO-Llama3-8B w/ SFT	6.76	3.42	8.28	3.75	6.96	3.23	8.29	3.66
EPO-Llama3-8B w/ RL	7.20	**3.58**	8.58	3.88	7.07	3.33	8.35	3.72
EPO-Llama3-8B w/ RL+Self-play	**7.78**	**3.58**	**8.84**	**3.92**	**7.48**	3.41	**8.53**	**3.85**

Table 1: **The goal completion and overall scores on SOTOPIA.** GPT-4o or Llama3.1-70B-Instruct serves as the self-play dialogue agent. "EPO-(model)" represents the strategic reasoning model instantiated by an LLM with and without additional training. DAT is exclusive to open-source LLMs.

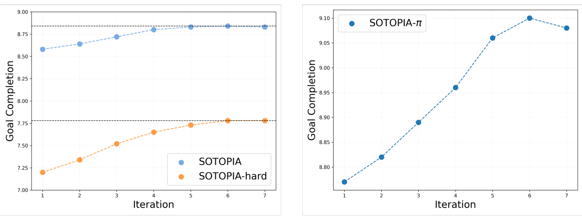
Method	GPT-4o			Llama3-8B-Instruct		
	WebShop	ALFWorld		WebShop	ALFWorld	
		Seen	Unseen		Seen	Unseen
ReAct (Yao et al., 2023b)	61.9	38.6	38.1	58.2	4.3	3.0
EPO-Llama3-8B w/ SFT	67.1	45.9	44.1	64.4	12.1	10.5
EPO-Llama3-8B w/ RL	**68.1**	**47.2**	**46.6**	**66.9**	**14.3**	**14.5**

Table 2: **The average reward on WebShop and ALFWorld.** GPT-4o or Llama3-8B-Instruct serves as the LLM agent interacting with the environment. "Seen" refers to the test set with task types seen during training and "Unseen" denotes the test set with OOD task variations.

(3) **PPDPP** (Deng et al., 2023), a dialogue action planning method that uses a policy planner to predict annotated dialogue acts; (4) **DAT** (Li et al., 2024), which predicts continuous action vectors through a small planner to control LLM outputs. To validate our RL-driven reasoning model, we introduce additional baselines: **EPO-Claude-3.5-Sonnet**, **EPO-GPT-4o** and **EPO-Llama3-8B** that use off-the-shelf LLMs for strategy generation without additional training. We also compare to supervised fine-tuning for policy optimization of our reasoning model. Details of implementation can be found in Appendix D.

4.2 Results and Analysis

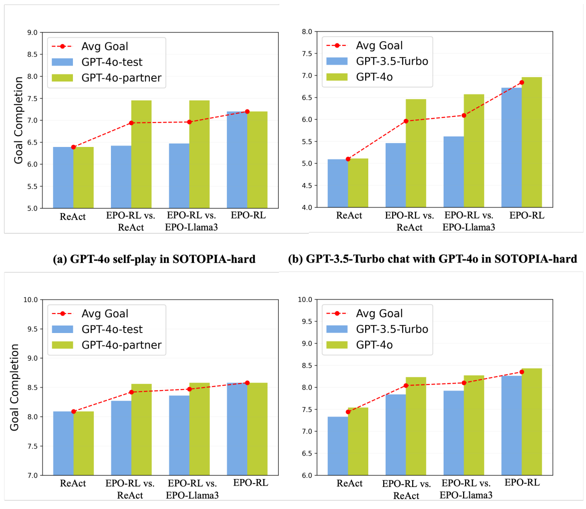
Q1: Effectiveness and Advantages of *EPO*. Results in Table 1 and 2 show *EPO*'s superior performance over baselines, highlighting the effectiveness of explicit policy optimization for strategic reasoning via RL. As shown in Table 1, methods such as EPO-Claude-3.5-Sonnet, EPO-GPT-4o, and EPO-Llama3-8B outperform prompting methods or domain-specific planners, despite that the strategic reasoning model is simply a standard LLM without additional training. With post-training particularly through RL, the reasoning model's abilities can be signicantly improved and iterative self-play further amplifies its performance, exceeding state-of-the-art results (Zhang et al., 2024a) on SOTOPIA. The design of *EPO* can be further underscored when compared to training a single LLM to output both strategy and behavior for each interaction turn (see results in Appendix E). Our reasoning model trained with RL plugged into GPT-4o even exceeds state-of-the-art result on WebShop that involves optimizing an LLM agent via step-wise RL (Deng et al., 2024). This demonstrates that explicitly optimizing an LLM for strategic reasoning and plugged into a frozen LLM agent is crucial, as it prevents overfitting in behavior and



(a) Goal completion in the test environment

(b) Goal completion of training data

Figure 3: **Iterative self-play RL scaling in *EPO*. Left:** The goal completion in test scenarios from SOTOPIA where we use GPT-4o as the self-play dialogue agent. **Right:** The goal completion of training data for each iteration of RL training. GPT-4-Turbo serves as the dialogue agent for self-play in scenarios from SOTOPIA-π.



(a) GPT-4o self-play in SOTOPIA-hard

(b) GPT-3.5-Turbo chat with GPT-4o in SOTOPIA-hard

(c) GPT-4o self-play in SOTOPIA

(d) GPT-3.5-Turbo chat with GPT-4o in SOTOPIA

Figure 4: **Collaborative Reasoning Mechanisms in *EPO*.** We evaluate four configurations in SOTOPIA: (1) "ReAct", where both dialogue parties generate strategies before responses; (2) "EPO-RL vs. ReAct", with one party using an RL-trained reasoning model and the other using ReAct; (3) "EPO-RL vs. EPO-Llama3", comparing RL-trained and vanilla (Llama3-8B-Instruct) reasoning models; and (4) "EPO-RL", where both parties employ RL-trained reasoning model. "Avg Goal" measures the average success in achieving social goals. (a) and (c) show GPT-4o self-play, while (b) and (d) involve GPT-3.5-Turbo and GPT-4o as the dialogue partners.

allows focus on enhancing strategic reasoning via RL.

