Enhancing Cooperation in the IPD with Learning and Coalitions

Ana Peleteiro[1], Juan C. Burguillo[1], and Ana L. Bazzan[2]

[1]Telematics Engineering Department
University of Vigo, 36310-Vigo, Spain
[2] Instituto de Informatica
Universidade de Rio Grande do Sul, 15064-Porto Alegre, RS, Brazil
{apeleteiro,J.C.Burguillo}@det.uvigo.es
bazzan@inf.ufrgs.br

Abstract. The spatial version of the Prisoner's Dilemma (PD) has been used and analyzed to understand the role of local interactions in the emergence and maintenance of cooperation. In this paper we investigate the benefits of using reinforcement learning in this game. Agents can learn to join coalitions or not, as well as decide about which action to take, both when belonging to a coalition or when playing independently. We perform experiments where agents play the spatial IPD and use two learning techniques (Learning Automata and Q-Learning) to decide which actions perform and to see if cooperation and coalition formation emerges. The experiments and results provided in this paper show that by the use of learning algorithms the agents can learn which is the best behavior. Besides, we also show that by allowing formation of coalitions, the cooperation rate increases.

Keywords: Game Theory, Iterative Prisoner's Dilemma, Reinforcement Learning, Agent-Based Simulation

1 Introduction

Game theory provides useful mathematical tools to understand the possible strategies that self-interested agents may follow when choosing an action. The context of evolution of cooperation has been extensively studied seeking general theoretical frameworks like the Prisoner's Dilemma (PD) [2]. In this seminal work, Axelrod has shown that cooperation can emerge in a society of individuals with selfish motivations. An interesting spatial version of the PD have been suggested and analyzed in [10] trying to understand the role of local interactions in the emergence and maintenance of cooperation.

This paper focuses on the spatial interaction along generations of cooperators and defectors by means of computer simulations and reinforcement learning. Following the spatial configuration for the PD proposed by [10], we consider an agent population placed on a square lattice and simulate the dynamics of the population by means of agents (referred indistinctively also as cells in this text).

The interaction between the agents is modeled as an n-person game playing the PD iteratively (IPD). Every agent may behave as a defector or a cooperator when playing isolated, but they also can join in coalitions.

The aim of these simulations is to investigate whether the use of learning techniques (we employ Learning Automata and Q-Learning) and coalition formation enhances the emergence and maintenance of cooperation, when compared to the behavior of individual agents. The main question analyzed here is the emergence of coalitions as a way to support cooperation in a defection-prone environment.

The question of learning in the IPD has been partially addressed in [15] and by [17]. In particular, in this paper, Sandholm and Crites present an empirical study of reinforcement learning in the IPD. Their conclusion regarding the use of QL is that clear cooperation seldom emerged in experiments with two Q-learners even though the discount factor was set high to stimulate cooperation.

Therefore, the main contribution this paper is the exploration of the behavior of coalitions in multi-agent spatial games, playing the IPD by means of reinforcement learning techniques such as Learning Automata (LA) and Q-Learning (QL).

The remainder of the paper is structured as follows: firstly, in Sect. 2 we present the Prisoner's Dilemma. After, in Sect. 3, we describe the approach we followed in our game. In Sect. 4 we explain the learning techniques the agents use, and in Sect. 5 and in Sect. 6 we describe the scenario and the results of our experiments. Finally, in Sect. 7 we present the conclusions and future work.

2 Prisoner's Dilemma

Game Theory is a branch of applied mathematics that helps to understand the strategies that selfish individuals may follow when competing or collaborating in games and real scenarios. The concept of evolution of cooperation has been successfully studied using theoretical frameworks like the Prisoner's Dilemma (PD) [2], which is one of the most well-known strategy games. It models a situation in which two entities have to decide whether to cooperate or defect, but without knowing what the other is going to do.

