Governing the Commons with Multi-Agent Reinforcement Learning

A Single-, Two- and Multi-Agent Reinforcement Learning model of common-pool resource appropriation

A dissertation presented for the degree of
Bachelor of Arts and Sciences: Sciences & Engineering

Department of Arts and Sciences
Candidate nr: YSBB5
University College London
United Kingdom
May 2019

Abstract

In the natural environment, groups of individuals are often faced with the problem of governing common-pool resources. Various studies have shown that stable local communities with limited influence of external forces in fact managed to sustain common resources such as fisheries, irrigation systems or grazing pastures over centuries. Nevertheless, the challenge arises when the problem is on a global scale. International cooperation failed to manage large-scale resources such as freshwater or marine ecosystems, ultimately leading to the depletion of biological diversity. A number of research projects in the field of social sciences, in particular, behavioral game theory, have been conducted to uncover aspects underlying human behavior making the cooperation possible on a local scale. A plethora of that work focuses on game theory based laboratory experiments. Although a number of such experiments contributed to deepening our understanding of common-pool resource appropriation, the results failed to capture the importance of spatial and sequential dynamics crucial for the human decision-making process. Aimed at bridging this gap, this work proposes a multi-agent reinforcement learning framework, allowing for the meticulous observation of the emergent behavior. Employing deep multi-agent reinforcement learning, we show how, through trial-and-error the artificial agents learn to sustainably govern the commons in a single-, two- and multi-agent setting. The general-sum Markov game created for the purpose of the dissertation mimics the common-pool resource appropriation problem enabling for flexible manipulation of agents' and environment's parameters. The experiments conducted by deploying independently learning agents in a number of specifically designed environments provide detailed insights on the range of emergent social outcomes and show that advanced human-like cognitive capabilities are not necessarily required for sustainable resource management. The report concludes with potential avenues for research using the framework established here and explicitly states the limitations of the study.

Preface

A distinguishing characteristic of human intelligence is that it encapsulates years of social interactions. Emerging from constant competition and cooperation over the course of centuries, it is a result of cumulative cultural evolution. Reflecting on this provided me with a new vision and a drive to better understand human behavior through building and observing the interactions between intelligent systems. Focusing on cooperation and competition, this dissertation sheds light on how such collective behavior emerges, by proposing methods utilizing artificial agents.

In retrospect, my initial interest in the topic of emergent behavior was spurred by my first-year economics and game theory classes. Though intriguing, I felt that the models of human decision-making put forth within these modules were overly simplistic and based upon numerous strict underlying assumptions. Moreover, they did not bring me closer to understanding how these models were established. With neither psychology nor evolutionary biology providing a satisfactory explanation in my own readings, I turned to the field of artificial intelligence. This prompted me to pursue a variety of quantitative courses in and beyond my degree. Eventually, this evolved into an interest for the field of reinforcement learning, specifically multi-agent reinforcement learning, which explores how agents learn to make decisions in a multi-agent setting. As a capstone of my undergraduate studies, this work draws on the undertaken modules and developed interests, which have equipped me with the necessary skills to undertake the dissertation.

To address the research area holistically, the project joins together two inherently interdisciplinary concepts, Multi-Agent Reinforcement Learning and the Tragedy of the Commons. Multi-Agent Reinforcement Learning, stemming from computer science and artificial intelligence, is a field at the intersection of mathematics, economics and psychology, incorporating mathematical and programming methods into the problem of scarce resource allocation (economics) under the bounded rationality assumption (psychology). The second concept, the Tragedy of the Commons spans multiple disciplines and has seen its applications in philosophy, economics, environmental science as well as many others. Only by aligning these two separate ideas, one is able to study the emergence of collective behavior. Specifically, the dissertation investigates how naïve artificial agents learn to achieve sustainability in a single-, as well as multi-agent setting. This is done by proposing novel grid-world environments developed using python, in which the agents learn to manage a renewable resource using the DQN algorithm authored by DeepMind, recreated for the purpose of the study.

The aim of the dissertation is to show that advanced cognitive capabilities are not necessarily required to prevent a complete depletion of the resources, hereafter referred to as the 'tragedy', and offer a framework for further study of this concept. A newly coined term for the emerging area to which this study falls into has been termed multi-agent reinforcement learning in social dilemmas.

By aligning concepts of distinct disciplines, the dissertation's findings can be beneficial to both social sciences (e.g. social psychology) and reinforcement learning (e.g. architecture of novel environments and algorithms), offering a unique method for the study of human collective decision-making.

Acknowledgements

The utmost thanks go to my dissertation supervisor, Professor Jun Wang and Dr Manuela Dal Borgo for their guidance on this culminating research project of my undergraduate degree. The author would also like to thank Dr Haifeng Zhang for his advice throughout the project, Ying Wen and Dr Jakob Foerster for fruitful discussions in the lead-up to this work as well as motivation for future work.

Deep gratitude goes to Greg Surma for software engineering mentorship. Finally, Sasha Mouzin and Jolee Tung for assisting me with topic scoping and proofreading drafts. For this, I could not be more appreciative.

Contents

1 Introduction 6

2 Methods 10

2.1 Definitions and Notations 10

2.1.1 The Tragedy of the Commons 10

2.1.2 The commons game 12

2.1.3 Markov games 13

2.2 Learning Algorithms 15

2.2.1 Deep multi-agent reinforcement learning 16

2.3 Social outcome metrics 20

2.4 Simulation methods 21

3 Results 23

3.1 Single-agent case 23

3.1.1 Experiment 1: Basic single-agent environment 24

3.1.2 Experiment 2: Complex single-agent environment 24

3.2 Two-agent case 25

3.2.1 Experiment 3: DQN agent versus DQN agent 26

3.2.2 Experiment 4: DQN agent versus random agent 27

3.3 Multi-agent case 28

3.3.1 Experiment 5: Multi-agent model of the Tragedy of the Commons 28

4 Discussion 31

4.1 Contributions to the field 31

4.2 Limitations 32

4.3 Future work 33

4.3.1 Implementation of opponent-aware algorithms 33

4.3.2 Influence of various parameters on the emergent behavior 34

4.3.3 Impact of Ostrom's rules on governing the commons . . . 34

4.4 Conclusion 35

A Parameters and scores 41

B Tragedy of the Commons - supplementary proof 42

Chapter 1

Introduction

Game theoretic models have provided a framework for understanding social interactions and have been fruitfully applied to a range of disciplines such as social sciences (Camerer, 2003, 2011), biology (Broom and Cannings, 2010) and artificial intelligence (Russel and Norvig, 2012). Abundant literature has documented how the study of such models have aided convergence and stabilization of socially preferable outcomes, offering insights into mechanisms of incentivizing individual agents and whole societies (Westley et al., 2016). Nevertheless, such models, often tend to reduce the model of the environment to a simplistic matrix, general-sum and constant-sum games. An example is the Iterated Prisoner's Dilemma, widely used in social science (Axelrod, 1998) and reinforcement learning (Busoniu et al., 2010) experiments. While the Iterated Prisoner's Dilemma offers a clear framework for understanding social dilemmas and serves as an important benchmark, the player's actions are limited to two actions: (1) cooperate or (2) defect, usually taken simultaneously. Therefore, failing to capture important spatial and temporal dynamics of the game. Similarly, the model of common-pool resource appropriation, often referred to as the Tragedy of the Commons, also suffers from such limitations (Hardin, 1968).

The core idea behind the common-pool resource (CPR) appropriation problem has been motivated by the natural environment, where natural resources, such as fisheries, irrigation systems or groundwater basins, are often common-pool, making it costly to exclude potential beneficiaries from accessing them (Ostrom et al., 1994). With the resource yielding finite blows of benefits, one's incentive to free ride (i.e. to avoid paying for the good), leads to the exhaustion of the good, hence the 'tragedy'. To take the example of common grazing pasture, this can be considered as a situation where the agent is faced with the decision to either act in self-interest adding one more animal to his herd, thus contributing to the depletion of the resource (here, grazing pasture), or cooperate by restraining from increasing the population of his herd. Since the reward goes entirely to the appropriator, and the cost is split equally among all the herdsmen on the pasture, in accordance with economic theory, a rational agent shall continue to increase the population of its herd as long as the gain from increasing the population size by one unit exceeds the marginal social cost associated with the depletion of the resource. Furthermore, even if a particular agent were to realise that draining the CPR leads to a socially sub-optimal outcome, it cannot single-handedly prevent the tragedy by altering its own behavior. Therefore, the agents always prefer to exploit the resource, regardless of its current state which in terms of game theory approach results in a Nash deficient equilibrium (Smith et al., 2007) and in theory, grim predictions on the cooperation over CPRs. In his publication conceiving the concept of the Tragedy of the Commons, Hardin (1968) concluded the essay with the phrase ”The population problem has no technical solution; it requires a fundamental extension in morality.”. This dissertation returns to this idea discussing relevant literature and proposing a framework using multi-agent reinforcement learning for the study of this problem.

Over the course of centuries, this issue of collective action for sustaining common-pool resources has inherently shaped history (Ostrom, 2015). Today, it remains equally important, with resources such as tropical forests, freshwater and marine ecosystems being regularly depleted. Despite such a bleak prognosis, the real-world provides a number of examples where stable local communities with limited influence of external forces managed to successfully sustain the CPR for a substantial amount of time, therefore preventing the tragedy (Dietz et al., 2003). Examples include irrigation systems in medieval Spain, communal tenure in mountain meadows and forests in Switzerland and Japan, as well as many others. Worth noting is that sustainable CPR management has been documented in a range of societies functioning within different natural and cultural settings (Ostrom, 2015).

A wide variety of studies have been conducted to uncover the underlying mechanisms which enabled the promotion of collective behavior preventing the tragedy. This includes institutional and policy design (Ostrom, 2015), evolutionary biology (Rankin et al., 2007), behavioral research (Camerer, 2011), and recently reinforcement learning (Perolat et al., 2017). Some of the research even discusses its applications in international agreements, such as the Montreal Protocol on stratospheric ozone (Parson, 2003). However, as pointed out by Perolat et al. (2017), the majority of the game-theoretic experimental research in this area has been focused on examining the choice of how much to appropriate (Ostrom et al., 1994). While such a framework has been largely successful, as argued by Leibo et al. (2017) in his work on sequential social dilemmas, it omits a number of real-world aspects, crucial for the common pool resource appropriation problem outlined below:

1. Common-pool resource appropriation happens over a course of time (i.e. is temporally extended).
2. Cooperativeness rarely is an atomic decision (e.g. cooperate or defect)

and may be a graded quantity.

