Training Cooperative Agents for Multi-Agent Reinforcement Learning

Extended Abstract

Sushrut Bhalla
University of Waterloo
Waterloo, Ontario
sushrut.bhalla@uwaterloo.ca

Sriram G. Subramanian
University of Waterloo
Waterloo, Ontario
s2ganapa@uwaterloo.ca

Mark Crowley
University of Waterloo
Waterloo, Ontario
mcrowley@uwaterloo.ca

ABSTRACT

Deep Learning and back-propagation has been successfully used to perform centralized training with communication protocols among multiple agents in a cooperative environment. In this paper we present techniques for centralized training of Multi-Agent (Deep) Reinforcement Learning (MARL) using the model-free Deep Q-Network as the baseline model and message sharing between agents. We present a novel, scalable, centralized MARL training technique, which separates the message learning module from the policy module. The separation of these modules helps in faster convergence in complex domains like autonomous driving simulators. A second contribution uses the centrally trained model to bootstrap training of distributed, independent, cooperative agent policies for execution and thus addresses the challenges of noise and communication bottlenecks in real-time communication channels. This paper theoretically and empirically compares our centralized training algorithms to current research in the field of MARL. We also present and release a new OpenAI-Gym environment which can be used for multi-agent research as it simulates multiple autonomous cars driving cooperatively on a highway.

KEYWORDS

MARL; Multi-Agent Reinforcement Learning; Reinforcement Learning; MultiAgent Systems; Autonomous Driving

ACM Reference Format:
Sushrut Bhalla, Sriram G. Subramanian, and Mark Crowley. 2019. Training Cooperative Agents for 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

We propose two centralized training algorithms for MARL environments using DQN as the baseline. The first approach extends the idea of using communication channels for message sharing as proposed in [2] and extends it to multi-agent same discrete time-step communication, where the communication protocol is trained using back propagation [5]. The second approach introduces a broadcast network which generates a single broadcast message for all agents in the environment and thus reduces channel bandwidth and memory requirements of the algorithm. In the real world driving environment, communication channels will not always be reliable and the messages could be corrupted by noise or delayed due to latency. Another contribution uses our centralized cooperative policy to bootstrap training of a decentralized cooperative policy. We evaluate our methods against current state of the art techniques in MARL on a multi-agent autonomous driving environment. We have developed an OpenAI Gym environment [1] which simulates multiple autonomous and adversary cars driving on a highway.

Effective communication channels in MARL can be trained using backpropagation [2, 5]. [5] employs a message sharing protocol where an aggregated message is generated by averaging the messages from all agents and passing it back as an input to the agents along with their observation's hidden state representation to compute the final action-values. This Iterative Message Sharing (**IMS**) is iterated P times in a single discrete time-step of the environment before the final action for all agents at that time-step is computed. Differentiable Inter-Agent Learning **DIAL** [2] also trains communication channels, through back-propagation, for sequential multi-agent environments. However, the messages exchanged between the agents are from the past time-steps. This causes a sub-optimal convergence as we show in our experiments section. Our work differs from these approaches in two ways. (a) We remove the iterative network structure of communication protocol and replace it with a feed-forward neural network. (b) We use the centralized structure during training only and train a decentralized policy for execution as the communication among agents in our environment is not guaranteed.

2 METHODS

2.1 Multi-Agent Message Sharing Network

MA-MeSN uses a similar network structure to DIAL except we train agents to learn a negotiation message sharing protocol in the same discrete time-step. Training with messages from the same discrete time-step reduces the complexity of the message when compared to a message generated based on past observations. MA-MeSN treats the communication as a speech act and thus it separates the message generation network from the policy network. The message network is optimized by applying the averaged policy gradients from the policy networks of all agents. The reduced complexity of the generated messages along with task decoupling (speech act and agent policy) allows for faster convergence, stability during training, and a better final policy when compared to DIAL.

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.

2.2 Multi-Agent Broadcast Network

MA-BoN is designed as an extension to IMS to achieve a cooperative MARL policy for homogeneous agents with a reduced communication channel throughput, when compared to previous approaches. MA-BoN structure is similar to IMS except we replace the averaging module with a feed-forward neural network which is trained to generate a single broadcast message (vs multiple rounds of messages in IMS). All agents generate an embedding of their private observation which is concatenated to generate a unified message using a shared broadcast network. This message is passed back to the agents which compute a policy using the private observation and the broadcast message. The central hub for message generation improves the throughput of the network with a reduced number of trainable parameters.

2.3 Cooperative Distributed Behavior Cloning

CoDBC is used to achieve a fully decentralized (independent) policy. MA-BoN and MA-MeSN can be partially decentralized by using a Gumbel-Softmax [4] approximation of the continuous message signal. To achieve communication free cooperative policy, we bootstrap the training of decentralized policy by sampling trajectories from centralized (MA-MeSN/MA-BoN) policy. We use imitation learning to train a neural network to imitate the policy of the expert (MA-MeSN).

3 EXPERIMENTS AND RESULTS

Driving Environment. We have developed a multi-vehicle driving simulator which simulates multiple autonomous and adversary vehicles driving on a highway. The adversary's objective is to hit the closest car and all cooperative autonomous cars must avoid crashes. The MARL agents receive a hidden observation of the environment and a private reward based on distance from the closest agent but don't know which car is autonomous or adversary. The agents can communicate using a discrete limited bandwidth channel.

