Navigating the N-Person Prisoner's Dilemma: From the Tragic Valley to the Collaborative Hill

Chris Tcaci[1] and Chris Huyck[1]

Middlesex University, London NW4 4BT UK
M00674787@mdx.ac.uk and c.huyck@mdx.ac.uk
https://cwa.mdx.ac.uk/chris/chrisroot.html

Abstract. The N-Person Iterated Prisoners' Dilemma (N-IPD) is an excellent environment to explore collaboration. This paper shows that the voting mechanism is crucial in determining whether sets of agents collaborate, or defect. When each agent can vote against each other agent individually, the agents become cooperative much more easily, ascending the Collaborative Hill. When the agents vote against have only one vote each round, they tend to defect, descending into the Tragic Valley. This is shown with static decision policies, and with policies that learn using reinforcement learning. Fortunately, when agents retain enough history, they can become collaborative even with one vote each round. This is all due to the shape of the reward space.

Keywords: N-Person Prisoners' Dilemma · Reinforcement Learning · Emergence of Cooperation · Multi-Agent Systems.

1 Introduction

The Prisoners' Dilemma (PD) serves as a foundational paradigm in game theory, illustrating the conflict between individual rational self-interest and mutually beneficial collective action [1]. In its simplest form, two individuals, unable to communicate, must independently choose whether to cooperate or defect. While mutual cooperation yields a good outcome for both, each agent has an individual incentive to defect, often leading to both defecting, which yields a poor outcome for both. The Iterated Prisoners' Dilemma (IPD), where the game is played repeatedly, opens the door for cooperation to emerge through strategies based on reciprocity, as famously demonstrated by Axelrod's tournaments where Tit-for-Tat proved remarkably successful [1].

However, many real-world social and economic dilemmas—ranging from managing common resources to international climate agreements—involve more than two interacting parties. The N-Person Iterated Prisoners' Dilemma (N-IPD) generalizes the IPD to scenarios with N agents [2,3].

This paper first provides a brief background on the N-IPD and relevant learning approaches (section 2). It then describes simulations with static policies (section 3) and simulations with reinforcement learning (section 4). Section 5 is a brief discussion of simulations that enable systems with neighbourhood voting to escape the Tragic Valley. This is followed by a discussion and conclusion.

2 Background and Related Work

This section briefly reviews some key concepts from game theory, focusing on the N-Person Prisoners' Dilemma (N-IPD), and introduces the common agent strategies and Multi-Agent Reinforcement Learning (MARL) approaches relevant to this paper.

2.1 The Prisoners' Dilemma

The Prisoners' Dilemma [4,1] is a widely used to study collaboration. There are two prisoners and they are given the option to turn the other in. So both have the option to collaborate, and not turn the other in, or defect and turn the other in. That is each has a choice C or D. Each is given a reward based on their decision and the decision of the other. If they both collaborate, they both are given R, a reward. If both defect, they are both given P, a punishment. If one defects, and the other collaborates, the defector is given T, a temptation, and the collaborator is given S, a sucker's payoff.

		Prisoner 1	
		Collaborate	Defect
Prisoner 2	Collaborate	$R_{1,2}$	$T_1 S_2$
	Defect	$T_2 S_1$	$P_{1,2}$

Table 1. The table represents the outcomes for the four scenarios when two prisoners vote. When both collaborate, both get R, and when both defect, both get P. When one collaborates and one defects, the defector gets T and the collaborator gets S payoff.

Axelrod and Hamilton [1] restrict the values so that equations 1 and 2 are followed. A standard set of values, used below, is $T = 5$, $R = 3$, $P = 1$ and $S = 0$.

$$T > R > P > S \tag{1}$$

$$R > (S + T)/2 \tag{2}$$

The IPD merely plays the tournament over and over. This gives the agents a chance to develop their own policy.

