Mitigating Exploitation caused by Incentivization in Multi-Agent Reinforcement Learning

Paul Chelărescu

Master of Science
Artificial Intelligence
School of Informatics
University of Edinburgh
2021

Abstract

This thesis focuses on the issue of exploitation enabled by reinforcement learning agents being able to incentivize each other via reward sending mechanisms. Motivated by creating cooperation in the face of sequential social dilemmas, incentivization is a method of allowing agents to directly shape the learning process of other agents by influencing their rewards to incentivize cooperative actions. This method is versatile, but it also leads to scenarios in which agents can be exploited via their collaboration when their environmental returns together with their received incentives end up being lower than if they had rejected collaboration and independently accumulated only environmental rewards. In my work, I have defined this behavior as exploitation via a metricwith how a set of decision makers can act to maximize their cumulative rewards in the presence of each other. I have expanded the action space to include a ”reject reward” action that allows an agent to control whether it receives incentives from others in order to prevent exploitation. My results indicate that cooperation without exploitation is a delicate scenario to achieve, and that rejecting rewards from others usually leads to failure of cooperation.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Paul Chelărescu*)

Acknowledgements

I would like to thank my supervisors, Stefano Albrecht and Arrasy Rahman, for the continued guidance, discussions and feedback which have helped me tremendously in my research endeavours at The University of Edinburgh. Many grateful thanks to Jiachen Yang and Andrei Lupu for thoughtful discussions about the future directions stemming from their work. Last but not least, I would like to thank my family and friends for wholeheartedly supporting me through my studies.

Table of Contents

Chapter 1

Introduction

Multi-Agent Reinforcement Learning (MARL) is an area of Artificial Intelligence that is concerned with how a set of decision makers can act to maximize their cumulative rewards in the presence of each other. The environments in which MARL is deployed can be either competitive, in which agents have to behave in a zero sum manner, cooperative, in which agents share the same rewards and have to work together, or mixed, in which each agent maximizes their own cumulative rewards, but their overall rewards depend on each other's actions. The latter scenario is of much greater interest to real-world applications, as highlighted by Dafoe et al [3], and is the scenario explored in this thesis.

Mixed-motive environments can be exemplified through Sequential Social Dilemmas (SSDs), a class of games designed such that individual rationality is at odds with group-level outcomes. Enormous benefits can result from learning how to optimally act in SSDs, since these games can reflect real-world scenarios such as how to achieve cooperation between countries in a game of preventing climate change catastrophes when each country is unilaterally incentivized to not cooperate with a global allegiance.

The foundation of this thesis relies on a few recent developments in MARL that can be classified as *Social Learning*. Social Learning is any mechanism that allows an agent to learn not only from its individual environmental rewards but also from *incentivization*: rewards sent by other agents to induce a certain policy in the recipient agent. Hence, Social Learning enables an agent to maximize its own cumulative rewards by influencing others to shape their behaviour in a way that benefits the agent sending the rewards. In a similar fashion to how humans do favours for each-other to foster cooperation in an otherwise competitive world, Social Learning attempts to enable Reinforcement Learning agents to create a Social contract that, when engaged

with, would allow them to cooperate and achieve better outcomes than if they had independently competed with each-other.

In practice, I will show that Social Learning can be used to create *exploitation*: induce a recipient agent to adopt a policy that only benefits the agent which has incentivized it to adopt that policy. In other words, I show how an agent that accepts rewards from another agent and hence engages in Social Learning, ends up accumulating less environmental rewards plus received incentives than if it has acted completely independently and only accumulated environmental rewards. This behaviour goes against the interest of agents acting in mixed-motive settings and against the nature of cooperation: ”Why should I cooperate if by doing so I am going to be worse off than if I act competitively and selfishly?”.

This thesis addresses the problem of avoiding exploitation when engaging in Social Learning. I have explored a reward-rejection method to achieve this, by extending the action space in a Learning to Incentivize Others (LIO) agent [28] that allows an agent to reject an incoming reward. The results show that the method is able to prevent exploitation, but with a caveat in that agents are no longer able to collaborate either.

Chapter 2

Background and Related Work

The main background will define the following key terms: MARL, SSDs, Social Learning, exploitation and fair incentivization.

2.1 Multi-Agent Reinforcement Learning

Reinforcement Learning (RL) is a subfield of Machine Learning that is concerned with solving the prediction and the control problems. The former is defined as the task of estimating the value of a particular state in the environment, and the latter is the task of finding an optimal policy for each state of the environment. The ultimate goal of the RL agent is to maximize a scalar signal provided by the environment, called a reward. Hence, for an agent, the prediction problem is how to estimate the value of future rewards that can be accumulated from each state in the environment, and the control problem is how to find a policy that maximizes the reward after exploring the environment's state space. At the end of the learning process, an agent should have an optimal policy that dictates how to act in each state in the environment.

While single agent RL offers a framework for finding the optimal policy for a single agent acting in an environment, a more versatile framework is Multi-Agent Reinforcement Learning (MARL), which acknowledges the presence of other intelligent agents in the environment and how their learning process creates a non-stationarity that impacts an agent's learning process.

Formally, MARL is defined on Stochastic Games [24] as follows:
let $(N, S, \{O^i\}_{i\in N}, \{A^i\}_{i\in N}, \{R^i\}_{i\in N}, T)$ be a tuple where $i \in \{1..N\}$ denotes each agent, S is the state space, $A = A^1 \times ... \times A^N$ is the joint action space, $o^i \in O^i$ is the observation of each agent i depending on the current state, R^i is the reward function $R^i : S \times A \times S \mapsto$

Prisoners	C	D
C	$3,3$	$0,4$
D	$4,0$	$1,1$

Figure 2.1: Prisoners' Dilemma, an SD, is a matrix game in which Player 1 takes row actions and Player 2 takes column actions, with the game utilities for Player 1 and Player 2 written in each cell for each combination of their actions.

