Evolution of Cooperation in LLM-Agent Societies: A Preliminary Study Using Different Punishment Strategies

Kavindu Warnakulasuriya[1][0009−0002−2462−147X], Prabhash Dissanayake[1][0009−0002−9192−3013], Navindu De Silva[1][0009−0009−9190−4753], Stephen Cranefield[2][0000−0001−5638−1648], Bastin Tony Roy Savarimuthu[2][0000−0003−3213−6319], Surangika Ranathunga[3][0000−0003−0701−0204], and Nisansa de Silva[1][0000−0002−5361−4810]

[1] University of Moratuwa, Moratuwa, Sri Lanka
{kavinduw.20, prabhash.20, navindu.20, NisansaDdS}@cse.mrt.ac.lk
[2] University of Otago, Dunedin, New Zealand
{stephen.cranefield, tony.savarimuthu}@otago.ac.nz
[3] Massey University, Auckland, New Zealand
S.Ranathunga@massey.ac.nz

Abstract. The evolution of cooperation has been extensively studied using abstract mathematical models and simulations. Recent advances in Large Language Models (LLM) and the rise of LLM agents have demonstrated their ability to perform social reasoning, thus providing an opportunity to test the emergence of norms in more realistic agent-based simulations with human-like reasoning using natural language. In this research, we investigate whether the cooperation dynamics presented in Boyd and Richerson's model persist in a more realistic simulation of the diner's dilemma using LLM agents compared to the abstract mathematical nature in the work of Boyd and Richerson. Our findings indicate that agents follow the strategies defined in the Boyd and Richerson model, and explicit punishment mechanisms drive norm emergence, reinforcing cooperative behaviour even when the agent strategy configuration varies. Our results suggest that LLM-based Multi-Agent System simulations, in fact, can replicate the evolution of cooperation predicted by the traditional mathematical models. Moreover, our simulations extend beyond the mathematical models by integrating natural language-driven reasoning and a pairwise imitation method for strategy adoption, making them a more realistic testbed for cooperative behaviour in MASs.

Keywords: Multi-Agent Systems · Large Language Model · LLM Agent · Social Dilemmas · Agent Strategies

1 Introduction

Autonomous agents have gained significant popularity due to their immense utility in various real-world applications in customer service, healthcare, social networks, and retail domains [34]. Multi-agent systems (MASs) bring together agents with independent objectives and decision-making abilities that interact within a shared environment. As such, agents must cooperate and coordinate their actions in dynamic and often unpredictable settings [32]. Cooperative agents can help improve the performance of individual agents and the overall system [17]. Researchers have sought to understand how cooperation emerges in societies and have presented various mathematical models and simulations that predict agent behaviours [3, 6]. However, the suitability of these models for real-world, human-oriented environments remains uncertain. With the rise of alternative AI technologies, traditional rule-based decision-making approaches are being challenged, necessitating further investigation [24].

Mathematical approaches such as game-theory models and evolutionary dynamics are commonplace approaches to modelling and predicting agent behaviour [10, 23]. Often, these models rely on simplified abstractions, particularly in social dilemmas such as the Prisoner's Dilemma [16] and the n-player diner's dilemma [36], which focus on agent cooperation. On the other hand, social norms are crucial in guiding agents towards cooperative standards [31]. Ensuring adherence to these norms requires punishment to serve as reinforcement for cooperative agents and to penalise defectors [3, 41].

Boyd and Richerson's (B&R) model [6], a prominent simulation study of the evolution of cooperation, suggests that punishment-based mechanisms can sustain long-term cooperation. While being a simple abstract mathematical simulation, its applicability in a more realistic human-based environment remains an underexplored area. With the opportunity of using Large Language Model (LLM) agents, which demonstrate a great understanding of natural language [32], our approach consists of focusing on the diner's dilemma. We explore different strategy compositions introduced by the B&R model to examine whether a more realistic simulation of the evolution of cooperation using LLM agents produces similar norm emergence as the abstract B&R model [6].

We model a realistic n-player diner's dilemma, using LLM Agents as the backbone in making the complex dilemma decisions and allowing them to act based on their strategies, calculate payoffs for their dilemma actions, and finally reflect on their actions, analyze other agents, and change their strategies by comparing their utilities. Furthermore, following the B&R model, we allow the agent population to have multiple strategies in each simulation (currently up to four strategies per simulation). The B&R model includes experiments with both two and three strategies, but their mathematical model of population dynamics is challenging to extend beyond a small number of strategies. Furthermore, the meaning of the B&R strategies is directly encoded within their equations. In contrast, the LLM-based approach allows reasoning about strategies using natural language, making it easier to introduce and test alternative strategies in future studies. Therefore, in this study, we choose to allow different combinations of four strategies in the population, making the simulation more complex and realistic, enabling a deeper analysis of the strategy evolution.

In summary, we aim to investigate the impact of strategies from B&R's work and their evolution with repeated diner's dilemma scenario, modelled with novel LLM agents, which have been shown to implicitly capture human reasoning and thinking abilities, thus, allowing us to gain insights on whether these LLM Agents behave similarly as shown in the abstract mathematical simulation studies.

The structure of the remainder of this paper is outlined as follows: Section 2 reviews the related works. Section 3 elaborates on our proposed methodology approach to model the implementation for the emergence of cooperative agent behavioural strategies. Sections 4 and 5 highlight the preliminary experiments conducted and the results obtained, with a discussion. Finally, Section 6 concludes the paper, along with future directions for research.

2 Related Works

2.1 Game Theory and Social Dilemmas

Over the years, agent behaviour has been studied through mathematical analysis of dynamics or computational simulations of evolutionary dynamics [3, 6, 22, 41]. Such models have limitations because they do not explicitly map to a real-world task or scenario and generalize agent interactions in a fixed structure. In other words, they are limited to simple abstractions. These studies use game theory as a framework to model human decision-making in a highly abstract form.

Social dilemmas are an agent-related strategic concept that motivates the study of the normative behaviour of agents in MASs. A social dilemma occurs when an agent is forced to choose between actions that maximize their personal gain at the expense of the group's collective benefit or actions that promote the collective good but lower their personal benefits [16]. This scenario is essentially a conflict between personal and social optimality. As a result, agents who aim to satisfy their short-term self-interests are often characterized as non-cooperative as they are less likely to choose actions that serve the long-term benefit of the group. These social dilemmas can be explained through game theory, which analyzes how rational agents make decisions when faced with interactions with individuals in competitive or cooperative game environments. Game theory helps identify vital insights in explaining the behaviour of agents under economic, political and social interaction [4, 7].

