CSE7642 Project: Correlated Q-Learning

Shaikh Shamid

Department of Computer Science, Georgia Institute of Technology

April 23, 2018

Abstract

This report presents the pitfalls of reproducing a research work by Amy Greenwald and Keith Hall[1] correlated Q-learning (CE-Q), a multi-agent reinforcement learning framework based on the correlated equilibrium solution concept. They showed that the algorithms learns correlated equilibrium policies of a markov game, also proved that certain variants of correlated-Q learning are guaranteed to converge to stationary correlated equilibrium policies in zero-sum and common-interest games.

1 Multi-agent learning algorithms

A multi-agent learning algorithm learns equilibrium policies in general-sum Markov games, just as Q-learning converges to optimal value function V^* in Markov decision processes (single-agent) as follows

$$Q(s,a) = (1-\alpha)Q(s,a) + \alpha\left[R(s,a) + \gamma V(s')\right]; V(s) = \max_{a \exists A} Q(s,a) \tag{1}$$

For two-player zero-sum stochastic game (SG), Littman[3] suggests the minimax-Q learning algorithm, in which V is updated with the minimax of the Q values for a player

$$V_1(s) = \max_{p_1 \exists \pi(A)} \min_{a_2 \exists A_2} \sum_{a_1 \exists A_1} \sum_{a_1 \exists A_1} p_1(a_1) Q_1(s,(a_1,a_2)) \tag{2}$$

As a generalization of Q-learning to general-sum games, Hu and Wellman[2] propose an algorithm called Nash-Q that converges to Nash equilibrium policies under certain (restrictive) conditions. But in general there are many Nash equilibria,and therefore the Nash payoff may not be unique. Littmans[4] explicit friend-or-foe-Q (FF-Q) algorithm always converges for a restricted classes of games: e.g., two-player, constant-sum Markov games, which exhibit minimax equilibria (foe-Q); e.g., coordination games with uniquely-valued equilibria

(friend-Q) as follows:

$$Friend : V_1(s) = \max_{a_1 \exists A_1, a_2 \exists A_2} Q_1(s, (a_1, a_2) \tag{3}$$

$$Foe : V(s) = \max_{p_1 \exists \pi(A_1)} \min_{a_2 \exists A_2} \sum_{a_1 \exists A_1} p_1(a_1) Q(s, (a_1, a_2) \tag{4}$$

CE-Q generalizes both Nash-Q and FF-Q: in general-sum games, the set of correlated equilibria contains the set of Nash (and thus, coordination) equilibria; in constant-sum games, where Nash and minimax equilibria coincide, the set of correlated equilibria contains the set of minimax equilibria.

multiQ(game $\Gamma, f, g, , i$)
Inputs game Γ, selection mechanism f, decay schedule g, learning rate , agent i
Output values V , Q-values Q, joint policy π^{i*}
Initialize Q-values Q, state s, action profile a
for $t \leftarrow 1$ **to** T **do**
    simulate action a_i in state s
    observe action profile a_{-i}, rewards $R(s, a)$, and next state $s^{'}$
    select $\pi_s^{i*} \exists f(Q(s'))$
    for all agents j
    (a) $V_j(s') = \sum_a \exists A_{s'} \pi_{s'}^{i*} Q_j(s', a)$
    (b) $Q_j(s, a) = (1 - \alpha) * Q_j(s, a) + \alpha * [(1 - \gamma) R_j(s, a) + \gamma * V_j(s)]$
    choose actions $a_i^{'}$ (on- or off-policy)
    update $s = s'$, $a = a'$
    decay α via g
end

Algorithm 1: Multi-agent Q-Learning.

Correlated-Q learning is an instantiation of multi-agent Q-learning, with the equilibrium selection mechanism f defined as follows: at state s, given the one-shot game Q(s), select an equilibrium π_s that satisfies the following constraints: for all $i \exists N$ and for all $a_i, a_i \exists A_i$,

$$\sum_{a_{-i} \exists A_{-i}(s)} \pi_s(a) Q_i(s, a) \geq \sum_{a_{-i} \exists A_{-i}(s)} \pi_s(a) Q_i(s, (a_{-i}, a_i')) \tag{5}$$

Out of four variants of correlated-Q learning, we will use *utilitarian* class in which we maximize the sum of all agents rewards: at state s as follows

$$max_{\pi_s \exists \Delta(A(s))} \sum_{j \exists N} \sum_{a \exists A(s)} \pi_s(a) Q_j(s, a) \tag{6}$$

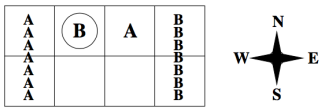

Figure 1: Soccer game

2 The soccer game

In this section, we describe soccer[3], a zero-sum game for which there do not exist deterministic equilibrium policies. The soccer field is a grid. The circle represents the ball. There are two players, whose possible actions are N, S, E, W, and stick. The players actions are executed in random order. If this sequence of actions causes the players to collide, then only the first moves. But if the player with the ball moves second, then the ball changes possession.2 If the player with the ball moves into a goal, then he scores +100 if it is in fact his own goal and the other player scores 100, or he scores 100 if it is the other players goal and the other player scores +100. In either case, the game ends. In this simple soccer game, there do not exist deterministic equilibrium policies, since at some states there do not exist deterministic equilibria. For example, at the state depicted in fig 1, any deterministic policy for player B is subject to indefinite blocking by player A. But if player B employs a nondeterministic policy, then player B can hope to pass player A on his next move.