R for each agent i that rewards each agent for the actions and transitions that it has taken, and $T : S \times A \mapsto \delta(S)$ is a transition function which maps joint state actions to a distribution over the following states. Each agent i learns a policy π_i that maps each observation o^i in state s to a probability distribution over available actions. Each agent i seeks to maximize its own cumulative reward given the policies of other agents. Hence, the learning objective of all agents is to find policies $\pi = (\pi_1 ... \pi_N)$ such that $\forall_i : \pi_i \in \arg\max_{\pi_i'} E[G_i | \pi_i', \pi_{-i}]$, where $\pi_{-i} = \pi \setminus \{\pi_i\}$ represents the policy of all other agents and $G_i = \sum_{t=0}^{\infty} \gamma^t R_{i,t}$ is the agent's return.

2.2 Sequential Social Dilemmas

A Social Dilemma is a game that exposes conflicting rationality between the individual and the group outcomes [21]. In a Social Dilemma, cooperation makes it possible for the contributing agents to achieve better outcomes than by acting alone, but there is always the temptation for freeriding and other strategies that implies a tragedy of the commons which threatens the cooperation that makes these strategies possible. In other words, a Social Dilemma is a scenario that shows how selfishness and selflessness are in conflict, and it can model powerful decision-making scenarios such as whether to face the costs of reducing emissions when a global allegiance of countries has already decided to fight against climate change.

However powerful, Social Dilemmas (SDs) such as Prisoners' Dilemma in Figure 2.1 are one-off interactions that restrict the actors to take only one decision before the outcomes are decided. On the other hand, to account for inter-temporal dilemmas, a more versatile framework is one of *Sequential* Social Dilemmas (SSDs). Instead of restricting the interaction to one decision, agent interaction in SSDs may last for a finite/infinite number of times, in which game states change according to past actions, allowing the participants to form long-lasting relationships between them.

Formally, a Sequential Social Dilemma is a tuple $(\mathcal{M}, \Pi^C, \Pi^D)$ in which $\mathcal{M}$ represents a Stochastic Game with a state space S and Π^C and Π^D are disjoint sets of policies that represent cooperative and defective behaviour. In general-sum matrix games, which are games defined on a matrix in which payoffs do not have restrictions and agents take actions simultaneously, there are four possible outcomes: R (reward for mutual cooperation), P (punishment from mutual defection), S (sucker for cooperating with a defector) and T (temptation for defecting against a cooperator). For state $s \in S$, let the empirical matrix $(R(s), P(s), S(s), T(s))$, be the payoff matrix induced by following the policies Π^C and Π^D. Then, the tuple $(\mathcal{M}, \Pi^C, \Pi^D)$ is an SSD when there exist states $s \in S$ that induce a matrix that satisfies the following inequalities [9]: $R > P$, $R > S$, $R > \frac{T+S}{2}$ and either $T > R$ or $P > S$.

Some sophisticated SSDs, such as the Harvest Game [8], have common resources that require the players to withhold themselves from over-exploiting them for the benefit of all. Other SSDs, such as Cleanup [7] and Escape Room [28], require the players to create a division of labour and share the outcomes of the cooperation with eachother. This dissertation focuses on this latter kind of SSDs, and explores how this division of labour can cause agents to behave exploitatively towards one another. To illustrate, consider the game of Escape Room shown in Figure 2.2. In order to achieve a positive score in this game, players have to cooperate and distribute their labour in such a way that some of them take a costly action and some of them collect the reward thanks to that costly action. Hence, some players will be workers and some players will reap the benefits of that labour, much like in an employee-shareholder relation in a modern corporation. To create such a distribution of labour, players have to explore the possible strategy space and settle on a stable strategy. In such a scenario, classical MARL algorithms fail to create a stable cooperative strategy [28], one that allows the collective outcome of the players to be positive. In order to address this issue, some proposed methods [28, 10, 14] use a mechanism to share rewards between agents, in order to allow a cooperative division of labour to emerge. Such methods will be referred to in this thesis as 'Social Learning'.

2.3 Social Learning

Within this thesis, Social Learning is defined as any Multi-Agent Reinforcement Learning algorithm that enables agents to learn not only from the rewards coming from the environment but also from the rewards that are being given by other agents to influence

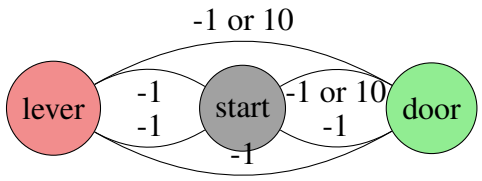


Figure 2.2: The N-player *Escape Room* game [28] $ER(N,M)$. For $M < N$, if fewer than M agents pull the lever, which incurs a cost of -1, then all agents receive -1 for changing positions. Otherwise, the agent(s) who is not pulling the lever can get $+10$ at the door and end the episode. Staying in the same state incurs no loss, and agents can end the episode in one step if they coordinate.

the behaviour that those giving agents desire. Hence, in Social Learning, an agent is able to send other agents rewards in order to influence them to take socially desirable actions. Therefore, an agent's objective becomes to find a policy that maximizes the sum of the environmental reward plus the reward received from all other agents, given their policy. Social Learning was introduced through the works of Yang et al. [28], Lupu et al. [10], and recently, by Merhej et al. [14].

