Improving on the reinforcement learning of coordination in cooperative multi-agent systems

Spiros Kapetanakis and Daniel Kudenko
Department of Computer Science
University of York, Heslington
York, YO10 5DD, U.K.
{spiros, kudenko}@cs.york.ac.uk

Abstract

We report on an investigation of reinforcement learning techniques for the learning of coordination in cooperative multi-agent systems. These techniques are variants of Q-learning (Watkins, 1989) that are applicable to scenarios where mutual observation of actions is not possible. To date, reinforcement learning approaches for such *independent* agents did not guarantee convergence to the optimal joint action in scenarios with high miscoordination costs. We improve on previous results (Claus and Boutilier, 1998) by demonstrating that our extension causes the agents to converge almost always to the optimal joint action even in these difficult cases.

1 Introduction

Learning to coordinate in cooperative multi-agent systems is a central and widely studied problem, see, for example Lauer and Riedmiller (2000); Boutilier (1999); Claus and Boutilier (1998); Sen and Sekaran (1998); Sen et al. (1994); Weiss (1993). In this context, coordination is defined as *the ability of two or more agents to jointly reach a consensus over which actions to perform in an environment*. We investigate the case of *independent* agents that cannot observe one another's actions, which often is a more realistic assumption.

In this investigation, we focus on reinforcement learning, where the agents must learn to coordinate their actions through environmental feedback. To date, reinforcement learning methods for independent agents (Tan, 1993; Claus and Boutilier, 1998; Sen et al., 1994) did not guarantee convergence to the *optimal* joint action in scenarios where miscoordination is associated with high penalties. We investigate variants of Q-learning (Watkins, 1989) in search of improved convergence to the optimal joint action. More specifically, we investigate the effect of the temperature function on Q-learning with the Boltzmann action-selection strategy. We validate our results experimentally and show that the convergence probability is greatly improved over other approaches.

Our paper is structured as follows: we first introduce a common testbed for the study of learning coordination in cooperative multi-agent systems and describe an especially difficult variant with high miscoordination costs. We then introduce variants of Q-learning and discuss the experimental results. We finish with an outlook on future work.

2 Single-stage coordination games

A common testbed for studying the problem of multi-agent coordination is that of repeated cooperative single-stage games (Fudenberg and Levine, 1998). In these games, the agents have common interests i.e. they are rewarded based on their joint action and all agents receive the same reward. In each round of the game, each agent chooses an action. These actions are executed simultaneously and the reward that corresponds to the joint action is broadcast to all agents.

A more formal account of this type of problem was given by Claus and Boutilier (1998). In brief, we assume a group of n agents $\alpha_1, \alpha_2, \ldots, \alpha_n$ each of which have a finite set of *individual actions* A_i which is known as the agent's *action space*. In this game, each agent α_i chooses an individual action $action \in A_i$ from its action space to perform. The action choices make up a *joint action* which is associated with a unique reward. Upon execution of their actions all agents receive the reward that corresponds to the joint action. For example, Table 1 describes the reward function for a simple cooperative single-stage game. If agent 1 executes action b and agent 2 executes action a, the reward they receive is 5. Obviously, the optimal joint-action in this simple game is (b, b) as it is associated with the highest reward of 10.

		Agent 1	
		a	b
Agent 2	a	3	5
	b	0	10

Table 1: A simple cooperative game reward function.

Our goal is to enable the agents to learn optimal coordination from repeated trials. To achieve this goal, one can use either *independent* or *joint-action* learners. The difference between the two types lies in the amount of information they can perceive in the game. Although both types of learners can perceive the reward that is associated with each joint action, the former are unaware of the existence of other agents whereas the latter can also perceive the actions of others. In this way, joint-action learners can maintain a model of the strategy of other agents and choose their actions based on the other participants' perceived strategy. In contrast, independent learners must estimate the value of their individual actions based solely on the rewards that they receive for their actions.

A popular technique for learning coordination in cooperative single-stage games is one-step Q-learning which is due to Watkins (1989), a reinforcement learning technique. Since the agents in a single-stage game are stateless, we need a simple reformulation of the general Q-learning algorithm such as the one used by Claus and Boutilier (1998). Each agent maintains a Q value for each of its actions. These values are updated after each step of the game according to the reward received for the action. The value $Q(action)$ provides an estimate of the usefulness of performing action $action$. We apply Q-learning with the following update function:

$$Q(action) \leftarrow Q(action) + \lambda(r - Q(action))$$

where λ is the learning rate $(0 < \lambda < 1)$ and r is the reward that corresponds to choosing action $action$.