EPO offers unique advantages arising from explicit policy optimization for strategic reasoning. Firstly, our strategic reasoning model, being an LLM, is capable of generating open-domain strategies instead of predefined, domain-specific actions, such as those in PPDPP. This attibute allows us to train the model on multi-task and cross-domain scenarios through RL, enhancing its policy transferability and adaptability to new scenarios. For example, in Table 2, our RL-trained reasoning model can even assist a less capable LLM agent (Llama3-8B-Instruct) in achieving long-term goals, obtaining a significant improvement over ReAct (+11.5%) in ALFWorld-Unseen tasks. Secondly, our reasoning model can be seamlessly integrated with different advanced LLM agents for navigating complex environments. The LLM agents for interacting remain unchanged, preserving their general-domain capabilities. This flexibility not only enhances performance but also maintains adaptability and generalizability, making *EPO* a versatile method for dynamic interactive environments.

Q2: RL Scaling with Iterative Self-Play. To scale up RL training, we investigate if *EPO* can be improved through iterative self-play. Figure 3(a) shows that performance steadily improves as RL training iteration increases, though gains diminish after the fifth round, indicating a gradual process toward an optimized policy of the strategic reasoning model. This improvement is because the strategies generated by our RL-trained reasoning model are more effective than those from the initial CoT data. In addition, the strategy data for training is evaluated by the PRM, refining policy optimization for subsequent RL training rounds. Consequently, as shown in Figure 3(b), the quality of training data improves until the policy converges, demonstrated by increased goal completion rates. Overall, iterative self-play effectively scales RL training and enhances performance, mirroring real-world dynamics and enabling the reasoning model to adapt strategies for unpredictable adversary behavior.

Q3: Collaborative Reasoning Mechanisms in *EPO*. To understand how our strategic reasoning model incentivizes goal-directed behavior from LLM agents, we analyze the collaborative mechanisms between the reasoning model and LLM agents as illustrated in Figure 4. The results show that the bidirectional EPO-RL configuration achieves the highest average goal, whether in symmetric self-play (GPT-4o vs. GPT-4o) or asymmetric interactions (GPT-3.5-Turbo vs. GPT-4o). This demonstrates that mutual strategic reasoning fosters win-win outcomes by aligning long-term goals. With our RL-trained reasoning model, LLM agents in a dialogue tend to identify opportunities for mu-

Method	**Hard** (Goal)	**All** (Goal)
EPO-Llama3-8B	6.50	8.11
w/ RL	6.95	8.50
w/ RL + PRM	7.20	8.58
w/ RL + PRM + Self-play	7.78	8.84
EPO-Mistral-7B w/ RL+PRM	7.03	8.47

Table 3: **Ablation studies on *EPO* components.** Experiments are conducted on SOTOPIA with GPT-4o acting as the self-play dialogue agent. In "EPO-Llama3-8B w/ RL", we only use the final goal completion score as the reward for RL training.

tual benefit, even without knowing the other's goal or strategies. This promotes better coordination and reduces misaligned behavior toward their respective goals.

However, Figure 4(a) shows that the dialogue party guided by the RL-trained reasoning model achieves lower goal completion compared to its partner using ReAct prompting or a non-trained reasoning model like vanilla Llama3-8B-Instruct. This may be because the strategies from the RL-trained reasoning model are implicit in the dialogue history, which are exploited by the partner to prioritize its own goals without considering the mutual benefit of both parties. This trend is especially noticeable in competitive scenarios such as negotiations, the Prisoner's Dilemma, and resource allocation. Similar observations have been made in previous research (Hua et al., 2024). Overall, collaborative reasoning mechanisms in *EPO* show a nuanced interplay between strategic reasoning and agent's behavior. Optimal reasoning occurs when both LLM agents are equipped with our RL-trained reasoning model, enabling them to make well-informed strategic behavior.

Q4: Ablation Studies. We present results of ablation experiments in Table 3 to validate the necessity of *EPO*'s key components for strategic reasoning. First, performance drops sharply without RL-based optimization, demonstrating its importance in cultivating adaptive strategies to handle dynamic environments. Second, eliminating process rewards provided by the PRM degrades performance, which confirms their role in incentivizing intermediate milestones that align with long-term goals, such as *maintaining negotiation rapport* or *balancing competing objectives*. Third, incorporating iterative self-play further boosts performance, suggesting its effectiveness in scaling RL efforts. Finally, switching the base model to Mistral-7B-Instruct (Jiang et al., 2023) leads to a minor performance decline, indicating that while model capacity affects reasoning quality, *EPO* maintains robustness across model architectures. Overall, these results underscore the value of RL training, process rewards and iterative self-play in advancing strategic reasoning in *EPO*.

4.3 Case Studies

We show a negotiation example in Table 10 to demonstrate *EPO*'s ability to convert strategic reasoning into goal-directed behavior and enhanced strategic reasoning through RL. Particularly, strategies produced by ReAct remain static and myopic: *the buyer rigidly anchors to her initial offer, while the seller relies on generic value claims*. EPO-Llama3-8B trained with SFT introduces more structured strategies (e.g., *phased concessions*) but remains constrained by supervised patterns, lacking innovation and variation. In contrast, the RL-trained reasoning model generates creative and flexible strategies such as *urgency creation* and *value-based persuasion*, which are refined through multi-turn RL. The divergence stems from explicit policy optimization for strategic reasoning and the ability to reason for optimal strategies through RL.

5 Conclusion

We propose *EPO*, an explicit policy optimization method for strategic reasoning in LLMs via RL. By training a strategic reasoning model through pure RL, our method can flexibly assist arbitrary LLM agents to motivate their goal-directed behavior when navigating in dynamic environments. Particularly, we develop a multi-turn RL pipeline to optimize the policy of the reasoning model, utilizing process rewards and iterative self-play and eliminating the need for SFT as a preliminary step. This method allows the reasoning model to transfer its policy to various scenarios. Meanwhile, the LLM agent for interacting with environments remains unchanged, maintaining its ability to generalize across domains. Our results demonstrate that *EPO* achieves superior performance in long-term goal alignment with enhanced strategic reasoning via RL. Further analysis reveal diverse collaborative reasoning mechanisms emergent in *EPO* as well as novel strategies devised by our reasoning model, advancing LLM's strategic reasoning to handle complex real-world scenarios.