Nowadays, the PD game has been applied to a huge variety of disciplines: economy, biology, artificial intelligence, social sciences, e-commerce, etc. Table 1 shows the general PD form matrix, which represents the rewards a player obtains depending on its own action and on the opponent's one. In this matrix, T means the Temptation to defect, R is the Reward for mutual cooperation, P the Punishment for mutual defection and S the Sucker's payoff. To be defined as a PD, the game must respect the following constraints: $T > R > P > S$ and $2R > T + S$.

Given these constraints, in the IPD, the optimal action if both players know that they are going to play exactly N times is to defect (it is the Nash equilibrium of the game) [14]. In particular, in any one round (or "one-shot") game, choosing action D is a Nash equilibrium because it rewards the highest payoff for agent A_i no matter what the opponent chooses; the same applies to A_j. However, the

Table 1: General Prisoner's Dilemma Matrix.

	Player A_j Cooperates	Player A_j Defects
Player A_i Cooperates	R, R	S, T
Player A_i Defects	T, S	P, P

combined action DD (i.e. both play the Nash equilibrium) is very poor socially speaking. This *combined* payoff is maximized if both cooperate. However, when players play an indefinite or random number of times, cooperation can emerge as a game equilibrium.

3 The Game

The approach we follow in this paper is a spatial game where every agent may interact with several neighbors at a time. We consider a spatial structure of the population, i.e., interaction among agents is locally restricted to their neighbors. The approach presented here is based on [10].

3.1 Spatial Distribution

For the spatial distribution of the cells we consider a two-dimensional square lattice consisting of N nodes, in which each cell is ruled by an agent (Fig. 1).

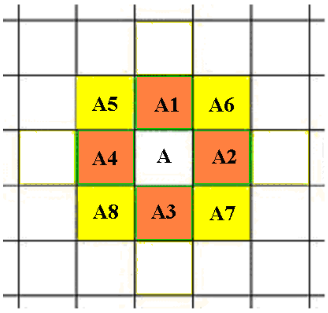


Fig. 1: Cell agent (A) and two neighborhoods: first with 4 cells A1,...,A4, and second with 8 cells A1,...,A8

If we let every node in the system interact with the remaining (N-1) nodes, we have a panmictic population, i.e., a population where all the individuals are potential partners. But, in many real contexts like geography, biology, MANETs or social networks, each node interacts mainly with a set of local neighbors. In this paper, we are mainly interested in the spatial effects of the PD game. In our case, each agent A_i interacts only with the m closest agents (in evolutionary game theory this is called a m-person game). Each game is played by the n players ($n = m + 1$) simultaneously, and the payoff for each player depends on the number of cooperators and defectors in the neighborhood. Thus, we consider that each cell agent A_i interacts only with the agents in its neighborhood and can perform two actions against them: defect (D) or cooperate (C). Fig. 1 shows a cell agent A and two possible neighborhoods, which are defined depending on the distance that a cell is allowed to use to play with other cells. In this paper we consider these two possible neighborhoods: 4-neighbors or 8-neighbors.

3.2 Agent Roles

In [7, 10] every agent may behave as a defector or a cooperator playing isolated, but in the work presented here agents can create coalitions. Therefore, there are two possible roles in our MAS:

1. Independent cells: ruled by its own agent that may behave as a cooperator or a defector with their neighbors depending on its own decisions. They can join a group in their neighborhood or remain independent.
2. Coalition cells: these are the cells that belong to a coalition. The action they perform is decided by all the members of the coalition. They can become independent whenever they want.

3.3 Agent Actions

The aim of the paper is to show that the formation of coalitions enforces the cooperation among the agents, so we have two possible scenarios: one not allowing coalitions and the other allowing them. In the former case, the agents can only decide about which action they play, i.e., if they are cooperators or defectors. In the later case, the cells also decide about the action to play, but they can also join or disjoin coalitions.

In the coalition scenario, there are three possible actions to perform: join/remain in a coalition, go/remain independent playing C, go/remain independent playing D. If an agent decides to perform the first action, if it is joining a new group, it has to decide to which one to join. This decision depends on the historical gains of the surrounding groups or cells, i.e., the cell joins the group or an isolated cell that have had higher benefits in the past.