In a single-agent learning scenario, Q-learning is guaranteed to converge to the optimal action independent of the action-selection strategy. In other words, given the assumption of a stationary reward function, Q-learning will converge to the optimal policy for the problem. However, in a multi-agent setting, the action-selection strategy becomes crucial for convergence to *any* joint action. A major challenge in defining a suitable strategy for the selection of actions is to strike a balance between exploring the usefulness of moves that have been attempted only a few times and exploiting those in which the agent's confidence in getting a high reward is relatively strong. This is known as the *exploration/exploitation problem*.

The action selection strategy that we have chosen for our research is the Boltzmann strategy (Kaelbling et al., 1996) which states that agent α_i chooses action $action$ with a probability based on its current estimate of the usefulness of that action. In the case of Q-learning, the Q values act as the agent's estimates of the usefulness of an action so the probability for action selection is based on the function:

$$P(action) = \frac{e^{\frac{Q(action)}{T}}}{\sum_{action' \in A_i} e^{\frac{Q(action')}{T}}}$$

Specifically, we have concentrated on a proper choice for the temperature function T. This function provides an element of randomness in the way that actions are chosen: high values in temperature encourage exploration since small variations in Q values become less important whereas low temperature values encourage exploitation. The value of the temperature can be decreased over time as exploitation takes over from exploration. It has been shown (Singh et al., 2000) that convergence to a joint action can be ensured if the temperature function adheres to certain properties. However, we have found that there is more that can be done to ensure not just convergence to *some* joint action but convergence to the *optimal* joint action, even in the case of independent learners.

In our study, we focus on the *climbing game* which is due to Claus and Boutilier (1998). This focus is without loss of generality since the climbing game is representative of coordination problems with high miscoordination penalty in multi-agent systems and is, therefore, especially difficult to solve. This game is played between two agents. The reward function for this game is included in Table 2:

		Agent 1		
		a	b	c
	a	11	-30	0
Agent 2	b	-30	7	6
	c	0	0	5

Table 2: The climbing game table.

For each agent, it is difficult to converge to the optimal joint action (a, a) because of the negative reward in the case of miscoordination. For example, if agent 1 plays a and agent 2 plays b, then both will receive a negative reward of -30. Incorporating this reward into the learning process can be so detrimental that both agents tend to avoid playing the same action again. In contrast, when choosing action c, miscoordination is not punished so severely. Therefore, in most cases, both agents are easily tempted by action c. The reason is as follows: if agent 1 plays c, then agent 2 can play either b or c to get a positive reward (6 and 5 respectively). Even if agent 2 plays a, the result is not catastrophic since the reward is 0. Similarly, if agent 2 plays c, whatever agent 1 plays, the resulting reward will be at least zero. From this analysis, we can see that the climbing game is a sufficiently complex problem for the study of coordination. It includes heavy miscoordination penalties and "safe" actions that are likely to tempt the agents away from the optimal joint action.

3 Experimental results

This section contains our experimental results. Each subsection describes a variant of Q-learning for coordination games where the agents have no social awareness.

3.1 Exponential temperature

Typically, reinforcement learning experiments that use a temperature function to control how much exploration and exploitation an agent performs during the learning are set up so that the value of the temperature starts from an initial value and decreases over time. Exponential decay in the value of the temperature is a popular choice. This way, the agent learns until the temperature reaches some lower limit. The experiment then finishes and results are collected. The temperature limit is normally set to zero which may cause complications when calculating the action-selection probabilities with the Boltzmann function. To avoid such problems, we have set the temperature limit to 1 in our experiments.

Although reinforcement learning experiments that use an exponentially decaying temperature function are quite common, the effect that the parameters of the temperature function have on the learning have not been explored sufficiently.

In our analysis of exponential temperature functions, we use the following family of functions:

$$T(x) = e^{-sx} * \mathrm{max\_temp} + 1$$

where x is the number of iterations of the game so far, s is the parameter that controls the rate of exponential decay and $\mathrm{max\_temp}$ is the value of the temperature at the beginning of the experiment. Varying the parameters allows a detailed specification of the temperature function. A sample of 5 of these choices have been plotted in figure 1.