3. Decisions to cooperate and restrain from acting in self-interest emerge over time, rarely arising simultaneously.

4. Seldom an agent has perfect information about state of the world. The decisions are usually made based on the partial observability.

5. The abundance of resources, enabling players to penalize each other and other parameters of the game contribute to the non-linear evolvement of player's strategies.

This work proposes an extension of the model developed by Perolat et al. (2017), with the emergence of collective action playing a crucial role. Newly programmed game models are then used for the experimental analysis of the model. Similarly, as in previously mentioned work, the CPR appropriation model could be divided into two sections; a CPR game environment capturing spatial and temporal dynamics designed to mimic real-world commons problem (Janssen and Ostrom, 2008, Janssen, 2010, 2014) and a multi-agent system of an independently learning artificial agents utilizing multi-agent reinforcement learning. The main motivation behind the study is to complement current research on the CPR appropriation by observing the emergence of collective action and its stability. We monitor how cooperation between simultaneously learning agents and their best responses evolve through trial-and-error over time in different environments. The framework presented here also enables the manipulation of numerous agent and environment parameters influencing players behavior. The results of learning dynamics are then thoroughly analysed using specifically designed social metrics, enabling for the evaluation of the social outcomes from the ethical standpoint (i.e. Gini coefficient, peacefulness).

The game design used as a model of CPR, has been inspired by numerous behavioral experiments by Janssen (2010; 2014, with Ostrom, 2008). Where the participant is placed in the interactive environment and faces the problem of maximizing his reward in a CPR appropriation setting. An advantage of such a framework is the ability to constantly monitor the behavior of the player and benchmark the results against other participants. This work also benefits from this aspect but employs artificial instead of human agents as decision-makers.

The behavior of each of the artificial agents is modelled by deep reinforcement learning. Reinforcement learning (RL) seeks to determine the agent's actions based on its state of the environment so as to maximise its cumulative reward. In other words, RL enables an agent to learn to pick an appropriate action given its current state and experience (Sutton and Barto, 1998). Seen as a candidate theory for animal-habit learning (Niv, 2009), RL has recently enjoyed notable success with solving complex tasks and overperforming human agents in multiple games such as Go (Silver et al., 2016) or Atari (Mnih et al., 2016), previously thought to be unattainable for machines. Furthermore, as previously mentioned, its structure allows for the observation of the learning process of agents, something not yet entirely possible with human agents.

For the purpose of the analysis outcome metrics measuring overall performance, inequality and other aspects are drawn and extended from the work of Perolat et al. (2017). Obtained results are then benchmarked against existing research on that topic and evaluated. Apart from these results, the present work also outlines possible avenues for further research and development, as well as states the limitations of the adopted approach.

While the approach proposed in this dissertation enriches the current framework and provides novel methods on the investigation of the collective action in CPR appropriation, it is important to note that this is also a highly reductionist model. A significant limitation is the rationality of the artificial agents, as in the real-world human and animal agents tends to often deviate from the rational choice, characterized by the so-called bounded rationality (Lara, 2015). Nevertheless, the compatibility of obtained results with previous studies offers a promising possible research avenue, applying multi-agent reinforcement learning to a plethora of social dilemmas. Thereby, contributing to the promotion of socially preferable outcomes, which could potentially be used for the design of policies and institutions. On the other side of the spectrum, work here presented argues that temporally extended social dilemmas such as the Tragedy of the Commons could be more suitably tested with multi-agent reinforcement learning algorithms.

The dissertation is organised as follows: Chapter 2 devoted to methods introduces the reader to the main paradigm of the Tragedy of the Commons and the mechanisms of the game environment, mimicking the problem of CPR appropriation. Further, in this section, the Markov Games framework used for modelling the game is described, together with the workings of deep reinforcement learning algorithms, determining agent's behavior. Chapter 3 presents the results of the study with respect to different environments and discusses the obtained outcomes with respect to relevant literature. Finally, Chapter 4 provides an overview of the dissertation, its contributions, implications for future studies as well as outlines its assumptions and limitations.

Chapter 2

Methods

Introducing the methodology followed in the study, this chapter gives an outline of the research techniques and concepts employed throughout the work. It provides necessary definitions and the research design that was chosen for the purpose of this research, including the reasoning behind such choice. In this chapter, the author aims to make the reader familiar with the concept of the study summarizing the Tragedy of the Commons from the game theoretic perspective and describing how it could be modelled with multi-agent reinforcement learning.

2.1 Definitions and Notations

2.1.1 The Tragedy of the Commons

To further introduce the reader to the concept of the tragedy of the commons, the formal of the game is presented below (Gibbons, 1992).

Following the example of the grazing pasture outlined in the introduction, assume there are n herders eligible to graze their cows on the common pasture. Each herder grazes a_i cows, hence the total number of cows grazed on the pasture is $A = a_1 + ... + a_n$. The cost of maintaining a cow is c and depends only on the number of cows grazed. Therefore one herder's cost of grazing a_i cows amounts to $a_i c$. Herder's value from grazing a cow, considering the total amount A grazed on the pasture is $v(A)$ per cow.

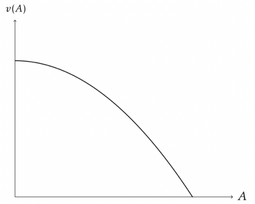


Figure 2.1: The relationship between $v(A)$ and A

Due to the limitations of CPR such as grazing pastures, there is a certain limit of cows that can be grazed on the field. We can think of this as that a cow requires a given amount of grass to survive, thus there can only be A_{max} cows grazed on the field. Therefore $A_{max} : v(A) > 0$ for $A < A_{max}$, but $v(A) = 0$ for $A \geq \mathrm{A}_{max}$. This implies that grazing an additional cow is valuable if and only if the total number of cows A is smaller than the limit A_{max}. Inspecting the properties of the function $v(A)$ for $A < A_{max}$ by taking its first and second derivative, we can see that $v'(A) < 0$ and $v''(A) < 0$, shown in figure 2.1. This means that the value per cow diminishes with more cows grazed on the pasture.

The herders are simultaneously deciding how many cows to graze, hence the action space of farmer i is $[0, \infty]$ (assuming cows are continuously divisible). This gives us the payoff of individual farmer i from grazing a_i cows as:

$$a_i v(a_1 + ... + a_{i-1} + a_i + a_{i+1} + ... + a_n) - ca_i \tag{2.1}$$

Where the value from grazing a_i cows depends on $a_i + a_{-i}$ standing for the number of cows grazed a_i by the farmer i and all other farmers a_{-i}.

Seeking to find the Nash equilibrium, where herders' best responses coincide, we maximise the equation (1) with respect to a_i. Assuming all other farmers also maximise their own payoffs, we obtain the first-order condition:

$$v(a_1 + a^*_{-i}) + a_i v'(a_i + a^*_{-i}) - c = 0 \tag{2.2}$$

where a^*_{-i} is everyone-that-is-not-farmer-$i's$ maximisation strategy. To find the number of cows grazed in the field, conditional on all of the n herders maximizing, we get:

$$v(A^*) + \frac{1}{n} A^* v'(A^*) - c = 0 \tag{2.3}$$

where $A^* = na^*_i = a^*_i + a^*_{-i}$, which has been done to be able to compare social and personal optima. This gives us the number of cows grazed on the pasture, A^*, when all the herders maximise their payoff, but we also need to find the social optimum, where we seek to optimise the collective payoff of all the herders. We proceed to do that similarly as for the single case of farmer i. However, now instead of a_i we consider A, the number of cows owned collectively by the herders. Hence, our optimisation problem becomes:

$$\max_{0 \leq A < \infty} Av(A) - Ac, \tag{2.4}$$

which through differentiation with respect to A and equating to 0 results in:

$$v(A^{**}) + A^{**} v'(A^{**}) - c = 0. \tag{2.5}$$

Here, A^{**} represents the collective optimum number of cows grazed on the pasture. Using equations (2.3) and (2.5) it can be proved, that $A^* > A^{**}$, implying that the number of cows grazed in Nash equilibrium resulting from each herder maximising his own payoff is higher than the one in the socially optimal outcome, making the Nash equilibrium a deficient one. The proof for $A^* > A^{**}$ can be found in the Appendix B.

Putting it in context; when all herders consider only their own incentives, the pasture (a CPR) becomes depleted. This is due to the fact that each individual does not take into account the effect of him adding an additional cow on other farmers' choices. Such a situation leads to *tragedy* and is detrimental to all of the herders. Much of the literature focused on the factors contributing to the successful governance of the commons such as institutional design, defining clear group boundaries, using graduated sanctions for those violating set rules and others. A robust summary of governing the commons has been authored by Ostrom et al. (1994) and included the principles crucial for sustaining CPRs. The next section focuses on describing a framework enabling further study of these rules.

2.1.2 The commons game

To mimic the CPR problem, the game environment proposed in the work of Perolat et al. (2017) has been recreated, making use of interactive setting described by Janssen (2010; 2014, with Ostrom, 2008). Here, the environment refers to a holistic term encompassing rules and characteristics of the game. The learning agents are deployed in an environment, with the goal of collecting "apples" (resources). Meaning that the agents receive a reward when collecting an apple. However, the regrowth of apples depends on the spatial configuration of the resource, with the higher density leading to a higher regrowth rate. Such a setting encourages agents to sustainably exploit the environment, as complete resource depletion diminishes future reward. Nevertheless, drawing on the economic theory described in the previous section, agents shall continue to appropriate the resource as long as the marginal gain from collecting the additional unit is bigger than the marginal social cost. Hence, to prevent the tragedy of the commons, thus lower future payoffs, the vast majority of the agents should learn to cooperate.

In order to learn more about the factors contributing to the cooperation, the game has been created, so as to enable manipulation of the parameters. The parameters could be divided between the environment based and agent-based. The environment parameters include resource abundance, referring to (1) the regrowth of the apples and (2) density of its allocation (how many apples are possible for collection). Furthermore, it is also possible to manipulate its size and space configuration such as placing of the map boundaries. The agent's parameters among others include its observability (what agent can see determines its state), cognitive capability (defined as the architecture of neural networks responsible for the agent's learning) and stored and refreshed with every iteration batch of experience used to train the agent (also known as replay buffer). Moreover, each agent also has the ability to punish other agents by using the laser beam as an action. Each agent is able to direct such a beam in a straight line of certain width along its current orientation. If the agent is hit by the beam (i.e. is in the area of other agent's beam) it is immediately removed from the game for N_{tagged} time frames. Being hit by the beam does not add nor subtract any reward for the tagging and tagged agent. However, it 'freezes' the hit agent preventing it from making any actions. This introduces punishment as a boundary constraint, as an agent can penalize other agents who wander to close to its position. Through controlling the N_{tagged} time frames as well as the length and width of the beam we can monitor how altering the penalty affects the outcome.

