Dynamic Skill Selection for Learning Joint Actions

Extended Abstract

Enna Sachdeva
Oregon State University, Corvallis, Oregon
sachdeve@oregonstate.edu

Shauharda Khadka
Microsoft, Seattle, Washington
skhadka@microsoft.com

Somdeb Majumdar
Intel Labs, San Diego, California
somdeb.majumdar@intel.com

Kagan Tumer
Oregon State University, Corvallis, Oregon
kagan.tumer@oregonstate.edu

ABSTRACT

Learning in tightly coupled multiagent settings with sparse rewards is challenging because multiple agents must reach the goal state simultaneously for the team to receive a reward. This is even more challenging under temporal coupling constraints - where agents need to sequentially complete different components of a task in a particular order. Here, a single local reward is inadequate for learning an effective policy. We introduce MADyS, Multiagent Learning via Dynamic Skill Selection, a bi-level optimization framework that learns to dynamically switch between multiple local skills to optimize sparse team objectives. MADyS adopts fast policy gradients to learn local skills using local rewards and an evolutionary algorithm to optimize the sparse team objective by *recruiting* the most optimal skill at any given time. This eliminates the need to generate a single dense reward via reward shaping or other mixing functions. In environments with both spatial and temporal coupling requirements, we outperform prior methods and provides intuitive visualizations of its skill switching strategy.

KEYWORDS

Multiagent Coordination; Reinforcement Learning; Evolutionary Algorithm; Dynamic Skill Selection

ACM Reference Format:
Enna Sachdeva, Shauharda Khadka, Somdeb Majumdar, and Kagan Tumer. 2021. Dynamic Skill Selection for Learning Joint Actions: Extended Abstract. In *Proc. of the 20th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2021), Online, May 3–7, 2021*, IFAAMAS, 3 pages.

1 INTRODUCTION

Real-world multiagent tasks often require agents to collaborate with varying spatial and temporal coupling requirements [2, 3, 6, 11]. While spatial coupling requires multiple agents' simultaneous presence at a given location, temporal coupling requires agents to perform multiple sub-tasks in a fixed sequence. The sparsity of the shared team objective significantly increases with high spatial or temporal coupling. It is infeasible to learn effective coordination strategies by relying only on such a sparse global reward [9].

One potential solution to partially address this problem is reward shaping [1, 15], where a dense reward is heuristically designed to allow agents to learn an effective coordination policy. Recent work on D++ [12] provides stepping stone rewards to address the credit assignment problem in domains with spatial coupling requirements. However, designing such rewards requires a functional form of global rewards, and their effectiveness is limited by the sparsity of global reward. While a local approximation of difference evaluations could address this, it relies on agents generating useful training data by stumbling upon the goal state [5, 13]. LIIR [7] addresses the structural credit assignment problem by *learning* agent-specific rewards to enable coordinated exploration. MERL [10] addresses spatially coupled multiagent coordination tasks with sparse rewards by using dense proxy rewards. However, these approaches suffer in domains with extremely sparse rewards requiring both spatial and temporal coupling, as the likelihood of agents simultaneously stumbling upon the right joint action at the right time is extremely small.

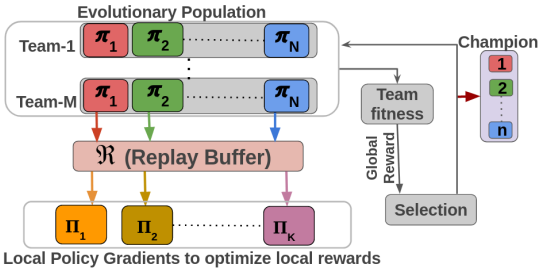


Figure 1: MADyS integrates local and global rewards.