The Tit-for-Tat (TFT) strategy proved to be most successful. The strategy is to collaborate when the opponent collaborates, then defect in the round after they defect. Note that the TFT strategy is not a Nash equilibrium [5], indicating that it can make cooperation difficult to maintain [6].

2.2 The N-Person Prisoner's Dilemma (N-IPD)

The Iterated Prisoners' Dilemma (IPD[) serves as a model for understanding cooperation. While Axelrod's work highlighted the success of TFT, extending strategies like TFT from two-agent encounters to multi-agent scenarios (N-Person IPD or N-IPD) reveals a more complex strategic landscape [7]. In the N-IPD, the overall reward increases linearly as the number of collaborators increase [8]. However, in any round the individual reward is greater when the agent defects. No matter what votes the other agents make, the payoff is higher for any agent to defect in on round than to collaborate.

When there are N agents (and they all have only one vote), an individual agent's payoff is determined by its own choice and the total number of other agents in the group who chose to cooperate. This structure represents scenarios where individual actions contribute to a shared outcome, and direct one-to-one reciprocity is obscured. The payoff for an agent who cooperates is calculated by equation 3 and one that defects is calculated by equation 4, where n_{oc} is the number of other cooperators. T, R, P, and S are the same as the two agent dilemma, and the simulations below use the default values, $T = 5$, $R = 3$, $P = 1$ and $S = 0$.

$$S + (R - S) \times (n_{oc}/(N - 1)) \tag{3}$$

and for a defector as equation

$$P + (T - P) \times (n_{oc}/(N - 1)) \tag{4}$$

There are two distinct interaction models in N-IPD environments. The first is the pairwise voting model. In this setup, agents can vote for each of the other $N - 1$ agents. This is, in essence, a series of independent two agent IPD games. Each round each of the N agents plays against the $N - 1$ others. An agent's total score is the sum of payoffs from all its interactions. This model emphasizes direct, one-to-one accountability, where the actions of one agent in a pair directly affect the other, and responses can be specifically targeted.

The second is the neighbourhood voting model. All N agents make a single choice (cooperate or defect) simultaneously as part of one collective group. In this case payoff comes directly from equations 3 or 4.

2.3 Agent Strategies and Adaptations

There are several static policies that are used in the simulations below. They are static in the sense that they perform by the same rules each time. These are the always collaborate strategy, the always defect strategy, the random strategy, and the TFT strategy. The random strategy just flips a coin to decide whether to collaborate or defect. In the N-Person game the Tit-for-Tat (TFT) strategy makes a probabilistic decision based on the number of collaborators in the last round. That is, if the number of collaborating agents in the prior round is κ, and there are $N - 1$ other agents, the TFT agents randomly selects collaborate

$\kappa/(N-1)$ of the time. An additional variant of the TFT agent, the TFT-E agent, explores; that is, a given percentage of the time, the TFT-E agent merely flips a coin to determine whether it collaborates or defects.

2.4 Reinforcement Learning in Multi-Agent Systems (MARL)

Reinforcement Learning (RL) is a class of machine learning where agents learn to make sequences of decisions by interacting with an environment and receiving feedback in the form of rewards or punishments [9]. Standard Q-Learning is a foundational RL algorithm that learns the value of taking a particular action in a given state. However, when applied to multi-agent systems, where multiple agents are learning simultaneously, standard RL algorithms face significant challenges, primarily due to the non-stationarity of the environment; each agent's policy changes as it learns, thereby changing the environment from the perspective of other agents [10]. This can destabilize learning and prevent convergence to cooperative equilibria. To address these issues within the N-IPD context, prior work has explored more advanced multi-agent reinforcement learning (MARL) techniques like Hysteretic Q-Learning [11] and WoLF-PHC [12], which aim to endow agents with more sophisticated learning capabilities.

