ASSESSING COOPERATION IN MULTI AGENT REINFORCEMENT LEARNING W.R.T COORDINATION AND ENVIRONMENTAL HETEROGENEITY

Anonymous authors
Paper under double-blind review

ABSTRACT

In cooperative multi-agent reinforcement learning, a team of many agents works together to achieve a common goal. Different environments or tasks may require varying degrees of coordination among agents in order to achieve the goal in an optimal way. Further, from a practical perspective, an environment may have spatially varying properties, such as varying dynamics, distribution of obstacles, etc. which need to be taken into account by the agents, in order to solve the task efficiently. We term this spatial variation of properties as "environmental heterogeneity" (also simply referred to as "heterogeneity"). Different environments may have different levels of heterogeneity, a property which has not been addressed sufficiently in the existing literature. We provide formal definitions of "coordination level" and "heterogeneity level" of an environment and present **HECO-Grid** - a suite of multi-agent RL environments, that facilitates empirical evaluation of different MARL approaches across different levels of coordination and environmental heterogeneity by providing a quantitative control over coordination and heterogeneity levels of the environment. Further, we propose a coordinated communication method called **HECO-PPO** that enables agents to work efficiently in high-coordination and high-heterogeneity environments through a differentiable, shared, and bottlenecked communication channel and dynamic selection of policies from a shared pool of policies. We evaluate HECO-PPO and compare it's performance against baselines IPPO and MAPPO on HECO-Grid. Our results show that HECO-PPO consistently outperforms the baselines across different tasks and different heterogeneity and coordination levels.

1 INTRODUCTION

Multi-Agent Reinforcement Learning (MARL) studies the problem of sequential decision-making in an environment with multiple actors. Unlike most single-agent RL settings, MARL is faced with the unique problem of changing environment dynamics which makes it difficult to learn optimal behavior policies. In many settings, MARL manifests itself in the form of cooperative tasks which require efficient coordination among the individual actors in order to achieve a common goal. This entails the ability of agents to communicate with each other. To facilitate communication and overcome environment non-stationarity, the Centralized Training Decentralized Execution paradigm (CTDE) (Yu et al., 2021a; Lowe et al., 2017b; Rashid et al., 2018a; Wang et al., 2020; Das et al., 2019) was introduced which consists of allowing the agents to access each other's states during training only whereas, during testing, each agent takes actions conditioned on its own state. Communication becomes especially necessary when the agents only have access to a partial view of the environment. Another challenge in practical multi-agent learning problems is heterogeneity in the environment. In this paper, we study spatial heterogeneity, where the environment can be different with respect to dynamics, structure (e.g. different distribution of obstacles), etc. in different regions. Existing MARL literature studies heterogeneity in the context of heterogeneous agents, i.e. agents having different abilities Samvelyan et al. (2019), Berner et al. (2019). However, heterogeneity in environments has not been studied widely. Monolithic policies aren't best suited to tackle these problems where the properties of the environment can vary significantly from one region to other. We formally define two properties of an environment: **heterogeneity**, which is the level of heterogeneity of the environment

and **coordination** which is the level of coordination required amongst agents to solve the task at hand. We present formal definitions of **heterogeneity** and **coordination** in Section 3.2. The difficulties of environments can vary based on the amount of heterogeneity and the level of coordination required to solve it.

In order to investigate the effects of coordination and environmental heterogeneity on multi-agent reinforcement learning, we need to systematically analyze the performance of different approaches on varying levels of these two factors. Recently, several benchmarks have been proposed to investigate the emergent coordination and communication abilities of MARL approaches, however, there exist few suits of environments that allow us to do so in a controlled fashion and no suit which allows systematically varying the heterogeneity of the environment. A quantitative control over required coordination and heterogeneity levels of the environment can also facilitate testing the generalization and transfer properties of MARL approaches across different levels of coordination and heterogeneity.

