A New Framework for Multi-Agent Reinforcement Learning – Centralized Training and Exploration with Decentralized Execution via Policy Distillation

Extended Abstract

Gang Chen
Victoria University of Wellington
Wellington, New Zealand
aaron.chen@ecs.vuw.ac.nz

ABSTRACT

Multi-agent deep reinforcement learning demands for highly coordinated environment exploration among all the participating agents. Previous research attempted to address this challenge through learning centralized value functions. However, the common strategy for every agent to learn their local policies directly may fail to nurture inter-agent collaboration and can be sample inefficient whenever agents alter their communication channels. To address these issues, we propose a new framework known as *centralized training and exploration with decentralized execution via policy distillation*. Guided by this framework, we will first train agents' policies with shared global component to foster coordinated and effective learning. Locally executable policies will be derived subsequently from the trained global policies via policy distillation.

KEYWORDS

Deep Reinforcement Learning; Multi-Agent Learning; Policy Distillation

ACM Reference Format:
Gang Chen. 2020. A New Framework for Multi-Agent Reinforcement Learning – Centralized Training and Exploration with Decentralized Execution via Policy Distillation. In *Proc. of the 19th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2020), Auckland, New Zealand, May 9–13, 2020,* IFAAMAS, 3 pages.

1 INTRODUCTION

Many practical applications of *deep reinforcement learning* (DRL) algorithms naturally involve more than one interdependent learners [1, 3]. In fact, partitioning the problem domain according to the role played by each agent can result in a similar effect as imposing hierarchical abstractions in both time and space, contributing positively to improved learning efficiency and effectiveness. MADRL also stands for a major paradigm for training agents to interact with each other productively [9].

Researchers have attempted to tackle MADRL problems by using single-agent DRL algorithms with success [10]. Despite of promising results, *environment non-stationarity* hinders stable DRL because individual agents can no longer perceive their environment as being stationary since it is also influenced by other agents' activities. This issue has triggered the wide adoption of the fundamental paradigm known as *centralized training with decentralized execution* (CTDE) [4, 6].

Our research is inspired by the understanding that effective MADRL demands for coordinated environment exploration among all the participating agents [7]. However, the common strategy for agents to directly learn their *local policies* with restricted access to local observations may fail to meet this requirement. Such lack of observability can be mitigated with *inter-agent communication* that allow agents to share instant messages and expand their environment knowledge. However, this will inevitably lead to increased learning complexity, especially when agents are struggling to learn how to communicate effectively and how to maximize their *expected long-term rewards* at the same time.

In order to achieve coordinated environment exploration and high sample efficiency, we propose a new framework known as *Centralized Training and Exploration with Decentralized execution via policy Distillation* (i.e., CTEDD) as an extension of CTDE to promote *global information sharing*. Before building up their local policies, agents first train their *global policies* to process full state input through a shared global *deep neural network* (DNN). Such global DNN paves the way for coordinated action sampling and environment exploration. Specifically, with the help of a maximum-entropy RL technique [2], all agents will learn to collectively decide when to explore aggressively and when to focus on exploiting policies learned so far, nurturing balanced tradeoff between exploration and exploitation.

Guided by CTEDD, we further propose to adopt the *policy distillation* technique [8] to derive locally executable policies for every agent from trained global policies, enabling parallel training of local policies with flexible inter-agent communication capabilities. A recently developed algorithm termed MADDPG [6] will serve as the baseline algorithm in this paper. Building on MADDPG, we will explore the key advantages of CTEDD over the prevalent approach of learning agents' local policies directly. Particularly, empirical comparison with MADDPG confirms that CTEDD is more sample efficient and effective, while guaranteeing decentralized execution of the learned policies.

2 METHODS

Identical to MADDPG, our algorithm aims to learn locally executable policies for each agent in a multi-agent system. However,

Proc. of the 19th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2020), B. An, N. Yorke-Smith, A. El Fallah Seghrouchni, G. Sukthankar (eds.), May 9–13, 2020, Auckland, New Zealand.

instead of training these local policies directly, our algorithm realizes the idea of CTEDD by dividing policy training into two consecutive stages, i.e., the *centralized training and exploration* stage and the *policy distillation* stage.