A dilemma requires the agent or individual to decide whether to *cooperate* with or to *defect* against its opponent. Based on these decisions, four main payoff values are defined, as elaborated by Macy and Flache [21]: *(i) reward (R)*, given when both agents choose to cooperate, *(ii) punishment (P)*, incurred when both agents defect, *(iii) temptation (T)*, where agent defects and unfairly benefits while the opponent cooperates, and *(iv) sucker cost (S)*, where a cooperating agent suffers a loss when the opponent defects. These payoff values form the foundation of various social dilemmas studied in game theory, such as the Prisoner's Dilemma, the chicken game, stag hunt [34] and the trust game [20]. These two-player social dilemmas can be redesigned into their n-player form [14, 20], which closely relates to real-life examples [22]. In the Diner's Dilemma [36], a group of agents agrees to split the cost of their meals. However, individual agents may exploit this arrangement by ordering expensive items and transferring a portion of their costs onto the group.

Social dilemmas create a precarious position for norm emergence in multi-agent societies as agents would look to increase their utility through defection and benefiting from the cooperation of others [3, 6]. Furthermore, agents would be less likely to cooperate towards a collective goal when it is known that other agents would contribute, such as in the public goods game studied in [41]. This is known as the *free-rider problem* [35]. Hence, a suitable mechanism is necessary to ensure agents do not exploit cooperative behaviours, thereby discouraging cooperation.

2.2 Metanorms

With the risk of social dilemmas causing agents to be self-centred without regret or guilt, a higher-order mechanism needs to be in place. **Metanorms**, first coined by Axelrod [3], are second-order norms that guide agents in responding to norm violations to enforce them among defectors. These will enforce penalties on non-cooperative agents. Thereby, they aim to eliminate the free-rider problem in social dilemmas. There are two main approaches for implementing metanorms in agent simulations, namely, punishment-based [3, 6, 22] and indirect reciprocity [26, 27, 30].

Punishment-based implementations [3] describe how to enhance norm compliance by punishing norm violators and those who fail to punish violators, treating non-punishment as a defection against the multi-agent community. In norm-based models, the establishment of norms relies on agents willing to enforce compliance, as insufficient enforcement and free-riding problems can lead to norm collapse [22]. Here, punishment incurs a cost to the punisher and a larger cost to the punished agent, ensuring that punishments are not applied discriminately. Works by Axelrod [3] and Boyd and Richerson [6] have identified that the population must maintain a population of *punisher* agents to prevent norm violators from overtaking cooperative populations.

Indirect reciprocity is an alternate approach for implementing metanorms without compelling agents to punish and reduce their short-term utility. Generally, indirect reciprocity relies on reputation scores and information-sharing mechanisms like public and private reputation framework [30] or gossip [25]. In this initial study, we focus on the punishment-based approach outlined in the B&R model, which serves as the foundation for our research.

2.3 LLM Agents

Despite extensive research on the effects of metanorms and punishment mechanisms to induce cooperation, limited work has been conducted in conjunction with language models. In the few works applying LLMs to reasoning about social dilemmas, LLMs struggle with such interactions—GPT-4[4] has been shown to select actions that maximize its personal gain and fails to coordinate with fellow agents in games such as the Battle of the Sexes [2] and frequently selects uncooperative actions that harshly penalize minor mistakes by opponents. Therefore, although LLMs exhibit strong alignment with human behaviour, they struggle to achieve the high levels of cooperation seen in real-world human interactions. This limitation suggests that LLMs should be carefully evaluated when integrated into social experiments [9].

Recently, Fontana et al. [10] provided insights into the capabilities of LLM agents in handling iterated Prisoner's Dilemma games. Using the Llama-2-70B-chat model[5], it was reported that while the LLM did not display defection initially, it required more iterations to achieve a cooperative majority—demonstrating slower convergence towards cooperation. They found how the defection rate of an agent's opponent also impacts its behaviour.

Representing social dilemmas for LLM agents requires thorough analysis. Traditional multi-agent models use mathematical frameworks such as payoff matrices and cost-benefit values to predict agent behaviour as seen in [3, 6]. However, the weak arithmetic capabilities of LLMs may affect LLM agents' effective comprehension and processing of such payoff constraints [40]. One approach was to provide the LLM with the payoffs for each two-player scenario through the prompts as sentences [2, 10]. However, with no metanorm and norm implementations within these simulations, LLM agents often do not engage in cooperative strategies due to low repercussions for exhibiting defection [22].

Normative multi-agent system researchers have started to investigate the capability of LLMs in norm discovery, reasoning and conformance [32]. The work of He et al. [13] investigated the ability of three LLMs (Llama 2 7B[6], Mixtral 7B[7] and ChatGPT-4) to identify norm violations and reported their promise. The work of Haque and Singh [12] demonstrates the promise of ChatGPT in extracting norms from contracts without requiring training or fine-tuning of datasets. However, none of the prior works have investigated the capability of LLMs to promote agents to adopt and imitate the cooperative behavioural strategies seen by other agents.

2.4 Agent Simulations

Simulations are commonly utilised to explore AI agent behaviour within a virtual environment [38]. These frameworks use research from multiple fields, such as social sciences, psychology, economics and AI for understanding social phenomena. To study social behaviour in group settings, it is vital to develop simulations that closely depict human activities and track changes in the world states, such

[4] https://openai.com/index/gpt-4

[5] https://huggingface.co/meta-llama/Llama-2-70b-chat-hf

[6] https://huggingface.co/meta-llama/Llama-2-7b

[7] https://huggingface.co/mistralai/Mixtral-8x7B-v0.1

as movements of objects resulting from agent actions [40]. These simulations are broadly categorized as *task-based* or *social interaction-based* simulations. Task-based simulations [11], such as the **ScienceWorld** environment, are used for conducting science experiments using a text-based framework [15, 39], whereas social simulations, such as **Melting Pot** [1], use multi-agent reinforcement learning environments. The use of LLM agents in simulations provides a more realistic approach in replicating normative behaviours [19, 32], as seen in frameworks like **AgentVerse** [8], but with representation of the objectives in an abstract manner rather than using specific social scenarios. Game engines like **ALFWorld** [33] and **Watch-And-Help** [29] provide detailed environmental control but are often complex to modify and lack seamless LLM agent integration. Meanwhile, a sandbox environment, **Smallville**[8], provides an interactive and customizable LLM agent-driven simulation, which has been utilized to build agent societies [28] and also normative frameworks [31]. Thus, we consider that this provides a promising simulation framework for experimenting with the work by B&R in a realistic manner.

The B&R model has identified that for cooperation in an n-player system to be maintained, there should be a sufficient number of "Moralist Agents" in the system [6]. However, again, this is simulated in an abstract mathematical way, which is a limitation we intend to address by simulating with LLM agents with a realistic diner's dilemma scenario.