It is important to note that the proposed environment is solely focused on the problem of CPR appropriation involving the allocation of the flow; i.e. the regrowth of the resource. Thus, omitting the question of the supply of the stock of the CPR.

2.1.3 Markov games

The CPR game described above is modelled using Markov games framework (Littman, 1994) as a general-sum simultaneous move Markov game. Each agent in the environment chooses an action based on its partial observation determined by the game state space. Through interaction with the environment over the course of training, the agent must learn an appropriate policy. While in the single-agent case this is limited solely to the learning environment's properties, in the multi-agent scenario agents also need to learn to effectively cooperate or exclude other agents from collecting the resource to prevent ending up in a sub-optimal outcome. At each step of the game, the agent receives an individual reward (either 0 or 1) for its action and utilizes the experience to learn the optimal policy, thus maximizing its total reward.

As the settings evaluated in this work include single-, two- and N-player setting, the technical description of the Markov game below describes N-player option which could be easily narrowed down to the single- and two-player settings.

The N-player partially observable Markov game $\mathcal{M}$ is defined by a finite set of states $\mathcal{S}$. Each player's d-dimensional view on the state space is specified by the observation function $O : \mathcal{S} \times \{1,...,N\} \to \mathbb{R}^d$ and player's i individual observation space written as $\mathcal{O} = \{o^i | s \in \mathcal{S}, o^i = O(s,i)\}$. At each state, the agent's action space remains fixed with player's i action space denoted by $\mathcal{A}^i$. The transition from one state to another is a result of actions $a^1,...,a^n \in \mathcal{A}^1,...,\mathcal{A}^n$ of all players N and follows the stochastic transition function $\mathcal{T} : \mathcal{S} \times \mathcal{A}^1 \times ... \times \mathcal{A}^n \to \Delta(\mathcal{S})$. Here, $\Delta(\mathcal{S})$ represents the set of discrete probability distributions over $\mathcal{S}$. Player's i individual reward is defined as $r^i : \mathcal{S} \times \mathcal{A}^1 \times ... \times \mathcal{A}^n \to \mathbb{R}$.

Putting the notation in context, each agent interacts with the environment and through this experience, independently learns a long-term reward maximizing behavior based on its own rewards $r^i(s, a^1, ..., a^N)$ and observations $o^i = O(s, i)$. The behavior of the agent here referred to as the policy (Sutton and Barto, 1998) is understood as a set of rules used by an agent to decide what actions to take and denoted as $\pi^i : \mathcal{O}^i \rightarrow \Delta(\mathcal{A}^i)$, which is written as $\pi(a^i|o^i)$. For simplicity, a set of all the agents' actions, observations and policies at a particular time-step t are written respectively as $\vec{a_t} = (a^1_t, ..., a^N_t)$, $\vec{o_t} = (o^1_t, ..., o^N_t)$ and $\vec{\pi_t}(.|\vec{o_t}) = (\pi^1_t(.|o^1_t), ..., \pi^N_t(.|o^N_t))$. Using the notation above, we can define each agent's payoff it is looking to maximize as follows:

$$V^i_{\vec{\pi}}(s_0) = \mathbb{E}\Bigg[\sum_{t=0}^{\infty} \gamma^t r^i(s_t, \vec{a_t}) | \vec{a_t} \sim \vec{\pi_t}, s_{t+1} \sim \mathcal{T}(s_t, \vec{a_t})\Bigg]. \tag{2.6}$$

To elaborate on the formula above, the value function $V^i_{\vec{\pi}}(s_0)$ denotes agent's i expected return ($\mathbb{E}$) if it starts in the very first state of the environment s_0 and then act according to a particular policy π ever after. In this case, the value function is defined as the expected infinite-horizon discounted return $\sum_{t=0}^{\infty} \gamma^t r^i(s_t, \vec{a_t})$, which in other words is the sum of all rewards obtained by an agent discounted by how far off the future, here determined by the time-step t are obtained by a factor $\gamma \in (0, 1)$. The simple idea behind the discount factor, stemming from economics (Dutta, 1991) is that a reward now is valued higher than the same reward in the future. Moreover, it also is mathematically convenient as on the contrary to an infinite-horizon sum of rewards which is likely to not converge to a finite value, an infinite-horizon discounted sum of rewards does converge to a finite value under reasonable conditions. Therefore facilitating equation approximation. The reward function $r^i(s_t, \vec{a_t})$ depends on the current state of the world s_t and a set of all actions taken by the agents $\vec{a_t}$ at the time-step t. Finally, the sum of the rewards are based on agents' actions sampled from the policy π_t, and the next state of the world is sampled from the stochastic transition function $\mathcal{T}(s_t, \vec{a_t})$. That is, agents' rewards depend on the state they are in s_t and all their actions $\vec{a_t}$. While the next state is determined by the stochastic transition function $\mathcal{T}(s_t, \vec{a_t})$ containing rules and probabilities on going from one state to another. The goal of the agent is to discover strategy that leads to subsequently appropriate states, thus maximizing its total reward.

The catch is that one agent's reward and the state of the world is determined not only by its actions and the environment, which is stochastic by itself, but also by the actions of all the other agents. Thus, impeding agent's convergence to an optimal policy. In order to solve this problem, the agents employ learning algorithm such as deep Q-learning (Mnih et al., 2015, Mousavi et al., 2018) described in the next section.

2.2 Learning Algorithms

Single- and multi-agent learning has been the topic of extensive literature, with the majority of it focused on the problem of finding and implementing an optimal policy (Shoham et al., 2007, Busoniu et al., 2010, Mousavi et al., 2018, Albrecht and Stone, 2018). An example of that is a wide variety of work devoted to solving matrix games (Axelrod, 1998, Sandholm and Crites, 1996) as well as more broadly constant-sum and significantly more challenging general-sum cases. In notation by Shoham et al. (2007), such an approach towards the multi-agent learning problem is called a *prescriptive view* and aims to answer the question of ”what should agent do?”. The *prescriptive view*, whose applications’ include planning, resource management, robotics and others (Mousavi et al., 2018) encompasses a broad area of research which has become a highly active field in the last years (Busoniu et al., 2010). A particular rise has been noted in deep multi-agent reinforcement learning, where a number of novel algorithms, methods and benchmark environments have been developed (Samvelyan et al., 2019, Foerster et al., 2017, Lowe et al., 2017). Much of that work concerns the topic of opponent modelling (Albrecht and Stone, 2018), including recursive reasoning (Wen et al., 2019), Bayesian inference (Chalkiadakis and Boutilier, 2002, Foerster et al., 2018), actor-critic methods (Lowe et al., 2017) and others (Bloembergen et al., 2015, Han and Gmytrasiewicz, 2018, Rabinowitz et al., 2018, He et al., 2016). This will be further elaborated on in the *Discussion* chapter.

On the other side of the spectrum, there is the *descriptive view*, which unlike the previously discussed *prescriptive view*, studies the learning dynamics and emergent behavior of agents in the presence of other agents. In other words, we focus on the question of ”what social effects emerge when each agent uses a particular learning rule?”. Following this view, instead of designing novel learning algorithms, the experiments conducted in this work aim to make use of already existing algorithms and provide insights on the emergent social outcomes, resulting from the game. Furthermore, we are careful to avoid making unrealistic assumptions implied in some studies on the emergence of multi-agent coordination (Yu et al., 2015) such as agent’s knowledge of the Markov models underlying the game.

Thanks to recent advancements in the field of reinforcement learning, we are able to study the learning dynamics using deep multi-agent reinforcement learning. This is motivated not only by its abilities and recent successes in solving complex problems but also by its resemblance to the theory of animal habit-learning (Niv, 2009). The algorithm is constructed so as to provide the agents with a myopic (i.e. short-sighted) view, using selective memory uniformly sampled from its set of last experiences to choose an appropriate action and update its strategy. The details of the algorithm are described in the following section.

2.2.1 Deep multi-agent reinforcement learning

In order to maximize its long-term future reward, a reinforcement learning agent interacts with the environment through trial-and-error aiming to improve its performance over the course of iterations. A commonly used algorithm is the deep Q-network (DQN) (Mnih et al., 2015). The DQN has been a breakthrough in dealing with imperfect information, partially observed games with a relatively large action, and state space observable from raw input feature planes. Thanks to its architecture, the algorithm has been able to form and follow long-term strategies, often requiring thousands of steps.

Q-learning

The core of the DQN is the Q-learning (Watkins and Dayan, 1992). The Q-learning agent interacts with the environment and learns the action-value function $Q(s,a)$, specifying how good the taking of an action is at a particular state and corresponding to a complete plan of action. Through trial-and-error, Q-learning constructs a memory table $Q[s,a]$ storing the Q-values for all possible combinations of s and a. In other words, we sample action from the current state. After a move, we record reward R and the resulting new state s'. Using the memory table, we choose the subsequent action a' with maximum $Q(s',a')$. Hence, taking a single move a and the reward from it R creates a one-step look ahead $R+Q(s',a')$, giving us the target:

$$target = R(s,a,s') + \gamma \max_{a'} Q_k(s',a') \tag{2.7}$$

where k is the iteration number. Putting it in context of the game, the long-term target refers to the sustainability, maximizing agents' rewards. The full Tabular Q-learning algorithm with α denoting the learning rate can be found below.