Q-learning [13] is a system that learns by building a table of results from prior experience, called a Q-table. For example, in section 4, three agents participate, and make decisions. If one is a Q-learning agent, it can build a table of, for example, the last two rounds. Each round has eight possible outcomes 2^3, so there are $2^6 = 64$ cells to fill. Additionally, the Q-learning agent has an explore option, so that no matter what the tables say, it will try a random move a small percentage of the time.

Another parameter in the Q-learning update rule is the discount factor This value, set between 0 and 1, determines the present value of future rewards, prioritising immediate gratification over long-term gains. An agent with a discount factor of 0 only optimizes for the immediate reward of the current action. Conversely, a discount factor approaching 1 causes the agent to weigh future rewards heavily enabling it to learn more long-term strategies. A sufficiently high discount factor is often a necessary precondition for a learning agent to overcome the immediate incentive to defect and discover the long-term benefits of collaboration [9].

3 The Collaborative Hill and the Tragic Valley

The simplest extension to the two person IPD is the three person IPD. Simulations on this task show that the voting mechanism largely determines whether the agents converge on a cooperative solution (the Collaborative Hill) or whether they defect (the Tragic Valley).

The first set of simulations uses pairwise voting, and tournaments are run with agents with static policies. Fifty rounds are performed on all of the combinations of agents with the static policies of always defect, always collaborate, Tit-for-Tat (TFT), Tit-for-Tat with exploration (TFT-E), and random.

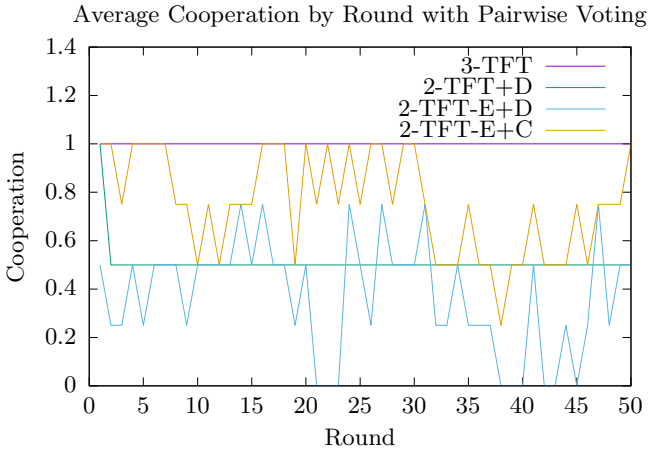

Fig. 1. Results from four static 50 round tournaments with pairwise voting. 3-TFT refers to 3 Tit-for-Tat agents that start cooperating; note that they continue to cooperate. 2-TFT+D refers to two TFT agents with an agent that always defects; this leads to system where the TFT agents collaborate with each other and defect against the defecting agent. 2-TFT-E+D refers to two TFT agents with 10% exploration and an always defect agent; this leads to a system where the TFT agents have some collaboration but mostly defect. 2-TFT-E+C refers to two TFT agents with 10% exploration and an agent that always cooperates; the TFT agents mostly collaborate, but there is some defection.

Figure 1 show the results of four different sets of three agents with static policies: three TFT agents, two TFT with 10% exploration (TFT-E) and one always defect, two TFT agents and one always defect agent, and two TFT-E agents and one always collaborate agent. The vertical axis refers to how often the two or three TFT or TFT-E agents voted to collaborate, cooperation. The TFT agents start off by collaborating, and the three TFT system continues to collaborate. The two TFT agents with the always defect agent always collaborate with each other, but (after an initial collaboration), always defect against the always defect agent. This leads to a lower average payoff, but is still largely collaborative. Both of these give horizontal lines in figure 1.

The TFT-E agents behave more stochastically, as sometimes they change their decisions. The pair with the collaborative agent largely collaborate, and the pair with the always defect agent largely defect. Below (figure 3) it is shown that the ratio is 75% and 25%.