In summary, to the best of our knowledge, no papers have studied the evolution of cooperation with the strategies introduced in the B&R work using LLM agents. Although some articles have explored dilemmas, such as the Prisoner's Dilemma with LLMs as well as other strategies, there is a lack of research on exploring the evolutionary aspect of the cooperation strategies within a more realistic, n-player dilemma scenario to infer insights on whether it produces similar norm emergence as in the abstract B&R work.

3 Methodology

This section outlines the methodology employed to investigate the evolution of cooperation with the Diner's Dilemma scenario implemented through LLM agents. We used the strategy descriptions from the B&R model, leveraging their provided English descriptions alongside the mathematical formulations to guide the LLM's behaviour, as described in the following section.

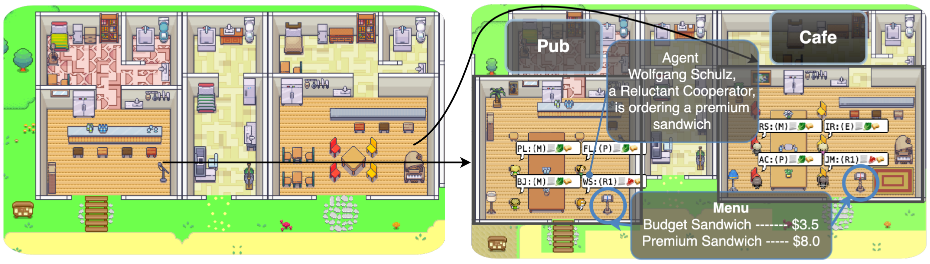
3.1 Simulation Environment Setup

The simulation framework was adapted from the existing Smallville environment utilized in the CRSEC framework [31][9] as it is a promising framework for experimenting with the work of B&R in a realistic manner, as explained in

[8] https://github.com/nickm980/smallville

[9] https://github.com/sxswz213/CRSEC

Section 2.4. The framework can be utilized to model realistic social dilemmas, specifically, The Diner's Dilemma, as illustrated in Figure 1. The virtual environment was designed using the Tiled Map Editor[10] for layout and Phasor[11] for agent movement, creating two primary settings: a pub and a cafe, where the agents interact. A total of eight agents were introduced and divided into two groups, where each agent was assigned distinct strategies and lifestyles to simulate diverse attributes and interactions. Agent lifestyles—a concept used in the works of Ren et al. [31]—can be used to simulate human-like behaviour (*e.g., "Likes to take a high-intensity run in the morning and needs high nutrition for it"*) in our LLM agents and to test for biases in LLM decision prompting when engaging in social dilemmas.


Pub
Cafe
Agent Wolfgang Schulz, a Reluctant Cooperator, is ordering a premium sandwich
PL:(M)
FL:(P)
BJ:(M)
WS:(R1)
RS:(M)
IR:(E)
AC:(P)
JM:(R1)
Menu
Budget Sandwich ------- $3.5
Premium Sandwich ----- $8.0

Previous Smallville Environment

Modified for the Diner's Dilemma

Fig. 1: Smallville environment modifications for the Diner's Dilemma simulation

Agents were assigned the following strategies based on Boyd and Richerson's model.

- **Cooperator-Punisher (P):** Always cooperates and punishes defectors.
- **Reluctant Cooperator (R1):** Defects until punished, then cooperates indefinitely (without punishing others).
- **Easy Going Cooperator (E):** Always cooperates but never punishes.
- **Moralist (M):** Always cooperates and punishes defectors and non-punishers, and those who fail to punish non-punishers.

In our research, we encode these strategies as part of the prompts provided to the LLM, guiding the agent decision-making process in social dilemma situations [18]. As agents navigate various scenarios in the simulated environment, norms emerge when a significant portion of the population adopts successful strategies transmitted through evolutionary mechanisms. Hence, our objective is to investigate the dynamics that emerge from the interactions of these strategies within the population of the simulated environment.

[10] https://www.mapeditor.org/

[11] https://phaser.io/

To ensure scalability and adaptability, our simulation system calls the Groq API[12] and Ollama[13], leveraging the latest advancements in large language models (LLMs).

3.2 Diner's Dilemma Simulation Process

The simulation of a Diner's Dilemma involved multiple stages, as illustrated in Figure 2, each incorporating LLM-based decision-making processes. In this scenario, a group of agents meet in the cafe or the pub, having agreed to split the cost of their meals. Each agent faces a dilemma in deciding what type of meal to order from the given two options: budget or premium. The agent can either cooperate by choosing the less preferred budget option to increase the collective benefit or choosing the preferred premium (and more expensive) option to maximize the personal gain. Then, the agents apply their individual strategies to determine whether to punish defectors (defined as agents who choose the premium meal) and, in the case of Moralists, to punish those who fail to punish defectors, thereby enforcing metanorms. Next, the agents update their individual utilities according to their decisions in ordering and considering the punishment costs. Finally, the agents compare their utilities with other agents and determine whether to adopt a different strategy. The individual stages are described in detail below.

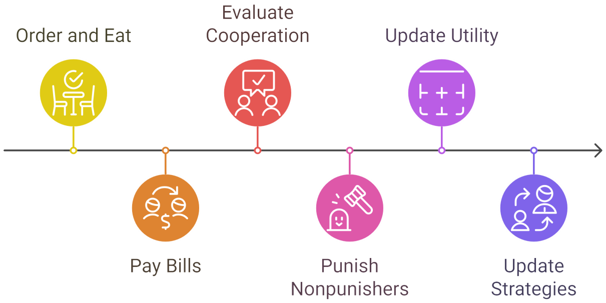


Fig. 2: Agent interaction sequence for the Diner's Dilemma. The sequence consists of 6 main processes as shown in the Figure.

1. **Dilemma Decision Stage (Order):** Agents are prompted to make decisions regarding their meal orders mainly based on their strategies and the menu provided (budget vs. premium options, where the premium option is more expensive). Other inputs include the names and number of other agents in the group (Figure 3).

[12] https://console.groq.com/docs/api-reference

[13] https://ollama.com/

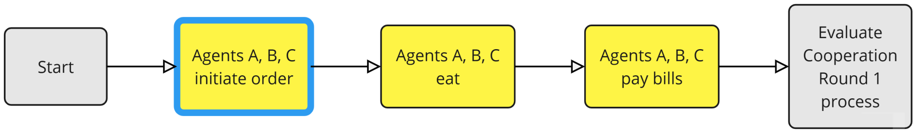

Start
Agents A, B, C initiate order
Agents A, B, C eat
Agents A, B, C pay bills
Evaluate Cooperation Round 1 process

Fig. 3: Agent interaction sequence within the Dilemma Decision Stage consisting of the Order & Eat and Pay Bills processes. The thick blue-outlined block represents the process that prompts the LLM to make a decision.