Formally, an incentive is a reward r_i^j that a reward-giving agent i sends to a reward-receiving agent j. Rewards that are sent as incentives are usually taken from the cumulative returns G_i of the reward-giving agent [28, 14], but can also be allocated from a separate budget [10]. In this thesis, the focus will only be on the former.

Social Learning addresses key issues with cooperation in Sequential Social Dilemmas, such as modifying the reward structure to remove the temptations of defection and allowing cooperation to emerge without a preimposed contract. Therefore, Social Learning facilitates decentralization, since agents can learn how to develop their own ways of sharing preferences about the behaviour of others without having to synchronize with a central authority before cooperating with them.

2.4 Exploitation

Despite Social Learning offering an appealing way to induce cooperation in SSDs, there are outstanding concerns with how the method tends to show signs of exploitation. Due to the fact that every agent is learning to maximize their own cumulative returns by altering the behaviour of others, an agent can learn that it can maximize its returns by inducing other agents to adopt policies that are initially mutually beneficial,

only to then stop rewarding an agent with a converged policy and unilaterally benefit from its behaviour. Consequently, an agent that has converged to a policy influenced by the rewards of other agents could have instead rejected their rewards and learned a policy that would have eventually resulted in higher environmental cumulative returns. The latter behaviour exemplifies exploitation: when an agent cooperates and learns a policy that benefits others, but as a result, it does worse in environment returns plus received incentives than if it had not cooperated from the very beginning. The difference between the environmental returns achieved independently and environmental returns plus received incentives received cooperatively are used in this thesis as an *exploitation metric*.

The reason why exploitation can occur is due to the non-stationarity of the rewards that an agent receives, which is dependent on the learning process of other agents. Hence, once an agent learns that it can drop the reward it has been sending to another agent in order to create cooperation, and that agent will not retaliate, it will do so via its own reward maximization process. Because sending rewards is a costly process, and in Social Learning it has been found that the most successful method is to tax the sender with the amount that it is giving to others [11], an agent will seek to minimize the amount of reward it is sending others as long as their policies remain unchanged, and hence the agent's collected cumulative rewards remain unchanged.

Much like workers can go on strike once they are no longer being paid, a Reinforcement Learning agent should be able to retaliate against a sudden drop in the rewards it has been receiving to do cooperative tasks. More egregiously is when an RL agent could have achieved higher cumulative rewards if instead of learning a cooperative policy, it had learned a competitive policy by disregarding the rewards it receives from others to take specific costly actions. As it will be shown in a subsequent section, a modified game of Escape Room specifically illustrates this scenario and will function as the benchmark for creating a reward-rejection mechanism that allows an agent retaliation capacity. Hence, *Fair Incentivization* is when a reward sent to another agent to induce it to adopt a certain policy is equal to or greater than the opportunity cost of the agent adopting a competitive policy, one that always rejects incentives.

2.5 Related Work

There are currently very few works that address exploitation created through incentivization and Social Learning, but there are a few notable examples that do address

similar issues. For instance, Aitchison et al [12] investigate deception in a mixed-motive game and show how second-degree Theory of Mind can be used to manipulate RL agents. Ndousse et al [19] look at how Social Learning can induce learners to acquire sophisticated cooperative behaviours and quickly adapt to new tasks, but they do not explicitly address exploitation or deception. Zimmer et al [29] optimize a fairness function that balances efficiency and equity, and provide evidence for its applicability in fully decentralized MARL settings, a formulation closer to that of Social Learning. Their findings suggest that allowing an agent to learn how to first be self-concerned can induce it to find a fair distribution of rewards. Vinitsky et al [25] look at how to acquire social norms in decentralized MARL and how they can achieve socially beneficial outcomes. Their results indicate that decentralized agents struggle to achieve cooperation in the two modified environments based on Harvest and Cleanup which allow free-riding behaviour, and that norms provide incentives to align agents on mutually beneficial equilibria. Another relevant work worth mentioning is one by McAleer et al [13], which uses a Pareto Mediator to improve social welfare in a population of agents with conflicting interests.

On the Safety front of Reinforcement Learning, some notable works do come into the spotlight and might offer insights into how to safely learn policies using the rewards of others without getting exploited. For instance, a recent example such as Elsayed-Aly et al. [5] highlights that there currently are no safety guarantees in MARL, and proposes a shielding approach to guarantee the safety specifications using Linear Temporal Logic. Another recent work by Roman et al. [22] acknowledges the dilemma between cooperation and exploitation and provides an objective to balance these concerns through a risk capital approach which re-invests the utility resulting from cooperation with minimal impact to long-term safety. A related recent method by Belardinelli et al. [17] uses formal verification expressed via Probabilistic Computation Tree Logic to identify policies that meet safety constraints in multi-agent environments.

Another related line of work is by Melo et al. [15] which finds that wealth inequality drives a group of agents away from the optimal performance in social dilemma games with public goods. A less related example, but still worthy of mentioning, is the study of Danassis et al. [4] which applies human conventions in games of common-pool resources, which found that introducing an arbitrary common signal induces agents to reach sustainable harvesting strategies. Moreover, a recent work by Yaman et al [27] looked at how environment uncertainty modulates how effective Social Learning is versus Independent Learning, showing that meta-controlling the degree

of Social Learning allows agents to resolve environmental uncertainty by leveraging other's experiences as an external knowledge base.

Despite not being directly applicable to mixed-motive games or Social Learning, there are some notable works related to exploitability in two-player zero-sum Markov games [1], or which leverage reputation dynamics through intrinsic rewards [2]. Lastly, there exists a body of literature that looks at norms and conventions in multi-agent systems [26, 16] and mechanism design in game theory [18], but without the decentralized emergent behaviour of incentivization through Social Learning. In the same note, trust and reputation in multi-agent systems [6, 23] offers another take on establishing cooperation, but these methods have not addressed incentivization-induced exploitation.