The second set of simulations uses neighbourhood voting, with each agent getting one vote, and the results are shown in figure 2. The first set is three TFT agents with 10% exploration. The remaining three sets are the same static policies as figure 1. The TFT agents with the always defect initially vote to

collaborate, but then quickly move to always defect. They descend into the Tragic Valley.

The agents with exploration have a great deal of randomness. When one selects an unusual option, which happens 10% of the time for each, the overall system moves away from attractor states. With other TFT agents, this means the other agent will now become much more likely to vote this way in the next round. So, these systems are always quite volatile. The three TFT-E agents largely collaborate, the two TFT-T agents with the always collaborate agent largely collaborate, and the two TFT TFT-T agents with the always defect agent largely defect.

The TFT-E systems move up and down, with the pair with the collaborative agent being more collaborative, and the pair with the defecting agent being less collaborative. Below (figure 3) it is shown that the ratio is 80% and 20%.

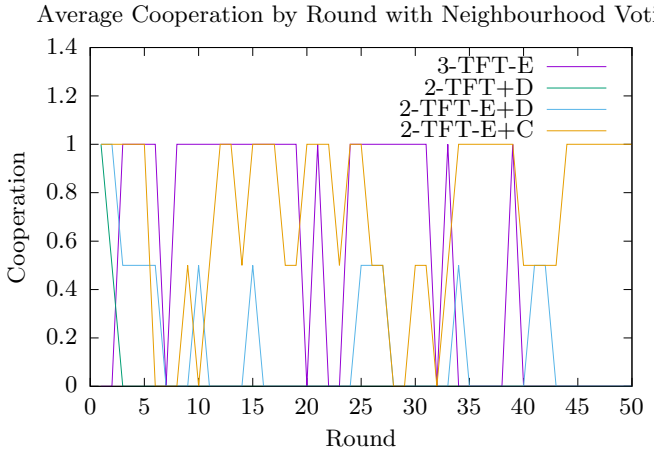

Fig. 2. Results from four 50 round tournaments with static agents with neighbourhood voting. 3-TFT-E refers to 3 Tit-for-Tat agents with 10% exploration that start cooperating. 2-TFT+D refers to two TFT agents with an agent that always defects; this quickly descends to no cooperation. 2-TFT-E+D refers to two TFT agents with exploration and an always defect agent; this system has some cooperations but mostly defects, but more than with pairwise voting. 2-TFT-E+C refers to two TFT exploration agents and an agent that always cooperates; this cooperates about half the time.

The reason behind the difference between the two voting mechanisms is that the reward space for each agent reinforces collaboration with with pairwise voting. Each agent can punish particular defectors and reward particular collaborators. This is why the agents can all get better results and ascend the Collaborative Hill. On the other hand, with neighbourhood voting, each agent can only reward or punish in aggregate. The reward structure is that each agent

does better by defecting, and any collaboration tends to give worse immediate results. This reward space and the agents' ability to only weakly influence other agents draws the agents into the Tragic Valley where agents do not cooperate.

Note that without exploration, the static policies move into an attractor state. When all three agents are TFT, if they all collaborate, they will continue to do so; not shown is the case when the initial choice is random. In the one of eight times when all three agents defect, the system starts in the attractor state when they all defect, and continue to defect. In the case where there is mixed voting, the system will move about until it gets to one of the two attractor states.

Figure 3 compares the TFT-E agents with voting mechanisms. These are similar to the runs in figures 1 and 2, but here the results reflect an average of 100 runs. It indicates that the pairwise voting strategy is more likely to collaborate against always defect.