2 MADyS: MULTIAGENT LEARNING WITH DYNAMIC SKILL SELECTION

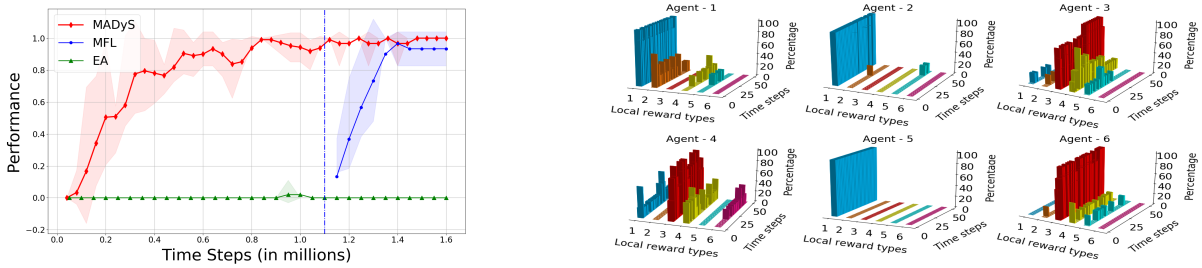
We introduce *MADyS*, a bi-level optimization framework that leverages a portfolio of semantically meaningful local rewards to coordinate agents across time and space. Each local reward corresponds to a basic skill and is based on domain knowledge.

Figure 1 depicts the general flow of *MADyS*. We initialize an EA population of M teams, each consisting of N agents. Each agent, π_n, is a neural network *skill picker* that maps a local observation to one of K skills. Separately, we initialize one *skill learner* for each of K as Π_k, which is a neural network that maps local observations to primitive action to execute in the environment when an agent picks this skill k. When an agent in the EA population picks a skill k, consecutively, skill learner Π_k generates the primitive to execute in the environment.

For every action that an agent takes in the environment, it receives a **vector** of local rewards as $r = [r_1, r_2, ...r_K]$, which captures

Proc. of the 20th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2021), U. Endriss, A. Nowé, F. Dignum, A. Lomuscio (eds.), May 3–7, 2021, Online.

(a) Configuration: 2 agents with spatial coupling of 2, and temporal coupling of 2. Agent-1 selects *"go to POI-A"* followed by *"go to POI-B"*, whereas Agent-2 learns to stay close to Agent-1 by picking *"go to agent"* for all time steps.



(b) Configuration: 6 agents with spatial coupling of 3, and temporal coupling of 2. Agent-1, Agent-2 and Agent-5 form a team of 3 and pick *"go to POI-A"*, and Agent-3, Agent-4 and Agent-6 form another team of 3 and pick *"go to POI-B"*.

Figure 2: Training curves (left) and histograms (right) showing the distributional shift of local rewards, for various spatial and temporal coupling. The vertical dotted blue line denotes the time steps required to pre-train the skills for the MFL baseline.

how good or bad that action is for *all* K skills. The experiences gathered during the EA rollouts are stored in a shared replay buffer $\mathcal{R}$, as tuples $< s, a, s', r >$.

EA pushes each agent to select the skill most likely to maximize the team reward. Concurrently, the skill learner Π_k is trained using a gradient-based optimizer to maximize r_k by sampling a random mini-batch from the shared replay buffer. The shared replay buffer allows for increased information extraction from each agent, facilitating exploration maximization and sample efficiency. The concurrent learning of low-level skills and agent policies to optimize team objectives allows agents to learn skills from experiences driven towards optimizing the global reward.

We test MADyS on a simulated robot exploration domain [14, 15]. The environment consists of homogeneous agents and several types of Points of Interests (POIs), denoted as $A, B, \ldots$. The task is to observe each POI type by a team of i agents- characterized by spatial coupling, in a specific order of POI types ($A \rightarrow B$)- characterized by temporal coupling. The team gets a reward of 1 when it fulfills the temporal and spatial coupling and 0 otherwise.

3 RESULTS

We compare the performance of MADyS with a standard evolutionary algorithm (EA) [4] operating directly on the individual agent actions, as well as with Multi-fitness Learning (MFL) [14], as shown in Fig. 2. In MFL, EA searches over actions generated by agents that have been pre-trained on local skills only, without access to the team objective - thus, EA simply learns to pick an optimal pre-trained skill. While the original MFL paper adopted EA to learn local skills separately, we allow our MFL agents to be pre-trained using PG and a shared replay buffer. These modifications to MFL agents equalize the skill learning modules in MFL and MADyS and ensure that any sample efficiency gains we observe come purely from the joint optimization of local and global objectives in MADyS and not from implementation differences of the common components. We refer to this modified implementation as MFL. Both MFL and MADyS utilize EA to select skills rather than low-level actions. However, in MADyS, local skills are learned concurrently with the global optimization, making the overall process more sample efficient. We use TD3 [8] as the PG method to optimize local rewards for both MADyS and pre-training skills for our baseline *MFL*. We conduct experiments over 5 statistically independent runs with random seeds from 2000, 2004 and report the average performance with error bars showing 95% confidence interval. All scores reported are compared against the number of environment steps.