Some of the existing benchmarks based on online multiplayer games, attempt to move away from the toy-like grid world setting for multi-agent RL environments in favor of more realistic environments, by making use of high dimensional observation and action spaces, continuous action space, challenging dynamics, and partial observability Vinyals et al. (2017), Berner et al. (2019), **?**. These benchmarks focus on decentralized control in cooperative tasks. and agents are heterogeneous (i.e., different types of agents having different abilities). In principle, it is possible to vary the levels of coordination and heterogeneity since the difficulties of different environments vary. However, there is so well-defined notion of the two concepts and these can't be varied in a controlled fashion. MeltingPot, which was recently proposed in Leibo et al. (2021) focuses test time generalization abilities of a group of agents (called the *focal population*) to different tasks. It uses the idea of using passive agents (called the *background population*) to create different tasks for the *focal population*. Different tasks are created by varying the size of and distribution of policies of the *background population*. MeltingPot includes a wide range of scenarios: competitive games, games of pure common interest, team based competitive games, mixed motion games which stress test the coordination abilities of the agents. However, similar to other benchmarks, there is no systematic decomposition and quantitative notion of the concepts of *coordination* and environmental heterogeneity.

To address these limitations, we propose HECOGrid - a novel procedurally generated Multi Agent Benchmark built upon MARLGrid Ndousse et al. (2021) consisting of four different environments, which enables testing algorithms across different coordination and environmental heterogeneity with quantitative notions of coordination and heterogeneity. Each environment consists of N agents and M treasures, where the goal is to maximize the total number of treasures picked up by the agents in one episode. c agents are required to pick up a single treasure, where c is the coordination level. The environment is spatially divided in to h zones, and the environment dynamics vary from zone to zone. h is the level of heterogeneity of the environment. c, h, N and M are controllable parameters using which which an environment is defined. This also allows us to test generalization across different coordination and heterogeneity levels. Table 1 presents a qualitative comparision between HECOGrid and other commonly used Multi Agent RL environment suites.

To enable efficient training of MARL agents when levels of required coordination and environmental heterogeneity are high, we propose a novel algorithm call HECO-PPO which has coordinated bottlenecked inter-agent communication and pool a policies agents can dynamically choose from. Communication has been shown to be essential component of multi agent systems attempting to solve tasks requiring coordination among agents. Hence, multi-agent tasks requiring very little coordination among the agents can be easily tackled by approaches which do not facilitate communication among the agents. For eg. independent learning approaches such as IPPO Witt et al. (2020b), IQL Tan (1997), etc. However, these often fail in scenarios where high level of coordination is required. Several algorithms propose different communication strategies which help tackle this problem. Recent approaches explore end-to-end training procedure based on a differentiable communication channel Das et al. (2019), Sukhbaatar et al. (2016), Gupta et al. (2017), Singh et al. (2018). We propose a novel communication scheme wherein agents communicate by reading from and writing to a **Stateful Active Facilitator (SAF)** which acts as a latent bottleneck for aggregating from and disseminating information to agents. We describe an approach in which the communication channel is *active* in that it can interpret and transform signals from one or more agents, and it is *stateful* in that its interpretations can depend on the recent history of messages transmitted. Because we endow the channel itself with intelligence, message communication complexity is reduced from quadratic to linear in N. In essence, the channel is a specialist agent aiming to *facilitate* coordination among the

other agents. Our approach also has the ability to generalize w.r.t. to the number of agents present in the environment, unlike most of the existing approaches, since it uses a *pool* of policies which can be shared among arbitrary number of agents.

On the other hand research in modular deep learning Goyal et al. (2021a), Goyal et al. (2021c), Goyal et al. (2021b), Rahaman et al. (2021) has shown that different neural modules trained on a common objective lead to the emergence of specialist neural modules which help in improving performance via decomposition of the task. We hypothesize that a similar form of modularity can also be helpful in tackling the problem of heterogeneity. Instead of each agent using individual monolithic policies, we propose the use of a *pool* of policies which is shared across different agents. At each time-step, each agent selects one policy from the pool which is used to determine the next action. This selection is conditioned on the current state of the agent and the message received by the agent from the SAF during training and only on the current state of the agent during test time. Hence, our approach falls in the class of Centralized Training, Decentralized Execution learning algorithms, since we do away with the SAF during test time.