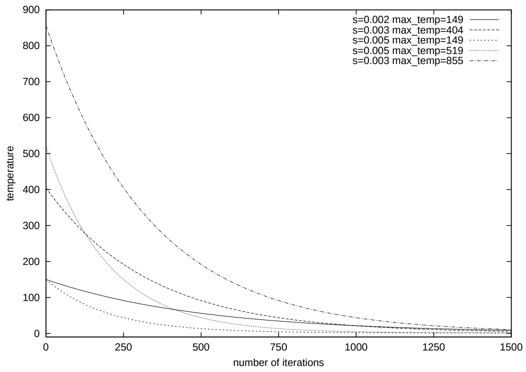

Figure 1: Exponential temperature functions for different s and $\mathrm{max\_temp}$ values.

For a given number of iterations, we experimented with a variety of $s, \mathrm{max\_temp}$ combinations. To motivate this point we use the climbing game and set the length of the experiment to 1000 iterations. We repeat each experiment 1000 times to ensure adequate confidence in the results. We compare the following parameter combinations:

- $s = 0.01 \quad \mathrm{max\_temp} = 499$
- $s = 0.006 \;\; \mathrm{max\_temp} = 499$
- $s = 0.01 \quad \mathrm{max\_temp} = 999$
- $s = 0.006 \;\; \mathrm{max\_temp} = 999$

The results from these experiments are included in Tables 3 to 6. Note that greater values of s mean that the temperature is decaying more rapidly.

	a	b	c
a	193	0	0
b	0	113	275
c	0	0	419

Table 3: Results with $s = 0.01$ $\mathrm{max\_temp} = 499$.

	a	b	c
a	192	0	0
b	0	114	279
c	0	0	414

Table 4: Results with $s = 0.006$ $\mathrm{max\_temp} = 499$.

	a	b	c
a	202	0	0
b	0	121	274
c	0	0	403

Table 5: Results with $s = 0.01$ $\mathrm{max\_temp} = 999$.

	a	b	c
a	154	0	1
b	0	98	331
c	9	0	405

Table 6: Results with $s = 0.006$ $\mathrm{max\_temp} = 999$.

Tables 3 to 6 contain the number of times out of the 1000 repetitions of the experiment that a joint action was reached after 1000 moves. For example, in table 3, joint action (c, b) was reached 275 times. The success ratio of these experiments is defined as the number of times the agents converged to the optimal joint action (a, a) over the total number of repetitions. These experiments are only a small sample, nevertheless, they show that, for a given length of the experiment, variation in the s and $\mathrm{max\_temp}$ parameters of the exponential function do not have a significant impact on convergence to the optimal joint action.

3.2 FMQ heuristic

The climbing game is a particularly difficult coordination game. This is not only due to the high miscoordination penalty that is associated with joint actions (a, b) and (b, a). It is also due to the relative safety provided

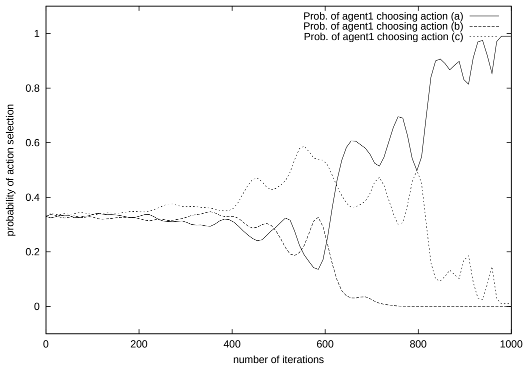

Figure 2: Probabilities for action-selection.

by action c for both agents. Figure 2 depicts the action-selection probabilities for agent1 during a successful run of the experiment for 1000 moves with $s = 0.006$ and $\mathrm{max_temp} = 499$.

It is clear from Figure 2 that action c is consistently the most probable action until a dominates it. This happens after approximately 600 moves. In fact, in most unsuccessful runs of the same experiment, action c is the most probable action throughout. What is needed, in this case, is a way to influence the learning towards the optimal joint action. Since independent agents are devoid of social awareness, from the agent perspective, the learning scenario consists only of actions and rewards: each agent knows what actions it plays and what the corresponding rewards are. There are two ways to influence the learning towards the optimal joint action: by changing the Q-update function and by changing the action-selection strategy.