4 CONCLUSION

In this paper, we introduced MADyS - a framework that allows a team of agents to coordinate and solve complex tasks involving spatial and temporal coupling. MADyS targets a class of multiagent coordination problems where agents need to learn to decompose a long-horizon task into several sub-tasks, each of which requires different sub-strategies. MADyS solves this by allowing multiagent teams to dynamically select from local policies trained on different dense local objectives to optimize a sparse global objective. It outperforms all baselines tested on a set of complex coordination problems with several spatial and temporal coupling requirements.

ACKNOWLEDGMENTS

This work was partially supported by NSF (IIS-1815886), AFOSR (FA9550-19-1-0195), and Intel.

REFERENCES

[1] Adrian K Agogino and Kagan Tumer. 2008. Analyzing and visualizing multiagent rewards in dynamic and stochastic domains. *Autonomous Agents and Multi-Agent Systems* 17, 2 (2008), 320–338.

[2] Jen Chung, Damjan Miklić, Lorenzo Sabattini, Kagan Tumer, and Roland Siegwart. 2019. The impact of agent definitions and interactions on multiagent learning for coordination. In *AAMAS'19 Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*. International Foundation for Autonomous Agents and Multiagent Systems, 1752–1760.

[3] Jen Jen Chung, Carrie Rebhuhn, Connor Yates, Geoffrey A Hollinger, and Kagan Tumer. 2019. A multiagent framework for learning dynamic traffic management strategies. *Autonomous Robots* 43, 6 (2019), 1375–1391.

[4] Carlos A Coello, Gary B Lamont, David A Van Veldhuizen, et al. 2007. *Evolutionary algorithms for solving multi-objective problems*. Vol. 5. Springer.

[5] Mitchell Colby, Theodore Duchow-Pressley, Jen Jen Chung, and Kagan Tumer. 2016. Local approximation of difference evaluation functions. In *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems*. International Foundation for Autonomous Agents and Multiagent Systems, 521–529.

[6] Kurt Dresner and Peter Stone. 2005. Multiagent traffic management: An improved intersection control mechanism. In *Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems*. 471–477.

[7] Yali Du, Lei Han, Meng Fang, Ji Liu, Tianhong Dai, and Dacheng Tao. 2019. LIIR: Learning Individual Intrinsic Reward in Multi-Agent Reinforcement Learning. In *Advances in Neural Information Processing Systems*. 4403–4414.

[8] Scott Fujimoto, Herke van Hoof, and David Meger. 2018. Addressing function approximation error in actor-critic methods. *arXiv preprint arXiv:1802.09477* (2018).

[9] Shariq Iqbal and Fei Sha. 2019. Coordinated Exploration via Intrinsic Rewards for Multi-Agent Reinforcement Learning. *arXiv preprint arXiv:1905.12127* (2019).

[10] Shauharda Khadka, Somdeb Majumdar, and Kagan Tumer. 2019. Evolutionary Reinforcement Learning for Sample-Efficient Multiagent Coordination. *arXiv preprint arXiv:1906.07315* (2019).

[11] Marin Lujak, Alberto Fernández, and Eva Onaindia. 2020. A Decentralized Multi-Agent Coordination Method for Dynamic and Constrained Production Planning. In *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems*. 1913–1915.

[12] Aida Rahmattalabi, Jen Jen Chung, Mitchell Colby, and Kagan Tumer. 2016. D++: Structural credit assignment in tightly coupled multiagent domains. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 4424–4429.

[13] Kagan Tumer and Matt Knudson. 2010. Aligning agent objectives for learning and coordination in multiagent systems. *PerAdA Magazine* (2010).

[14] C. Yates, R. Christopher, and K. Tumer. 2020. Multi-Fitness Learning for Behavior-Driven Cooperation. In *Proceedings of Genetic and Evolutionary Computation Conference (GECCO)*. Cancun, Mexico.

[15] Logan Yliniemi and Kagan Tumer. 2014. Multi-objective multiagent credit assignment through difference rewards in reinforcement learning. In *Asia-Pacific Conference on Simulated Evolution and Learning*. Springer, 407–418.