When the cell has joined a group, it has to vote, with the rest of the members, to decide how they behave against the agents not belonging to the group (henceforth called *outsiders*), i.e., each agent had chosen before which is the best action to perform in their own experience. Therefore, all the cells vote to agree the coalition behavior, and the action performed is the one obtaining the highest number of votes.

Inside a coalition, the agent does not have to decide which is the behavior against its coalition mates. We could consider agents betraying their coalition members, but as a first approach, in this paper we assume that the agents cannot behave as defectors towards their coalition members as all the actions are public.

The other possible actions a cell might perform when we allow coalitions are: go/remain independent playing C and go/remain independent playing D. With this, the agents decide if they want to remain independent, or, in the case that they belong to a coalition; if they want to become independent, choosing also which action they will perform: being a defector or a cooperator.

4 Methods: Reinforcement Learning Techniques

We consider two popular reinforcement learning techniques that the agents use to decide the actions to perform: Q-learning (QL) and Learning Automata (LA).

4.1 Q-Learning (QL)

Q-Learning is a model-free technique popular in RL exactly because it does not require that the agent have access to information about how the environment works. QL works by estimating values for pairs state–action (the Q-values), which are numerical estimators of quality for a given pair of state and action. More precisely, a Q-value $Q(s, a)$ represents the maximum discounted sum of future rewards an agent can expect to receive if it starts in state s, chooses action a and then continues to follow an optimal policy.

The QL algorithm approximates $Q(s, a)$ as the agent acts in a given environment. The update rule for each experience tuple $\langle s, a, s', r\rangle$ is as in Eq. 1, where α is the learning rate and γ is the discount for future rewards. If all pairs state-action are visited infinitely often during the learning process, then QL is guaranteed to converge to the correct Q-values with probability one [16].

$$Q(s, a) \leftarrow Q(s, a) + \alpha \; (r + \gamma \, max_{a'} \, Q(s', a') - Q(s, a)) \tag{1}$$

4.2 Learning Automata (LA)

In the case of LA, the agent just considers its history and selects its next action depending on its experience and payoffs. We use the L_{R-I} scheme [12] as defined in Eq. 2, where $\alpha \in [0, 1]$.

The L_{R-I} scheme is defined as following:

$$\begin{array}{rrcl} & p^{i,t+1} & = & p^{i,t} + \alpha(1 - p^{i,t}) \\ \forall_{j \neq i}: & p^{j,t+1} & = & p^{j,t}(1 - \alpha) \end{array} \tag{2}$$

In these equations, α is a (small) learning factor. The first rule is used to reinforce the action chosen if it performed better than its alternatives in the considered state. At the same time, we apply the second rule to the other actions, decreasing its probability. In the next round, the agent will choose its new strategy using the updated probabilities.

4.3 Action selection and states

Both LA and QL keep and update a vector containing the probabilities of performing a concrete action in a state. We need to use an action selection mechanism to explore all the set of possible actions in every state. In this paper we use the epsilon-greedy method: with probability ϵ an agent will explore the action space, i.e., an action is chosen depending on its probabilities. Besides, with probability $1 - \epsilon$ the agent selects the action with the highest probability. Along the simulation, epsilon is decreased to reduce the exploration, allowing the agent to select the most successful actions.

We want agents to learn which action to take in each state, thus we define the state of an agent as the number of neighbors that cooperate and defect. For example, if we consider that an agent's neighborhood is eight, one state could be four defectors and four cooperators. However, to compute the state in every round is not trivial, since the agents cannot know a priori what the other players are going to play (they do not know the actual state until they actually play). To solve this problem, we use the previous actions played to derive the current state; we then compute which is the best action to play for the current state.