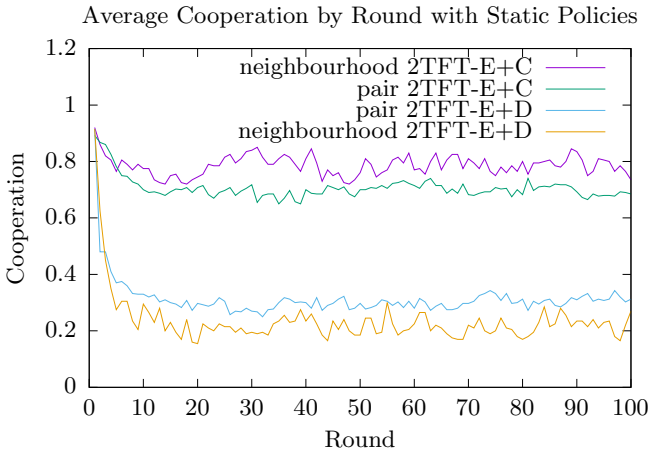

Fig. 3. Results from the average of 100 50 round tournaments with four different sets of agents. All the sets have two Tit-for-Tat (TFT) agents with 10% exploration. Two have a third agent that always defects, and two have one that always cooperates. The other variable is voting (pairwise or neighbourhood).

Adaptive exploration is when the TFT-E agent's exploratory behaviour gradually declines to 0% from 10%. The exploratory TFT agents decline to 0% with the defecting partner, and grow to 100% when they have adaptive exploration. See table 2 lines 1 and 2, where the exploration rate arrives at 0 at 300 rounds.

4 Reinforcement Learning Agents

Using standard Q-learning agents with only two rounds of history, agents become cooperative with pairwise voting, and defect with neighbourhood voting.

Figure 4 shows this. Over time the neighbourhood voting descends toward 20% cooperation while the pairwise voting agents remain at around 80%. The static TFT agents follow their counterparts with learned policies.

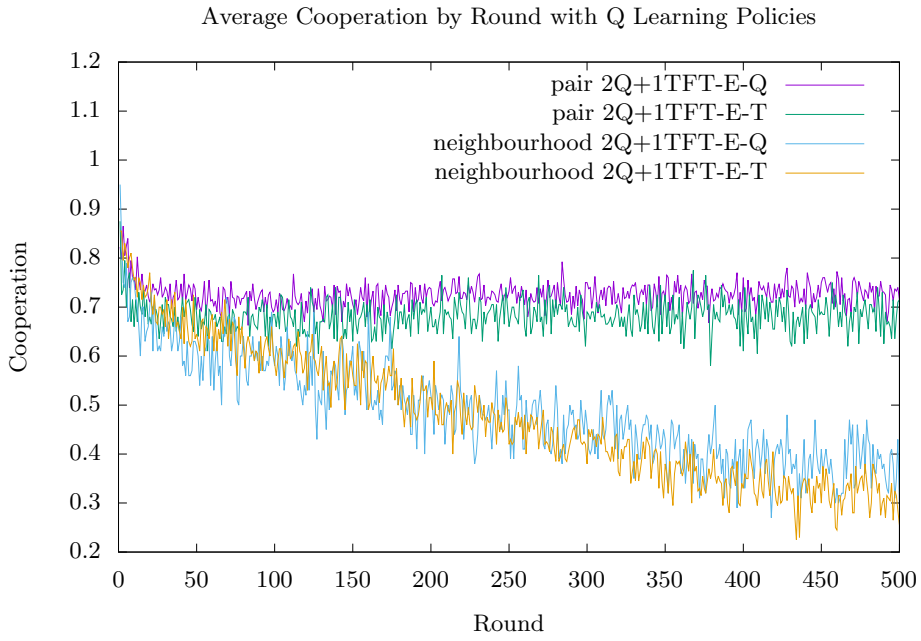

Fig. 4. Results from the average of 100 500 round tournaments with two different sets of agents. Both sets are two Q-Learning agents with one Tit-for-Tat agent with exploration. They differ by pairwise vs. neighbourhood voting, and the average of the Q-Learning agents have a line, and the TFT-E agent has a line. This clearly shows the pairwise agents remaining collaborative while the neighbourhood agents descend into the Tragic Valley.