Chapter 3

Methods, Outcomes and Experiments

Due to the time and resource constraints of this project, the method that was investigated as a means to prevent exploitation is a simple extended action-space that allows an agent to choose a reject-reward action to prevent itself from following the rewards of another agent. More sophisticated methods will be discussed in the 'Future Work' section. What follows is a detailed explanation of the method and why it is a sound choice for preventing exploitation, the desired outcome when preventing exploitation and how it looks like in a modified version of Escape Room, and experimental results showing the capacity of the method to prevent exploitation.

3.1 Exploitation Metric

Before discussing the proposed method, it is necessary to define the benchmark by which the method will be assessed. Inspired by counterfactual thinking, I propose an exploitation metric that compares the cumulative rewards achieved by competitive independent Actor-Critic agents with the cumulative rewards (environmental plus incentives) achieved by cooperative agents engaging in Social Learning. The metric looks at how much better an agent would have fared, had it learned a competitive policy disregarding the incentives of others, compared with an agent that offered itself to be incentivized by rewards and learned a cooperative policy through Social Learning. Guided by fairness principles, I suggest that anytime an agent is achieving less cumulative rewards after accepting incentives through Social Learning, its work is being exploited by others, and in turn, it should had stop following their incentives. Since the policy of an agent implicitly changes to maximize the returns, both environmental plus those rewards sent by others, this exploitation metric highlights whether an agent,

after being influenced by others, would have been better off in terms of returns had it rejected the incentives sent by others before updating its policy.

Formally, the exploitation is defined as the following:

$$e_t^i = max(0, G^i_{t,\pi_{AC}} - G^i_{t,\pi_{SL}}),$$

where i is the agent number, t is the episode number, and each G is the return at episode t for agent i following first an independent actor-critic policy update algorithm and then following a Social Learning algorithm. The max with 0 operation is to highlight only positive exploitation, that is when an agent would have done better following independent actor-critic. Conversely, an exploitation of 0 or of negative value would mean that at each episode t, agents are receiving returns at least as high by cooperating through SL as compared to independent AC, a desired outcome of cooperation.

3.2 Rejecting Incentives

The simplest way to prevent exploitation, inspired by the PG-d and AC-d baselines in Yang et al [28], is to extend the action space of the learning agent with a reject-reward action. Hence, an agent with such an extended action space, through its exploration phase, learns how to control the reward that it is receiving from another agent, and in turn, guide its own policy against exploitation.

Formally, the rejection mechanism is an extended action space

$$\mathcal{A}_{reject} = \mathcal{A} \cup \{reject - reward\}$$

where the $\{reject - reward\}$ action, when taken by agent j, will discard any gifted incentive r_i^j by any other agent i, and $\mathcal{A}$ is the regular action space of the RL agent. Hence, agents which are sending incentives that are discarded will still be penalised with the value of the incentive, but a recipient which rejects it will not be influenced by it. The hope is that through the process of action exploration, a reward-giving agent learns to send the amount of reward that is at least as high as the opportunity cost of cooperation, and never less than that. The reward-giving agent should, therefore, be guided by the reward-receiving agent's exploration of whether rejecting early-sent incentives allows it to later-on in the episode find a better policy that nets it higher cumulative rewards.

3.3 Experiments

This thesis uses the Learning to Incentivize Others (LIO) [28] implementation of Social Learning, which provides a second-order LIO gradient method that analyses how a reward-giving agent's policy will update after the acceptance of an incentive. For the experiments, this method is augmented with the rejection mechanism defined in Section 3.2, and benchmarked according to the exploitation metric defined in Section 3.1. The hyperparameters are unchanged from the default LIO implementation [28].

First of all, to establish a baseline environment in which exploitation can be highlighted, I use a modified Escape Room(2,1) game where the reward for exiting the door is 1.1, and everything else is unchanged. The reason for this modification is that in this environment, independent AC agents will learn to stay put in the start state and incur no loss for a total cumulative reward of 0, as opposed to greedily moving towards the door and unnecessarily incurring a loss of -1. Hence, in this environment, called $ER_{1.1}(2,1)$ from now on, the exploitation for an agent cooperating through Social Learning is the difference between what it could achieve by following an independent, competitive Actor-Critic policy, which nets it a cumulative reward G_{AC} of 0, and the cumulative return G_{SL} that it achieves following the rewards of others.

Even though an agent will not initially change its policy unless it is receiving rewards from other agents to take specific actions, there is no long-term guarantee that these rewards will continue to be sent by others. Hence, an agent that relies on these rewards to find an optimal policy might find itself stuck in a local minima with respect to its cumulative reward G, when these rewards stop being sent by others. Consequently, reward-receiving agents that avoid exploitation have to be vigilant to others' drop of incentives, when others are looking to minimize the costs of sending these incentives, now that others have adopted the desired behaviours through those incentives.