5 Scenarios

We have tested the following scenarios for the spatial IPD game:

1. LA: with or without coalitions
2. QL: with or without coalitions

In cases of scenarios without coalitions, an agent learns to act as a cooperator or as a defector. In cases where coalitions are allowed, the agent learns if it is better to be in a coalition or to be an independent cell (and in this case it learns to cooperate or defect). Besides, if a cell is in a coalition, it also has to learn which is the best action to play against the *outsiders*. By presenting these two scenarios, we want to show how the cells learn to choose among different behaviors by joining to other agents allowing them to get bigger gains.

6 Results

In this section we present the most relevant results obtained in the different scenarios considering three IPD matrices. The experiments were performed in a Pentium (R) Dual-Core CPU, E5200 2.50 GHz, 3.50 GB RAM. As we said before, we use LA and QL, and each agent plays against the eight or four closest neighbors. In the experiments we have used the following matrices, where the values refer to [[R R] [S T] [T S] [P P]]:

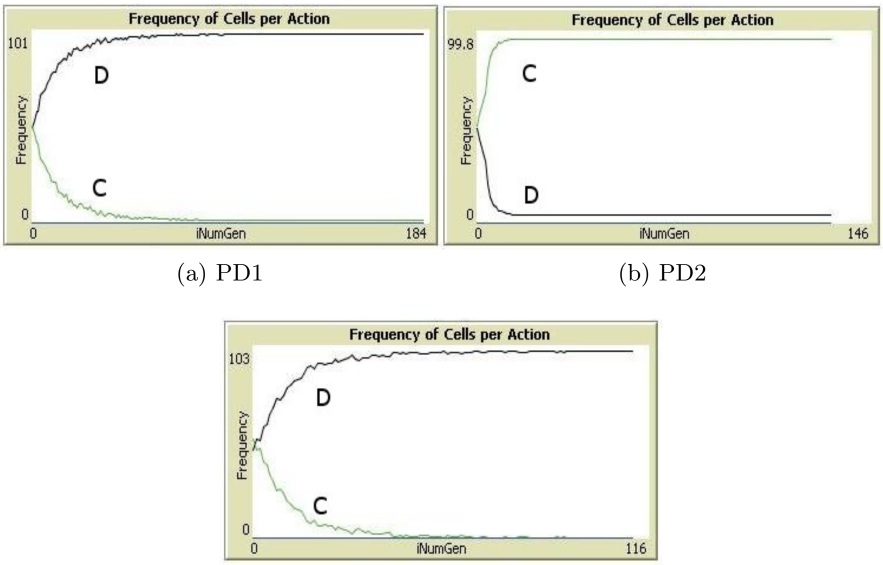
- PD1: [[5 5] [0 9] [9 0] [1 1]]
- PD2: [[8 8] [5 3] [3 5] [1 1]]
- PD3: [[3 3] [0 5] [5 0] [1 1]]

PD1 is a matrix biased to defection, since as we see, the payoff if a cell defects and the opponent cooperates is very high. In PD2, the matrix does not satisfy the conditions for the PD matrix values, but we use it to bias the matrix to cooperation. Finally, PD3 has the standard PD values for R, S, T, and P.

6.1 Scenario without coalitions

In this section we present the results obtained when using the learning algorithms without coalitions. The graphs only show the results of using the LA algorithm and with 8-neighbors, since results are quite similar for QL with 8-neighbors and for LA and QL with 4-neighbors.

In Fig. 2 we show the different frequency of cells performing an action when using the three different matrix. The defectors are represented in black, and the cooperators in green/grey.



(a) PD1

(b) PD2

(c) PD3

Fig. 2: No coalitions: cooperators (C) and defectors (D) along time.

We can see that in the case of PD1 (Fig. 2a), all cells tend to defect, since in the beginning, if playing against a cooperator, defection yields the maximum reward. This way, cells learn that defecting is the best option. When using the standard PD matrix (Fig. 2c), we see that all cells end up defecting as well, since with this matrix the cells are more tempted to defect; therefore as the gain of a defector is higher, the learning phase drives to defecting.