While all of the figures describe results based on collaboration, the agents are deciding based on payoff. The best payoff an agent can receive in a three agent system is 5; this is when the agent defects and both of the other agents collaborate. In the system with neighbourhood voting, a Q-Learning agent and two always collaborate agents, the Q-Learning learns to defect. At round 10,000, the agent has an accumulated payoff of 4.84 while it almost never cooperates.

The simulations described above are based on three agent systems. The same results largely apply to systems with more agents. Table 2 lines 3a to 5b show the results of systems with 5, 7 and 25 agents. These all have one Q-Learning agent and the rest are TFT agents. These lines show how the systems with

neighbourhood voting descend into the Tragic Valley, while the pairwise voting systems ascend the Collaborative Hill.

Table 2. Example Results: Average Cooperation Value over last 50 rounds of run. The second column is the length of the run.

	Rounds	Voting	Agents 1 & 2 (N-1)	Cooperation	Agent 3 (N)	Cooperation
1	200	Neighbour	TFT-E Adapt	98.72%	Collaborate	100.0%
2	200	Neighbour	TFT-E Adapt	0.38%	Defect	0.0%
3a	5000	Pairwise	4 TFT	99%	Q-Learning	93%
3b	5000	Neighbour	4 TFT	68%	Q-Learning	68%
4a	5000	Pairwise	6 TFT	98%	Q-Learning	94%
4b	5000	Neighbour	6 TFT	56%	Q-Learning	48%
5a	5000	Pairwise	24 TFT	99%	Q-Learning	92%
5b	5000	Neighbour	24 TFT	20%	Q-Learning	12%
6a	5000	Pairwise	Q-Learning	64.1%	Random	50%
6b	5000	Neighbour	Q-Learning	19.6%	Random	50%
7	5000	Neighbour	Q Learning	71.8%	TFT	72.4%

The authors were surprised that they have found no papers explicitly stating that pairwise voting leads to largely collaborative performance.

5 Escaping the Tragic Valley with Enhanced Reinforcement Learning

There are a wide range of adjustments to standard Q-Learning that enable an agent in the N-IPD to be cooperative in many situations. Obviously, the agents interacting with the Q-Learning agent matter. If all of the other agents are static always defect agent, an extreme mechanism is needed. One such mechanism is having the agent learn from cooperation rather than payoff. In this case, as it is the only agent that can cooperate, then it should learn to cooperate. Including overall cooperation being a component in the evaluation metric for learning will be reasonably easy as the value for defecting and collaborating are the same (0).

Of course, what is usually desired is for the learning agent or agents to learn to cooperate because that, in the long run, gives better payoff. Several additions can help including optimistic initialisation, adaptive exploration, larger Q-tables, and a discount factor that encourages long-term gains.

With optimistic initialisation, the agent's Q-table is initialised with optimistic values for unexplored state-action pairs [9]. This encourages the agent to thoroughly explore its options before committing to a potentially suboptimal defect-heavy strategy.

With adaptive exploration, the learning agent employs a decaying exploration strategy. It explores more at the beginning of a tournament and gradually reduces its random exploration rate to exploit the knowledge it has gained, allowing for convergence to a stable policy.

Q-tables can, at least theoretically, be built to consider any number of past states. Unfortunately, the size of the table grows exponentially in the number of states with a large exponent. However, increasing from two states of history to three states has enabled Q-Learning agents with a large discount factor to cooperate in simulations with neighbourhood voting (see table 2 line 7).

The discount factor in standard Q-Learning considers longer-term gains the closer it is to 1. So, having values such as 0.99 often lead to Q-Learning agents that collaborate in neighbourhood voting N-IPD tasks.

Of course, the Q-Learning agent is more likely to collaborate when the other agents are collaborative such as TFT agents and other Q-Learning agents. These enhancements can transform an agent from a reactive learner into one that can perceive and respond to the emergent dynamics of the system.

6 General Discussion

The simulations show that pairwise voting leads to a cooperative system, and that neighbourhood voting leads to systems where the agents defect. This shows the Tragic Valley (neighourhood voting) and the Collaborative Hill (pairwise voting).