Benchmark	**Cooperative**	**Partial Obs.**	**Image Obs.**	**Coordination Control**	**Heterogeneity Control**
SMAC	✓	✓	×	×	×
MeltingPot	✓	✓	✓	✓	×
MPE	✓	×	×	×	×
SISL	✓	×	×	×	×
DOTA 2	✓	✓	×	×	×
HECOGrid	✓	✓	✓	✓	✓

Table 1: Comparision between HECOGrid, SMAC Vinyals et al. (2017), MeltingPot Leibo et al. (2021), MPE Lowe et al. (2017a), SISL Gupta et al. (2017) and DOTA2 Berner et al. (2019)

We show that our approach consistently outperforms the eshtablished baselines in MARL on environments across different coordination and heterogeneity levels. We use MAPPO and IPPO as the baselines. Further ablation studies show that the information bottleneck (SAF) is the key to good performance across different coordination levels, whereas the the pool of policies is the key to good performance across different levels of environment heterogeneity.

1.1 Our contribution

Our contribution in this study include

- Formally define level of coordination of cooperative tasks and standardize the definition of environmental heterogeneity in full cooperative multi-agent reinforcement learning settings.
- Construct a set of benchmark cooperative MARL environments with adjustable coordination and heterogeneity levels
- Propose coordinated inter-agent communication to enable agent have good performance in cooperative games with high level of coordination
- Propose a pool of policy the agents can dynamically choose from to enable agent have good performance in cooperative games with high level of environmental heterogeneity

2 Related Work

Centralized Learning Decentralized Execution (CLDE). These approaches are among the most commonly adopted variations for MARL in cooperative tasks. They usually involve a centralized critic which takes in global information, i.e. information from multiple agents, and decentralized policies which are guided by the critic. Foerster et al. (2018) use the standard centralized critic decentralized actors framework with a *counterfactual baseline*. Lowe et al. (2017a) and Yu et al. (2021b) propose the extension of single-agent DDPG and single-agent PPO respectively, to a multi-agent framework with a centralized critic and decentralized policies during training and completely decentralized execution. Li et al. (2021) uses an information theory-based objective to promote

novel behaviors in CLDE-based approaches. Value Decomposition (Sunehag et al. (2018), Rashid et al. (2018b), Wang et al. (2020), Mahajan et al. (2019)) approaches learn a factorized action-value function. Sunehag et al. (2018) proposes Value Decomposition Networks (VDN) which simply add the action-value function of each agent to get the final action value function. Rashid et al. (2018b) uses a mixing network to combine the action-value functions of each agent in a non-linear fashion.

Communication in MARL. Several approaches use emergent communication protocols for MARL. Communication involves deciding the message to be shared and determining how the message sending process is implemented. Foerster et al. (2016) and Sukhbaatar et al. (2015) have done work on learnable inter-agent communication protocols. Jiang & Lu (2018) first proposed using attention for communication where attention is used for integrating the received information as well as determining when communication is needed. Das et al. (2019) uses multiple rounds of direct pairwise inter-agent communication in addition to the centralized critic where the messages sent by each agent are formed by encoding it's partial observation, location information, etc., and the messages received by each agent are integrated into it's current state by using a soft-attention mechanism. Kim et al. (2020) uses intentions represented as encoded imagined trajectories as messages where the encoding is done via a soft-attention mechanism with the messages received by the agent. Wang et al. (2021) trains a model for each agent to infer the intentions of other agents in a supervised manner, where the communicated message denotes the intentions of each agent. The above mentioned approaches require a computational complexity which is quadratic in the number agents. Our approach has a computational complexity which is linear in the number of agents. Moreover, we show that our approach is able to outperform several standard baselines using messages which can be computed as simply the encoded partial observations of each agent. (Weis et al., 2020) Developed a transformer based multi-agent reinforcement learning method that models decision as a sequential model.