In the case of PD2 (Fig. 2b), all cells end up cooperating, since the matrix favors this behavior.

6.2 Scenario with Coalitions

In this section we present the results obtained when cells may join or form coalitions, and they use LA and QL to decide which action to take.

In Fig. 3 we plot the percentage of cells that cooperate, for the case where LA is used. Graphs in the left represent the percentage of cells which are independent and cooperators (green line), individual and defectors (black line) or belonging to a coalition (blue line). The figures in the right represent the frequency of cells that, belonging to a coalition, decide to defect against the *outsiders* (black line) and the ones that decide to cooperate (green line).

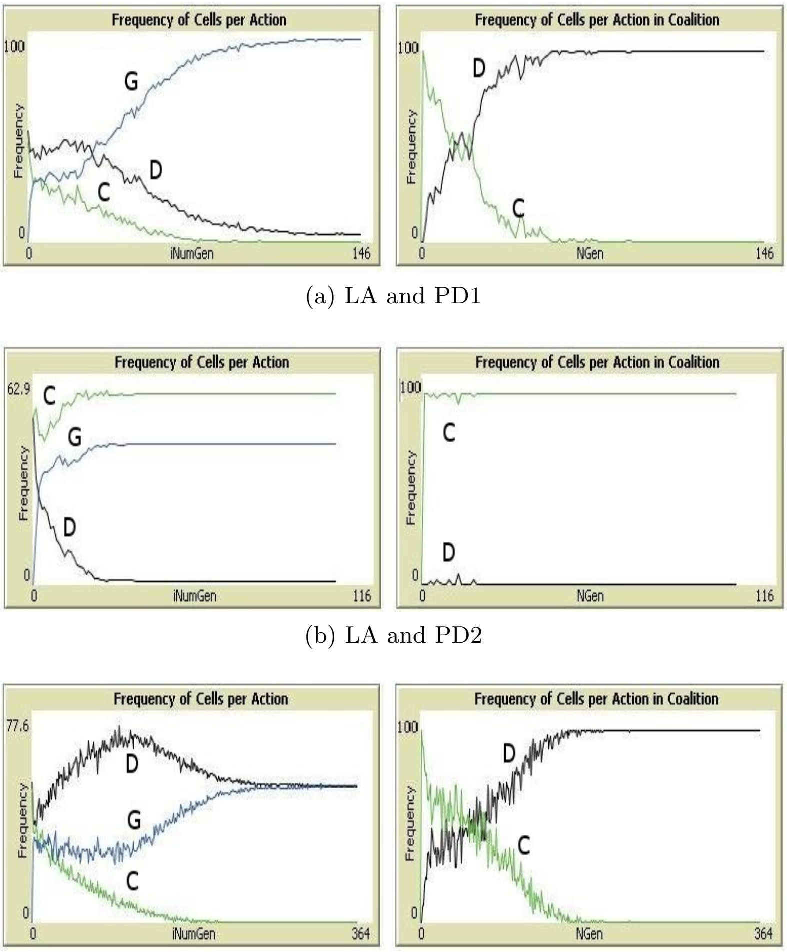
In Fig. 3a we see that, when using the PD1 matrix, cells tend to join a coalition and play D against the outsiders. This is a coherent result, since mutual cooperation yields a high reward. Thus, joining cooperative coalitions results in big gains. Besides, defecting against the outsiders leads to high rewards (9 points). This is why cells learn to defect against outsiders.

In Fig. 3c we observe that in the standard PD case (PD3), the number of defectors is similar to the number of cells belonging to a coalition. This happens because in this case the reward of mutual cooperation is not as high as in the PD1 matrix. Besides, we also observe that the internal behavior of the coalition cells against the outsiders is to defect.

Finally, in Fig. 3b, with the PD2 matrix, we see that the number of cooperating cells exceeds the number of the ones belonging to a coalition. However, we also observe that the internal behavior of the coalition is to cooperate against the outsiders. In this case cells learn that cooperating (no matter if within a coalition or playing independently) provides the highest benefit.