Deep Q-network with experience replay and target network

Although the Q-learning algorithm successfully solves a variety of simple tasks and games, if the combinations of states and actions are too large, the memory and the computation requirement for Q will be too high. Hence the algorithm fails to perform in complex real-world problems. DQN addresses this issue, using deep neural networks to approximate $Q(s,a)$. Hence, instead of remembering the Q-value, the algorithm generalizes the approximation of the Q-value function. Nevertheless, the Q-value function approximation is not the sole solution. If we were to update the parameters with every step (as in Tabular Q-learning), the algorithm would not converge to an optimal strategy. This is due to the fact that in reinforcement learning, both the input (i.e. agent's observations) and the target change constantly during the process of training. Hence updating the Q target in every step would artificially magnify it, resulting in the algorithm ”chasing its own tail”, by recursively updating the target. To address this problem, we slow down the changes in both input and output,

Algorithm 1 Tabular Q-learning

```
Start with $Q_0(s,a)$ for all $s$,$a$
Get initial state $s$
for $k = 1, 2, ...$ till convergence do
    Sample action $a$, get next state $s'$
    if $s'$ is terminal then
        $target = R(s,a,s')$
        Sample new initial state $s'$
    else
        $target = R(s,a,s') + \gamma \max_{a'} Q_k(s',a')$
    end if
    $Q_{k+1}(s,a) \leftarrow (1-\alpha)Q_k(s,a) + \alpha[target]$
    $s \leftarrow s'$
```

allowing the function to evolve. This is done through the introduction of (1) experience replay and (2) target network. Experience replay stores the T last experienced transitions $\{(s,a,r_i,s')_t : t = 1, ...T\}$ of the agent in replay memory $\mathcal{D}$, where the transitions include its state, action, reward and next state. At every time-step, the algorithm randomly samples N_b (here, 8) transitions from the replay memory and uses it to update its policy. The constantly refreshed replay buffer (also called experience replay) allows the Q-network to adapt to the changing data distribution resulting from the learning process. Moreover, it also makes the DQN significantly more data efficient. The second improvement is the creation of the target network. Having two deep networks (initial network and target network), we can use one to retrieve the Q- values and the second one to include all the updates in the training. The target network is updated every N^- episodes. This addresses the second problem of the DQN, as we avoid chasing a moving target and restrain the function from changing too quickly. Using the parameters of the initial network θ_i and target network θ_i^- we can formulate the loss of the new Q-learning update at iteration i with the following function:

$$L_i(\theta_i) = \mathbb{E}_{(s,a,r,s')\sim\mathcal{U}(\mathcal{D})}\left[\left(r + \gamma max_{a'} Q(s',a';\theta_i^-) - Q(s,a;\theta_i)\right)^2\right] \tag{2.8}$$

with $(s,a,r,s') \sim \mathcal{U}(\mathcal{D})$ standing for experience sampled uniformly from the replay buffer D. In summary, the algorithm stores its recent memory in the replay buffer D and at every time-step, randomly selects N_b transitions from it and uses them to choose an appropriate action. Its target (here, sustainability) is updated every N^- episodes (1 episode = 1000 time-steps).

In our game, each agent has its own deep Q-network with experience replay and target network, expressed as $Q_i : \mathcal{O}_i \times \mathcal{A}_i$. The observation of a player i are denoted by $o_i = O(s,i)$. Although players' observations differ, for the sake of simplicity, we use $Q_i(s,a) = Q_i(O(s,i),a)$. All of the agents, use a ϵ-greedy policy with diminishing over iterations exploration rate ϵ. A crucial aspect and a subject of extensive literature (Ozcan et al., 2011), the trade-off between exploration and exploitation determines how much the agent should devote to exploring new, possibly better strategies instead of exploiting the currently best-known action.

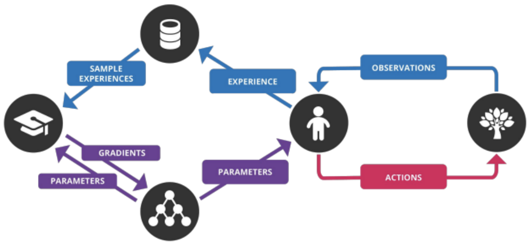


Figure 2.2: Graph depicturing the relationships between different parts of the DQN (Mnih, 2015).

While spending too much time on the exploration might result in the lost opportunity cost (as we could possibly start exploiting the best-known strategy earlier), a small exploration value might trap the agent in a local minimum. Hence ending with a sub-optimal outcome. The solution is the previously mentioned ϵ-greedy policy adopted by the agents during the learning process. With $\mathcal{U}(\mathcal{A}_i)$ standing for a sample from the uniform distribution over action space of the player, we can parameterize such policy of the i'th agent by

$$\pi_i(s) = \begin{cases} \mathcal{U}(\mathcal{A}_i) & \text{with probability } \epsilon \\ argmax_a \in \mathcal{A}_i Q_i(s,a) & \text{with probability } 1-\epsilon. \end{cases} \tag{2.9}$$

There is a number of methods for controlling the ratio of exploration and exploitation in agent learning (Ozcan et al., 2011). This includes linear, non-linear and other techniques. Here, during the training process, the exploration rate decays linearly with every time-step. Finally, the policy update given a sample of transitions from the replay buffer of each agent is formulated as

$$Q_i(s,a) \leftarrow Q_i(s,a) + \alpha \left[r_i + \gamma max_{a' \in \mathcal{A}_i} Q_i(s',a') - Q_i(s,a) \right] \tag{2.10}$$

with the update being controlled by the learning rate α.

Learning Independence

To treat agents as separate entities, their learning is independent of each other. Therefore one agent has no notion of similarity to other agents and regards other players as part of the non-stationary environment. Although this may impede the convergence to an optimal solution, it allows us to observe the

Algorithm 2 DQN algorithm with experience replay and target network

```
Initialize replay memory $D$ to capacity $N_r$, $N_b$ - training batch size (number of transitions uniformly sampled from the replay memory)
Initialize action-value function $Q$ with random weights $\theta$
Initialize target action-value function $\hat{Q}$ with weights $\theta^- = \theta$ and target action-value function update frequency $N^-$
for episode $= 1, 2, ..., M$ do
    Initialize frame sequence $x \leftarrow ()$
    for $t = 1, 2..., T$ do
        Set $s \leftarrow x$
        With probability $\epsilon$ select a random action $a_1$
        Otherwise select $a_t = argmax_a Q((s_t), a; \theta)$
        Execute action $a_t$ in emulator and observe reward $r_t$ and image $x' = s$
        Set $s_{t+1} = s$ and add the transition tuple $(s_t, a, r, s_{t+1})$ to D replacing the the oldest tuple if $|D| \geq N_r$
        Sample a random mini-batch of $N_b$ tuples $(s_t, a, r, s_{t+1}) \sim \mathcal{U}(\mathcal{D})$
        Set $y_j = \begin{cases} \mathrm{r}_j & \text{if episodes terminates at step } j+1 \\ \mathrm{r}_j + \gamma max_{a'} \hat{Q}(s_{j+1}, a'; \theta^-) & \text{otherwise} \end{cases}$
        Perform a gradient descent step on $\left(y_j - Q(s_j, a_j; \theta)\right)^2$ with respect to the network parameters $\theta$
        Replace target parameters $\theta^- \leftarrow \theta$ every $N^-$ steps
    end for
end for
```

emergence of cooperation between separate learning agents as without it, the problem would resemble a coordination planning problem, following the prescriptive view. Furthermore, drawing on the work of Leibo et al. (2017, 2018), and (Perolat et al., 2017), as agents do not infer beliefs or use recursive reasoning to predict actions of other agents, the independence assumption could be seen as a way of bounded rationality. Where the bounded rationality term refers to the idea that human agents often tend to deviate and follow non-rational heuristic rules in decision-making (Simon, 1972).

Nevertheless, this also is a severe limitation of the model, as humans use recursive reasoning on an everyday basis. To provide an example, a farmer willing to determine the price he should set on his crop, will intuitively think of the other farmers on the market and use his predictions to set an optimal price. We will return to this limitation in the final chapter, discussing potential alternatives.

2.3 Social outcome metrics

The value function (i.e. the algorithm performance), the standard metric used in problems of prescriptive view does not allow to fully analyse the resulting learning dynamics. Therefore, to fully characterize the emergent behavior in the commons game, we draw on metrics proposed by (Perolat et al., 2017). The metrics include four key criteria measuring (1) sum of all rewards denoted as efficiency, (2) inequality (3) sustainability and (4) peacefulness. Using such indicators allows us to fully track the system, thus providing the data required for the analysis and benchmark of the results.

Formalizing the metrics, let us consider N independent players, T the number of time-steps in an episode, $t = 1, 2, ..., T$ and the sequence of rewards and observations of agent i denoted respectively as $\{r_i^t | t = 1, 2, ..., T\}$ and $\{o_i^t | t = 1, 2, ..., T\}$. We formulate the metrics as:

(1) *Efficiency (E)* - designed to measure the total sum of rewards of all of the players in the environment in a given episode. Given that the sum of rewards of i'th agent in episode e is denoted as R_e^i, the efficiency metric is defined as $E_e = \sum_{n=1}^{N} R_e^i$.

(2) *Inequality (I)* - based on the Gini coefficient (Gini, 1912), this metric allows for tracking the statistical distribution of the rewards between the agents. The Gini coefficient in a particular episode is defined as $I_e = 1 - \frac{\sum_{i=1}^{N} \sum_{j=1}^{N} |R_e^i - R_e^j|}{2N \sum_{i=1}^{N} R_e^i}$.

(3) *Sustainability (S)* - lets us monitor the depletion of the resource through monitoring the time-step at which the episode ends, allowing us to see how often and how quickly the agents manage to completely deplete the resource. We can represent it as $S_e = T_e$.

(4) *Peacefulness (P)* - defined as the number of time-steps at which the agents were not 'frozen' by other agents' beam (i.e. untagged agent steps), measures the effective aggressiveness of the agents. $P_e = \frac{NT - \sum_{i=1}^{N} \sum_{j=1}^{N} I(o_t^i)}{T}$ where $I(o) = \begin{cases} 1 & \text{if } o = \text{tagged} \\ 0 & \text{otherwise.} \end{cases}$

2.4 Simulation methods

Concluding the methods chapter, this section describes the details of the simulations conducted for the purpose of this study. Drawing strongly on the material of previous sections, it presents a holistic view on the environments and parameters of the games, where the agents were deployed.