Option-critic in Multi-agent Hierarchical Reinforcement learning A classical approach within the Hierarchical Reinforcement Learning (HRL) Pateria et al. (2021) literature is modeling agents' intentions as options Sutton et al. (1999), temporally extended actions that aim to achieve subgoals in a finite time horizon. Recently, Bacon et al. (2017) proposed an end-to-end option-critic architecture capable of learning both options and the related policy. However, despite the advantages brought by using options, due to the temporally-extended nature of options, agents' responses can be inconsistent when the environment or other agents' behaviour change. To tackle this problem, Han et al. (2019) proposed a dynamical termination scheme which allows an agent to flexibly terminate its current option. Although both option-critic and our approach use a pool of actors, while in the former case actors model options, in the latter one actors model policies, preventing agents' inconsistent behaviours. Moreover, although within the option-critic framework the optimality of the learned hierarchical policy can be theoretically guaranteed Chakravorty et al. (2020), the learned options cannot be easily transferred to other tasks Pateria et al. (2021). Furthermore, many works proposed within the option-critic framework (Klissarov et al. (2017),Riemer et al. (2018),Khetarpal et al. (2020)) perform poorly on sparse reward tasks (Bagaria & Konidaris (2020),Nachum et al. (2018)), while experimental results show that our approach presents comparable performances with SOTA baselines even on that case (e.g., Waterworld Environment Gupta et al. (2017))

Independent learning. Independent Learning in MARL consists of each agent optimizing its policy locally using its observation and in the absence of any communication or centralized controller (as in CTDE). These approaches mainly consist of the extension of single-agent RL approaches to multi-agent settings where each agent learns independently using local observations, considering other agents as part of the environment. Tan (1993) proposed Independent Q-Learning (IQL) where each agent independently learns it's own action-value function. Witt et al. (2020a) demonstrates that PPO, when used for independent learning in multi-agent settings (called Independent PPO or IPPO) is in fact capable of beating several state of the art approaches in MARL on competitive benchmarks such as StarCraft and can hence serve as a strong baseline.

3 Problem settings,Notations

In this section, we describe the system used for cooperative multi-agent reinforcement learning in this study and its notations, and give a formal definition of coordination and heterogeneity levels in a cooperative MARL task

3.1 NOTATIONS

In this work, we consider a multi-agent version of Markov decision processes (MDPs) with partially observable Markov environments (POMDP). The environment is defined as $(\mathcal{N}, \mathcal{S}, \mathcal{O}, \mathcal{A}, \Pi, R, \gamma)$. $\mathcal{N} = \{1, ..., N\}$ denotes the set of $N > 1$ agents. $\mathcal{S}$ describes all possible states of all agents. $\mathcal{A} = A_1 \times \cdots \times A_N$ denotes the joint action space and $a_{i,t} \in A_i$ refers to the action of agent i at time step t. $\mathcal{O} = O_1 \times \cdots \times O_N$ denotes the set of partial observation where $o_{i,t} \in O_i$ stands for partial observation of agent i at time step t. Π is the set of policies available to the agents. To choose actions, agent i uses a stochastic policy $\pi_{\theta_i} : O_i \times A_i \mapsto [0, 1]$. Actions from all agents together produces the transition to the next state according to transition function $\mathcal{T} : (s_t, a_{1,t}, \ldots, a_{N,t}) \mapsto s_{t+1}$ where $s_t \in \mathcal{S}$ is the joint state at timestep t. $R : \mathcal{S} \times \mathcal{A} \mapsto \mathbb{R}$ is the global reward function conditioned on the joint state and actions. At timestep t, the agent team receives a reward $r_t = R(s_t, a_{1,t}, \ldots, a_{N,t})$ based on the current total state s_t and joint action $a_{1,t}, \ldots, a_{N,t}$. γ is the discount factor for future rewards. In this study, we focus on cooperative MARL with partial observations.

3.2 DEFINITION: COORDINATION LEVEL OF N COOPERATIVE MULTI-AGENT REINFORCEMENT LEARNING TASK

In cooperative MARL environment where all agents share the same reward, some tasks require more coordination among agents while others require less.

Here, we formally define the coordination level required in a cooperative MARL task. Recall that $R : \mathcal{S} \times \mathcal{A} \mapsto \mathbb{R}$ is the global reward function conditioned on the joint state and actions. At time step t, the agent team receives a reward $r_t = R(s_t, a_t)$ based on the current total state of all agents s_t and joint action a_t. If we assume we can decompose $s_t = U(\{s_{it}\})$ where $i \in [1..N]$ is agent index and $\{s_{it}\}$ is the collection of states of each agent.