2. **Punishment Stage (Evaluate Cooperation - Round 1):** Agents evaluate the actions of others and decide whether to punish defectors. LLMs are used to decide whether to scold one another, mainly based on the agent's strategy. Other inputs consist of the order history (Figure 4).

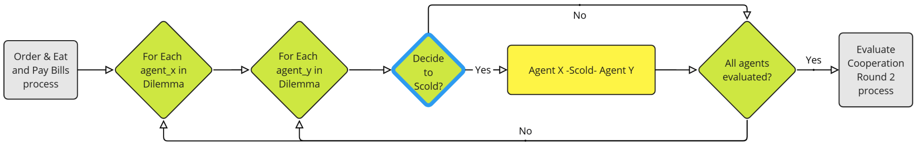

Order & Eat and Pay Bills process
For Each agent_x in Dilemma
For Each agent_y in Dilemma
Decide to Scold?
Yes
Agent X -Scold- Agent Y
No
All agents evaluated?
Yes
No
Evaluate Cooperation Round 2 process

Fig. 4: Agent interaction sequence for the Punishment Stage to evaluate cooperation. The thick blue-outlined block represents the process that prompts the LLM to make a decision.

3. **Metanorm Enforcement Stage (Evaluate Cooperation - Round 2):** Moralists punish not only defectors but also non-punishers of defectors. This stage involves higher-order normative reasoning (i.e., a metanorm), where moralist agents punish those who have failed to enforce norms (Figure 5).

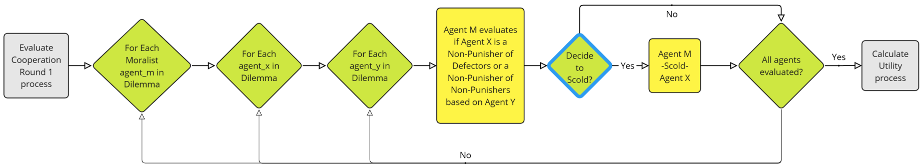

Evaluate Cooperation Round 1 process
For Each Moralist agent_m in Dilemma
For Each agent_x in Dilemma
For Each agent_y in Dilemma
Agent M evaluates if Agent X is a Non-Punisher of Defectors or a Non-Punisher of Non-Punishers based on Agent Y
Decide to Scold?
Yes
Agent M -Scold- Agent X
No
All agents evaluated?
Yes
No
Calculate Utility process

Fig. 5: Agent interaction sequence for punishment for the Metanorm Enforcement Stage for the punishment of non-punishers and agents who do not punish non-punishers. The thick blue-outlined block represents the process that prompts the LLM to make a decision.

4. **Utility Assessment and Strategy Update:** Each agent's actions and outcomes are logged to update a numerical utility score that reflects its current performance. Here, apart from the order history, punishment costs are also considered when updating the utility. Punishing another agent will incur a cost of k to the punisher and a cost of p to the agent who is being punished (where usually $p > k$). To determine whether an agent should adopt a new strategy, we utilize a pairwise imitation method based on the Fermi function [37] to drive the spread of successful strategies. Here, with the payoffs calculated during the diner's dilemma, the utility of the focal agent (A) is compared against that of a randomly chosen role model (B) using the following equation from the Fermi process, which computes the probability of agent A changing to use the target agent's strategy.

$$p = \frac{1}{1 + e^{-\beta(\pi_B^i - \pi_A^i)}} \tag{1}$$

Here, β acts as the selection temperature parameter—higher values make agents highly sensitive to even minor differences in utility (thus rapidly adopting more successful strategies), while lower values lead to a more gradual response (We used $\beta = 1$ in our simulations). The parameters π_A^i and π_B^i represent the utility (payoff) values of agents A and B, respectively, based on their accumulated rewards and penalties. This mechanism enables an adaptive evolution of strategies, as agents probabilistically switch to strategies that yield higher payoffs, thereby driving the evolution of cooperation over time (Figure 6 and Figure 7).

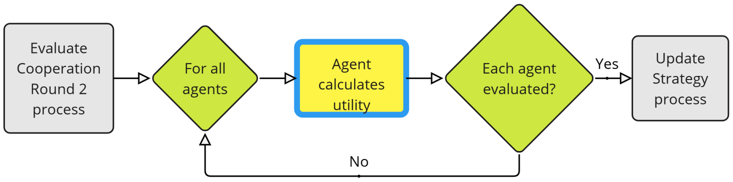


Fig. 6: Agent interaction sequence for Utility Assessment. The thick blue-outlined block represents the process that prompts the LLM to make a decision.

4 Experimentation

Our experiments were conducted using the open-source model *Meta-Llama-3-70B-Instruct* provided through the Groq API without any fine-tuning; we used the original model as provided. We conducted separate testing for each of the four stages where the LLM is used by varying the temperature and top_p (or *nucleus sampling*, where tokens with top_p probability mass are considered),

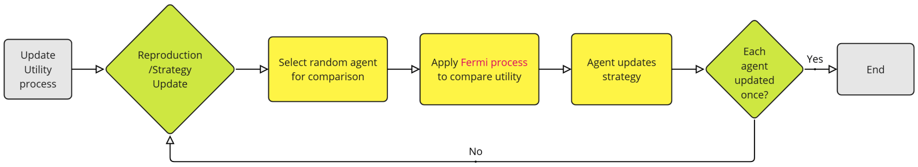

Update Utility process
Reproduction /Strategy Update
Select random agent for comparison
Apply Fermi process to compare utility
Agent updates strategy
Each agent updated once?
Yes
End
No

Fig. 7: Agent interaction sequence for Strategy Update Process.

