Actor-Critic Algorithms for Constrained Multi-agent Reinforcement Learning*

Extended Abstract

Raghuram Bharadwaj Diddigi
Indian Institute of Science
Bangalore, India
raghub@iisc.ac.in

D. Sai Koti Reddy
IBM Research
Bangalore, India
saikotireddy@in.ibm.com

Prabuchandran K.J.
Amazon-IISc Postdoctoral Fellow
Bangalore, India
prabuchandra@iisc.ac.in

Shalabh Bhatnagar
Indian Institute of Science
Bangalore, India
shalabh@iisc.ac.in

ABSTRACT

Multi-agent reinforcement learning has gained lot of popularity primarily owing to the success of deep function approximation architectures. However, many real-life multi-agent applications often impose constraints on the joint action sequence that can be taken by the agents. In this work, we formulate such problems in the framework of constrained cooperative stochastic games. Under this setting, the goal of the agents is to obtain joint action sequence that minimizes a total cost objective criterion subject to total cost penalty/budget functional constraints. To this end, we utilize the Lagrangian formulation and propose actor-critic algorithms. Through experiments on a constrained multi-agent grid world task, we demonstrate that our algorithms converge to near-optimal joint action sequences satisfying the given constraints.

KEYWORDS

Constrained Reinforcement Learning; Multi-agent Learning; Actor-Critic Algorithms; Cooperative Stochastic Game.

ACM Reference Format:
Raghuram Bharadwaj Diddigi, D. Sai Koti Reddy, Prabuchandran K.J., and Shalabh Bhatnagar. 2019. Actor-Critic Algorithms for Constrained Multi-agent Reinforcement Learning. In *Proc. of the 18th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2019), Montreal, Canada, May 13–17, 2019,* IFAAMAS, 3 pages.

1 INTRODUCTION

In the reinforcement learning (RL) paradigm, an agent interacts with the environment by selecting actions in a trial and error manner. The agent receives rewards for the chosen actions and the goal of the agent is to learn to choose actions so as to maximize a certain long-run reward objective. Many real world problems however cannot be considered in the context of single agent RL owing to which the study of the multi-agent RL framework has emerged [5]. In this paper, we consider the fully cooperative multi-agent setting which has gained popularity in recent times [6–9, 11, 12].

In many real-life multi-agent applications one often encounters constraints specified on the joint action sequence taken by the agents. Under this setting, the goal of the agents is to obtain the optimal joint action sequence that minimizes a long-run objective function while meeting the constraints. These problems are studied as "Constrained Markov Decision Process" (C-MDP) [1–4, 10] for the single agent RL setting and as "Constrained stochastic game (C-SG)" for the multi-agent RL setting respectively. In this work, our goal is to develop multi-agent RL algorithms for the setting of constrained cooperative stochastic games.

2 MODEL

A stochastic game is an extension of the single agent Markov Decision Process to multiple agents. A stochastic game is described by the tuple $(n, S, A_1, ...A_n, T, C, \gamma)$ where n denotes the number of agents participating in the game, S denotes the state space of the game, A_i, denotes the action space of agent i, $i = 1, \dots, n$, $C : S \times A_1 \times ... \times A_n \times S \rightarrow \mathbb{R}$ denotes the common cost incurred by the agents for the joint action profile $(a_1, a_2, \dots, a_n)$, $a_i \in A_i$, $i \in \{1, 2 \dots, n\}$, $T : S \times A_1 \times ... \times A_n \times S \rightarrow [0, 1]$ denotes the probability transition mechanism and $\gamma \in (0, 1]$ is the discount factor. Let $X_t \in S$ denote the state of the game at time t. Assume that X_0 is sampled from an initial distribution D. Let $\pi_i : S \times A \rightarrow [0, 1]$ be the stochastic policy followed by the agent i. Let $P : S \times A_1 \times ... \times A_n \times S \rightarrow \mathbb{R}$ be the single stage cost function for the common penalty cost and α be a certain prescribed penalty threshold.