Now we show results when the cells use QL. In Fig. 4 we can see the corresponding graphs, which follow the same schema as the ones in Fig. 3.

The results obtained are similar to the ones obtained with LA, except for the following issues: in Fig. 4a (left) we see that the number of defectors is not as low as when using LA. Besides, in Fig. 4b (left), the number of cooperators and cells belonging to a coalition is similar, whereas in the LA case, the number of cooperators exceeded the number of cells that belong to coalitions. Finally, regarding the PD3, when using QL, the number of cells belonging to the coalition exceeds the number of defectors, while when using LA they appear in a similar percentage. This might be because the learning with LA is faster, and if the agents defect they get higher benefits in short term, being this learned in the first stages. We see this in the beginning of Fig. 3c (left): there is a peak of defectors that decreases with time to reach a similar number of cells belonging to a coalition. As we have seen in the previous experiments, by using learning algorithms, coalitions emerge, and cells self-organize to obtain the highest benefit. Besides, by changing the payoff matrix, we observe that the behavior of the cells varies to adapt to the conditions that these biased matrices impose. Finally, the results obtained with two different learning algorithms produce similar plots.



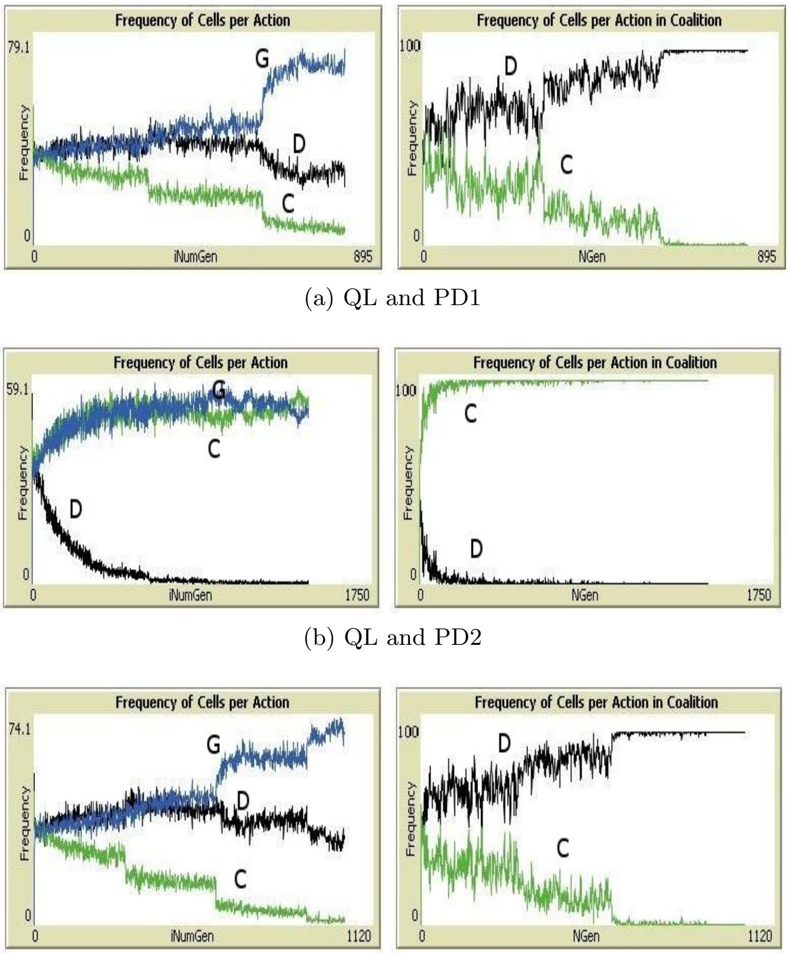
(a) LA and PD1

(b) LA and PD2

(c) LA and PD3

Fig. 3: LA with coalitions. Plots in the left represent the percentage of independent cells playing C, independent cells playing D and cells belonging to a coalition (G). Plots in the right represent cells that belong to a coalition and play C or D against the outsiders.