Lauer and Riedmiller (2000) describe an algorithm for multi-agent reinforcement learning which is based on the *optimistic* assumption. In the context of reinforcement learning, this assumption implies that an agent makes any action it finds suitable expecting the other agent to choose accordingly. More specifically, the optimistic assumption affescts the way Q values are updated. Under this assumption, the update rule for playing action α defines that $Q(\alpha)$ is only updated if the new value is greater than the current one. Incorporating the optimistic assumption into Q-learning solves the climbing game every time. This fact is not surprising since the penalties for miscoordination, which make learning optimal actions difficult, are neglected as their incorporation into the learning tends to lower the Q values of the corresponding actions. Such lowering of Q values is not allowed under the optimistic assumption so that all the Q values eventually converge to the maximum reward corresponding to that action for each agent. In this way, using the optimistic assumption solves the climbing game.

The optimistic assumption is a heuristic that applies to the Q update function. Similarly, one can define heuristics that apply to the action-selection strategy. For example, we define the *Frequency Maximum Q value* (FMQ) heuristic. First, we augment the agent's internal status by maintaining 3 values for each of its actions $\alpha \in A_i$:

① $\mathrm{action\_count}(\alpha)$ holds the number of times the agent has chosen α in the game

② $\mathrm{maxR}(\alpha)$ holds the maximum reward encountered so far for choosing action α

③ $\mathrm{count\_maxR}(\alpha)$ holds the number of times that the maximum reward has been received as a result of playing action α

The expected value function in the Boltzmann strategy is now:

$$EV(\alpha) = Q(\alpha) + k * \mathrm{freq}(\mathrm{maxR}(\alpha)) * \mathrm{maxR}(\alpha)$$

where k is a weight which controls the importance of the FMQ heuristic in the action-selection and $\mathrm{freq}(\mathrm{maxR}(\alpha))$ is the frequency of receiving the maximum reward corresponding to an action. $\mathrm{freq}(\mathrm{maxR}(\alpha))$ is defined as:

$$\mathrm{freq}(\mathrm{maxR}(\alpha)) = \frac{\mathrm{count\_maxR}(\alpha)}{\mathrm{action\_count}(\alpha)}$$

Informally, the FMQ heuristic carries the information of how frequently an action produces its maximum corresponding reward. Table 7 contains the results that were obtained using the FMQ heuristic with the climbing game. These results were achieved with an exponentially decaying temperature $(s = 0.006, \mathrm{max\_temp} = 499)$ and $k = 10$ over 1000 iterations of the experiment for 1000 moves. Note that the success ratio with the FMQ heuristic is 99.8% whereas the same settings gave a success ratio of only 19.2% with the normal exponentially decaying temperature function (see Table 4).

	a	b	c
a	998	0	0
b	0	2	0
c	0	0	0

Table 7: Results with the FMQ heuristic in the climbing game.

4 Validation

To experimentally validate the FMQ heuristic and compare it to the optimistic assumption Lauer and Riedmiller (2000), we introduce a variant of the climbing game which we term *the partially stochastic climbing game*. This version of the climbing game differs from the original in that one of the joint actions is now associated with a stochastic reward. The reward function for the partially stochastic climbing game is included in Table 8.

Joint action (b, b) yields a reward of 14 or 0 with probability 50%. The partially stochastic climbing game is

		Agent 1		
		a	b	c
	a	11	-30	0
Agent 2	b	-30	14/0	6
	c	0	0	5

Table 8: The partially stochastic climbing game table.

functionally equivalent to the original version. This is because, if the two agents consistently choose their b action, they receive the same overall value of 7 over time as in the original game.

Using the optimistic assumptionLauer and Riedmiller (2000) on the partially stochastic climbing game consistently converges to the suboptimal joint action (b, b). This because the frequency of occurrence of a high reward is not taken into consideration at all. In contrast, the FMQ heuristic shows much more promise in convergence to the optimal joint action. It also compares favourably with normal Q-learning with an exponential temperature function. Tables 9, 10 and 11 contain the results from 1000 experiments with the exponential function, the optimistic assumption and the FMQ heuristic respectively. In all cases, the parameters are: $s = 0.006$, $\text{max\_temp} = 499$ and, in the case of FMQ, $k = 10$.

	a	b	c
a	212	0	3
b	0	12	289
c	0	0	381

Table 9: Results with exponential temperature.

	a	b	c
a	0	0	0
b	0	1000	0
c	0	0	0

Table 10: Results with optimistic assumption.

	a	b	c
a	988	0	0
b	0	4	0
c	0	7	1

Table 11: Results with the FMQ heuristic.