CTEDD first initializes and trains global policy networks and a centralized Q-network. All networks receive full state input. We design the neural network architecture for the global policies $\{\tilde{\pi}_i\}_{i=1}^N$ to satisfy two key requirements: (1)$\{\tilde{\pi}_i\}_{i=1}^N$ must support easy sharing of full state information across all agents A_i ($1 \leq i \leq N$); and (2) the network must allow efficient policy training and local action selection by individual agents. Driven by the two requirements, a mixed network architecture involving both global and local components has been developed, as illustrated in Figure 1. This design significantly reduces the total number of trainable parameters in the entire network structure. Building on this policy network design, we decide to train the policy parameters in the shared global DNN in Figure 1 via DDPG [5]. Similar to MADDPG, for stable learning, CTEDD makes use of the target networks for the global DNN in Figure 1 as well as the Q-network.

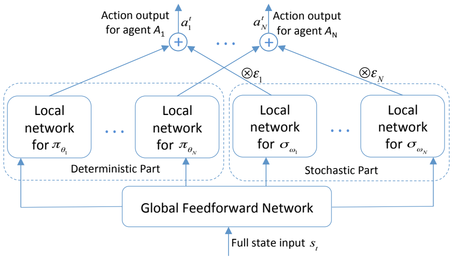


Figure 1: The global policy network design with mixed global and local components.

In order to train the policy parameters $\{\omega_i\}_{i=1}^N$ associated with the local components in Figure 1, we keep track of the most recently collected environment samples since the last learning iteration in a batch $\mathcal{B}$. Policy gradients can be subsequently estimated based on $\mathcal{B}$. Specifically, aimed at promoting effective and coordinated environment exploration, maximum-entropy DRL techniques will be employed by CTEDD to train $\{\omega_i\}_{i=1}^N$ along the direction below:

$$\begin{aligned}\Delta\omega_i \propto & \frac{1}{\|\mathcal{B}\|}\sum_{\mathcal{B}} \nabla_{\omega_i} \log \tilde{\pi}_i(s_t, \{a_j^t\}_{j=1}^N) Q^{\tilde{\pi}}(s_t, \{a_j^t\}_{j=1}^N) \\ & + \alpha \sum_{\mathcal{B}} \nabla_{\omega_i} \mathcal{H}(\{\tilde{\pi}_j(s_t, \cdot)\}_{j=1}^N)\end{aligned} , \quad (1)$$

where α is the *entropy regularization factor* and $\mathcal{H}(\{\tilde{\pi}_j(s_t, \cdot)\}_{j=1}^N)$ is the Shannon entropy for action sampling across all agents. Moreover $\nabla_{\omega_i} \log \tilde{\pi}_i(s_t, \{a_j^t\}_{j=1}^N)$ can be simplified to $\nabla_{\omega_i} \log \tilde{\pi}_i(s_t, a_i^t)$ and $\nabla_{\omega_i} \mathcal{H}(\{\tilde{\pi}_j(s_t, \cdot)\}_{j=1}^N)$ can be simplified to $\nabla_{\omega_i} \mathcal{H}(\tilde{\pi}_i(s_t, \cdot))$, since ω_i only affects local action selection by agent A_i.

To fulfill the learning goal, the second stage of CTEDD builds locally executable policies for every agent from the trained global policies. The network design of the local policies and the communication channels that connect agents' local policies together have been depicted in Figure 2. As shown, the local policy networks $\{\hat{\pi}_{\theta_i'}\}_{i=1}^N$ consist of two successive parts. Part 1 is responsible for message generation based on agents' local observations. Part 2 produces the final action output based on all messages received. We order all state samples chronologically according to when they were sampled. State samples obtained at earlier times will be used first to train local policies, followed by samples collected at later times. For every batch of samples $\mathcal{B}$ just retrieved from the buffer, the policy parameters $\{\theta_i'\}_{i=1}^N$ of all local policies will be trained for a few iterations to minimize the loss $\mathcal{L}_D$ defined in (2) below:

$$\mathcal{L}_D = \frac{1}{\|\mathcal{B}\|}\sum_{\mathcal{B}}\sum_{i=1}^{N} \left\| \pi_{\theta_i}(s_t) - \hat{\pi}_{\theta_i'}(o_i(s_t), \{m_{j\neq i}\}_{j=1}^N) \right\|_2^2 . \quad (2)$$

This process will continue until all samples in the buffer have been consumed.