However, a further difference between the use of LA and QL relates to the spatial distribution of cells in the grid that are cooperating and defecting. In Fig. 5 we show the distribution of coalitions and independent cells in the scenario at the end of the previous simulations. In this figure, defectors are black, cooperators green, and the cells belonging to the different coalitions have other colors (unique for each coalition). We can see that agents tend to group in order to get more benefits. Depending on the payoff matrix and the learning algorithm,



(a) QL and PD1

(b) QL and PD2

(c) QL and PD3

Fig. 4: QL with coalitions. Plots in the left represent the percentage of independent cells playing C, independent cells playing D, and cells belonging to a coalition (G). Plots in the right represent the cells that belong to a coalition and play C or D against the outsiders.

the shape and size of the groups vary, as we can see comparing Fig. 5e, where there is only one group and Fig. 5f, where the number of groups is higher.

7 Conclusions and future work

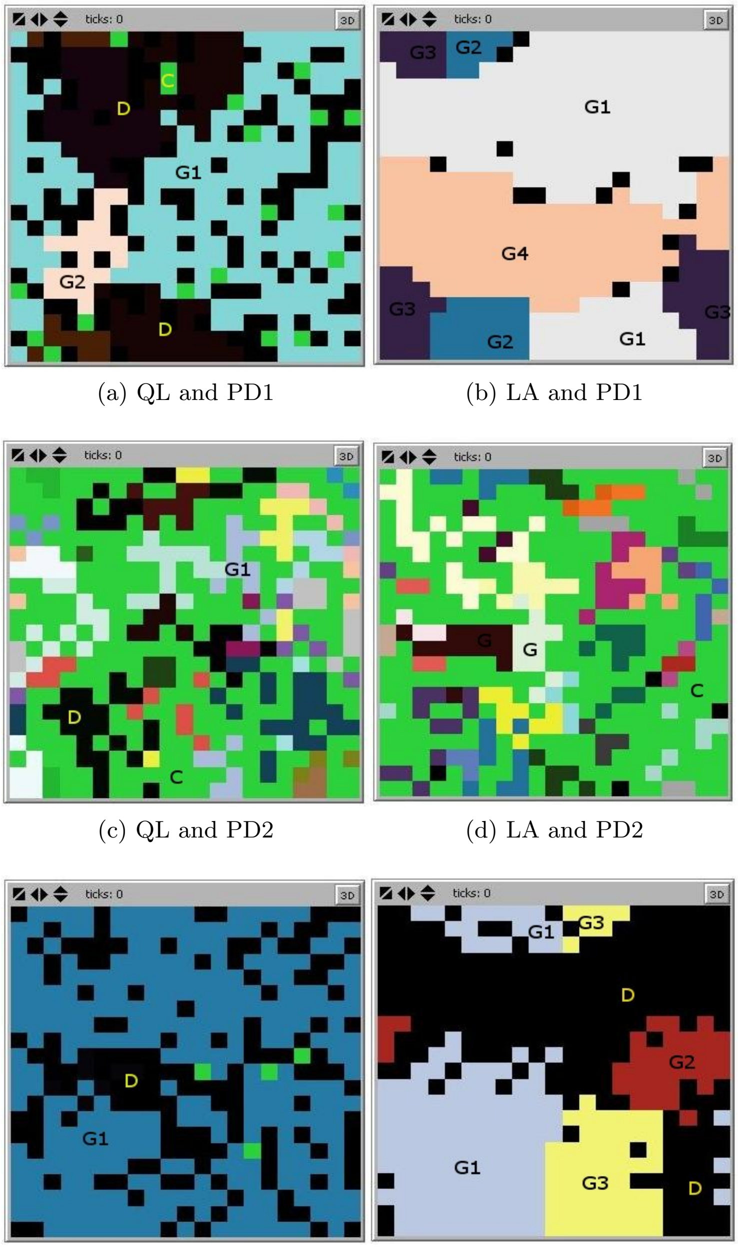
In this paper we have presented the use of reinforcement learning to study the interaction of agents playing the spatial IPD when the formation of coalitions is allowed. The main contribution of the paper is the exploration of the behavior of coalitions in multi-agent spatial games, playing the IPD by means of reinforcement learning techniques, in scenarios with and without coalitions.