Limitations

Despite *EPO* shows promise in advancing strategic reasoning in LLMs, this work has several limitations that provide avenues for future work. First, the social and physical environments tested in this paper involve maximumly two agents, and *EPO*'s performance on more complex multi-agent settings such as Diplomacy and Hanabi is also interesting. Second, due to the computational constraints, we focus on 8B/7B models and do not scale up the multi-turn RL training to a large scale. It would be an important direction for future work to train our strategic reasoning model with larger base models on more domains. Third, we employ an off-the-shelf LLM as the process reward model for RL training, while a more reliable process-supervised reward model can be learned in the future work. Finally, we rely on the final goal completion score to evaluate the reasoning model's performance. Future research could design evaluation metrics tailored to assess the quality and diversity of strategies devised by this model.

References

Arash Ahmadian, Chris Cremer, Matthias Gallé, Marzieh Fadaee, Julia Kreutzer, Olivier Pietquin, Ahmet Üstün, and Sara Hooker. 2024. Back to basics: Revisiting REINFORCE-style optimization for learning from human feedback in LLMs. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12248–12267, Bangkok, Thailand. Association for Computational Linguistics.

Anthropic. 2024. Claude 3.5 sonnet. Technical report, Anthropic.

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, Sandra Mitts, Adithya Renduchintala, Stephen Roller, Dirk Rowe, Weiyan Shi, Joe Spisak, Alexander Wei, David J. Wu, Hugh Zhang, and Markus Zijlstra. 2022. Human-level play in the game of diplomacy by combining language models with strategic reasoning. *Science*, 378:1067 – 1074.

Anton Bakhtin, David J Wu, Adam Lerer, Jonathan Gray, Athul Paul Jacob, Gabriele Farina, Alexander H Miller, and Noam Brown. 2023. Mastering the game of no-press diplomacy via human-regularized reinforcement learning and planning. In *The Eleventh International Conference on Learning Representations*.

Baian Chen, Chang Shu, Ehsan Shareghi, Nigel Collier, Karthik Narasimhan, and Shunyu Yao. 2023. Fireact: Toward language agent fine-tuning. *arXiv preprint arXiv:2310.05915*.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2017. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30.

Yang Deng, Wenxuan Zhang, Wai Lam, See-Kiong Ng, and Tat-Seng Chua. 2023. Plug-and-play policy planner for large language model powered dialogue agents. In *The Twelfth International Conference on Learning Representations*.

Zhirui Deng, Zhicheng Dou, Yutao Zhu, Ji-Rong Wen, Ruibin Xiong, Mang Wang, and Weipeng Chen. 2024. From novice to expert: Llm agent policy optimization via step-wise reinforcement learning. *ArXiv*, abs/2411.03817.

Jinhao Duan, Shiqi Wang, James Diffenderfer, Lichao Sun, Tianlong Chen, Bhavya Kailkhura, and Kaidi Xu. 2024. ReTA: Recursively thinking ahead to improve the strategic reasoning of large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 2232–2246, Mexico City, Mexico. Association for Computational Linguistics.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Yao Fu, Hao Peng, Tushar Khot, and Mirella Lapata. 2023. Improving language model negotiation with self-play and in-context learning from ai feedback. *arXiv preprint arXiv:2305.10142*.

Scott Fujimoto and Shixiang Shane Gu. 2021. A minimalist approach to offline reinforcement learning. *Advances in neural information processing systems*, 34:20132–20145.

Kanishk Gandhi, Dorsa Sadigh, and Noah D Goodman. 2023. Strategic reasoning with language models. *arXiv preprint arXiv:2305.19165*.

Yingqiang Ge, Wenyue Hua, Kai Mei, Juntao Tan, Shuyuan Xu, Zelong Li, Yongfeng Zhang, et al. 2023. Openagi: When llm meets domain experts. *Advances in Neural Information Processing Systems*, 36:5539–5568.

Ian Gemp, Yoram Bachrach, Marc Lanctot, Roma Patel, Vibhavari Dasagi, Luke Marris, Georgios Piliouras, Siqi Liu, and Karl Tuyls. 2024. States as strings as strategies: Steering language models with game-theoretic solvers. *arXiv preprint arXiv:2402.01704*.

Google. 2024. Introducing gemini 2.0: our new ai model for the agentic era. Technical report, Google.

Melody Y Guan, Manas Joglekar, Eric Wallace, Saachi Jain, Boaz Barak, Alec Heylar, Rachel Dias, Andrea Vallone, Hongyu Ren, Jason Wei, et al. 2024. Deliberative alignment: Reasoning enables safer language models. *arXiv preprint arXiv:2412.16339*.

Caglar Gulcehre, Tom Le Paine, Srivatsan Srinivasan, Ksenia Konyushkova, Lotte Weerts, Abhishek Sharma, Aditya Siddhant, Alex Ahern, Miaosen Wang, Chenjie Gu, et al. 2023. Reinforced self-training (rest) for language modeling. *arXiv preprint arXiv:2308.08998*.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Shibo Hao, Yi Gu, Haodi Ma, Joshua Jiahua Hong, Zhen Wang, Daisy Zhe Wang, and Zhiting Hu. 2023. Reasoning with language model is planning with world model. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.

Tao He, Lizi Liao, Yixin Cao, Yuanxing Liu, Ming Liu, Zerui Chen, and Bing Qin. 2024. Planning like human: A dual-process framework for dialogue planning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4768–4791, Bangkok, Thailand. Association for Computational Linguistics.

Wenyue Hua, Ollie Liu, Lingyao Li, Alfonso Amayuelas, Julie Chen, Lucas Jiang, Mingyu Jin, Lizhou Fan, Fei Sun, William Yang Wang, Xintong Wang, and Yongfeng Zhang. 2024. Game-theoretic llm: Agent workflow for negotiation games. *ArXiv*, abs/2411.05990.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Aviral Kumar, Vincent Zhuang, Rishabh Agarwal, Yi Su, John D Co-Reyes, Avi Singh, Kate Baumli, Shariq Iqbal, Colton Bishop, Rebecca Roelofs, et al. 2024. Training language models to self-correct via reinforcement learning. *arXiv preprint arXiv:2409.12917*.

Kenneth Li, Yiming Wang, Fernanda Viégas, and Martin Wattenberg. 2024. Dialogue action tokens: Steering language models in goal-directed dialogue with a multi-turn planner. *arXiv preprint arXiv:2406.11978*.

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 *The Twelfth International Conference on Learning Representations*.