All of the games implemented for the purpose of this study are 2D grid-world based games developed using the python language. Environments presented here consist of two single-player games, one two-player game and one multi-agent game. The creation of single- and two-player games have been done to observe how the agents converge to a sustainable strategy in a less complex environment. Moreover, due to significantly higher computational costs of a multi-agent scenario, it was successfully utilized for the purpose of model tuning (i.e. choosing appropriate hyperparameters such as learning rate, replay buffer size and other). The action-space $\mathcal{A}^i \in \mathbb{R}^8$ in the multi-agent setting consisted of 8 actions: (1) move forward, (2) move backwards, (3) move left, (4) move right, (5) turn left, (6) turn right, (7) shoot beam and (8) stay. The action space in the single-agent setting has been the same, with the exception of the action (7) shoot beam exclusion, unnecessary for this particular context. Observations of the agent depend on the environment size, its position in the grid and the true state of the game s_t. The observation function is defined as $O(s,i) \in \mathbb{R}^{3 \times X \times Y}$ (RGB) where X stands for how far does the observation window extends ahead of the agent and Y for sideways observation window [1]. In any environment and at any stage of the game the agent i had only partial observability (i.e. could not see a whole grid). Furthermore, the beam used in the multi-agent setting could not extend beyond the agent's observation window. The agents, apart from being able to see appropriately coloured squares for apples (resource), map boundaries, beam, were also able to see its direction and its current position appeared blue in their own view, on the contrary to other agents' positions showing up in red. The default episode length in every environment was 1000 time-steps t. Unless the game terminated earlier due to the complete depletion of the resource. The default N_{tagged} determining the time for which the agents hit by the opponents' beam were frozen was 25 time-steps. Finally, the per-time-step respawn probability p_t of the apple (resource) depends on the number of non-collected apples in a ball of a radius

[1] $X, Y > 0$ and Y is an odd number extending for the same number of squares both sides.

of 2 around its location a. Denoting the number of the resource stock around a as $B = B_2(a)$, we formulate the probability as a function of B by:

$$p_t(B) = \begin{cases} 0 & \text{if } B = 0 \\ 0.01 & \text{if } B = 1 \text{ or } 2 \\ 0.05 & \text{if } B = 3 \text{ or } 4 \\ 0.1 & \text{if } B > 4. \end{cases} \tag{2.11}$$

The agent learns to take appropriate actions by taking a preprocessed image of its observation window, depicting the true state of the world s_t. The image is then flattened and used as an input to the feed forward neural network (LeCun et al., 2015) which is effectively used to approximate the $Q-$value function of an agent and can be loosely regarded as the 'brain' of an agent. Due to the simplicity of the grid-world games, the Convolutional Neural Nets (CNNs) (LeCun et al., 1998) were not necessary here. The default neural network consisted of two hidden layers, each with 32 units interleaved with rectified linear (ReLU) layers. The final layer had 8 (in the two- or multi-agent setting) or 7 units (single-agent setting), one for each action. To ensure sufficient exploration, agents used an ϵ-greedy policy decaying linearly over 500000 timesteps ($\approx$ 500 episodes), with ϵ-max being 1 and ϵ-min equal to 0.1. During policy evaluation performed at the end of training the $epsilon$ was set to 0.05. The per-time-step discount rate amounted to $\gamma = 0.995$. The network has been optimized with *Adam* optimizer and the target network was updated every three episodes. On the contrary to the default DQN in the initial release by (Mnih et al., 2015) which used *Huber loss*, here the mean square error loss (MSE) was applied, due to the more stable convergence it offered (van Hasselt et al., 2016).

Crucially, the training has been re-run multiple times, in order to find suitable parameters. While it is possible that other set of parameters could provide similar or superior training results in terms of performance, smoothness or sample efficiency, the parameters above offer a relatively good outcome. To ensure benchmark is possible, all of the agents regardless of setting have used the same set of parameters. The full set of parameters can be found in the appendix. The training has been conducted on the local computer and Google Cloud Platform due to the computational costs. The environment has been built enabling others to change game parameters such as N_{tagged} time frames and the code can be found online[2]. In the upcoming weeks, the repository is to be organized and released as an open-source environment for research purposes.

[2] https://github.com/macwiatrak/cpr-dqn

Chapter 3

Results

This chapter describes the experiments conducted and presents its findings. In the order from the least to the most complex, single-, two- and multi-agent examinations are outlined. Each experiment section includes brief characterization of the employed environment and the observed behavior of the agent. Visual aids depicting environment maps and agents' training return curves, (where each curve is the mean of multiple runs and the shaded area surrounding it is the variance) are offered in every section.

3.1 Single-agent case

In order to show how the agent learns to sustainably manage the resource and restrain from over-depletion, two different environments were created. First one (basic) with a relatively easier optimal strategy has been created for hyperparameter tuning and algorithm testing. The second one (complex) tested the agent's capability to converge to a significantly more challenging strategy.

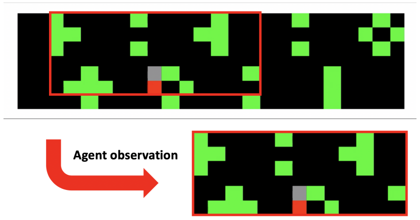


Figure 3.1: Single-agent complex environment map with agent's observation.

The observation window for both environments was $O(s,i) \in \mathbb{R}^{3\times6\times13}$. Hence the agent could see six grid squares ahead (including the row of its current position on the grid) and six grid squares from side to side. As mentioned in the section on simulation methods, the agents in the single-agent environment were not able to use the beam action. The size of both environments utilized here was 24 × 7 grid squares.

3.1.1 Experiment 1: Basic single-agent environment

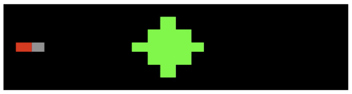


Figure 3.2: Single-agent basic environment map.

In the first environment, the spatial configuration of the apples was configured in such a way, that the apples are concentrated in one area on the grid. This structure enabled the agent to relatively easily discover where the reward yielding resource is placed on the grid. Moreover, the regrowth rate has also been extraordinarily high as the apples were directly neighbouring with each other. Nevertheless, as a consequence of such resource concentration, it was also easy for the agent to completely deplete the resource in a short amount of time, thus reducing the total reward.

The results of such configuration are visible on the agent's training curve in the environment. We can see that an initial phase of exploration represented by the steady rise in the first episodes is followed by a sudden drop in the performance around episode number 300. This corresponds to the phase in training when the agent learns where the resource is located, subsequently depleting it and leading to a decline in rewards. Successive episodes show how the agent manages to escape the *tragedy* and converge to an optimal, sustainable strategy. The video shows the learned strategy after 1000 episodes[1].

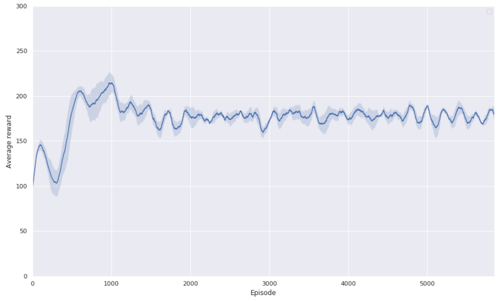


Figure 3.3: Single-agent basic environment; training curve averaged over multiple runs.

3.1.2 Experiment 2: Complex single-agent environment

On the contrary to the single-agent environment introduced above, the apples on this map were spread evenly on the grid and concentrated in five unit cross-shaped centres. Another property of this environment was that the average reward of the random policy was similar to the one were the agent greedily collects the apples prompting its depletion. Therefore, the agents did not record

[1] https://youtu.be/dNutUOUpCjI

a dip in reward during training as in the basic single-agent case. Instead, the training curve increases steadily, finally converging to the optimal strategy. Interestingly, the optimal policy requires the agent to circulate around the grid in a given pattern. The video to which the link can be found in the footnote shows the discussed learned policy after 1000 episodes[2]. It should be also noted that the policies shown on the video include minimal epsilon during training amounting to 0.1. This means that on average 10% of agent's actions are random. Therefore non-optimal behavior is to some extent justified.

Successful convergence in this environment, which can be viewed as more arduous in comparison to the basic one, has proven that the agent can learn to sustainably manage a resource in a single-agent case. The next environments pose a significantly bigger challenge through the introduction of additional agents.

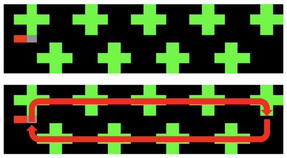


Figure 3.4: Single-agent complex environment map (top) with the optimal path marked by red arrows (bottom).

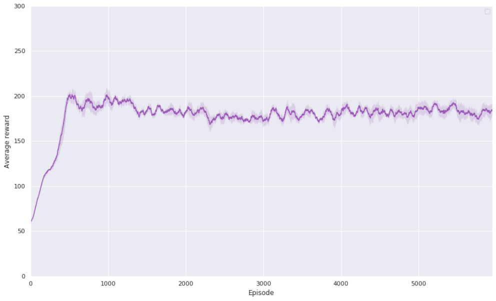


Figure 3.5: Single-agent complex environment; training curve averaged over multiple runs.

3.2 Two-agent case

To simulate the two-player game, we have used the complex environment map, used in the single-agent case. The player's observation window was the same $O(s,i) \in \mathbb{R}^{3\times6\times13}$ (RGB), with the players' starting positions being at the opposite sides of the map. Additionally, the agents were able to use the beam, which when used extended to 5 grid squares in the direction faced by the tagger and was 3 squares wide. In this section, two experiments were conducted, (1) two DQN agents and (2) DQN agent with a random agent.

[2] https://youtu.be/Wq4Zs4W2Y9k

3.2.1 Experiment 3: DQN agent versus DQN agent

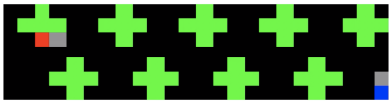


Figure 3.6: Two-player map with the agent one in red and agent two in blue.

In this experiment, two independently learning agents were trying to maximize its reward. The main metric we are using here is the cumulative return (sum of rewards obtained by both agents) curve. Looking at the graph we can see that the shape of the curve resembles the one in the single-agent case in the complex map. Nevertheless, the mean of the cumulative reward for an episode is approximately 15 points higher (8%) in the two-agent case in comparison to the single-agent scenario. Furthermore, the average reward of both players over 6000 episodes in this environment is almost equal to one another.

Analysing the recording[3] from this scenario, we can see that the agents learn not to excessively use the beam and limit to using beam only when close to the opponent. This, however, does not mean that the agent restrains from using beam at all times. More, that they prefer to focus on collecting the resource, rather than chasing the opponent in order to tag it. Moreover, the agents seem to split the map and spend the majority of the time harvesting the apples in this particular territory, rather than circling the environment as in the single-agent case. Concluding, although the strategies on the same map differ between single- and two-player scenarios, this experiment shows that the DQN agents are able to sustainably manage the resource even when they have to compete for it with other players.

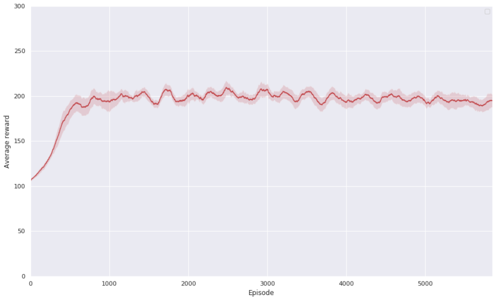


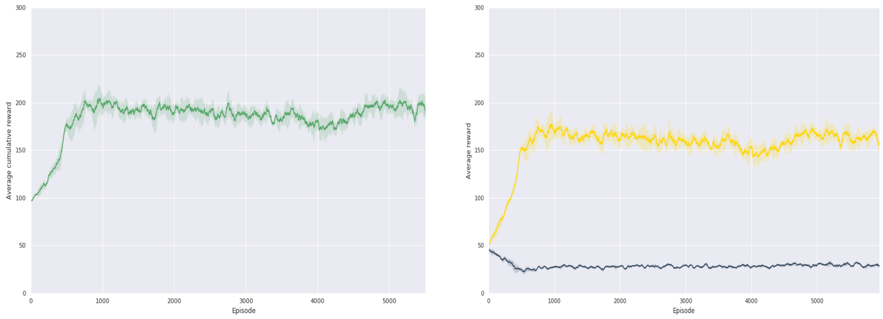
Figure 3.7: Two-player environment; DQN versus DQN on a complex map; training curve averaged over multiple runs.