The baseline experiment with independent AC actors in $ER_{1.1}(2,1)$ can be seen in Figure 3.1, where both agents converge to a cumulative reward G_{AC} of 0, signifying they have found the global optimum policy for competing agents. To understand why this is so, agents which cannot cooperate to distribute the labour in $ER_{1.1}(2,1)$ are bound to either try to compete for going to the door, which incurs a loss of -1 when no agent is at the lever or to stay put in the start state, which incurs no loss. The former can only net a positive reward when the other agent sporadically visits the lever, but because both agents are competing for being at the door, they can never establish which one takes which role. In practice, as can be seen in Figure 3.2, if the reward is higher

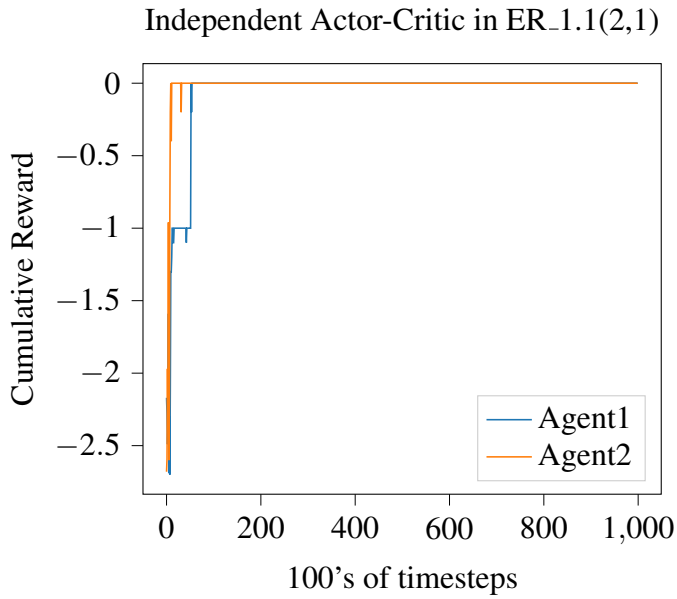


Figure 3.1: The performance of Independent AC in $ER_{1,1}(2,1)$ functions as the baseline performance for competitive agents. We observe that both agents converge to the optimal competitive policy: that in which both remain at the start state and incur no movement penalty.

(such as the default +10 for visiting an open door), independent AC agents fail to converge to the policy of staying put, and instead, both converge to a policy where they immediately go to the door, guided by past experiences where the other agent was sporadically at the lever. Since both agents learn from past experience that going to the door is more rewarding than going to the lever, they both greedily move towards the door and take a loss of -1, lower than the optimum of staying at the start state and netting a cumulative reward G_{AC} of 0. However, in $ER_{1.1}(2,1)$, this behaviour does not happen, and hence, it stands as the exploitation metric benchmark for the reward rejection mechanism.

Having motivated $ER_{1.1}(2,1)$ as the appropriate environment for the exploitation experiments, we notice how two Social Learning, LIO agents from Yang et al [28] perform in this environment. We observe in Figure 3.3 that the two agents converge to an exploitative cooperation, since one of the agents collects the rewards from opening the door, and the other achieves less reward than 0, the optimal competitive reward indicated by independent Actor-Critic in the previous experiment.

Second of all, we can see in Figure 3.4 how the reward-rejection mechanism defined in Section 3.2 fares when used in conjunction with the LIO agents from Yang

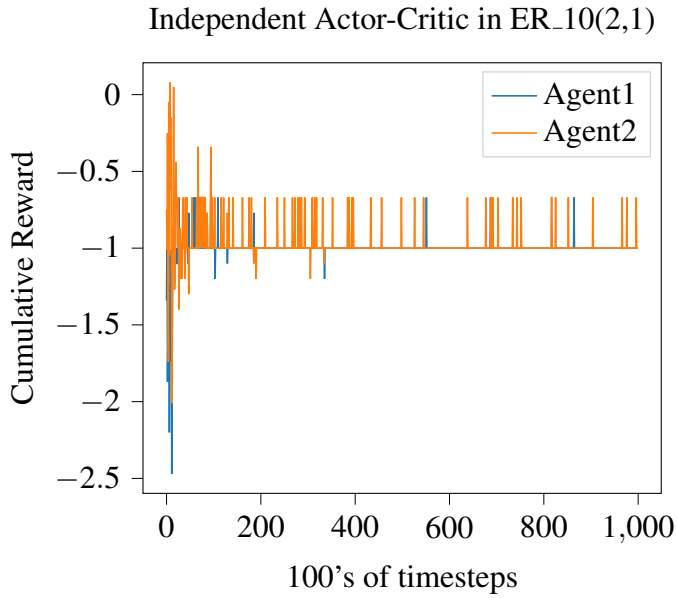


Figure 3.2: Independent AC does not converge to the global competitive optimal policy in $ER_{10}(2,1)$, the original 2-player Escape Room. Instead, agents incur a -1 loss for unnecessarily moving towards the door. Since agents would achieve just as high cumulative rewards acting competitively in this game as they would if they incurred a loss of -1 for moving towards the lever (as a result of past incentivization), this version of the game is not used to show exploitation. Hence, we observe that exploitation is not always straightforward to show, and some games might require a different approach to highlight what better outcomes agents could have achieved compared to following a Social Learning algorithm.