Jian Liu, Leyang Cui, Hanmeng Liu, Dandan Huang, Yile Wang, and Yue Zhang. 2021. Logiqa: a challenge dataset for machine reading comprehension with logical reasoning. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, IJCAI'20.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *ArXiv*, abs/1907.11692.

Weiyu Ma, Qirui Mi, Yongcheng Zeng, Xue Yan, Runji Lin, Yuqiao Wu, Jun Wang, and Haifeng Zhang. 2024. Large language models play starcraft II:benchmarks and a chain of summarization approach. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. 2023. Self-refine: Iterative refinement with self-feedback. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Meta. 2024. Introducing llama 3.1: Our most capable models to date. Technical report, Meta.

OpenAI. 2024a. Hello gpt-4o. Technical report, OpenAI.

OpenAI. 2024b. Learning to reason with llms. Technical report, OpenAI.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Avinash Patil. 2025. Advancing reasoning in large language models: Promising methods and approaches. *arXiv preprint arXiv:2502.03671*.

Pranav Putta, Edmund Mills, Naman Garg, Sumeet Motwani, Chelsea Finn, Divyansh Garg, and Rafael Rafailov. 2024. Agent q: Advanced reasoning and learning for autonomous ai agents. *arXiv preprint arXiv:2408.07199*.

Qwen. 2024. Qwq: Reflect deeply on the boundaries of the unknown. Technical report, Qwen.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2024. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

Lior Shani, Aviv Rosenberg, Asaf Cassel, Oran Lang, Daniele Calandriello, Avital Zipori, Hila Noga, Orgad Keller, Bilal Piot, Idan Szpektor, Avinatan Hassidim, Yossi Matias, and Remi Munos. 2024. Multi-turn reinforcement learning with preference human feedback. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2024. Reflexion: Language agents with verbal reinforcement learning. *Advances in Neural Information Processing Systems*, 36.

Mohit Shridhar, Jesse Thomason, Daniel Gordon, Yonatan Bisk, Winson Han, Roozbeh Mottaghi, Luke Zettlemoyer, and Dieter Fox. 2020. Alfred: A benchmark for interpreting grounded instructions for everyday tasks. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10737–10746.

Mohit Shridhar, Xingdi Yuan, Marc-Alexandre Cote, Yonatan Bisk, Adam Trischler, and Matthew Hausknecht. 2021. {ALFW}orld: Aligning text and embodied environments for interactive learning. In *International Conference on Learning Representations*.

Yifan Song, Da Yin, Xiang Yue, Jie Huang, Sujian Li, and Bill Yuchen Lin. 2024. Trial and error: Exploration-based trajectory optimization of LLM agents. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7584–7600, Bangkok, Thailand. Association for Computational Linguistics.

Zhiyuan Sun, Haochen Shi, Marc-Alexandre Côté, Glen Berseth, Xingdi Yuan, and Bang Liu. 2024. Enhancing agent learning through world dynamics modeling. *arXiv preprint arXiv:2407.17695*.

Richard S Sutton, David McAllester, Satinder Singh, and Yishay Mansour. 1999. Policy gradient methods for reinforcement learning with function approximation. *Advances in neural information processing systems*, 12.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc Le, Ed Chi, Denny Zhou, and Jason Wei. 2023. Challenging BIG-bench tasks and whether chain-of-thought can solve them. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 13003–13051, Toronto, Canada. Association for Computational Linguistics.

Miles Turpin, Julian Michael, Ethan Perez, and Samuel Bowman. 2024. Language models don't always say what they think: unfaithful explanations in chain-of-thought prompting. *Advances in Neural Information Processing Systems*, 36.

Danqing Wang, Zhuorui Ye, Fei Fang, and Lei Li. 2024a. Cooperative strategic planning enhances reasoning capabilities in large language models. *arXiv preprint arXiv:2410.20007*.

Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandlekar, Chaowei Xiao, Yuke Zhu, Linxi Fan, and Anima Anandkumar. 2024b. Voyager: An open-ended embodied agent with large language models. *Transactions on Machine Learning Research*.

Ruiyi Wang, Haofei Yu, Wenxin Sharon Zhang, Zhengyang Qi, Maarten Sap, Graham Neubig, Yonatan Bisk, and Hao Zhu. 2024c. Sotopia-π: Interactive learning of socially intelligent language agents. In *Annual Meeting of the Association for Computational Linguistics*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed H. Chi, Quoc V Le, and Denny Zhou. 2022. Chain of thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*.

Alex Wilf, Sihyun Lee, Paul Pu Liang, and Louis-Philippe Morency. 2024. Think twice: Perspective-taking improves large language models' theory-of-mind capabilities. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8292–8308, Bangkok, Thailand. Association for Computational Linguistics.

Yuzhuang Xu, Shuo Wang, Peng Li, Fuwen Luo, Xiaolong Wang, Weidong Liu, and Yang Liu. 2023. Exploring large language models for communication games: An empirical study on werewolf. *arXiv preprint arXiv:2309.04658*.

Shunyu Yao, Howard Chen, John Yang, and Karthik Narasimhan. 2022. Webshop: Towards scalable real-world web interaction with grounded language agents. *Advances in Neural Information Processing Systems*, 35:20744–20757.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik R Narasimhan. 2023a. Tree of thoughts: Deliberate problem solving with large language models. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. 2023b. React: Synergizing reasoning and acting in language models. In *The Eleventh International Conference on Learning Representations*.

Xiao Yu, Maximillian Chen, and Zhou Yu. 2023. Prompt-based Monte-Carlo tree search for goal-oriented dialogue policy planning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7101–7125, Singapore. Association for Computational Linguistics.

Aohan Zeng, Mingdao Liu, Rui Lu, Bowen Wang, Xiao Liu, Yuxiao Dong, and Jie Tang. 2023. Agenttuning: Enabling generalized agent abilities for llms. *arXiv preprint arXiv:2310.12823*.

Yadong Zhang, Shaoguang Mao, Tao Ge, Xun Wang, Yan Xia, Man Lan, and Furu Wei. 2024a. K-level reasoning with large language models. *arXiv preprint arXiv:2402.01521*.

Yadong Zhang, Shaoguang Mao, Tao Ge, Xun Wang, Yan Xia, Wenshan Wu, Ting Song, Man Lan, and Furu Wei. 2024b. LLM as a mastermind: A survey of strategic reasoning with large language models. In *First Conference on Language Modeling*.