[3] https://youtu.be/4ODv5ygWmek

3.2.2 Experiment 4: DQN agent versus random agent

Having tested the case of two DQN agents, the next task was to evaluate whether a DQN agent is able to learn to successfully manage the resource with the presence of a less robust agent. For this reason, we have conducted an experiment, where the DQN agent, was competing for the resource with a random agent, whose probability of taking an action a was uniformly distributed over the action space, $a \sim \mathcal{U}(\mathcal{A})$. In other terms, a naïve DQN agent was competing for a resource with a completely irrational, *noise* random player.

Over the course of training, the DQN agent learns to dynamically adapt to the new environment which depends on the actions of the random agent. Interestingly, the DQN agent does not focus on excluding the random player from harvesting the apples, but on collecting and adjusting its policy, tagging it occasionally, when nearby. Thus allowing it to focus on the optimal circular policy, visible on a video[4]. The exclusion of the random agent is visible on the training reward curve where the average return of the random agent diminishes until ≈ 500 episode. Simultaneously, the return of the DQN agent surges during that time. Additionally, freezing the random agent using the beam helps in preventing from completely depleting the cross-shaped centres of apples, a highly sub-optimal outcome, which may happen when the agent is acting randomly.



(**a**) Cumulative reward of DQN and Random agent.

(**b**) Returns of DQN agent (yellow) and Random agent (black).

Figure 3.8: Two-player environment; DQN versus random on a complex map; training averaged over multiple runs.

[4] https://youtu.be/4ODv5ygWmek

3.3 Multi-agent case

Using the agents and its parameters evaluated on the single- and two-player cases, we now move to the multi-agent scenario. For its purpose, the environment has been enlarged to 24 × 12 grid squares. The map increase was followed by the increase in the observation window, with the new agent's observation $O(s,i) \in \mathbb{R}^{3\times7\times15}$ (RGB) extending 7 grid squares ahead and 7 side to side. The beam was adjusted accordingly to extend 6 grid squares ahead in the direction faced by the tagger with the width of 3. The default number of agents in the environment was 4, all starting from the same corner of the map. Additionally, the number of time-steps over which the *epsilon* decayed linearly was increased from 500000 (≈ 500 episodes) to 1000000 (≈ 1000 episodes) to reduce the probability of agents ending up in a local minimum.

3.3.1 Experiment 5: Multi-agent model of the Tragedy of the Commons

Contrary to the single- and two-agent cases, the returns do not grow steadily over the course of training (except a small dip in the performance in experiment 1). Instead, the agents spend a considerably longer time on the initial exploration phase characterized by the randomness of their actions. Interestingly, the time span around ≈ 500 episode, when the agents learn how to efficiently collect the apples, yet not deplete it proves to yield the highest results. The supplementary videos show agents' exploratory [5] and optimal policy [6]. The analysis of the emergent behavior using the social outcome metrics as well as game recordings could be described using the terminology developed by Perolat et al. (2017).

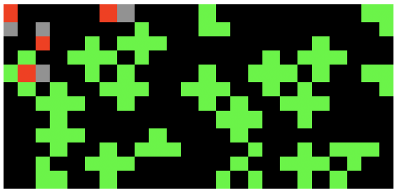


Figure 3.9: Multi-agent environment map, with the agents in red.

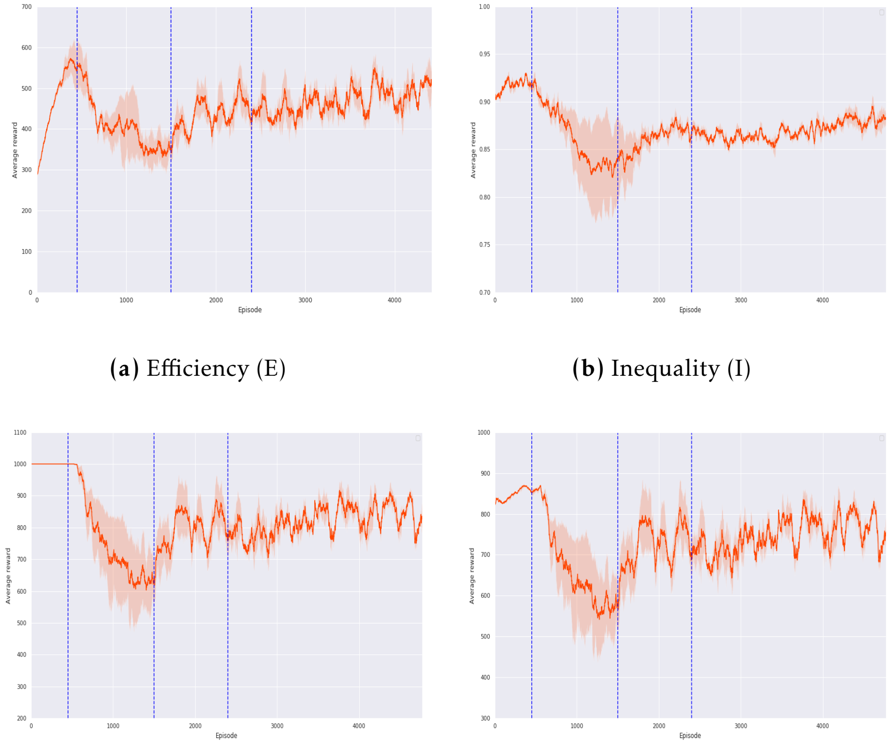
Emergent social outcomes

Analysing the outcome metrics characterizing agents' learning, over the course of training the game moves through 4 distinct phases. The stages are separated on the graph by blue vertical lines. The first one, to which we refer to as *exploration*, relates to the training from the beginning until ≈ 500 episode. In this phase, agents learn how to collect the resource and tend to restrain from excessive aggressiveness, with the peace metric steadily rising. Acting partly randomly, the agents do not lead to a complete depletion of the resource with the CPR stock maintaining a healthy, stable level. As a small gain in learning

[5] https://youtu.be/h1IWzxaNSdg

[6] https://youtu.be/Onb-ME-e5H4

leads to considerable improvement in the behavior, the equality metric slowly decreases. *Exploration* could be briefly summarized as a phase where agents gradually learn the properties of the environment, without being exceedingly greedy.



(a) Efficiency (E)

(b) Inequality (I)

(c) Sustainability (S)

(d) Peace (P)

Figure 3.10: Multi-agent environment (all DQN agents); social outcome metrics over the course of the training on a multi-agent map; training averaged over multiple runs.

The following phase, *tragedy* coincides with the end of the *epsilon* decay length. Spanning approximately between episodes 500 and 1500 marks a drastic decline in agents' returns (here, efficiency curve). The *tragedy* represents a stage, at which the agents learn how to effectively collect apples, thus leading to an exhaustion of the CPR. Accompanied by sudden drops in sustainability, equality and peacefulness, the phase shows how each agent tries to collect as many apples as possible, aggressively tagging other agents and not considering its regrowth. In terms of game theory, the agents act as rational agents should, continuing to deplete the resource.

The next phase, regarded as *combat* is characterized by largely fluctuating, however, steady rise in agents' efficiency. Starting around 1500 and finishing at ≈ 2400 episode denotes a stage, at which agents recognize the sub-optimal outcome arising from greedily collecting the resource and try to prevent that. Nevertheless, not in a peaceful manner. The extensive fluctuations in efficiency, sustainability and peace metrics show how agents aggressively try various methods to increase their returns. Not yet capable of highly efficient tagging of other agents and managing the CPR, the episode results exhibit high volatility. The only stable metric in this stage is the equality metric, which climbing after *tragedy*, stabilizes at around 0.86 Gini coefficient, indicating relatively high equality (in Gini coefficient 1 - perfect equality; 0 - perfect inequality). This stage could be understood as a phase at which in the agents realise faults in their behavior but are not capable of implementing the optimal policy.

As learning continues, the game moves into its final phase, *cultivation*, which begins at ≈ 2400 episode and continues indefinitely. With the efficiency, sustainability, and peacefulness metrics stabilizing, the agents manage to learn efficient tagging, not allowing the opponents to promptly deplete the resource. The inequality metric, maintains at a stable level of 0.87. This phase indicates, how agents learn to sustainably manage the resource with the use of the beam.

These emergent social outcomes characterized by 4 phases, shows the learning process of agents, which leads to sustainable CPR appropriation. Interestingly, the results above highlight the *combat* phase, not mentioned by Perolat et al. (2017). Its presence and importance has been confirmed by testing the game with multiple parameters including discount rate, loss and optimizer. In all cases, the *combat* phase remained part of the training. Another interesting finding is the fact that the highest returns were recorded during the *exploration* phase. This shows that the learning might not necessarily improve the returns. Further discussion of the results and its implications, can be found in the following chapter.

Chapter 4

Discussion

The conducted experiments manage to reproduce and extend the results obtained by the DeepMind team, simultaneously producing additional novel single- and two-player environments. Moreover, the created, open-sourced multi-agent framework can be treated as a base for further research in the field of multi-agent reinforcement learning in social dilemmas. The sections below summarize the results of the work, outlining its contributions, stating its limitations and potential avenues for future work. Finally, it concludes the dissertation with a brief synopsis of the research in the field of multi-agent reinforcement learning in sequential social dilemmas.

4.1 Contributions to the field

As an interdisciplinary study, this work is aimed at several audiences; the first one that we generalize to social sciences and the second one being the area of reinforcement learning, specifically multi-agent reinforcement learning. For the social sciences audience, the *descriptive view* offers an opportunity for monitoring agents' learning process. This dissertation did not aim to create a model which is inherently incentivized to follow a specific policy, adopted by multiple game-theoretic approaches (Axelrod, 1998, Nowak and Sigmund, 1993), but to create an environment fostering research into the circumstances under which the strategies emerge. This enables looking at how agents' often heuristic strategies evolve over the learning process and how in turn these strategies affect outcomes such as sustainability. Crucially for investigating the factors influencing the common-pool resource appropriation, this framework allows for the controlled study of the arising behavior. In the real-world, however, there are a variety of cultural, geographical and social factors, which are often difficult or impossible to recreate (Ostrom et al., 1994, Cox et al., 2010). While the interactive environment proposed and tested by the likes of Janssen et al. (2008, 2010, 2014) and Bednarik et al. (2019) provides the ability to compare and reproduce the results adding a valuable research dimension to the CPR problem, it does not grant the possibility of observing the impact of various parameters. Therefore, the framework proposed here introduces a new perspective to the Tragedy of the Commons literature. The specific parameters possible for manipulation are described in the upcoming section on future work.