3 Implementation and discussion

We believe that we successfully reproduce reproduce the results from the article. Figure2 shows that while the multiagent-Q learning algorithms Correlated-Q, Foe-Q, Friend-Q converge, but Q-learning does not. Our implementation of Q-learning is on-policy and -greedy, with $\epsilon = 0.01$ and the discount factor γ=0.9. In our case, the Q-value difference started to reduce after 9x10^5 iterations, but in the original paper Q-value had started to reduce after 7x10^5 iterations. We think this difference is due to the different α-decay schedule. We reduced the same $\delta\alpha$ in each iteration. The article mentioned this Q-value decrease is attributed to α getting smaller in each iteration.

For friend-Q learning it was an exact match with the article. We used exponential α-decay with initial α=0.2, where the Q-value difference quickly drops to zero around 0.5x10^5 iterations the learner is told to treat each other agent as either a "friend" who is helping to score in each iteration! Foe-Q algorithm provides strong guarantees on the learned policy, learner acts in a way that achieves its learned value independent of its opponents action choices, the sense of adversary is just to reduce Player As returns reflecting the zero-sum nature of the game. In this probabilistic game, we used linear programming to solve the probabilities for each action to take in each step. Then the current states Q value was

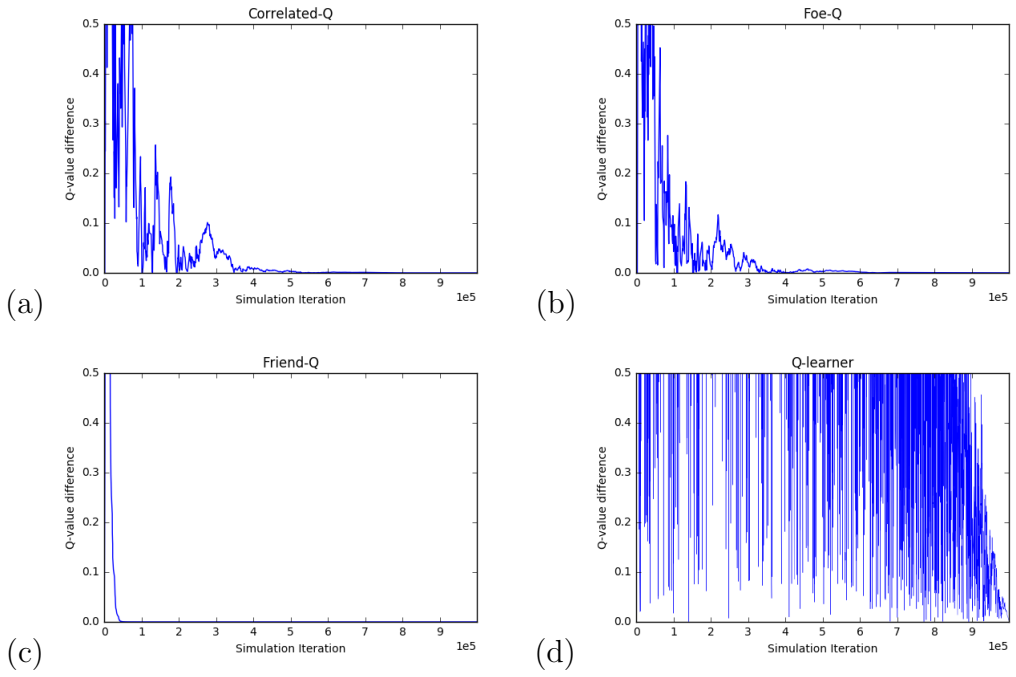

Figure 2: Convergence plot in the soccer game. All algorithms except Q-learning converge (a) Correlated-Q (b) Foe-Q (c) Friend-Q (d) Q-learning

updated using these probabilities multiplied by the Q values of those future actions and discounting for future states. For this implementation of Foe-Q learning the starting α was 0.2 with exponential decay and the discount factor γ=0.9. The Q-value difference profile of our implementation and the paper are not in a exact match, but the overall shape loops similar, and for both cases the difference drops to zero after $7\text{x}10^5$ iterations.

Similarly, for Correlated-Q learning we solve for a correlated equilibrium using linear programming so that it produces a higher reward for each player than acting independently. We then update the Q values of each state action pair using that reward. Other important sanity check I was able to reproduce for this report is that, both Foe-Q and Correlated-Q learning converge to the same Q values as the article states. In the figure 2 we see the Q value difference drops to zero near 600k iteration for both algorithms and the plots look similar!

In conclusion, the plots are very good match with the article. The choice of initial α, α-decay schedule and epsilon parameter is important for this reproduce work. The algorithms convergence was robust. We have experimented this robustness for other states as well.

References

[1] A. Greenwald and K. Hall. Correlated Q-learning. In Proceedings of the Twentieth International Conference on Machine Learning, pages 242249, 2003.

[2] J. Hu and M. Wellman. Nash Q-learning for general-sum stochastic games. Machine Learning Research, 4:10391069, 2003.

[3] M. Littman. Markov games as a framework for multi-agent reinforcement learning. In Proceedings of the Eleventh International Conference on Machine Learning, pages 157 163, July 1994.

[4] M. Littman. Friend or foe Q-learning in general-sum Markov games. In Proceedings of Eighteenth International Conference on Machine Learning, pages 322328, June 2001.