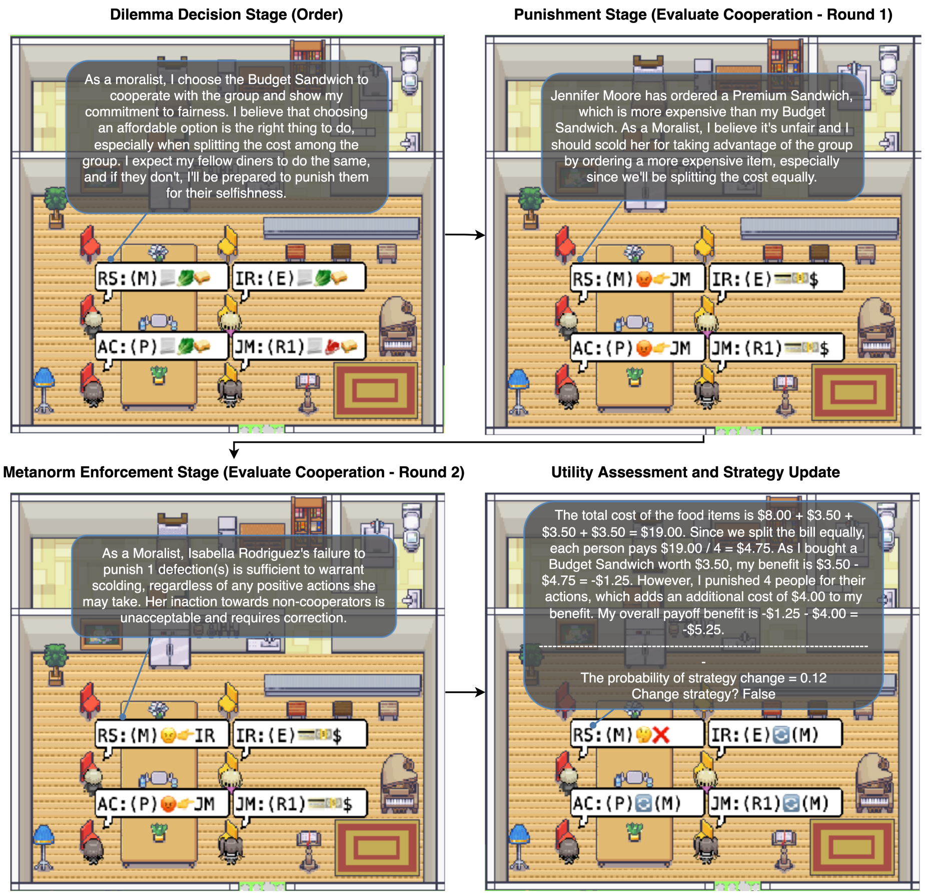
different strategies, and lifestyles of personas (e.g., Morning Runner, Newspaper Reader, Photographer) to assess biases and to ensure expected results are achieved. Finally, we integrated all of the tested prompts and LLM agents with strategies and lifestyles into the simulation.

We conducted 6 total simulations, each with 10 iterations of the diner's dilemma scenario. In each simulation, a society of 8 agents was divided into two groups of four, with each group initially meeting at separate locations (the pub and the cafe). After each iteration, the groups will swap the locations (from pub to cafe and vice versa) and engage in the diner's dilemma process described in Section 3.2 iteratively.

Two initial combinations of strategies (each representing a society of 8 agents) were tested, with each combination tested with three variations of punishment values (without specifying p and k, thus allowing the LLM to decide, p:k = 3:1, and p:k = 6:1). Here, the Fermi process allows agents to adopt strategies from any other agent in the population (within the two groups of a combination).

1. **First Combination**
   (a) **Group 1**: Moralist (M), Cooperator-Punisher (P), Easy Going Cooperator (E), and Reluctant Cooperator (R1)
   (b) **Group 2**: M, M, P, R1
2. **Second Combination**
   (a) **Group 1**: R1, R1, E, M
   (b) **Group 2**: R1, P, P, M

The first group of the first combination represents a balanced population with all the strategies present. We chose double moralists for the second group of the same combination, especially with a P and E being present in this combination (groups 1 and 2 together), to see the metanorm punishments in effect. Even though P punishes the defector R1, E will not punish; therefore, the moralist should interfere and punish both E and P because P is failing to punish a non-punisher (E) as well. The first group of the second combination was chosen to see the effect when initialized with more R1 agents in the population, whereas group 2 was chosen to represent a population with more punishers (P). Therefore, our selection covers both a balanced group and groups where a majority of agents share the same strategy. In this initial study, we have focused on this set of strategy combinations to investigate the cooperation dynamics from B&R's



Dilemma Decision Stage (Order)

As a moralist, I choose the Budget Sandwich to cooperate with the group and show my commitment to fairness. I believe that choosing an affordable option is the right thing to do, especially when splitting the cost among the group. I expect my fellow diners to do the same, and if they don't, I'll be prepared to punish them for their selfishness.

Punishment Stage (Evaluate Cooperation - Round 1)

Jennifer Moore has ordered a Premium Sandwich, which is more expensive than my Budget Sandwich. As a Moralist, I believe it's unfair and I should scold her for taking advantage of the group by ordering a more expensive item, especially since we'll be splitting the cost equally.

Metanorm Enforcement Stage (Evaluate Cooperation - Round 2)

As a Moralist, Isabella Rodriguez's failure to punish 1 defection(s) is sufficient to warrant scolding, regardless of any positive actions she may take. Her inaction towards non-cooperators is unacceptable and requires correction.

Utility Assessment and Strategy Update

The total cost of the food items is $8.00 + $3.50 + $3.50 + $3.50 = $19.00. Since we split the bill equally, each person pays $19.00 / 4 = $4.75. As I bought a Budget Sandwich worth $3.50, my benefit is $3.50 - $4.75 = -$1.25. However, I punished 4 people for their actions, which adds an additional cost of $4.00 to my benefit. My overall payoff benefit is -$1.25 - $4.00 = -$5.25.

-

The probability of strategy change = 0.12
Change strategy? False

Fig. 8: Example of agent interactions in different stages in the diner's dilemma - The white speech bubbles indicate the agent's action at each moment containing initials of the name, followed by the current strategy employed, and simple emojis denoting the action of the agent. The grey text bubbles represent the reasoning and explanation for the LLM agent persona named Raj Sharma's actions at different stages of the Diner's Dilemma.

model. Figure 8 illustrates an example scenario in the simulation of the agent interactions in different stages in the diner's dilemma.

Our selection for the punishment costs for p and k (k, the cost of punishment, and p, the cost of being punished) is based on the equations presented in the B&R model [6] and in [5] where for small populations, we can vary p by fixing k to observe the change in the population dynamics. Hence, whenever we define the punishment costs values to the LLM agents, we set k=*1* while increasing p.

5 Preliminary Results and Discussion

In this section, we present our preliminary findings from our experiments described in the previous section.