5 Discussion

The FMQ heuristic performs equally well in the partially stochastic climbing game and the original deterministic climbing game. In contrast, the optimistic assumption only succeeds in solving the deterministic climbing game. However, we have found a variant of the climbing game in which both heuristics perform poorly: *the fully stochastic climbing game*. This game has the characteristic that *all* joint actions is probabilistically linked with two rewards. The average of the two rewards for each action is the same as the original reward from the deterministic version of the climbing game so the two games are functionally equivalent. For the rest of this discussion, we assume a 50% probability. The reward function for the stochastic climbing game is included in Table 12.

		Agent 1		
		a	b	c
	a	10/12	5/-65	8/-8
Agent 2	b	5/-65	14/0	12/0
	c	5/-5	5/-5	10/0

Table 12: The stochastic climbing game table (50%).

It is obvious why the optimistic assumption fails to solve the fully stochastic climbing game. It is for the same reason that it fails with the partially stochastic climbing game. The maximum reward is associated with joint action (b, b) which is a suboptimal action. The FMQ heuristic, although it performs marginally better than normal Q-learning still doesn't provide any substantial success ratios.

6 Summary and Outlook

We have presented an investigation of techniques that can allow two agents that are unable to sense each other's actions to learn coordination in cooperative single-stage games. These technique are applicable to independent learners. However, there is still much to be done towards understanding exactly how the temperature function can influence the learning of optimal joint actions in this type of repeated games. In the future, we plan to specifically investigate the impact of the temperature function parameters on the learning.

Furthermore, since agents typically have a state component associated with them, we plan to investigate how to incorporate such coordination learning mechanisms in multi-stage games. We intend to further analyse the applicability of various reinforcement learning techniques to agents with a substantially greater action space. Finally, we intend to perform a similar systematic examination of the applicability of such techniques to partially observable environments where the rewards are perceived stochastically.

References

C. Boutilier. Sequential optimality and coordination in multiagent systems. In *Proceedings of the Sixteenth International Joint Conference on Articial Intelligence (IJCAI-99)*, pages 478–485, 1999.

Caroline Claus and Craig Boutilier. The dynamics of reinforcement learning in cooperative multiagent systems. In *Proceedings of the Fifteenth National Conference on Articial Intelligence*, pages 746–752, 1998.

Drew Fudenberg and David K. Levine. *The Theory of Learning in Games*. MIT Press, Cambridge, MA, 1998.

Leslie Pack Kaelbling, Michael Littman, and Andrew W. Moore. Reinforcement learning: A survey. *Journal of Artificial Intelligence Research*, 4, 1996.

Martin Lauer and Martin Riedmiller. An algorithm for distributed reinforcement learning in cooperative multi-agent systems. In *Proceedings of the Seventeenth International Conference in Machine Learning*, 2000.

Sandip Sen and Mahendra Sekaran. Individual learning of coordination knowledge. *JETAI*, 10(3):333–356, 1998.

Sandip Sen, Mahendra Sekaran, and John Hale. Learning to coordinate without sharing information. In *Proceedings of the Twelfth National Conference on Artificial Intelligence*, pages 426–431, Seattle, WA, 1994.

S. Singh, T. Jaakkola, M. L. Littman, and C Szpesvari. Convergence results for single-step on-policy reinforcement-learning algorithms. *Machine Learning Journal*, 38(3):287–308, 2000.

Ming Tan. Multi-agent reinforcement learning: Independent vs. cooperative agents. In *Proceedings of the Tenth International Conference on Machine Learning*, pages 330–337, 1993.

C. J. C. H. Watkins. *Learning from Delayed Rewards*. PhD thesis, Cambridge University, Cambridge, England, 1989.

Gerhard Weiss. Learning to coordinate actions in multi-agent systems. In *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence*, volume 1, pages 311–316. Morgan Kaufmann Publ., 1993.