We define the level of coordination needed for a RL task as how many agents are needed to work together to achieve team reward at time step t in the cooperative environment. Assuming team reward r_t can be linearly decomposed into $r_t = \gamma_1 \Psi^1(\{s_{it}, a_{it}\}) + \gamma_2 \Psi^2(\{s_{it}, a_{it}\}) + \gamma_3 \Psi^3(\{s_{it}, a_{it}\}) +$ ..... where Ψ^1 is the RL environment/task-specific function to calculate contribution of states and actions of single agents without any cooperation with other agents to the time step team reward , Ψ^2 is the RL environment/task-specific function to calculate contribution of cooperation between states and actions of a pair of agents to the time step reward.Ψ^k is the RL environment/task-specific function to calculate contribution of interactions between states and actions of k different agents to the time step reward. Members of $\{\Psi^1, \Psi^2, \Psi^3, \Psi^4....\}$ are mutually exclusive from each other and their sum is the team reward at time step t. $\gamma_1, \gamma_2...$ are the coefficient of each of the contribution term, In this study, we define the last γ which is not zero, as the level of coordination required by the MARL cooperative task.

3.3 DEFINITION:HETEROGENEITY LEVEL OF AN COOPERATIVE MULTI-AGENT REINFORCEMENT LEARNING TASK

Another aspects we like to explore is the heterogeneity of the RL environment.It is worth pointing out that heterogeneity of the RL environment is different from heterogeneity of agents or heterogeneity of policies as explored in previous studies.

Here we formally define heterogeneity of the environment. For simplification, we define this in a single agent RL environment, which can be easily unwrapped into multi-agent setting. The environment has state transition function $\mathcal{T} : (s_t, a_{1,t}) \mapsto s_{t+1}$ where $s_t \in \mathcal{S}$ is the state at time step t. Assuming s_t is independent from the position ν_t of the agent in the environment, if $\mathcal{T} : (s_t, a_{1,t}) \mapsto s_{t+1}$ holds identical for $\forall \nu$, then we define the environment as homogeneous (or not heterogeneous). On the other hand, if there are K different groups of positions $\{_k\}_k^K$ in which $\mathcal{T}_k$ are identical within the group and different among different groups, we define the environment as heterogeneous and K as the level of heterogeneity of the environment.

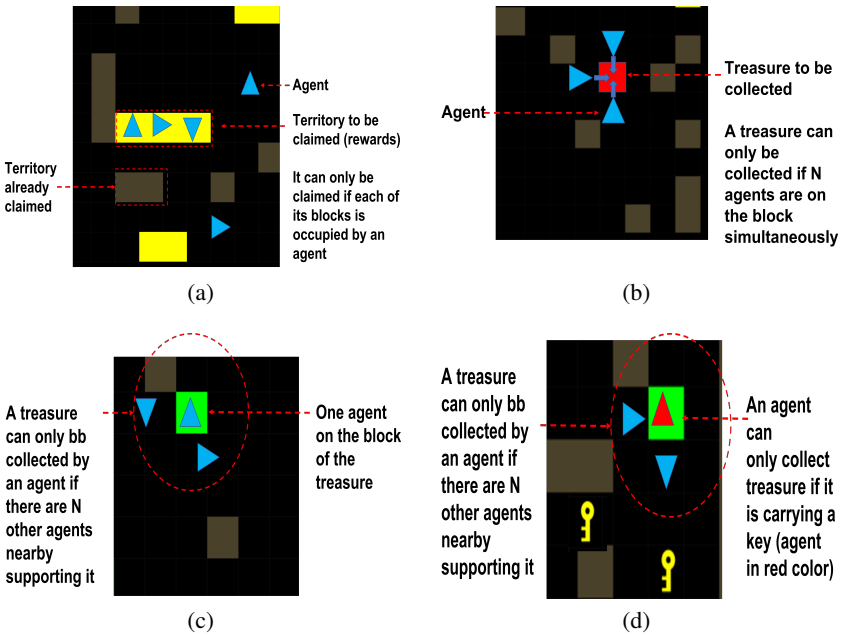


Figure 1: HECO-grids: We constructed four cooperative multi-agent reinforcement learning environments that allows control of coordination and environmental heterogeneity levels (a)"CompoundGoal" environment (b)"TeamTogether" environment (c) "TeamSupport" environment (d) "KeyForTreasures" environment