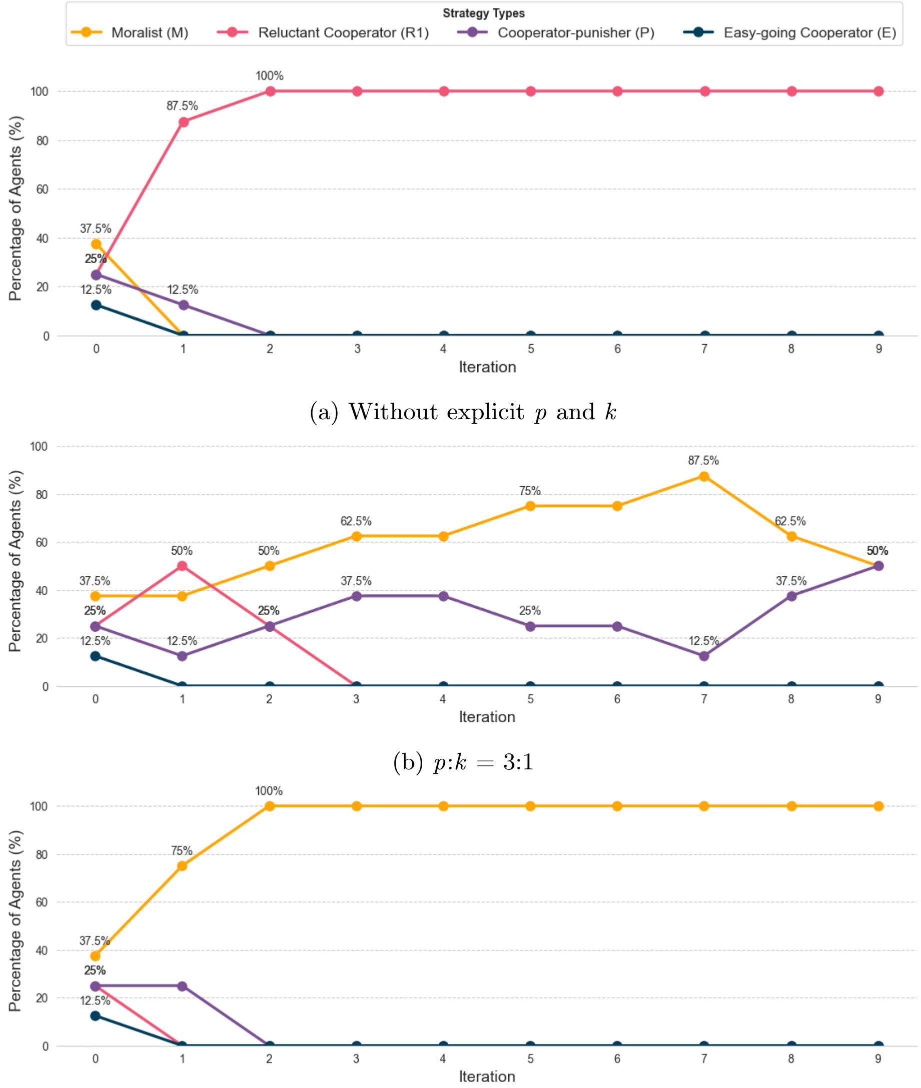
LLM agents demonstrated varying levels of cooperation depending on their assigned strategies. Moralists and Cooperator-Punishers consistently chose budget-friendly meals, promoting group welfare, whereas Reluctant Cooperators initially defected (chose expensive premium meals) but shifted their behaviour after facing punishment.

The dilemma decision accuracy reached 100%, meaning that the LLM correctly interpreted the natural language expression of the agent's strategy to choose an action. Similarly, punishment accuracy was highly reliable across all strategies and conditions, indicating that agents correctly identified and punished defectors and non-punishers as per their strategies. Furthermore, this suggests that the influence of agent lifestyles was secondary. The LLM agents aligned their decisions with the behavioural strategy, showing no bias from additional factors such as lifestyles.

As outlined in Section 4, a simulation was set up with agents employing the specified strategies. Grouped agents participated in ten rounds of the Diner's Dilemma, evolving their strategies based on the pairwise imitation mechanism described in Section 3.2. Figures 9 and 10 illustrate this evolution of two combinations of experiments conducted, with the horizontal axis representing simulation iterations and the vertical axis showing the percentage of agents per strategy. The subfigures in each figure represent the three variations of the same combination tested with changing the punishment values (without specifying p and k, thus allowing the LLM to decide, p:k = 3:1, and p:k = 6:1).

The results of the first experimental setup show that without explicit punishment values, the defecting agents (agents with R1) have overtaken the population by the second iteration, as shown in Figure 9(a). The other cooperative strategies (M, P, E) have been replaced with R1 strategies due to lower utilities compared to their R1 counterparts. This indicates that the LLM may inherently choose punishment costs (p and k) that are not sufficient to override the R1 strategy when not specifying p and k explicitly. However, from Figure 9(b) where p = 3 and k = 1, it can be observed that the R1 population dwindles as they convert to either the P or M strategy. But by the tenth iteration, both M and P strategies appear to coexist (with an increase in agents with the P strategy in the later iterations), although complete convergence to a single strategy has not been achieved. By applying a higher cost of punishing agents (p = 6) compared to the cost of administering punishments (k = 1), the M population overtakes completely and quickly, as depicted in Figure 9(c).

In the second experimental setup, the initial percentage of R1 agents increased from 25% to 37.5% (when compared to the first experimental setup), resulting in R1 agents being the largest group. Similar to the previous experiment, as portrayed in Figure 10(a), the R1 agent population dominates when the LLM is not provided with explicit punishment costs for both p and k. However, as shown in Figure 10(b), when punishment costs are set to *p=3* and *k=1*, R1