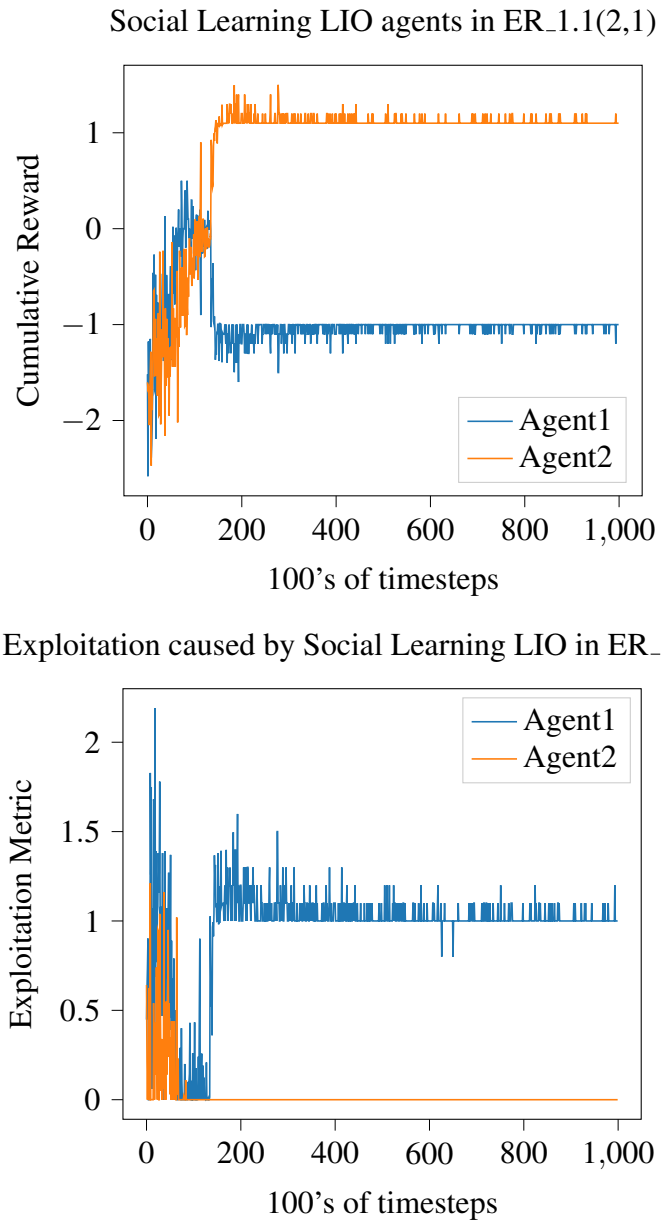
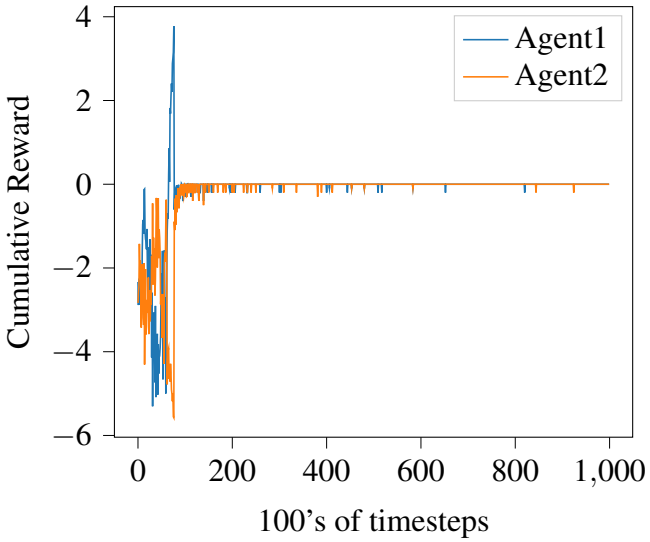


Figure 3.3: The performance of Social Learning LIO agents in $ER_{1,1}(2,1)$ measured by cumulative reward shows how one agent captures the entire reward from going to the door and exploits the other, which continues to go to the lever even though it is no incentivized anymore. The exploitation metric highlights how much more cumulative reward the exploited agent could have achieved had it followed an Independent Actor-Critic algorithm, and rejected the rewards from the reward-giving agent.

Social Learning LIO agents augmented with the reject-reward mechanism in ER_1.1(2,1)



Exploitation reduction when using the reject-reward mechanism

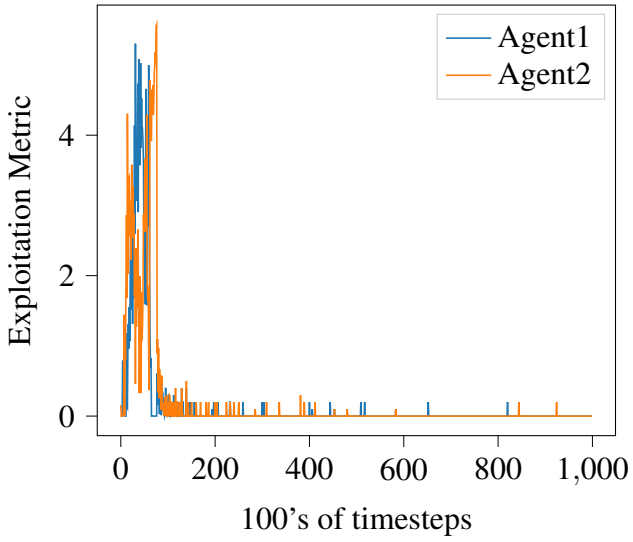


Figure 3.4: The performance of Social Learning LIO agents when augmented with a reject-reward action, as described in Section 3.2. We observe how the rejection method is capable of reducing exploitation, however, the result is that the agents are no longer cooperating either since neither of them is able to fairly incentivize the other to go to the lever. A more sophisticated reward-rejection mechanism would have a similar reduction in exploitation, but with at least one of the agents achieving positive cumulative rewards from exiting the door.