The objective of the agents in the cooperative stochastic game is to learn a joint constrained optimal policy $\pi^* = (\pi_1^*, \dots, \pi_n^*)$, i.e., the one that gives a solution to the following constrained optimization problem:

$$\min_{\pi} E\Big[\sum_{t=0}^{\tau-1} \gamma^t C(X_t, \pi(X_t), X_{t+1})\Big] \qquad \text{s.t} \tag{1}$$

$$E\Big[\sum_{t=0}^{\tau-1} \gamma^t P(X_t, \pi(X_t), X_{t+1})\Big] \leq \alpha, \tag{2}$$

where $\pi = (\pi_1, \dots, \pi_n)$, $\alpha \in \mathbb{R}$ is given and $\tau \geq 1$ denotes the number of time steps until the terminal state is reached. In order to solve for π^*, we consider the Lagrangian formulation of the multi-agent constrained setting [2, 4]. Let λ denote the Lagrange

[*] Equal contribution by the first three authors.

Proc. of the 18th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2019), N. Agmon, M. E. Taylor, E. Elkind, M. Veloso (eds.), May 13–17, 2019, Montreal, Canada. © 2019 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org). All rights reserved.

multiplier for the constraint. We define the Lagrangian cost function as follows:

$$L(\pi, \lambda) = E\Big[\sum_{t=0}^{\tau-1} \gamma^t \big(C(X_t, \pi(X_t), X_{t+1}) + \lambda P(X_t, \pi(X_t), X_{t+1})\big)\Big] - \lambda\alpha. \quad (3)$$

The optimal policy π^* and the corresponding optimal Lagrange parameter vector λ^* are obtained as follows:

$$(\lambda^*, \pi^*) = \arg\sup_{\lambda} \inf_{\pi} L(\pi, \lambda) \quad (4)$$

3 PROPOSED ALGORITHMS

To accomplish the task of finding the optimal Lagrange multipliers and optimal policy, we propose a Nested Actor-Critic (N-AC) architecture. In this setup, the inner (policy) actor-critic computes the policy of the agents while the outer (penalty) actor-critic computes the Lagrange parameters.

3.1 JAL N-AC

In JAL N-AC (Joint Action Learners N-AC), all the agents compute the optimal joint policy of all the agents in the SG. Therefore, the action space is the cartesian product of action spaces of all the agents.

The policy (inner) critic parameters θ_c are trained by minimizing the loss parameter: $L(\theta_c) = (r_t + \gamma V_{\theta_c}(X_{t+1}) - V_{\theta_c}(X_t))^2$ where $r_t = C(X_t, a_t, X_{t+1}) + \lambda P_t(X_t, a_t, X_{t+1})$ and $V_{\theta_c}(X_t)$ denotes the value function approximated by the critic architecture for the state X_t with reward r_t.

Having found the parameters θ_c for the policy critic, the policy actor utilizes these to compute policy gradients for improving the policy θ_π as follows:

$$\theta_\pi(t+1) = \theta_\pi(t) - a(t)(r_t + \gamma V_{\theta_c}(X_{t+1}) - V_{\theta_c}(X_t)) \nabla_{\theta_\pi} \log \pi(a_t|X_t)).$$

The penalty critic estimates the penalty value function parameters θ_p. These parameters are trained by minimizing the loss function $L(\theta_p)$ defined as

$$L(\theta_p) = (P(X_t, a_t, X_{t+1}) + \gamma V_{\theta_p}(X_{t+1}) - V_{\theta_p}(X_t))^2,$$

where $V_{\theta_p}(X_t)$ is the value function associated with the penalty cost P in the state X_t.

Finally, the Lagrange parameters are updated by the penalty actor by performing stochastic gradient ascent as follows ([2, 4]):

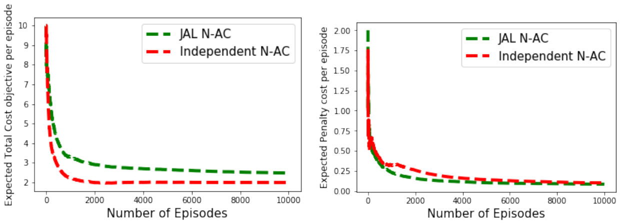
$$\lambda_{t+1} = max(0, \lambda_t + b(t)(V_{\theta_p}(X_t) - \alpha)),$$

where $b(t)$ is the step-size parameter and λ_t is the Lagrange parameter at time t.

3.2 Independent N-AC

In this algorithm, each agent has its own nested actor-critic architecture i.e., there are totally n nested actor-critic architectures. Each of the agents learns parameters of its nested actor-critic architecture separately and each agent i estimates its individual policy π_i. That is, each agent maintains its own policy actor-critic as well as penalty actor-critic network.