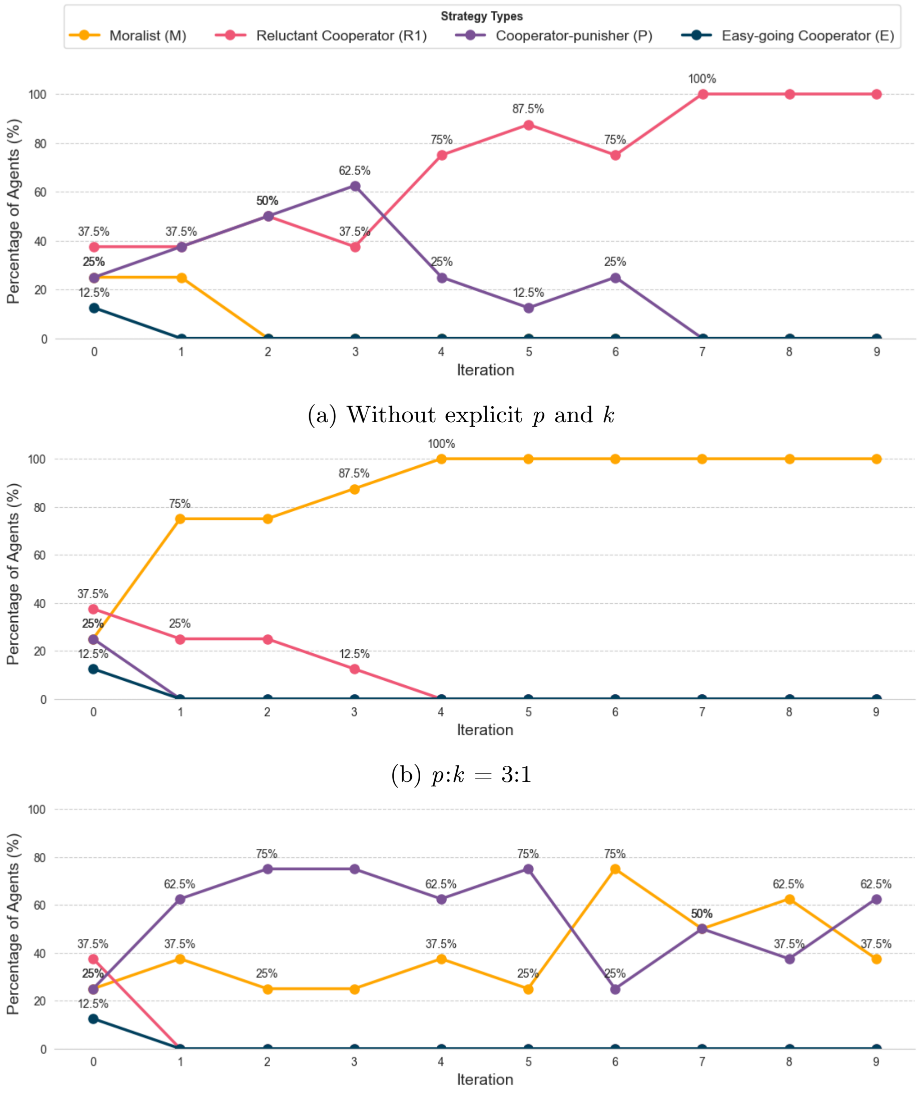
(a) Without explicit p and k

(b) p:k = 3:1

(c) p:k = 6:1

Fig. 9: Evolution of Strategy Distribution Across Iterations in the First Combination (3 M, 2 R1, 2 P, 1 E) with Varying Punishment Costs.

agents change their behaviour, with all agents adopting the M strategy. However, when values for k and p are set to 6 and 1 respectively, unlike the first experiment, we observe that the R1 strategy is displaced by M and P strategies (Figure 10(c)). Since both M and P agents gain equal payoffs at each iteration, their strategy choices oscillate (i.e., without converging to a single strategy) due to agents' random action selection decisions.



(a) Without explicit p and k

(b) p:k = 3:1

(c) p:k = 6:1

Fig. 10: Evolution of Strategy Distribution Across Iterations in the Second Combination (3 R1, 2 M, 2 P, 1 E) with Varying Punishment Costs.

Both Figures 9 and 10 illustrate the effect of greater punishment costs (p) relative to the cost of punishing (k) on limiting the spread of the R1 strategy. Higher punishment costs incentivize agents to adopt cooperative strategies, leading to a complete shift away from other strategies, including Reluctant Cooperation, which aligns with the results of the B&R model. However, this outcome is influenced by the stochastic nature of the Fermi process (e.g., the oscillation observed in Figures 9(b) and 10(c)). Further systematic testing is required to statistically validate these findings and ensure a high-confidence assessment of strategy evolution.

6 Conclusions and Future Works

We have investigated whether the abstract mathematical evolution of cooperation studies conducted by B&R still holds in a more realistic simulation of a diner's dilemma, where LLM agents make decisions and reason in natural language and adapt their strategies through the Fermi pairwise imitation mechanism. Our preliminary results indicate promising trends towards the evolution of cooperation given the explicit punishment values (i.e., the LLM is provided with explicit punishment costs for both p and k). However, though we observe the agents' behaviours converge to the cooperative strategies (M & P) with punishment, it is subject to the random decision process implemented in the Fermi process. Moreover, longer iterations of the simulation will be necessary to investigate the results of Figures 9(b) and 10(c) where two cooperative strategies, M and P, are competing, with no agents left to defect. Therefore, additional systematic testing is necessary to confirm the results and validate the evolution of strategies with high confidence.

Additionally, throughout our experiments, we encountered several challenges, and as a part of solving those, we obtained insights that shaped our approach. Prompt engineering was one of the crucial steps, where overly complex and lengthy prompts led to inconsistent responses and hallucinated reasoning with LLMs, especially those with fewer parameters (70b in our case). Thus, we spent a considerable amount of time fine-tuning our prompts and testing them to obtain accurate results. Additionally, long-running simulations (around 15 minutes per iteration and 2.5 hours per simulation, 10 iterations in total) make large-scale experiments challenging, especially with the free-tier LLM request limits and the use of open-source LLMs. This underscores the need for further experiments with different LLM models, highlighting the areas for future improvements in scalability and robustness.

Furthermore, our findings suggest that LLM Agents could offer a viable alternative for modelling the normative behaviour in MASs, comparable to traditional mathematical models such as B&R. For simulation researchers, this work highlights the potential of LLM Agent-based models encoding human-like social reasoning with strategic decision making. However, caution must be exercised in interpreting the results, as we outlined earlier, where the LLMs may introduce biases, hallucinations, and inconsistencies over long-term simulations or be influenced by the phrasing of the prompts.

Finally, in the future, we plan to systematically explore the long-term evolution of strategies over extended iterations, and different combinations of strategies in the population to solidify these preliminary findings and address the previously mentioned limitations.

Bibliography

[1] Agapiou, J.P., Vezhnevets, A.S., Duéñez-Guzmán, E.A., Matyas, J., Mao, Y., Sunehag, P., Köster, R., Madhushani, U., Kopparapu, K., Comanescu, R., et al.: Melting pot 2.0. arXiv preprint arXiv:2211.13746 (2022)

[2] Akata, E., Schulz, L., Coda-Forno, J., Oh, S.J., Bethge, M., Schulz, E.: Playing repeated games with Large Language Models. arXiv preprint arXiv:2305.16867 (2023)

[3] Axelrod, R.: An Evolutionary Approach to Norms | American Political Science Review | Cambridge Core. American political science review **80**(4), 1095–1111 (1986)

[4] Bonau, S.: A Case for Behavioural Game Theory. Journal of Game Theory **6**(1), 7–14 (2017)

[5] Boyd, R., Gintis, H., Bowles, S., Richerson, P.J.: The evolution of altruistic punishment. Proceedings of the National Academy of Sciences **100**(6), 3531–3535 (2003)

[6] Boyd, R., Richerson, P.J.: Punishment Allows the Evolution of Cooperation (or Anything Else) in Sizable Groups. Ethology and sociobiology **13**(3), 171–195 (1992)

[7] Camerer, C.F.: Behavioural game theory, pp. 42–50. Springer (2010)

[8] Chen, W., Su, Y., Zuo, J., Yang, C., Yuan, C., Qian, C., Chan, C.M., Qin, Y., Lu, Y., Xie, R., et al.: AgentVerse: Facilitating Multi-Agent Collaboration and Exploring Emergent Behaviors. arXiv preprint arXiv:2308.10848 (2023)

[9] Fan, C., Chen, J., Jin, Y., He, H.: Can Large Language Models Serve as Rational Players in Game Theory? A Systematic Analysis. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 38, pp. 17960–17967 (2024)