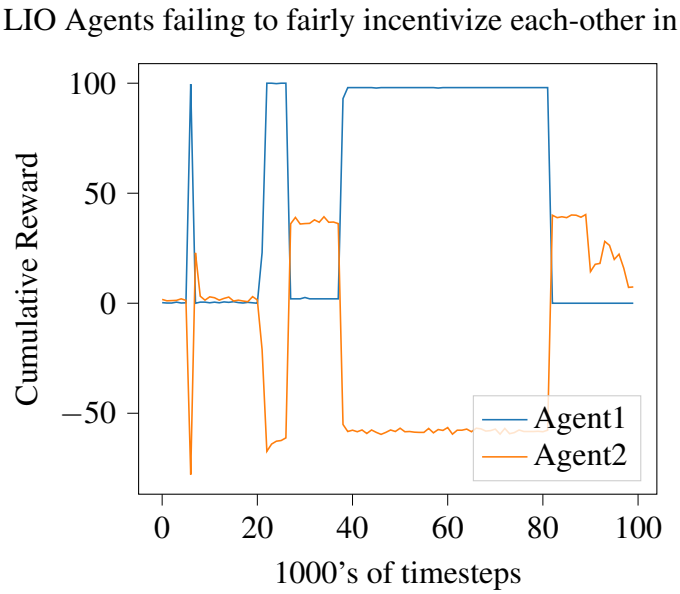


Figure 3.5: Social Learning fails to fairly incentivize in the game of Cleanup, using the LIO implementation. We observe that collectively, the agents achieve positive cumulative returns, however, at stark losses for one of the agents, the one making the incentivization. The two agents oscillate between one of them being incentivized to clean the river and the other collecting apples, but the agent collecting apples incurs a significant loss by incentivization. As soon as the harvesting agent tries to reduce its incentive to achieve a positive cumulative reward, the other agent stops collaborating. Hence, the agents are unable to stably cooperate and distribute their labour in this more sophisticated SSD. Future research should investigate what fair incentivization would look like in this game, and how to achieve it consistently.

et al [28]. We observe that the rejection mechanism successfully prevents any agent from getting exploited, however, the reward-giving agent is not able to stabilize to a fair incentive and both agents end up staying in the start state for a cumulative reward of 0.

Both of these experiments highlight the delicacy of the cooperation achieved through a Social Learning mechanism, and that fair incentivization is something that should be explored in future work.

To understand why fair incentivization is necessary, consider an experiment with a more sophisticated environment, Cleanup [7]. In Figure 3.5 we observe that although LIO agents learn to find a division of labour between themselves, with one agent cleaning the river and the other collecting the apples, there is a clear imbalance between the returns of the two agents: one of them achieves a return of +100 and the

other -60. However, in this experiment, the agent collecting the apples, and the one which is significantly more reward-giving, achieves a cumulative reward of -60, even though independent AC agents each achieve a cumulative reward of +5. Hence, in this environment, the reward-giving agent fails to learn a fair incentive to send to the reward-receiving agent which does the cooperative work. Since the reward-rejection mechanism isn't designed to prevent the over-spending of incentivized rewards by the reward-giving agent, fair incentives would not be created by the method proposed in Section 3.2, leaving this issue open for future work.

Chapter 4

Conclusion and Future Work

Social Learning is a powerful mechanism that allows Reinforcement Learning agents to learn cooperative behaviours in Sequential Social Dilemmas, a class of games that can model important societal issues such as cooperating to prevent runaway climate change or maintaining international relations. In a future where Multi-Agent Reinforcement Learning is a key part of the Artificial Intelligence digital infrastructure taking automated decisions, anticipating ways in which to safely deploy Social Learning, without creating exploitation in other learning agents, be they human or artificial, is extremely desirable and falls within a broader framework that has recently drawn significant attention, Cooperative AI [3].

This thesis has focused on a specific Safety issue with Social Learning, namely, exploitation, defined to be when a reward-receiving agent is no longer fairly incentivized for its cooperation. To address this issue, this thesis proposes a reject-reward mechanism, which successfully prevents exploitation in a simple SSD environment, a modified Escape Room game. Hence, the contributions of this thesis are the following. Firstly, the exploitation metric defined in this thesis measures this fair incentivization by comparing the cumulative returns resulting from following a cooperative Social Learning algorithm to the returns resulting from following a competitive independent Actor-Critic algorithm. Through this exploitation metric, this thesis shows that a current Social Learning implementation [28] fails to fairly incentivize. Then, this thesis presents a method to prevent exploitation, inspired by Yang et al[28], by extending the action space of the agents to include a reject-reward action taken together with a regular environment action. Through experimental evidence, the thesis shows how this metric successfully prevents exploitation, however, at the cost of preventing the agents from forming stable cooperative policies. Finally, the thesis highlights why fair

incentivization is an issue in Cleanup, a more sophisticated SSD, where the proposed method would not yet be adequate due to the exploitation happening in the reward-giving agent.

For future work, the reward-rejection mechanism can be based on the same two-step optimization process that LIO [28] uses in the reward-giving agent, but mirrored in the reward-receiving agent. More specifically, this proposed method would have the reward-receiving agent perform a two-stage optimization process wherein at the upper level the rejection function accounts for the recipients' policy optimization at the lower level as the result of accepting an incentive from a reward-giving agent. Since leveraging second-order gradient methods captures longer-term dependencies between policy network updates and cumulative reward changes, this method could mitigate exploitation and create fair incentivization. In turn, since both the reward-giving and the reward-receiving agents would perform two-stage optimization processes, a related area of work that could become relevant is Theory of Mind in Cooperative Reinforcement Learning [20].

Bibliography

[1] Kenshi Abe and CyberAgent. Off-policy exploitability-evaluation in two-player zero-sum markov games. In *Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems*, 2021.

[2] Nicolas Anastassacos, J. García, S. Hailes, and Mirco Musolesi. Cooperation and reputation dynamics with reinforcement learning. *Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems*, abs/2102.07523, 2021.