The second audience for which the study is beneficial is reinforcement learning. The obtained results show how learning may not necessarily lead to higher returns. As with the case of multi-agent environment where the agents' performance during the exploration phase exceeded its score at the later stages of training. Unlike widely used in multi-agent reinforcement learning environments taking on a *prescriptive view*, the framework captures a broader range of characteristics of resulting behavior. This highlights the need for establishing social dilemma based games as a benchmark for the reinforcement and multi-agent reinforcement learning algorithms. Moreover, it shows the importance of devising novel measurement metrics, as applied here social outcome metrics to general-sum Markov games. Nevertheless, the point is not to argue that the proposed environments are superior to the ones focusing solely on the metric of performance such as matrix games (i.e. prisoner's dilemma, stag hunt, chicken), or recently devised Hanabi (Bard et al., 2019) card game challenge. These games are inherent for assessing the performance of current state-of-the-art technique and the framework described here was created to complement, rather than replace them.

4.2 Limitations

Although this multi-agent reinforcement learning model of governing the commons provides a new spectrum on the CPR appropriation problem, it is a highly limited approach. Both environment-wise and agent-wise, the framework omits a range of crucial aspects. This section outlines and critically reflects on these limitations laying ground for the next section devoted to future work.

From the perspective of the environment, the game neglects the process of bargaining and negotiation that in the real world often leads to collective decisions (Carpenter, 2000). Real-world examples of successful CPR management show that establishing tools and institutions facilitating negotiations and bargaining is a key factor. Apart from helping with setting up rules as a result of negotiations, such institutions or tools often fulfil the role of monitoring, penalizing, conflict-resorting entities. Indeed, four of eight rules on governing the commons devised by Ostrom et al. (1994) summarizing the circumstances around sustainable CPR appropriation are directly related to the establishment and character of such institutions. Furthermore, the tested maps are very simplistic. While they serve as a reasonably good proof of concept, the literature points out at the considerable influence of the barriers, size and the spatial distribution on social outcomes, some of which could be implemented with the use of the current game environment.

Turning to the reductions connected with agents, the main limitation of the model is that agents lack recursive reasoning about each other's beliefs, actions and information. Human agents tend to rely on this heavily, especially in cooperation problems, basing their decision on a chain of beliefs, biases and heuristic rules. An example of that is the extensively documented impact of reciprocity on cooperation, determining the final outcome (Nowak and Sigmund, 2005, Hilbe et al., 2018). Furthermore, the proposed environment does not offer many insights into the process of self-organisation and collective behavior. Although the agents manage to successfully govern the commons, their behavior shows little sign of collective behavior, an essential factor in the documented examples of governing the commons. This is a major aspect differentiating the framework from a real-life scenario. The agents' parameters also pose a potential technical problem. While the initial tests, as well as external research (Leibo et al., 2017), show that monitoring how changing game parameter's (i.e. respawn probability of the resource, N_{tagged} time frames) affects the outcome gives interpretable and reliable results, the same cannot be entirely said for the hyperparameters of the agents' underlying model. Altering hyperparameters such as replay buffer size, batch size and learning rate provides strongly varying results. This is natural for the model, however, it could require a further investigation in order to better generalize its influence on the emergent behavior in the Tragedy of the Commons.

4.3 Future work

The established open-sourced framework offers multiple opportunities for further research in the areas of the Tragedy of the Commons, social dilemmas and reinforcement learning. Below possible avenues for development are listed, which either make use only of the available game architecture or require reasonable extensions to be made. It is important to note that the list below is limited and should be treated more as guidelines for future development.

4.3.1 Implementation of opponent-aware algorithms

While using independently learning algorithms such as the DQN can be seen as a form of bounded rationality and all studies investigating multi-agent reinforcement learning in social dilemmas shall consider testing independent algorithms in their experiments, applying novel opponent-aware algorithms could yield interesting results. It could be especially valuable to observe how multi-agent reinforcement learning algorithms with centralized learning and decentralized execution such as MADDPG (Lowe et al., 2017), LOLA (Foerster et al., 2017) or QMIX (Rashid et al., 2018) perform in comparison to a decentralized learning decentralized action algorithm, like PR2 (Wen et al., 2019), which employs probabilistic recursive reasoning to account for other players' behavior. This could not only appear beneficial for the purpose of learning what influences agents' behavior in the CPR problem and to what outcomes it leads but also to serve as a test and benchmark for the opponent-aware algorithms themselves. Another avenue is studying how these agents with differing underlying models play against each other.

4.3.2 Influence of various parameters on the emergent behavior

Agents' and environments' parameters have a massive influence on the players' behavior and the outcome of the game. In order to enable studying its impact the framework allows for seamless manipulation in environments' and agents' parameters including respawn probability rate, N_{tagged} time frames, length and width of the beam, size of the environment, agent's observation window, discount rate γ, *epsilon* max, min and *epsilon* linear decay length, learning rate α, replay buffer size N_r, training batch size N_b, target network update frequency N^- and neural network architecture. As shown by Leibo et al. (2017) and Austerweil et al. (2016), agents' parameter manipulation could help in advancing a number of hypotheses in social psychology, such as the emergence of collective action. Potential extensions include the influence of increasing the number of units in the neural network, which could be loosely interpreted as increasing agents' cognitive capabilities or raising the N_{tagged} time frames number.

4.3.3 Impact of Ostrom's rules on governing the commons

Major factors contributing to the successful governance of the commons has been summarized by Ostrom in a list of 8 broad rules (Ostrom, 2015). Evaluating the relative importance of some of these rules using the proposed framework could provide valuable insights and another spectrum to the studies of these principles (Cox et al., 2010).

Here, 3 of these rules are evaluated. (1) *Clearly defined boundaries* could be simulated placing barriers such as walls/boundaries or penalizing the agents who wander outside of the certain region. Reasonable proof of concept has already been tested (Perolat et al., 2017) and could be further extended. (2) *Monitoring*, instead of partial, the agent could have full observability allowing them to oversee the state of the resource (i.e. level of depletion) and behavior of other agents. Another avenue would be to allow agents to access a live episode scoreboard of other agents and enabling them to differentiate between them. The idea is that at a stage of the training, the agents would be able to recognize and penalize over-depleting players. (3) *Use of graduated sanctions for rule violators*, one could think of disabling the beam action for the players in the environment and for example introducing only one player with the ability to harshly penalize resource draining agents, similar to the approach introduced by Baumann et al. (2018). A separate design could look at automatically penalizing the agents with the return above a certain level in a given time-span.

4.4 Conclusion

This dissertation reproduces, enriches as well as outlines potential avenues for further improvements in the application of reinforcement learning research to social dilemmas, providing novel framework for the study of the sustainable common-pool resource appropriation. The general method of tracking emergent behavior used here, is widely applicable and could be utilized not only to extend the Tragedy of the Commons, but to a variety of other game theoretic models. Returning to the question posed at the beginning of this work, whether the problem of the commons ”[...] requires a fundamental extension in morality” (Hardin, 1968), through the use of artificial agents and trial-and-error learning, it is evident that this is not necessarily the case. Indeed, sustainable CPR management might not depend upon advanced human-like cognitive capabilities of the players, nor require any sort of morality.

Word count: 9,879
Number of math inlines: 251

Bibliography

Albrecht, S. V. and Stone, P. (2018), 'Autonomous agents modelling other agents: A comprehensive survey and open problems', *Artificial Intelligence* **258**(September 2017), 66–95.

Austerweil, J. L., Brawner, S., Greenwald, A., Hilliard, E., Ho, M., Littman, M. L., Macglashan, J. and Trimbach, C. (2016), How Other-Regarding Preferences Can Promote Cooperation in Non-Zero-Sum Grid Games, *in* 'AAAI Symposium on Challenges and Opportunities in Multiagent Learning for the Real World.'.

Axelrod, R. (1998), The evolution of strategies in the iterated prisoner's dilemma, *in* 'Evolutionary Computation: The Fossil Record'.

Bard, N., Foerster, J. N., Chandar, S., Burch, N., Lanctot, M., Song, H. F., Parisotto, E., Dumoulin, V., Moitra, S., Hughes, E., Dunning, I., Mourad, S., Larochelle, H., Bellemare, M. G. and Bowling, M. (2019), 'The Hanabi Challenge: A New Frontier for AI Research'.
URL: *http://arxiv.org/abs/1902.00506*

Baumann, T., Graepel, T. and Shawe-Taylor, J. (2018), 'Adaptive Mechanism Design: Learning to Promote Cooperation'.
URL: *http://arxiv.org/abs/1806.04067*

Bednarik, P., Linnerooth-Bayer, J., Magnuszewski, P. and Dieckmann, U. (2019), 'A Game of Common-pool Resource Management: Effects of Communication, Risky Environment and Worldviews', *Ecological Economics* .

Bloembergen, D., Tuyls, K., Hennes, D. and Kaisers, M. (2015), 'Evolutionary dynamics of multi-agent learning: A survey'.

Broom, M. and Cannings, C. (2010), 'Evolutionary Game Theory', *Encyclopedia of Life Sciences* pp. 503–505.

Busoniu, L., Babuska, R. and Schutter, B. D. (2010), Multi-agent reinforcement learning : An overview Multi-Agent Reinforcement Learning : An, *in* 'Innovations in Multi-Agent Systems and Applications – 1', Vol. 19.

Camerer, C. F. (2003), 'Behavioural studies of strategic thinking in games'.

Camerer, C. F. (2011), 'Progress in Behavioral Game Theory', *Journal of Economic Perspectives* .

Carpenter, J. P. (2000), 'Negotiation in the commons: Incorporating field and experimental evidence into a theory of local collective action', *Journal of Institutional and Theoretical Economics-Zeitschrift Fur Die Gesamte Staatswissenschaft* .

Chalkiadakis, G. and Boutilier, C. (2002), Coordination in Multiagent Reinforcement Learning: A Bayesian Approach, *in* 'Proceedings of the 2nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS-03)'.