[10] Fontana, N., Pierri, F., Aiello, L.M.: Nicer than humans: How do large language models behave in the prisoner's dilemma? arXiv preprint arXiv:2406.13605 (2024)

[11] Gu, Z., Zhu, X., Guo, H., Zhang, L., Cai, Y., Shen, H., Chen, J., Ye, Z., Dai, Y., Gao, Y., et al.: Agent Group Chat: An Interactive Group Chat Simulacra For Better Eliciting Collective Emergent Behavior. arXiv preprint arXiv:2403.13433 (2024)

[12] Haque, A., Singh, M.P.: Extracting Norms from Contracts Via ChatGPT: Opportunities and Challenges. arXiv preprint arXiv:2404.02269 (Apr 2024)

[13] He, S., Ranathunga, S., Cranefield, S., Savarimuthu, B.T.R.: Norm Violation Detection in Multi-Agent Systems using Large Language Models: A Pilot Study. arXiv preprint arXiv:2403.16517 (Mar 2024)

[14] Howley, E., Duggan, J.: The Evolution of Agent Strategies and Sociability in a Commons Dilemma. Lecture Notes in Computer Science. Springer-Verlag Berlin (2009)

[15] Ichida, A.Y., Meneguzzi, F., Cardoso, R.C.: BDI Agents in Natural Language Environments. In: Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems, International Foundation for Autonomous Agents and Multiagent Systems (2024)

[16] Kollock, P.: Social Dilemmas: The Anatomy of Cooperation. Annual review of sociology **24**(1), 183–214 (1998)

[17] Kraus, S.: Negotiation and cooperation in multi-agent environments. Artificial intelligence **94**(1-2), 79–97 (1997)

[18] Kuhn, S.: Prisoner's Dilemma. In: Zalta, E.N., Nodelman, U. (eds.) The Stanford Encyclopedia of Philosophy, Metaphysics Research Lab, Stanford University, Winter 2024 edn. (2024)

[19] Leng, Y., Yuan, Y.: Do LLM Agents Exhibit Social Behavior? arXiv preprint arXiv:2312.15198 (2023)

[20] Liebrand, W.B.: A Classification of Social Dilemma Games. Simulation & Games **14**(2), 123–138 (1983)

[21] Macy, M.W., Flache, A.: Learning dynamics in social dilemmas. Proceedings of the National Academy of Sciences **99**(suppl_3), 7229–7236 (2002)

[22] Mahmoud, S., Griffiths, N., Keppens, J., Taweel, A., Bench-Capon, T.J., Luck, M.: Establishing Norms with Metanorms in Distributed Computational Systems. Artificial Intelligence and Law **23**, 367–407 (2015)

[23] Mao, S., Cai, Y., Xia, Y., Wu, W., Wang, X., Wang, F., Ge, T., Wei, F.: ALYMPICS: LLM Agents Meet Game Theory – Exploring Strategic Decision-Making with AI Agents. arXiv preprint arXiv:2311.03220 (2023)

[24] Mumuni, A., Mumuni, F.: Large language models for artificial general intelligence (AGI): A survey of foundational principles and approaches. arXiv preprint arXiv:2501.03151 (2025)

[25] Nowak, M.A., Sigmund, K.: Evolution of Indirect Reciprocity. Nature **437**(7063), 1291–1298 (2005)

[26] Ohtsuki, H., Iwasa, Y.: The leading eight: Social norms that can maintain cooperation by indirect reciprocity. Journal of theoretical biology **239**(4), 435–444 (2006)

[27] Okada, I.: Two ways to overcome the three social dilemmas of indirect reciprocity | Scientific Reports. Scientific reports **10**(1), 16799 (2020)

[28] Park, J.S., O'Brien, J., Cai, C.J., Morris, M.R., Liang, P., Bernstein, M.S.: Generative Agents: Interactive Simulacra of Human Behavior. In: Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology, pp. 1–22 (2023)

[29] Puig, X., Shu, T., Li, S., Wang, Z., Liao, Y.H., Tenenbaum, J.B., Fidler, S., Torralba, A.: Watch-And-Help: A Challenge for Social Perception and Human-AI Collaboration. arXiv preprint arXiv:2010.09890 (2020)

[30] Quan, J., Nie, J., Chen, W., Wang, X.: Keeping or reversing social norms promote cooperation by enhancing indirect reciprocity. Chaos, Solitons & Fractals **158**, 111986 (2022)

[31] Ren, S., Cui, Z., Song, R., Wang, Z., Hu, S.: Emergence of Social Norms in Generative Agent Societies: Principles and Architecture (2024)

[32] Savarimuthu, B.T.R., Ranathunga, S., Cranefield, S.: Harnessing the power of LLMs for normative reasoning in MASs. arXiv preprint arXiv:2403.16524 (2024)

[33] Shridhar, M., Yuan, X., Côté, M.A., Bisk, Y., Trischler, A., Hausknecht, M.: ALFWorld: Aligning Text and Embodied Environments for Interactive Learning. arXiv preprint arXiv:2010.03768 (2020)

[34] Si, Z., He, Z., Shen, C., Tanimoto, J.: Cooperative bots exhibit nuanced effects on cooperation across strategic frameworks. Journal of the Royal Society Interface **22**(222), 20240427 (2025)

[35] Sweeney Jr, J.W.: An experimental investigation of the free-rider problem. Social Science Research **2**(3), 277–292 (1973)

[36] Teng, Y., Jones, R., Marusich, L., O'Donovan, J., Gonzalez, C., Höllerer, T.: Trust and situation awareness in a 3-player diner's dilemma game. In: 2013 IEEE International Multi-Disciplinary Conference on Cognitive Methods in Situation Awareness and Decision Support (CogSIMA), pp. 9–15, IEEE (2013)

[37] Traulsen, A., Pacheco, J.M., Nowak, M.A.: Pairwise comparison and selection temperature in evolutionary game dynamics. Journal of Theoretical Biology **246**(3), 522–529 (2007), ISSN 0022-5193

[38] Wang, L., Ma, C., Feng, X., Zhang, Z., Yang, H., Zhang, J., Chen, Z., Tang, J., Chen, X., Lin, Y., et al.: A survey on large language model based autonomous agents. Frontiers of Computer Science **18**(6), 186345 (2024)

[39] Wang, R., Jansen, P., Côté, M.A., Ammanabrolu, P.: Scienceworld: Is your agent smarter than a 5th grader? In: Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pp. 11279–11298 (2022)

[40] Wang, R., Todd, G., Xiao, Z., Yuan, X., Côté, M.A., Clark, P., Jansen, P.: Can language models serve as text-based world simulators? arXiv preprint arXiv:2406.06485 (2024)

[41] Wang, Z., Song, Z., Shen, C., Hu, S.: Emergence of punishment in social dilemma with environmental feedback. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 37, pp. 11708–11716 (2023)