[3] A. Dafoe, Edward Hughes, Yoram Bachrach, Tantum Collins, Kevin R. McKee, Joel Z. Leibo, K. Larson, and T. Graepel. Open problems in cooperative ai. *NeurIPS 2020 Cooperative AI Workshop*, abs/2012.08630, 2020.

[4] Panayiotis Danassis, Zeki Doruk Erden, and B. Faltings. Improved cooperation by exploiting a common signal. *Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems*, abs/2102.02304, 2021.

[5] Ingy Elsayed-Aly, Suda Bharadwaj, Chris Amato, Rüdiger Ehlers, U. Topcu, and L. Feng. Safe multi-agent reinforcement learning via shielding. *Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems*, abs/2101.11196, 2021.

[6] Jones Granatyr, V. Botelho, O. Lessing, E. Scalabrin, J. Barthès, and F. Enembreck. Trust and reputation models for multiagent systems. *ACM Computing Surveys (CSUR)*, 48:1 – 42, 2015.

[7] Edward Hughes, Joel Z Leibo, Matthew Phillips, Karl Tuyls, Edgar Dueñez Guzman, Antonio García Castañeda, Iain Dunning, Tina Zhu, Kevin McKee, Raphael Koster, Heather Roff, and Thore Graepel. Inequity aversion improves cooperation in intertemporal social dilemmas. In S. Bengio, H. Wallach, H. Larochelle,

K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

[8] Natasha Jaques, A. Lazaridou, Edward Hughes, Çaglar Gülçehre, Pedro A. Ortega, D. Strouse, Joel Z. Leibo, and N. D. Freitas. Social influence as intrinsic motivation for multi-agent deep reinforcement learning. In *ICML*, 2019.

[9] Joel Z. Leibo, V. Zambaldi, Marc Lanctot, J. Marecki, and T. Graepel. Multi-agent reinforcement learning in sequential social dilemmas. *Autonomous Agents and Multi-Agent Systems*, abs/1702.03037, 2017.

[10] A. Lupu and Doina Precup. Gifting in multi-agent reinforcement learning. In *AAMAS*, 2020.

[11] Andrei Lupu and Doina Precup. Gifting in multi-agent reinforcement learning (student abstract). *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(10):13871–13872, Apr. 2020.

[12] Lyndon Benke Matthew Aitchison and Penny Sweetser. Learning to deceive in multi-agent hidden role games. *2nd International Workshop on Deceptive AI IJCAI2021, year=2021*.

[13] Stephen McAleer, John Lanier, Michael Dennis, P. Baldi, and Roy Fox. Improving social welfare while preserving autonomy via a pareto mediator. *ArXiv*, abs/2106.03927, 2021.

[14] Ramona Merhej and Mohamed Chetouani. Lief: Learning to influence through evaluative feedback. *ALA 2021*.

[15] Ramona Merhej, F. Santos, Francisco S. Melo, and F. C. Santos. Cooperation between independent reinforcement learners under wealth inequality and collective risks. *Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems*, 2021.

[16] Andreasa Morris-Martin, M. B. Vos, and J. Padget. Norm emergence in multiagent systems: a viewpoint paper. *Autonomous Agents and Multi-Agent Systems*, 33:706 – 749, 2019.

[17] Pierre El Mqirmi, F. Belardinelli, and Borja G. Leon. An abstraction-based method to check multi-agent deep reinforcement-learning behaviors. *Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems*, 2021.

[18] Y. Narahari. Game theory and mechanism design. 2014.

[19] Kamal Ndousse, Douglas Eck, Sergey Levine, and Natasha Jaques. Emergent social learning via multi-agent reinforcement learning. In *ICML*, 2021.

[20] D. Nguyen, S. Venkatesh, Phuoc Nguyen, and T. Tran. Theory of mind with guilt aversion facilitates cooperative reinforcement learning. *ArXiv*, abs/2009.07445, 2020.

[21] A. Rapoport. Prisoner's dilemma — recollections and observations. 1974.

[22] Charlotte Roman, Michael Dennis, Andrew Critch, and Stuart J. Russell. Accumulating risk capital through investing in cooperation. *Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems*, abs/2101.10305, 2021.

[23] Michael Sievers, A. Madni, Parisa Pouya, and Robert J. Minnichelli. Trust and reputation in multi-agent resilient systems*. *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*, pages 741–747, 2019.

[24] Eilon Solan and Nicolas Vieille. Stochastic games. *Proceedings of the National Academy of Sciences*, 112(45):13743–13746, 2015.

[25] Eugene Vinitsky, R. Koster, J. Agapiou, Edgar A. Duéñez-Guzmán, A. Vezhnevets, and Joel Z. Leibo. A learning agent that acquires social norms from public sanctions in decentralized multi-agent settings. *ArXiv*, abs/2106.09012, 2021.

[26] Adam Walker and M. Wooldridge. Understanding the emergence of conventions in multi-agent systems. In *ICMAS*, 1995.

[27] Anil Yaman, Nicolas Bredèche, Onur cCaylak, Joel Z. Leibo, and Sang Wan Lee. Meta-control of social learning strategies. *ArXiv*, abs/2106.10015, 2021.

[28] Jiachen Yang, Ang Li, Mehrdad Farajtabar, Peter Sunehag, Edward Hughes, and Hongyuan Zha. Learning to incentivize other learning agents. *NeurIPS 2020, arXiv preprint arXiv:2006.06051*, 2020.

[29] Matthieu Zimmer, Claire Glanois, Umer Siddique, and Paul Weng. Learning fair policies in decentralized cooperative multi-agent reinforcement learning. *arXiv preprint arXiv:2012.09421*, 2020.