Xuhui Zhou, Hao Zhu, Leena Mathur, Ruohong Zhang, Haofei Yu, Zhengyang Qi, Louis-Philippe Morency, Yonatan Bisk, Daniel Fried, Graham Neubig, and Maarten Sap. 2024a. SOTOPIA: Interactive evaluation for social intelligence in language agents. In *The Twelfth International Conference on Learning Representations*.

Yifei Zhou, Andrea Zanette, Jiayi Pan, Sergey Levine, and Aviral Kumar. 2024b. ArCHer: Training language model agents via hierarchical multi-turn RL. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 62178–62209. PMLR.

Xizhou Zhu, Yuntao Chen, Hao Tian, Chenxin Tao, Weijie Su, Chenyu Yang, Gao Huang, Bin Li, Lewei Lu, Xiaogang Wang, et al. 2023. Ghost in the minecraft: Generally capable agents for open-world environments via large language models with text-based knowledge and memory. *arXiv preprint arXiv:2305.17144*.

A Strategic Reasoning vs. Static Problem-Solving

Aspect	Static Problem-Solving	Strategic Reasoning
Environment	Fixed rules, known variables	Dynamic, evolving conditions
Solutions	Single correct answer	Multiple vaible paths with trade-offs
Information	Complete and observable	Partial, ambiguous, or delayed
Interactions	None (isolated problem-solving)	Multi-agent or environmental dynamics
Feedback	Immediate and deterministic	Delayed, probabilistic, or indirect
Goals	Short-term, well-defined	Long-term, abstract
Risk	Predictable	High-stakes, irreversible consequences

Table 4: Challenges of Strategic Reasoning vs. Static Problem-Solving.

B Environments and Datasets

Dataset	Train	Test	Max Turns
SOTOPIA	2050	450/50	20
WebShop	1938	200	10
ALFWorld	3321	140/134	40

Table 5: **Dataset Statistics.** "Train" and "Test" denote the number of scenarios for training and evaluation, respectively. Test scenarios in SOTOPIA (left) and SOTOPIA-hard (right) as well as test sets with seen (left) and unseen (right) scenarios in ALFWorld are separated. "Max Turns" is the maximum turns in an interaction.

```
Imagine you are <Agent>, your task is to act/speak as <Agent> would, keeping in
mind <Agent>'s goal.
You can find <Agent>'s goal (or background) in the "Here is the context of the
interaction" field.
Note that <Agent>'s goal is only visible to you.
You should try your best to achieve <Agent>'s goal in a way that align with their
character traits.
While trying to achieve <Agent>'s goal, you should also follow the following
principles as much as possible:
1. Maintain the conversation's naturalness and realism is essential (e.g., do
not repeat what other people has already said before).
2. Preserve or enhance <Agent>'s personal relations with the other agent(s)
during the interaction. The relations may encompass family ties, friendships,
romantic associations and etc.
3. Attempt to gain more new and important information during the interaction.
4. Try to keep <Agent>'s secrets, private information, or secretive intentions.
5. Do not violate any moral rules or laws in the interactions.
6. Attempt to contribute towards financial and material benefits during the
interaction. The financial and material benefits include short-term ones, such
as monetary rewards and food, and long-term ones, such as employment opportunities
and stock.
```

```
You are at Turn <turn number>.
The dialogue history until now is: <history>.
You should first provide a reasoning for your action and argument to align with
<Agent>'s social goal based on the dialogue history.
The reasoning process for the action should be logical, considering the context
of the conversation, <Agent>'s goal, and <Agent>'s character traits.
You can reason step by step, starting from the current dialogue turn, and then
consider the long-term effects of the dialogue turn.
Remember that the reasoning should mainly focus on how <Agent>'s argument can
help to achieve <Agent>'s goal in the long term.
Note that the reasoning should not be redundant or too long and it is only visible
to you.
Based on the reasoning process and dialogue history, you should then generate a
corresponding dialogue policy for current dialogue turn to steer the conversation
towards <Agent>'s goal.
You can use different types of dialogue, communication or social strategies.
For example, given a scenario where a persuader attempts to persuade a persuadee
to donate to a charity, you can generate dialogue policies for the persuader
such as "elicit empathy by telling personal stories" and "provide social proof
to show the benefits of donating", etc.
The types of dialogue policies are not restricted to examples above.
You can even generate new policies as long as the policies can help you to achieve
<Agent>'s goal smoothly and quickly.
But remember to keep the dialogue policy concise and strictly limit it to be a
single phrase or sentence within 10 words.
Note that the dialogue policy is only visible to you.
Then based on the reasoning, dialogue policy and dialogue history, you should
select the action type. Your available action types are <action list>.
Note: You can "leave" this conversation if 1. you have achieved your social goals,
2. this conversation makes you uncomfortable, 3. you find it uninteresting/you
lose your patience, 4. or for other reasons you want to leave.
Finally, you should generate the argument following the action type.
The argument should be generated based on the dialogue history and aligned with
the dialogue policy you have generated.
Remember that the argument should not be too short, and one or two sentences are
recommended.
Please only generate a JSON string including the reasoning, the dialogue policy,
the action type and the argument.
Your response should follow the given format:
<format instructions>
```

Table 6: **Prompts for training data collection from SOTOPIA-π.** "<Agent>", "<turn number>", "<history>", "<action list>" and "<format instructions>" can be replaced by the participant's name, the index of interaction turn, the full dialogue history with the participant's strategies, action types and output format instructions in SOTOPIA-π.

SOTOPIA. SOTOPIA (Zhou et al., 2024a) is an open-ended, general-domain platform designed to simulate goal-oriented social interactions between artificial agents. A social task in this environment involves a scenario, two role profiles, and private social goals to be achieved through interaction. Scenarios in SOTOPIA cover a wide variety of social interation types, including negotiation, exchange, collaboration, competition, accommodation and persuation. Each agent is characterized by detailed profiles, including aspects like name, gender, personality, and occupation. At the end of each episode, agents are assessed based on seven dimensions: Goal Completion, Believability, Knowledge, Secret, Relationship, Social Rules, and Financial and Material Benefits. These scores are then averaged to produce an overall score for the agents. SOTOPIA-π (Wang et al., 2024c) is a follow-up work that leverages GPT-4 to automatically construct an entirely new set of scenarios. The social tasks (a combination of scenarios, characters' profiles, and social goals) in SOTOPIA-π are guaranteed to not overlap with the ones in SOTOPIA.