The results for centralized training of cooperative multi-agents are shown in Fig. 1(a) (averaged over 20 runs). The IMS and DIAL algorithm are able to avoid divergence because the messages passed are trained through back-propagation; however, the learning curve for DIAL and IMS is slower than MA-MeSN and MA-BoN. DIAL shows steady improvement in performance, however, the performance of the final policy is weak when compared to MA-MeSN, because the DIAL messages from the past cover a larger message space and thus can only train on samples of the current episode. The MA-BoN and MA-MeSN use step-based replay memory (z_i^t, a_i^t, r_i^t) along with $m_{-\mathrm{i}}^t$ which provides better indexing of the changing policies of other agents over time and thus allows for a more stable training algorithm. As a result, we see a stable learning curve with faster convergence properties than DIAL and IMS. The MA-BoN results show comparative performance to MA-MeSN with reduced communication steps ($|N| \times |N|$ to $|N|$).

The results for decentralized training are shown in Fig. 1(b). We compare the performance of our centralized MA-MeSN, decentralized CoDBC, independent DQN and independent DQN with stabilized experience replay (SER) [3]. As the treadmill environment does not explicitly reward agents for cooperation, we see poor performance from DQN and DQN with SER. DQN with SER

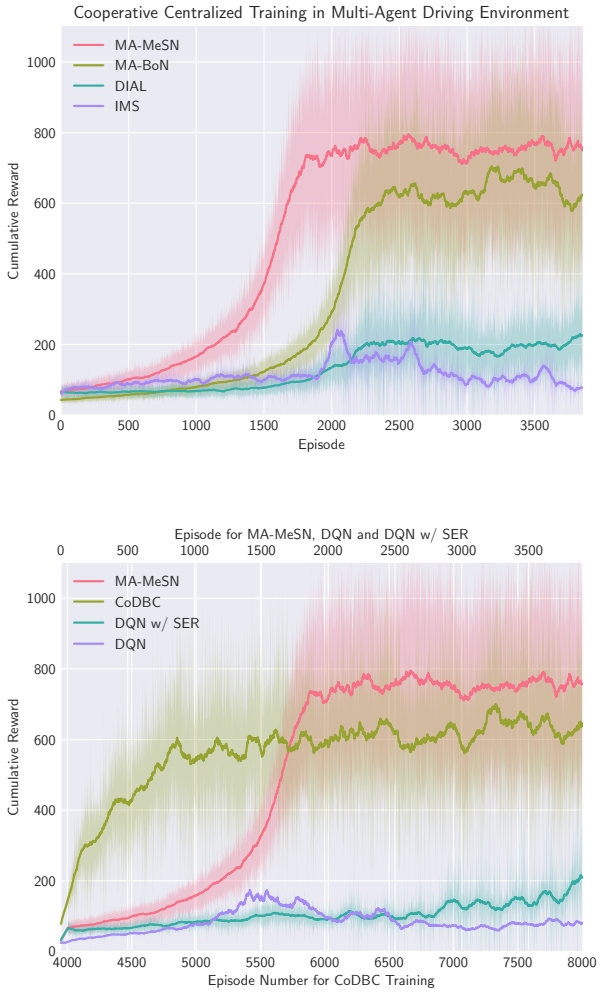

Figure 1: Cumulative reward for (a) Centralized (b) Decentralized training on the Driving Environment.

computes the weight of each sample's gradient using a linearly decaying function based on the episodes elapsed since a sample was collected. Thus, DQN with SER is able to prioritize its training on the latest samples which represent the latest policies of other agents and avoid divergence. The CoDBC method outperforms all other decentralized techniques while achieving cooperative behavior. The CoDBC algorithm is trained sequentially after MA-MeSN is fully trained; as opposed to the parallel curves shown in the figure. The CoDBC achieves an 86.6% accuracy compared to MA-MeSN. We freeze the final expert MA-MeSN policy for all agents before training the CoDBC policy, and mitigate the non-stationarity issue in MARL environments. This approach is ideal for real-time agents in MARL environments with a goal of cooperation as communication channels are unreliable and induce a time-latency.

REFERENCES

[1] Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. 2016. OpenAI Gym. (2016). arXiv:arXiv:1606.01540

[2] Jakob Foerster, Ioannis Alexandros Assael, Nando de Freitas, and Shimon Whiteson. 2016. Learning to communicate with deep multi-agent reinforcement learning. In *Advances in Neural Information Processing Systems*. 2137–2145.

[3] Jakob Foerster, Nantas Nardelli, Greg Farquhar, Phil Torr, Pushmeet Kohli, and Shimon Whiteson. 2017. Stabilising Experience Replay for Deep Multi-Agent Reinforcement Learning. In *ICML 2017: Proceedings of the Thirty-Fourth International Conference on Machine Learning*. http://www.cs.ox.ac.uk/people/shimon.whiteson/pubs/foerstericml17.pdf

[4] Eric Jang, Shixiang Gu, and Ben Poole. 2016. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144* (2016).

[5] Sainbayar Sukhbaatar, Arthur Szlam, and Rob Fergus. 2016. Learning Multiagent Communication with Backpropagation. In *Advances in Neural Information Processing Systems 29*, D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett (Eds.). Curran Associates, Inc., 2244–2252. http://papers.nips.cc/paper/6398-learning-multiagent-communication-with-backpropagation.pdf