Distributed Policy Iteration for Scalable Approximation of Cooperative Multi-Agent Policies

Extended Abstract

Thomy Phan
LMU Munich
thomy.phan@ifi.lmu.de

Kyrill Schmid
LMU Munich
kyrill.schmid@ifi.lmu.de

Lenz Belzner
MaibornWolff
lenz.belzner@maibornwolff.de

Thomas Gabor
LMU Munich
thomas.gabor@ifi.lmu.de

Sebastian Feld
LMU Munich
sebastian.feld@ifi.lmu.de

Claudia Linnhoff-Popien
LMU Munich
linnhoff@ifi.lmu.de

ABSTRACT
We propose Strong Emergent Policy (STEP) approximation, a scalable approach to learn strong decentralized policies for cooperative MAS with a distributed variant of policy iteration. For that, we use function approximation to learn from action recommendations of a decentralized multi-agent planning algorithm. STEP combines decentralized multi-agent planning with centralized learning, only requiring a generative model for distributed black box optimization. We experimentally evaluate STEP in two challenging and stochastic domains with large state and joint action spaces and show that STEP is able to learn stronger policies than standard multi-agent reinforcement learning algorithms, when combining multi-agent open-loop planning with centralized function approximation. The learned policies can be reintegrated into the multi-agent planning process to further improve performance.

KEYWORDS
multi-agent planning; multi-agent learning; policy iteration

ACM Reference Format:

1 INTRODUCTION
Cooperative multi-agent systems (MAS) are popular in artificial intelligence research and have many potential real-world applications like autonomous vehicles, sensor networks, and robot teams [4–6]. However, decision making in MAS is extremely challenging due to intractable state and joint action spaces as well as stochastic dynamics and uncertainty w.r.t. other agents’ behavior.

Centralized control does not scale well in large MAS due to the curse of dimensionality, where state and joint action spaces grow exponentially with the number of agents [1, 3–7]. Therefore, decentralized control is recommended, where each agent decides its individual actions under consideration of other agents, providing better scalability and robustness [4–7]. Decentralized approaches to decision making in MAS typically require a coordination mechanism to solve joint tasks and to avoid conflicts [3].

Recent approaches to learn strong policies are based on policy iteration and combine planning with deep reinforcement learning, where a neural network is used to imitate the action recommendations of a tree search algorithm. In return, the neural network provides an action selection prior for the tree search [2, 13]. This iterative procedure, called Expert Iteration (ExIt), gradually improves both the performance of the tree search and the neural network [2]. ExIt has been successfully applied to zero-sum games, where a single agent improves itself by self-play. However, ExIt cannot be directly applied to large cooperative MAS, since using a centralized tree search is practically infeasible for such problems [4, 5].

In this work, we propose Strong Emergent Policy (STEP) approximation, a scalable approach to learn strong decentralized policies for cooperative MAS with a distributed variant of policy iteration. For that, we use function approximation to learn from action recommendations of a decentralized multi-agent planner. STEP combines decentralized multi-agent planning with centralized learning, where each agent is able to explicitly reason about emergent dependencies to make coordinated decisions. Our approach only requires a generative model for distributed black box optimization.

2 METHOD
Given a Multi-agent Markov Decision Process (MMDP) \( M = (\mathcal{D}, \mathcal{S}, \mathcal{A}, P, R) \) [3] with a (finite) set of agents \( \mathcal{D} = \{1, \ldots, N\} \), a (finite) set of states \( \mathcal{S} \), a (finite) set of joint actions \( \mathcal{A} = \mathcal{A}_1 \times \cdots \times \mathcal{A}_N \), a transition probability function \( P : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \leftarrow [0,1] \), and a global reward function \( R : \mathcal{S} \times \mathcal{A} \leftarrow \mathbb{R} \), we extend the ExIt framework of [2, 13] to cooperative MAS to approximate a strong joint policy \( \hat{\pi}(s_t) = (\hat{\pi}_1(s_t), \ldots, \hat{\pi}_N(s_t)) \in \mathcal{A} \) for each state \( s_t \in \mathcal{S} \). For that, we use function approximation to learn from action recommendations of a decentralized multi-agent planner to approximate strong decentralized policies \( \hat{\pi}_i \) for each agent \( i \in \mathcal{D} \), which are combined into a strong joint policy \( \hat{\pi} \) for the MAS. The training procedure of STEP consists of a planning and a learning step.