Cox, M., Arnold, G. and Tomás, S. V. (2010), 'A review of design principles for community-based natural resource management'.

Dietz, T., Ostrom, E. and Stern, P. C. (2003), 'Struggle to Govern the Commons', *Science* .

Dutta, P. K. (1991), 'What do discounted optima converge to?: A theory of discount rate asymptotics in economic models', *Journal of Economic Theory* **55**(1), 64–94.

Foerster, J. N., Chen, R. Y., Al-Shedivat, M., Whiteson, S., Abbeel, P. and Mordatch, I. (2017), 'Learning with Opponent-Learning Awareness'.
URL: *http://arxiv.org/abs/1709.04326*

Foerster, J. N., Song, F., Hughes, E., Burch, N., Dunning, I., Whiteson, S., Botvinick, M. and Bowling, M. (2018), 'Bayesian Action Decoder for Deep Multi-Agent Reinforcement Learning'.
URL: *http://arxiv.org/abs/1811.01458*

Gibbons, R. (1992), *Game Theory For Applied Economists*.

Gini, C. (1912), *Variabilità e mutabilità*.

Han, Y. and Gmytrasiewicz, P. (2018), 'Learning Others ' Intentional Models in Multi-Agent Settings Using Interactive POMDPs', *The advances in Neural Information Processing Systems 31* (Nips), 69–76.

Hardin, G. (1968), 'The Tragedy of the Commons', *Science* **162**(June), 1243–1248.

He, H., Boyd-Graber, J., Kwok, K. and Daumé, H. (2016), 'Opponent Modeling in Deep Reinforcement Learning', **48**.
URL: *http://arxiv.org/abs/1609.05559*

Hilbe, C., Šimsa, Š., Chatterjee, K. and Nowak, M. A. (2018), 'Evolution of cooperation in stochastic games', *Nature* .

Janssen, M. (2014), 'The role of the state in governing the commons', *Environmental Science and Policy* **36**(4), 8–10.

Janssen, M. A. (2010), 'Introducing ecological dynamics into common-pool resource experiments', *Ecology and Society* **15**(2), 8.

Janssen, M. A. and Ostrom, E. (2008), 'TURFS in the lab: Institutional Innovation in Real-Time Dynamic Spatial Commons', *Inequality, Cooperation and Environmental Sustainability* .

Lara, A. (2015), 'Rationality and complexity in the work of Elinor Ostrom', *International Journal of the Commons* **9**(2), 573–594.

LeCun, Y., Haffner, P., Bottou, L. and Bengio, Y. (1998), Object Recognition with Gradient-Based Learning.

LeCun, Y., Hinton, G. and Bengio, Y. (2015), 'Deep learning', *Nature Methods* **13**(1), 35.

Leibo, J. Z., Perolat, J., Hughes, E., Wheelwright, S., Marblestone, A. H., Duéñez-Guzmán, E., Sunehag, P., Dunning, I. and Graepel, T. (2018), 'Malthusian Reinforcement Learning'.
URL: *http://arxiv.org/abs/1812.07019*

Leibo, J. Z., Zambaldi, V., Lanctot, M., Marecki, J. and Graepel, T. (2017), 'Multi-agent Reinforcement Learning in Sequential Social Dilemmas'.
URL: *http://arxiv.org/abs/1702.03037*

Littman, M. L. (1994), Markov games as a framework for multi-agent reinforcement learning, *in* 'Machine Learning Proceedings 1994'.

Lowe, R., Wu, Y., Tamar, A., Harb, J., Abbeel, P. and Mordatch, I. (2017), 'Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments'.
URL: *http://arxiv.org/abs/1706.02275*

Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D. and Riedmiller, M. (2016), 'Playing Atari with Deep Reinforcement Learning', *IJCAI International Joint Conference on Artificial Intelligence* **2016-Janua**, 2315–2321.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S. and Hassabis, D. (2015), 'Human-level control through deep reinforcement learning.', *Nature* .

Mousavi, S. S., Schukat, M. and Howley, E. (2018), 'Deep Reinforcement Learning: An Overview', *Lecture Notes in Networks and Systems* **16**, 426–440.

Niv, Y. (2009), 'Reinforcement learning in the brain', *Journal of Mathematical Psychology* .

Nowak, M. A. and Sigmund, K. (2005), 'Evolution of indirect reciprocity'.

Nowak, M. and Sigmund, K. (1993), 'A strategy of win-stay, lose-shift that outperforms tit-for-tat in the Prisoner's Dilemma game', *Nature* .

Ostrom, E. (2015), *Governing the commons: The evolution of institutions for collective action*.

Ostrom, E., Gardner, R. and Walker, J. (1994), Rules, Games, and Common Pool Source problems, *in* 'Rules, Games, and Common Pool Resources'.

Ozcan, O., Alt, J. and Darken, C. J. (2011), 'Balancing Exploration and Exploitation in Agent Learning', *Proceedings of the Florida Artificial Intelligence Research Society Conference* pp. 97–98.

Perolat, J., Leibo, J. Z., Zambaldi, V., Beattie, C., Tuyls, K. and Graepel, T. (2017), 'A multi-agent reinforcement learning model of common-pool resource appropriation', (Nips).
URL: *http://arxiv.org/abs/1707.06600*

Rabinowitz, N. C., Perbet, F., Song, H. F., Zhang, C., Eslami, S. M. A. and Botvinick, M. (2018), 'Machine Theory of Mind'.
URL: *http://arxiv.org/abs/1802.07740*

Rankin, D. J., Bargum, K. and Kokko, H. (2007), 'The tragedy of the commons in evolutionary biology'.

Rashid, T., Samvelyan, M., de Witt, C. S., Farquhar, G., Foerster, J. and Whiteson, S. (2018), 'QMIX: Monotonic Value Function Factorisation for Deep Multi-Agent Reinforcement Learning'.
URL: *http://arxiv.org/abs/1803.11485*

Russel, S. and Norvig, P. (2012), *Artificial intelligence—a modern approach 3rd Edition*.

Samvelyan, M., Rashid, T., de Witt, C. S., Farquhar, G., Nardelli, N., Rudner, T. G. J., Hung, C.-M., Torr, P. H. S., Foerster, J. and Whiteson, S. (2019), 'The StarCraft Multi-Agent Challenge'.
URL: *http://arxiv.org/abs/1902.04043*

Sandholm, T. W. and Crites, R. H. (1996), 'Multiagent reinforcement learning in the Iterated Prisoner's Dilemma', *BioSystems* .

Shoham, Y., Powers, R. and Grenager, T. (2007), 'If multi-agent learning is the answer, what is the question?', *Artificial Intelligence* .

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., Dieleman, S., Grewe, D., Nham, J., Kalchbrenner, N., Sutskever, I., Lillicrap, T., Leach, M., Kavukcuoglu, K., Graepel, T. and Hassabis, D. (2016), 'Mastering the game of Go with deep neural networks and tree search', *Nature* .

Simon, H. A. (1972), Theories of bounded rationality, *in* 'Decision and organization'.

Smith, C. A. B., Neumann, J. V. and Morgenstern, O. (2007), 'Theory of Games and Economic Behaviour', *The Mathematical Gazette* .

Sutton, R. S. and Barto, A. G. (1998), Sutton & Barto Book: Reinforcement Learning: An Introduction, Technical report.

van Hasselt, H., Guez, A., Hessel, M., Mnih, V. and Silver, D. (2016), 'Learning values across many orders of magnitude', (Nips).
URL: *http://arxiv.org/abs/1602.07714*

Watkins, C. J. and Dayan, P. (1992), 'Technical Note: Q-Learning', *Machine Learning* .

Wen, Y., Yang, Y., Luo, R., Wang, J. and Pan, W. (2019), 'Probabilistic Recursive Reasoning for Multi-Agent Reinforcement Learning', pp. 1–20.
URL: *http://arxiv.org/abs/1901.09207*

Westley, F. R., McGowan, K. A., Antadze, N., Blacklock, J. and Tjornbo, O. (2016), 'How game changers catalyzed, disrupted, and incentivized social innovation: Three historical cases of nature conservation, assimilation, and women's rights', *Ecology and Society* .

Yu, C., Zhang, M., Ren, F. and Tan, G. (2015), 'Emotional multiagent reinforcement learning in spatial social dilemmas', *IEEE Transactions on Neural Networks and Learning Systems* **26**(12), 3083–3096.

Appendix A

Parameters and scores

Experiment scores		
Experiment	Average score of random agent	Average score of DQN agent over training
Experiment 1	32.47	174.65
Experiment 2	59.53	177.45
Experiment 3	92.57	191.72
Experiment 4	28.9	184.07
Experiment 5	264.53	491.17

Agents' parameters							
Experiment	Replay buffer N_r	Training batch size N_b	Discount rate γ	Learning rate α	Epsilon [min,max]	Epsilon decay length (in steps)	Target network update N^-
Experiment 1	100000	8	0.995	0.00001	[0.1,1]	500000	3
Experiment 2	100000	8	0.995	0.00001	[0.1,1]	500000	3
Experiment 3	100000	8	0.995	0.00001	[0.1,1]	500000	3
Experiment 4	100000	8	0.995	0.00001	[0.1,1]	500000	3
Experiment 5	100000	8	0.995	0.00001	[0.1,1]	1000000	3 or 10

Appendix B

Tragedy of the Commons - supplementary proof

Proof of $A^{*} > A^{**}$

Proof by contradiction, assume $A^{*} \leq A^{**}$. If this holds, then $v(A^{*}) \geq v(A^{**})$ [See Figure 2.1] as $v' < 0$. Thus, implying that the relationship between $v(A)$ and A is negative, therefore the bigger the value of A, the smaller the value of $v(A)$. Furthermore, as $v'' < 0$, visible on Figure 2.1, $0 > v'(A^{*}) \geq v'(A^{**})$. Hence due to the nature of the relationship between $v(A)$ and A (concave), the slope of $A^{*}(v'(A^{*}))$ is smaller in absolute value than the slope of A^{**}. Thus the value of $v'(A^{**})$ is more negative compared to $v'(A^{*})$. Moreover, $\frac{A^{*}}{n} < A^{**}$.

Therefore, assuming $A^{*} \leq A^{**}$, $v(A^{*})$ will at all time be be larger than $v(A^{**})$, $v'(A^{*})$ will always be more positive compared to $v'(A^{**})$ and finally $\frac{A^{*}}{n} < A^{**}$. Working with these assumptions one can see that the left hand side of the equation (2.3) is strictly larger than the left hand side of the equation (2.5). As this is impossible as both equal zero, $A^{*} \leq A^{**}$ cannot hold proving $A^{*} > A^{**}$.