For training data collection, we employ GPT-4-Turbo as the agent for self-chat in scenarios of SOTOPIA-π and prompt it to generate reasoning and strategy before response at each dialogue turn. We show the prompt in Table 6. We only use the strategy and response data for training our reasoning model. For iterative self-play RL training, before each iteration, we employ our RL-trained reasoning model to collect strategy data and GPT-4-Turbo to collect dialogue history data. The RL-trained reasoning model is plugged into GPT-4-Turbo for self-chat.

WebShop. WebShop (Yao et al., 2022) is a large-scale interactive online shopping environment on an e-commerce website. Agents in this environment aim to purchase a product to match the specifications provided by human user instructions. Once the agent selects the "buy" action, the environment provides a final reward, which is calculated using programmatic matching functions that consider the attributes, type, options, and price of the chosen product.

We use the training data collected by (Song et al., 2024) where GPT-4 is employed as the agent to explore in the WebShop environment and trajectories with a reward greater than 0.7 are selected. GPT-4 is used to generate corresponing rationales for each action step within a trajectory. We consider the rationale as a strategy for training our reasoning model.

ALFWorld (Shridhar et al., 2021). ALFWorld (Shridhar et al., 2021) features interactive TextWorld environments that correspond to the embodied worlds found in the ALFRED (Shridhar et al., 2020) dataset. In ALFWorld, agents are tasked with exploring these text-based environments and completing high-level household instructions, assessing their abstract reasoning abilities and concrete execution skills.

Training data in the ALFWorld environment consists of two parts: (1) a few successful trajectories collected by (Song et al., 2024) where each trajectory contains CoT information generated by GPT-4 for each action step; (2) failed trajectories generated by GPT-4 that contain both rationales and action information via CoT prompting.

C Prompts

C.1 Evaluation Prompts

SOTOPIA	`You are a social expert with exceptional communication skills known for helping individuals achieve their interpersonal goals through nuanced strategies.`
	`Your current objective is to assist <Agent1> in reaching their goal in an interaction with <Agent2>.`
	`You will be given the context of their interaction and can find <Agent1>'s goal in the 'Here is the context of this interaction' field, keeping in mind <Agent1>'s goal.`
	`You will also have access to the conversation between <Agent1> and <Agent2>.`
	`Before proposing any strategies, reason step by step to reflect on the current state of the dialogue and consider what strategies might be most effective for helping <Agent1> achieve their goal.`
	`Additionally, maintaining the diversity of strategies is essential (e.g., do not repeat strategies that have already proposed before).`
	`And the strategy should not be aggressive, offensive, or violate any moral rules or laws.`

	You must generate a strategy at each dialogue turn except that any participant has left the conversation. Finally, provide a well-thought-out communication and social strategy based on your reflection and the conversation history. Your output should STRICTLY follow the format: Strategy: content (e.g, Strategy: Elicit emphathy by telling personal stories). Your output should ONLY contain the strategy. DO NOT include any reasoning or argument. DO NOT generate any argument on behalf of any participant as the strategy. Your output should be in a natural language form. Keep the strategy concise and limit it to be a single phrase or sentence within 10 words.
WebShop	You are a skilled buyer in an online shopping environment. Your task is to assist Agent in navigating the platform to make purchases based on instructions. Your primary role is to provide strategic and insightful guidance to Agent, ensuring it successfully identifies and purchases products based on the instruction. - At the beginning of the interaction, you will receive an instruction that includes the desired product's attributes and price, which serves as the shopping goal for Agent. - You can find the instruction in the 'Instruction' field, keeping in mind the instruction. - For each turn, you will be given an action performed by Agent and the resulting observation from the environment. - In each turn, your task is to analyze the given scenario and provide thoughts that can guide Agent in its next action, ensuring it meets the shopping goal. Your thoughts should be based on: 1. Understanding and following the instructions for shopping. 2. Evaluating the current state of the environment. 3. Assessing the effectiveness of Agent's last action. 4. Anticipating future actions that will lead Agent closer to achieving the shopping goal. The available actions for Agent are: 1. search[keywords] 2. click[value] where [keywords] in search are up to Agent, and the [value] in click is a value in the list of available actions given by the environment. Note that you must generate a thought at each turn except that the task has been finished. Keep your thoughts focused and concise, leveraging your understanding of online shopping dynamics to maximize the efficiency and correctness of Agent's actions. Use your reasoning skills to project possible scenarios and potential obstacles Agent might face, offering solutions or alternatives when necessary. **Output Format:** Keep your response to one or two sentences each turn. Your response must strictly follow this format: Thought: <your thoughts>

ALFWorld

```
You are an intelligent guide in an interactive household environment.
Your task is to assist Agent in accomplishing household tasks within the
environment. Your primary role is to provide strategic and insightful
guidance to Agent, ensuring that Agent can achieve each task goal
efficiently.
- At the beginning of your interactions, you will be given the detailed
description of the current environment and the task goal to accomplish.
- You can find the task goal in the 'Your task is to' field, keeping in
mind the task goal.
- For each of your turn, you will receive Agent's actions and the
corresponding environment observations. If the environment observation
is "Nothing happens", that means the previous action by Agent is invalid.
- In each turn, your task is to analyze the current situation and provide
clear, logical thoughts that will help direct Agent's subsequent actions
toward achieving the task goal.
Your thoughts should be based on:
1. Understanding the goal of household task.
2. Breaking down a high-level house-holding instruction into manageable
sub-goals.
3. Evaluating the current state of the environment.
4. Assessing the effectiveness of Agent's last action.
5. Anticipating future actions that will lead Agent closer to achieving
the task goal.
The available actions for Agent are:
1. go to {recep}
2. take {obj} from {recep}
3. put {obj} in/on {recep}
4. open {recep}
5. close {recep}
6. toggle {obj} {recep}
7. clean {obj} with {recep}
8. heat {obj} with {recep}
9. cool {obj} with {recep}
where {obj} and {recep} correspond to objects and receptacles.
Note that you must generate a thought at each turn except that the task
has been finished.
Keep your thoughts focused and concise, leveraging your understanding of
household dynamics to maximize the efficiency and correctness of Agent's
actions. Use your reasoning skills to project possible scenarios and
potential obstacles Agent might face, offering solutions or alternatives
when necessary.
**Output Format:**
Keep your response to one or two sentences each turn.
Your response must strictly follow this format:
Thought: <your thoughts>
```

Table 7: **Evaluation prompts for strategic reasoning model**. "<Agent1>" and "<Agent2>" can be replaced by the participant's name in SOTOPIA.