4 HECOGRID: MARL ENVIRONMENTS FOR DIFFERENT COORDINATION AND ENVIRONMENTAL HETEROGENEITY LEVELS

We developed a set of four cooperative MARL environments, which we collectively call "HECOGrid", that allows manipulation of coordination levels ,environmental heterogeneity levels, number of cooperative agents and several other properties to fill the gap in the community. The HECOGrid benchmarks include "CompoundGoal", "TeamSupport", "TeamTogether" and "KeyForTreasure" environments, which are described in details in this section. All four HECOGrid environments are derived and modified from MARLGrid Ndousse et al. (2021) which is developed based on Minigrid Chevalier-Boisvert & Willems (2018) environments. In this section, we describe each HECOGrid environment in detail.

4.1 "COMPOUNDGOAL" ENVIRONMENT

The cooperative team task in this environment is to "claim ownership" of as many territories as possible in a limited number of time steps. Each territory occupies certain number of squares and require one agent in each square in the territory simultaneously for the territory to be claimed. The rewards are distributed equally across all agents in the whole team. The level of coordination is defined as the maximum number of squares the a territory can occupy in the environment.For coordination level of 1, we design the environment so that territories occupy only one square and only require one agent to claim it, therefore $\gamma_1 > 0$ and $\gamma_{k>1} = 0$. For coordination level of 2, we design the environment so that territories occupy up to 2 squares which need to be collected by two agents working together, therefore $\gamma_2 > 0$ and$\gamma_{k>2} = 0$. For coordination level of κ, we design the environment so that environment so that territories occupy up to κ squares, therefore $\gamma_{Kappa} > 0$ and $\gamma_{k>\kappa} = 0$.

4.2 "TEAMSUPPORT" ENVIRONMENT

The cooperative team task in this environment is to collect as many treasures as possible in a limited number of time steps. Each treasure is presented as a bright box in the environment and became grey once collected. In order for an agent to collect a treasure, the agent need to step on to the box and with certain number of agents within a distance (set as 2 by default) to support to the collection. This number of agent required for collection support (including the agent that actually collects) is defined as level of coordination of the task.rewards are distributed equally across all agents in the whole team.

4.3 "TEAMTOGETHER" ENVIRONMENT

This environment is similar as "TeamSupport" except that agents can overlap with each other in the environment and,to collect the treasure, certain number of agents to be on the box simultaneously. As observed in our experiment in next sections, this is a more difficult task in sense of coordination compared with other HECO-grids environments.

4.4 "KEYFORTREASURE" ENVIRONMENT

This environment is similar as "TeamSupport" except that an agent can only collect a treasure if it is carrying a key and,to collect the treasure, certain number of agents to be on the box simultaneously. This additional step key searching increases the difficulty of the task. If an agent pickups a key, its color changes.

4.5 ENVIRONMENTAL HETEROGENEITY IN HECO-GRIDS

For all the four environments mentioned above, we implement the manipulation of environmental heterogeneity levels. In default setting each agent has an one-hot action space of size 5. In the default setting, action 1 means "turn right", action 2 means "turn left", action 3 means "move forward", action 4 means "pickup" and action 5 means "toggle"

For homogeneous environment (heterogeneity level = 1) any feasible actions leads to the same movement of the agents anywhere in the environment using the defaut setting.

When the heterogeneity level is $K > 1$, the environment is divided into K different rectangular zones of equal sizes. In each of the K different zones,the action of the same index lead to different movements of the agent in the environment, which corresponds to state transition function $\mathcal{T}_k$ Jin et al. (2022). In addition, we give different color to agents in different zones so the agent knows which zones it is in.

5 HECO-PPO

In this section, we describe our HECO-PPO method that consists of three component:a Stateful Active Facilitator (SAF) that facilitate inter-agent communication, a policy pool that allows agents to dynamically choose from, and KL divergence based regularization to maximize autonomy of agents. In the following subsections, we are going to explain each component in details.