We present the most relevant results indicating that using reinforcement learning algorithms lead agents to learn and to select actions that bring higher rewards. In our case, the different matrices and neighborhoods allow us to find several situations in which different agent behaviors may arise. Besides, by allowing the formation of coalitions, we explore the behavior that emerges; because the benefits that the cells obtain by joining are higher than those achieved when remaining independent.

As future work, we plan to use other algorithms, to include different kind of agents in the game and to apply the coalition formation to other more complex and realistic scenarios.

References

1. Binmore, K.: 1994. Game Theory. Mc Graw Hill.
2. Axelrod, R.: 1984. The evolution of Cooperation. Basic Books, New York.
3. Axelrod, R.: 2000. "On Six Advances in Cooperation Theory." Analyse und Kritik 22: 130-51.
4. Hoffmann, R.: 2000. Twenty years on: The evolution of cooperation revisited. Journal of Artificial Societies and Social Simulation, 3(2).
5. Feldman, M., Lai, K., Stoica I., and Chuang, J. 2004.: Robust Incentive Techniques for Peer-to-Peer Networks. ACM E-Commerce Conference (EC'04).
6. The official BitTorrent page, http://www.bittorrent.com
7. Schweitzer, F., Behera, L., and Mhlenbein, H.: 2002. Evolution of Cooperation in a Spatial Prisoner's Dilemma. Advances in Complex Systems, vol. 5, (2-3): 269-299.
8. Axelrod, R.: 1997. Building New Political Actors. The Complexity of Cooperation: Agent-based Models of Competition and Collaboration. Princeton University Press.
9. Costa-Montenegro, E., Burguillo-Rial, J.C., Gonzlez-Castao, F. J., and Vales-Alonso, J.: 2007. Agent-Controlled Sharing of Distributed Resources in User Networks. Computational Intelligence for Agent-based Systems (Studies in Computational Intelligence). Lee, Raymond S.T.; Loia, Vincenzo (Eds.). Volume 72. Springer.
10. Nowak, M.A., May, R.M.: 1992. Evolutionary games and spatial chaos. Nature 359: 826-829.
11. Langer, P., Nowak, M.A, Hauert, C.: 2008. Spatial invasion of cooperation. Journal of Theoretical Biology 250:634-641.
12. Narendra, K., Thathachar, M.: 1989. Learning Automata: An Introduction. Prentice-Hall, Englewood Cliffs, NJ.
13. Osborne, M. J.: 2003. An Introduction to Game Theory, Oxford University Press, USA.
14. Binmore, K.: 1994. Game Theory and the Social Contract Volume I: Playing Fair, The MIT Press: Cambridge, MA, USA.
15. Nguyen, D. and Ishida, Y.: 2009. Spatial Dilemma Strategies of Intelligent Agents: Coalition Formation in Environmental Game, International Conference on Knowledge and Systems Engineering, IEEE Computer Society, Los Alamitos, CA, USA
16. Watkins, C. J. C. H., and Dayan, P.: 1992. Q-learning. Mach. Learn. 8, 279292.
17. Sandholm, T.W. and Crites, R.H.: 1995. Multiagent Reinforcement Learning in the Iterated Prisoner's Dilemma, Biosystems (37): 147–166



(a) QL and PD1

(b) LA and PD1

(c) QL and PD2

(d) LA and PD2

(e) QL and PD3

(f) LA and PD3

Fig. 5: Graphical display and comparison of coalitions for QL and LA. D stands for defector, C cooperator and G group