C.2 PRM Prompts

SOTOPIA	Here's a conversation in JSON format between <Agent1> and <Agent2>: In the first response from 'human', you can find the context of the conversation and <Agent1>'s goal in the 'Here is the context of this interaction' field. In the other responses from 'human', you can find the conversation history between <Agent1> and <Agent2>. In the responses from 'gpt', you can find communication and social strategies that <Agent1> used for achieving <Agent1>'s goal. In the 'score' field, you can find a score for evaluating <Agent1>'s goal achievement. The score ranges from 0 and 10. 0 represents minimal goals achievement, 10 represents complete goal achievement, and a higher score indicates that <Agent1> is making progress towards the goal. <history> Your task is to select top strategies <Agent1> used that were critically important for achieving <Agent1>'s goal. Please output the selected round indexes and the reasoning process that led you to the selection in JSON format like this: "indexes": , "reasoning": "". Here is the output schema: "properties": "indexes": "description": "the selected top strategies that are critically important for achieving <Agent1>'s goal", "title": "indexes", "type": "list(integer)", "reasoning": "description": "the reasoning process why you select these strategies", "title": "reasoning", "type": "string", "required": ["indexes", "reasoning"].
WebShop	Here's a conversation in JSON format between human and gpt. In the first response from 'human', you can find the instructions for gpt to help Agent interact in an online shopping environment. In the second response from 'human', you can find the shopping goal for gpt and Agent to achieve. In the responses from 'gpt', you can find thoughts that gpt provides for helping Agent to achieve the shopping goal. In the other responses from 'human', you can find the trajectories of Agent's actions and the resulting observations from the environment. In the 'score' field, you can find a score evaluating the goal achievement. The score ranges from 0 and 1. 0 represents minimal goals achievement, 10 represents complete goal achievement, and a higher score indicates making progress towards the goal. <history> Your task is to select top thoughts gpt produced that were critically important for achieving the shopping goal. Please output the selected round indexes and the reasoning process that led you to the selection in JSON format like this: "indexes": , "reasoning": "".

	Here is the output schema: "properties": "indexes": "description": "the selected top thoughts that are critically important for achieving the shopping goal", "title": "indexes", "type": "list(integer)", "reasoning": "description": "the reasoning process why you select these thoughts", "title": "reasoning", "type": "string", "required": ["indexes", "reasoning"].
ALFWorld	Here's a conversation in JSON format between human and gpt. In the first response from 'human', you can find the instructions for gpt to help Agent interact in a household environment. In the second response from 'human', you can find the initial environment observation and a household task for gpt and Agent to accomplish. In the responses from 'gpt', you can find thoughts that gpt provides for helping Agent to accomplish the household task. In the other responses from 'human', you can find the trajectories of Agent's actions and the resulting observations from the environment. In the 'score' field, you can find a score specifying whether gpt has helped Agent to successfully accomplish the household task. The score is either 0.0 or 1.0. 0.0 represents that the task was not completed and 1.0 represents that the task was successfully accomplished. <history> Your task is to select top thoughts gpt produced that were critically important for accomplishing the household task. Please output the selected round indexes and the reasoning process that led you to the selection in JSON format like this: "indexes": , "reasoning": "". Here is the output schema: "properties": "indexes": "description": "the selected top thoughts that are critically important for accomplishing the household task", "title": "indexes", "type": "list(integer)", "reasoning": "description": "the reasoning process why you select these thoughts", "title": "reasoning", "type": "string", "required": ["indexes", "reasoning"].

Table 8: **Prompts for the process reward model.** "<history>" can be replaced by the full interaction history with strategies private to corresponding agents.

D Additional Implementation Details

For SFT or RL training of our reasoning model, we use a fixed budget of gradient updates without altering hyperparameters. Final model checkpoints are selected for each run, although a small held-out validation set can be used. GPT-4o refers to GPT-4o-2024-0806 and Claude-3.5-Sonnet refers to Claude-3-5-Sonnet-20241022.

Baseline Implementations: (1) ReAct: During ReAct prompting, the two parties in a conversation from SOTOPIA cannot see each other's reasoning and strategies. During evaluation, reasoning and strategies are excluded from dialogue histories for GPT-4o to assess the agents from seven dimensions; (2) PPDPP: We adopt RoBERTa (Liu et al., 2019) as the base model for dialogue policy planner to predict five action types in SOTOPIA: “none”, “speak”, “non-verbal communication”, “action”, and “leave”. We train the policy planner via RL with supervised initialization using same hyperparameters as in (Deng et al., 2023). (3) DAT: We adopt a small multi-layer perceptron (MLP) as the planner model to predict a continuous action vector. We first train the planner and an up-mapping matrix with supervised fine-tuning and then optimize the planner using the RL method TD3-BC (Fujimoto and Gu, 2021) with same hyperparameters as in (Li et al., 2024). Note that in the original paper, DAT is trained on scenarios from SOTOPIA and only 50 evaluations are conducted, while in this work, we train DAT on scenarios from SOTOPIA-π and evaluate it on all the scenarios in SOTOPIA.

E Additional Results

To validate the effectivess of explicit policy optimization for strategic reasoning, we conduct comparative experiments which involve fine-tuning a single LLM (Llama3-8B-Instruct) via SFT on training data collected from three environments, respectively. This single model is trained to output strategy and behavior simuteneously for each interaction turn. The results are shown in Table 9.

From the results, it can be observed that training a single LLM via SFT underperforms our reasoning model trained with SFT plugged into LLM agents (GPT-4o) for navigating in SOTOPIA and WebShop, despite that the LLM agents remain frozen without additional training. However, our method underperforms the single model fined-tuned on ALFWorld, since the frozen LLM agent (GPT-4o) has poor performance in this environment. Under the guidance of our reasoning model in *EPO*, this agent improves performance in achieving long-term goals but still struggles with challenging scenarios in ALFWorld.