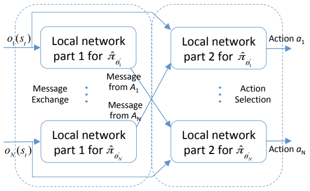
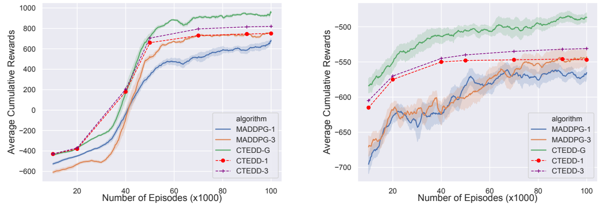

$o_1(s_t)$
Local network part 1 for $\hat{\pi}_{\theta_1'}$
Local network part 2 for $\hat{\pi}_{\theta_1'}$
Action a_1
Message Exchange
Message from A_1
Message from A_N
Action Selection
$o_N(s_t)$
Local network part 1 for $\hat{\pi}_{\theta_N'}$
Local network part 2 for $\hat{\pi}_{\theta_N'}$
Action a_N

Figure 2: The local policy network design.

3 EXPERIMENTS



(a) Push Forward

(b) Predator and Prey

Figure 3: The performance of CTEDD-G, CTEDD-1, CTEDD-3, MADDPG-1 and MADDPG-3 on two environments.

Experiments have been performed on several benchmarks[1]. Figure 3 presents some results that confirm that global policies (CTEDD-G) trained by CTEDD can outperform locally distilled policies (CTEDD-1 and CTEDD-3 with 1-bit and 3-bit communication channels, respectively), which clearly outperform local policies trained via MADDPG. CTEDD is also more sample efficient than MADDPG.

4 CONCLUSION

Effective DRL in complex multi-agent systems demand for highly coordinated environment exploration among all learning agents. This notion drove us to propose a new CTEDD framework to promote easy and effective sharing of global information. Our idea was realized by applying DDPG to training global policies approximated as DNNs with mixed local and global components. Meanwhile, a policy distillation technique was adopted by us to derive locally executable policies for each agent from well-trained global policies in a highly sample efficient manner.

[1] For more information, please check https://arxiv.org/abs/1910.09152

REFERENCES

[1] Gang Chen, Zhonghua Yang, Hao He, and Kiah Mok Goh. 2005. Coordinating multiple agents via reinforcement learning. *Autonomous Agents and Multi-Agent Systems* 10, 3 (2005), 273–328.

[2] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine. 2018. Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor. In *35th International Conference on Machine Learning*.

[3] P. Hernandez-Leal, B. Kartal, and M. E. Taylor. 2018. Is multiagent deep reinforcement learning the answer or the question? A brief survey. *arXiv preprint arXiv:1810.05587* (2018).

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

[5] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra. 2015. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971* (2015).

[6] R. Lowe, Y. Wu, A. Tamar, J. Harb, P. Abbeel, and I. Mordatch. 2017. Multi-agent actor-critic for mixed cooperative-competitive environments. In *Advances in Neural Information Processing Systems*. 6379–6390.

[7] L. Matignon, L. Jeanpierre, and A. I. Mouaddib. 2012. Coordinated multi-robot exploration under communication constraints using decentralized markov decision processes. In *AAAI conference on artificial intelligence*.

[8] A. A. Rusu, S. G. Colmenarejo, C. Gulcehre, G. Desjardins, J. Kirkpatrick, R. Pascanu, V. Mnih, K. Kavukcuoglu, and R. Hadsell. 2016. Policy distillation. In *International Conference on Learning Representations*.

[9] S. Sukhbaatar, Z. Lin, I. Kostrikov, G. Synnaeve, A. Szlam, and R. Fergus. 2017. Intrinsic motivation and automatic curricula via asymmetric self-play. *arXiv preprint arXiv:1703.05407* (2017).

[10] A. Tampuu, T. Matiisen, D. Kodelja, I. Kuzovkin, K. Korjus, J. Aru, J. Aru, and R. Vicente. 2017. Multiagent cooperation and competition with deep reinforcement learning. *PloS one* 12, 4 (2017), e0172395.