In the planning step, a decentralized planning algorithm is executed for each state \( s_t \in \mathcal{S} \) and executed to observe a new state \( s_{t+1} \in \mathcal{S} \) and a global reward \( r_t = R(s_t, a_t) \).
In the learning step, a parametrized function approximator \( f_\theta = (\hat{\pi}, \hat{V}) \) is used to approximate an optimal joint policy \( \pi^* \) by approximating optimal decentralized policies \( \pi_i^* \) for each agent \( i \in \mathcal{D} \) and the optimal value function \( V^* \). \( \hat{\pi}_i \) is approximated by minimizing the cross-entropy loss between \( p(a_t|i,s_t) \) and \( \hat{\pi}_i(a_t|i,s_t) \), while \( \hat{V} \) is approximated via temporal difference learning [14, 15].

\( f_\theta \) can be reintegrated into the planning step to further improve performance by providing an action selection prior \( \hat{\pi} \) similarly to [2, 13], a coordination mechanism to predict other agents’ behavior via \( \hat{\pi} \) [4], and a leaf state evaluator \( V \) to compensate for the limited search depth of the decentralized multi-agent planner [11]. The architecture of STEP is shown in Fig. 1.

### 3 RESULTS

We tested STEP in the Pursuit & Evasion domain (Fig. 2a and [16, 17]) with 2 agents and in the Smart Factory domain (Fig. 2b-c and [11]) with 4 agents. In the training phase, we applied STEP to both decentralized open-loop (DOLUCT) and closed-loop (DMCTS) planning, and compared the progress with different instances of DOLUCT using a random joint policy \( \hat{\pi} \) or a baseline value function of \( V(s_t) = 0 \) as well as a centralized open-loop version of DICE [9, 11]. In the test phase, we extracted the decentralized policies \( \hat{\pi}_i \) approximated with STEP after every tenth training episode and compared them with Distributed Q-Learning (DQL) [16] and Distributed Actor-Critic (DAC) [6]. We implemented two variants of each DQL and DAC, where one variant was trained on the global reward \( R \) and the other one was trained on a decomposed local reward similarly to [7].

The results are shown in Fig. 3. Fig. 3a and 3c indicate that open-loop planning algorithms like DOLUCT are especially suited for STEP, when the domains are too complex to provide sufficient computation budget as already noted for single-agent problems [8, 10, 12, 18]. The approximated policies \( \hat{\pi}_i \) of STEP with DOLUCT are able to clearly outperform standard multi-agent reinforcement learning algorithms like DQL and DAC in both domains. Providing a larger computation budget \( n_k \) seems to be beneficial when approximating strong decentralized policies with STEP as shown in Fig. 3b and 3d. The learned policies can be reintegrated into the planning process to further improve performance of the multi-agent planner as shown in Fig. 3a and 3c for DOLUCT and DMCTS.

### 4 CONCLUSION

We proposed STEP, a scalable approach to learn strong decentralized policies for cooperative MAS with a distributed variant of policy iteration by combining decentralized multi-agent planning with centralized learning, where each agent is able to explicitly reason about emergent dependencies to make coordinated decisions, only requiring a generative model for distributed black box optimization. Our results show that STEP is able to produce stronger policies than standard multi-agent reinforcement algorithms, which can be reintegrated into the planning process to further improve performance. For the future, we plan to address partially observable domains by combining multi-agent planning with deep recurrent reinforcement learning for cooperative MAS.
REFERENCES