4 EXPERIMENTS AND RESULTS



(a) Performance of algorithms in terms of minimizing the total cost objective

(b) Performance of algorithms in terms of minimizing the penalty cost objective

Figure 1: Performance of our proposed algorithms

We consider a constrained grid world experiment where there is a grid of size 4 × 4 and two agents in the grid. The objective of the agents is to learn the shortest path from any given source to a prescribed target, under the constraint that the maximum overlap in path is less than the penalty constraint of $\alpha = 0.1$. The state of each agent $s_i, i \in 1, 2$ is a vector of size 16, with 1 at the current position of the agent i and 0 at all other positions. The action of each agent in the grid includes moving up, down, left, right or staying in the same position, wherever applicable. In Figure 1, we can see that both the expected total cost and the expected penalty cost decrease as the number of episodes increase in both the algorithms. The expected total cost and the penalty cost are computed as the averages of total cost and penalty cost, respectively obtained over a single run of 10, 000 training episodes. In Table 1, we report the average penalty obtained by the converged policy over 10, 000 test episodes of both the proposed algorithms. We conclude that our algorithms compute near-optimal policies that not only minimize the total cost but also meet the penalty constraints.

Table 1: Performance of converged policy in constrained grid world

Algorithm	**Average Penalty of the converged policy**
JAL N-AC	0.0593
Independent N-AC	0.0656

5 ACKNOWLEDGEMENT

We acknowledge support from RBCCPS, IISc and DST, India. Prabuchandran K.J. acknowledges support from the Amazon-IISc Postdoctoral fellowship.

REFERENCES

[1] Eitan Altman. 1999. *Constrained Markov decision processes*. Vol. 7. CRC Press.
[2] Shalabh Bhatnagar. 2010. An actor–critic algorithm with function approximation for discounted cost constrained Markov decision processes. *Systems & Control Letters* 59, 12 (2010), 760–766.
[3] Shalabh Bhatnagar and K Lakshmanan. 2012. An online actor–critic algorithm with function approximation for constrained Markov decision processes. *Journal of Optimization Theory and Applications* 153, 3 (2012), 688–708.

[4] Vivek S Borkar. 2005. An actor-critic algorithm for constrained Markov decision processes. *Systems & control letters* 54, 3 (2005), 207–213.

[5] Lucian Busoniu, Robert Babuska, and Bart De Schutter. 2008. A comprehensive survey of multiagent reinforcement learning. *IEEE Transactions on Systems, Man, And Cybernetics-Part C: Applications and Reviews, 38 (2), 2008* (2008).

[6] Jakob Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. 2017. Counterfactual multi-agent policy gradients. *arXiv preprint arXiv:1705.08926* (2017).

[7] Jakob Foerster, Nantas Nardelli, Gregory Farquhar, Triantafyllos Afouras, Philip HS Torr, Pushmeet Kohli, and Shimon Whiteson. 2017. Stabilising experience replay for deep multi-agent reinforcement learning. *arXiv preprint arXiv:1702.08887* (2017).

[8] Jayesh K Gupta, Maxim Egorov, and Mykel Kochenderfer. 2017. Cooperative multi-agent control using deep reinforcement learning. In *International Conference on Autonomous Agents and Multiagent Systems*. Springer, 66–83.

[9] Landon Kraemer and Bikramjit Banerjee. 2016. Multi-agent reinforcement learning as a rehearsal for decentralized planning. *Neurocomputing* 190 (2016), 82–94.

[10] K Lakshmanan and Shalabh Bhatnagar. 2012. A novel Q-learning algorithm with function approximation for constrained Markov decision processes. In *Communication, Control, and Computing (Allerton), 2012 50th Annual Allerton Conference on.* IEEE, 400–405.

[11] Ryan Lowe, Yi Wu, Aviv Tamar, Jean Harb, Pieter Abbeel, and Igor Mordatch. 2017. Multi-agent actor-critic for mixed cooperative-competitive environments. In *Advances in Neural Information Processing Systems*. 6379–6390.

[12] Frans A Oliehoek, Matthijs TJ Spaan, and Nikos Vlassis. 2008. Optimal and approximate Q-value functions for decentralized POMDPs. *Journal of Artificial Intelligence Research* 32 (2008), 289–353.