Attention Actor-Critic algorithm for Multi-Agent Constrained Co-operative Reinforcement Learning

P. Parnika*,[1], Raghuram Bharadwaj Diddigi*,[2], Danda Sai Koti Reddy*,[3], Shalabh Bhatnagar[2]

[1] Mindtree Ltd., India, [2] IISc Bangalore, India, [3] Avesha Inc., India

Objective

Compute optimal actions for multiple agents working cooperatively to achieve a common goal while satisfying the constraints specified on their actions. To obtain optimal actions, we propose an actor critic algorithm that makes use of attentive critics in the constrained multi-agent RL setting.

Introduction

- In a multi-agent co-operative RL setting, multiple agents work towards a common goal in a common environment. In the real world, the choice of actions that can be taken by these agents is constrained.
- Hence, they have to learn actions that not only minimize the expected total discounted cost but also respect the constraints specified.
- We extend the Actor-Critic algorithm that makes use of attentive critics to the constrained multi-agent RL setting.
- By incorporating different attention modes, the agents can select useful information required for optimizing the objective and satisfying the constraints separately, thereby yielding better actions.
- A stochastic game is an extension of the single agent Markov Decision Process to multiple agents. A stochastic game is described by the tuple $< n, S, A, T, k, c_1, \ldots, c_m, \gamma >$.
- The objective of the agents in the cooperative stochastic game is to learn a joint constrained optimal policy $\pi^* = (\pi_1^*, \ldots, \pi_n^*)$, i.e., the one that gives a solution to the following constrained optimization problem:

$$\min_{\pi \in \Pi} \; J(\pi) = \; E\Big[\sum_{t=0}^{\tau-1} \gamma^t k(s_t, \pi(s_t))\Big] \tag{1}$$
$$\text{s.t} \quad E\Big[\sum_{t=0}^{\tau-1} \gamma^t c_j(s_t, \pi(s_t))\Big] \leq \alpha_j, \; \forall j \in 1, \ldots, m,$$

Let λ denote the Lagrange multiplier for the constraint. We define the Lagrangian cost function as follows:

$$L(\pi, \lambda) = E\Big[\sum_{t=0}^{\tau-1} \gamma^t \big(k(X_t, \pi(X_t)) + \sum_{j=1}^{m} \lambda_j c_j(X_t, \pi(X_t))\big)\Big] - \sum_{j=1}^{m} \lambda_j \alpha_j. \tag{2}$$

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

$$(\lambda^*, \pi^*) = \arg\sup_{\lambda} \inf_{\pi} L(\pi, \lambda) \tag{3}$$

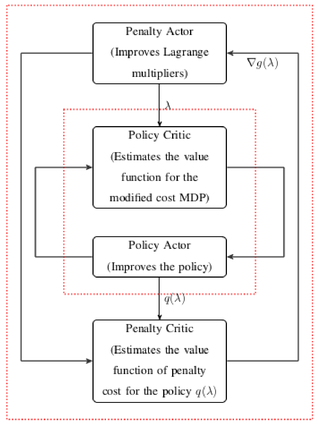

Figure 1:Constrained Actor-Critic [1]

MACAAC

The inner (policy) actor-critic computes the policy of the agents while the outer (penalty) actor-critic computes the Lagrange parameters.

- Each agent maintains a total of m+1 critics which use attention.
- The intuition here is, by having multiple critics with different attentions, each critic is especially able to attend to that information which is crucial in solving its objective.
- The critic parameters are updated by performing gradient descent on the MSE loss given by

$$\sum_{i=1}^{n} E[(Q^i(o, a) - y_i)^2], \tag{4}$$

where $y_i = r + \gamma E[Q^i(o^*, a^*) - \alpha \log(\pi_\theta(a_i^*|o_i^*))]$, o^*, a^* are the joint next state and actions of agents and α is known as the temperature coefficient.

- The policy parameters of each agent are updated by performing gradient descent using the gradient function given by:

$$E[\nabla_{\theta_i} \log(\pi_{\theta_i}(a_i|o_i))(-\alpha \log(\pi_{\theta_i}(a_i|o_i)) + Q_\psi^i(o, a) - b(o, a_{-i}))]. \tag{5}$$

- Finally, the Lagrange parameters $\lambda_j, \; j = 1, \ldots, m$ are updated by performing gradient ascent on the Lagrangian L (eq. 2)
- The critic and actor updates are performed on a faster time-scale compared to the Lagrange parameter updates. As a result, the critic and actor perceive Lagrange parameters as constants in their updates, thereby ensuring the convergence of the algorithm.

Experiments and Results

- We consider a constrained version of co-operative navigation.
- In the constrained version of Cooperative Navigation, there are 5 agents and 5 targets that are randomly generated in a continuous environment at the beginning of each episode.
- The objective of the agents is to navigate towards the targets in a cooperative manner such that all targets are covered.
- The cost at each time step is the sum of the distance to the nearest agent, over all the targets.
- We include a penalty of 1 when there is a collision between the agents (and 0 otherwise). The penalty threshold (α) is set to 3. Therefore, expected total penalty over all the episodes must be less than or equal to 3.
- For comparison purposes, we also implement the constrained version of MADDPG algorithm, which we refer to as 'MADDPG-C' and un-constrained version, which we simply refer to as 'Unconstrained'.

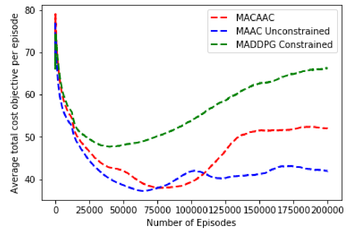

Figure 2:Expected total cost

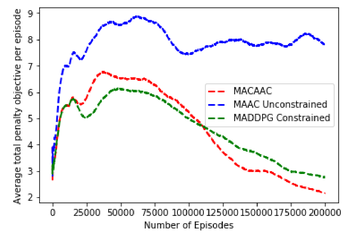

Figure 3:Expected total penalty

Discussion

- We observe that the total cost approaches convergence for all the three algorithms.
- The 'Unconstrained' algorithm achieves the smallest average cost as there is no penalty for collisions in this case. Therefore, the agents can move freely in the continuous space and navigate quickly towards the targets.
- We see that the average penalty comes down as the training progresses for the constrained algorithms, while for the 'unconstrained' algorithm it almost remains constant. This is the effect of Lagrange parameters that are learnt in the constrained setting.

References

[1] Raghuram Bharadwaj Diddigi, Sai Koti Reddy Danda, Prabuchandran K.J., and Shalabh Bhatnagar. Actor-critic algorithms for constrained multi-agent reinforcement learning. *arXiv preprint arXiv:1905.02907*, 2019.