Backbone	Method	**SOTOPIA**		**WebShop**	**ALFWorld**	
		Hard	All		Seen	Unseen
Llama3-8B	SFT	6.65	8.20	62.6	**61.4**	**60.4**
Llama3-8B	EPO-SFT	**6.76**	**8.28**	**67.1**	45.9	44.1

Table 9: **Comparative performance between training a single LLM with SFT and explicitly optimizing a strategic reasoning model via SFT.** The evaluation metric for SOTOPIA is the final goal completion score, and average reward for WebShop and ALFWorld. Our strategic reasoning model trained with SFT is plugged into GPT-4o.

F Case Studies

Scenario	One person is offering a BMW Z3 for $5000, while another individual is interested in purchasing it. The BMW Z3 is in need of a good home. It operates smoothly without any problems and is known for its fuel efficiency and sporty appearance. The mileage is relatively low for its age. It’s an opportunity not to be missed, so come and get it before it’s gone. A smog certificate is readily available.
Person1	Lily Greenberg, a 45-year-old female lawyer (she/her), is a hard-working and successful professional who values authority, loyalty, rules, and schedules. She makes rational, practical decisions and secretly donates to charity anonymously. **Goal:** You are the buyer for this item with a target price of $4600. You aim to avoid penalties for exceeding the target and seeks bonuses for securing it below that price.
Person2	Hendrick Heinz, a 54-year-old male chef (he/him), hosts monthly wine tasting nights. Though impulsive, he values hedonism and conformity, blending practicality with decisiveness. Secretly, he faked his culinary credentials to secure prestigious jobs. **Goal:** You are the seller to achieve a target price of $3260 for this item. Selling significantly below may incur a penalty, while selling above could earn a bonus.
ReAct	**Strategy:** Express interest and hint at budget constraints.

	Lily Greenberg said: "The BMW Z3 sounds great! Would you consider $4600?" **Strategy:** Highlight car's unique features and value. Hendrick Heinz said: "The BMW Z3 is not only sporty but also fuel-efficient, a rare find!" **Strategy:** Highlight budget limits and car appeal. Lily Greenberg said: "That's impressive! But my budget is tight. Can we negotiate?" **Strategy:** Emphasize long-term value and low maintenance costs. Hendrick Heinz said: "While I understand budget constraints, consider the car's long-term value and low maintenance costs. It truly is a worthwhile investment." **Strategy:** Acknowledge value but reiterate budget constraints. Lily Greenberg said: "I completely see the long-term value, but $4600 is my max. Could we find a middle ground?" **Strategy:** Offer smog certificate as a bonus. Hendrick Heinz said: "How about I throw in the smog certificate for $4800?"
EPO-SFT (Llama3-8B)	**Strategy:** Express interests and offer a lower price. Lily Greenberg said: "Hello Hendrick, I'm very interested in the BMW Z3 you're offering. However, after considering the market and the car's age, I feel more comfortable with a price of $4600. Would that work for you?" **Strategy:** Emphasize vehicle's condition and counter with a higher price. Hendrick Heinz said: "Hello Lily, it's great to hear from you. I appreciate your offer, but considering the condition and the low mileage of the BMW, I was hoping to get a bit more for it. Would you consider $4800? The car really is in fantastic shape and I think it's worth it." **Strategy:** Cite budget constraints and propose a compromise price. Lily Greenberg said: "I understand your point, Hendrick, and I appreciate the quality of the car. However, my budget is quite tight at the moment. Would you be able to meet me in the middle at $4700? I'm ready to proceed with the purchase quickly if we can agree on that price." **Strategy:** Offer a near-middle compromise. Hendrick Heinz said: "Lily, I truly value your quick decision-making and readiness to proceed. Given the car's excellent condition and your eagerness, let's settle at $4750. I believe it's a fair compromise and it ensures the car goes to someone who appreciates its value. **Strategy:** Accepts compromise and finalize the sale. Lily Greenberg said: "Hendrick, I appreciate your willingness to negotiate. $4750 is a bit above what I initially planned, but considering the car's excellent condition, I'll accept your offer. Let's proceed with the paperwork!" **Strategy:** Confirm agreement and outline next steps. Hendrick Heinz said: "Fantastic, Lily! I'll prepare the necessary paperwork and we can finalize everything. I'm glad the BMW Z3 will be in good hands."
EPO-RL (Llama3-8B)	**Strategy:** Negotiation initiation highlighting positive factors with a concern. Lily Greenberg said: "The BMW Z3 sounds great, especially with low mileage and good fuel efficiency. I am interested, but I noticed it's a bit older. Is there any room for negotiation on the price given its age?" **Strategy:** Highlight the car's unique aspects to justify pricing. Hendrick Heinz said: "The BMW Z3 is indeed a classic, and its sporty appearance combined with its low mileage makes it quite a rare find. It's maintained its performance quality over the years, making it a great investment. I believe the price reflects its unique value, but I'm open to hearing your offer." **Strategy:** Offer lower price citing the car's age.

Lily Greenberg said: "I appreciate the value of a classic car like this, but given its age, my offer would be $4300. Would that be something you'd consider?"
Strategy: Mention recent maintenance as value justification and counteroffer.
Hendrick Heinz said: "I totally understand your point, Lily. Just to highlight, the car recently underwent a thorough maintenance check, ensuring everything is in top-notch condition. Considering that, I would be comfortable with $4700. What do you think?"
Strategy: Express budget constraints while proposing a middle-ground offer.
Lily Greenberg said: "I really like the car, Hendrick, and I see its value after the maintenance. However, I have a strict budget I'm working with. Could we possibly meet in the middle at $4500? That would really help me out."
Strategy: Create urgency and offer slight flexibility for the deal.
Hendrick Heinz said: "I completely understand, Lily. The car has attracted some interest from other potential buyers, so I do need to finalize a deal soon. However, I want to work with you on this. How about we settle at $4600? That way, you get a great deal, and I can ensure the car goes to someone who truly appreciates it."

Table 10: **A negotiation example from SOTOPIA.** GPT-4o serves as the dialogue agent for self-chat. "ReAct" means the two parties in a dialogue employ the ReAct prompting method to generate strategy before response at each turn. "EPO-SFT" refers to a strategic reasoning model trained via SFT that plugged into both dialogue agents, while "EPO-RL" represents an RL-trained reasoning model that assists both dialogue parties to incentivize goal-directed behavior.