The difference stems from the reward space. If the agent knows every other agent's response, it will do better to select defect in almost every circumstance. The real difference only occurs when the agent can look back. With the two person version, TFT uses history from the last round to discourage defection. With $N > 2$ agents, history reamins important to allow collaboration to develop. The static policies that were explored, random, always defect, and always collaborate do not change based on history and their decisions are unaffected by the other agents.

Only TFT uses history, but it behaves differently with pairwise voting and neighbourhood voting because of the reward space. Its history is affected by other agents, individually in pairwise voting, and in aggregate for neighbourhood voting. When it can choose to punish or reward each individually, it climbs the Collaborative Hill. When it can only choose in aggregate, it can only reward or punish in aggregate, so other agents will free ride, and the overall effect will be a descent into the Tragic Valley.

The same is true with simple reinforcement learning agents. The reward space moves these agents up the Collaborative Hill with pairwise voting, and down the Tragic Valley with neighbourhood voting. Only when there is sufficient history can the agents learn to collaborate with neighbourhood voting.

There are several things of interest to explore for future work. One is to explore other games such as the volunteer's dilemma [8]. Another is merely to explore Q-Learning more rigorously. Of course, exploring other learning algorithms is also of interest.

7 Conclusion

The N-person iterated prisoners' dilemma poses two problems for collaboration. These problems differ based on how the agents vote. If each agent votes has one vote for each of the other agents, it is relatively easy for the agents to collaborate and climb the Collaborative Hill. When each agent has only one vote, then it is more difficult to choose collaborative policies due to the shape of the reward space and the agents usually descend into the Tragic Valley.

References

1. R. Axelrod and W. Hamilton, "The evolution of cooperation," *Science*, vol. 211(4489), pp. 1390–1396, 1981.
2. H. Hamburger, "N-person prisoner's dilemma," *Journal of Mathematical Sociology*, vol. 3(1), pp. 27–48, 1973.
3. R. Hardin, "Collective action as an agreeable n-prisoners' dilemma," *Behavioral science*, vol. 16(5), pp. 472–481, 1971.
4. A. Scodel, J. Minas, P. Ratoosh, and M. Lipetz, "Some descriptive aspects of two-person non-zero-sum games," *Journal of Conflict Resolution*, vol. 3(2), pp. 114–119, 1959.
5. J. Nash, "Equilibrium points in n-person games," *Proceedings of the national academy of sciences*, vol. 36(1), pp. 48–49, 1950.
6. C. Holt and A. Roth., "The Nash equilibrium: A perspective," *Proceedings of the national academy of sciences*, vol. 101(12), pp. 3999–4002, 2004.
7. R. Weil, "The n-person prisoner's dilemma: Some theory and a computer-oriented approach," *Behavioral Science*, vol. 11(3), pp. 227–234, 1966.
8. M. Archetti and I. Scheuring, "Coexistence of cooperation and defection in public goods games," *Evolution*, vol. 65(4), pp. 1140–1148, 2011.
9. R. Sutton and A. Barto, *Reinforcement learning: An introduction*. MIT press, 1998.
10. L. Busoniu, R. Babuska, and B. De Schutter, "A comprehensive survey of multi-agent reinforcement learning," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 38(2), pp. 156–172, 2008.
11. L. Matignon, G. Laurent, and N. Le Fort-Piat, "Hysteretic q-learning: an algorithm for decentralized reinforcement learning in cooperative multi-agent teams," in *International Conference on Intelligent Robots and Systems*, pp. 64–69, 2007.
12. M. Bowling and M. Veloso, "Agent-based modeling and simulation," in *Rational and convergent learning in stochastic games*, pp. 1021–1026, 2001.
13. C. Watkins, *Learning from delayed rewards Perceptual Generalisation*. PhD thesis, Cambridge, 1989.