5.1 INTER-AGENT COMMUNICATION THROUGH STATEFUL ACTIVE FACILITATOR

5.1.1 GENERATING THE MESSAGES

Each agent i receives a partial observation $o_{i,t}$ at each time step t. These observations are encoded into messages to be used for communication by a common encoder g_θ: $m'_{i,t} = g_\theta(o_{i,t})$; $m'_{i,t} \in \mathbb{R}^{d_m}$. We denote the set of messages generated by the agents at time step t by $\boldsymbol{M}_t$:

$$\boldsymbol{M}'_t = \{m'_{i,t} | 1 \leq i \leq N\}$$

5.1.2 WRITING INTO THE STATEFUL ACTIVE FACILITATOR

The messages $\boldsymbol{M}_t'$ generated in step one are distilled into a latent state which we term as a **Stateful Active Facilitator** or **SAF**. We represent the SAF's state at a time step t by $\boldsymbol{F}_t$. $\boldsymbol{F}_t$ consists of L slots $\{l_0, l_1, ..l_{L-1}\}$, each of dimension d_l.

$$\boldsymbol{F}_t \in \mathbb{R}^{L \times d_l}$$

The messages in $\boldsymbol{M}_t'$ compete with each other to write into each SAF's state slot via a cross-attention mechanism. The query in this case is a linear projection of the $\boldsymbol{F}_t$, i.e., $\widetilde{\boldsymbol{Q}} = \boldsymbol{F}_t \widetilde{\boldsymbol{W}}^q$, whereas the keys and values are linear projections of the messages $\boldsymbol{M}_t'$. SAF's state is updated as:

$$\boldsymbol{F}_t \leftarrow \text{softmax}\left(\frac{\widetilde{\boldsymbol{Q}}(\boldsymbol{M}_t \widetilde{\boldsymbol{W}}^e)^{\mathrm{T}}}{\sqrt{d_e}}\right) \boldsymbol{M}_t \widetilde{\boldsymbol{W}}^v$$

After this, self-attention is applied to the SAF using a transformer encoder tower constituting a perceiver encoder structure Jaegle et al. (2022).

5.1.3 READING FROM THE STATEFUL ACTIVE FACILITATOR

SAF makes the updated state available to the agents should they deem to use it. We again utilize cross attention to perform the reading operation. All the agents create queries $\boldsymbol{Q}_t^s = \{q_{i,t}^s | 1 \leq i \leq N\} \in \mathbb{R}^{N \times d_e}$ where $q_{i,t}^s = \boldsymbol{W}_{\text{read}}^q s_{i,t}$ and $s_{i,t}$ are encoded partial observations. Generated queries are matched with the keys $\boldsymbol{\kappa} = \boldsymbol{F}_t \boldsymbol{W}^e \in \mathbb{R}^{l \times d_e}$ from the updated state of SAF. As a result, the attention mechanism can be written as:

$$\boldsymbol{M}_t = \text{softmax}\left(\frac{\boldsymbol{Q}_t^s \boldsymbol{\kappa}^T}{\sqrt{d_e}}\right) \boldsymbol{F}_t \boldsymbol{W}^v \tag{1}$$

Here $\boldsymbol{M}_t = \{m_{i,t} | 1 \leq i \leq N\}$. Consequently, the read messages are used to define a more informative state $s_{i,t}^{\text{SAF}} = g_\phi([s_{i,t}, m_{i,t}])$, where g_ϕ is parameterized as a neural network. Finally, the new state $s_{i,t}^{\text{SAF}}$ is used by the critic to compute values.

5.2 POLICY SELECTION

In order to perform policy selection for each policy we define an associated signature key which is initialized randomly at the start of the training: $\boldsymbol{K}_t^{\Pi} = \{K_t^{\pi^u} | 1 \leq u \leq U\}$. These keys are matched against queries computed as deterministic function of the encoded partial observation $q_{i,t}^{\text{policy}} = g_{\text{psel}}(s_{i,t})$, where g_{psel} is parametrized as a neaural network.

$$\text{index}_i = \text{GumbelSoftmax}\left(\frac{q_{i,t}^{\text{policy}} (\boldsymbol{K}_t^{\Pi})^T}{\sqrt{d_m}}\right) \tag{2}$$

As a result of this attention procedure, agent i selects a policy π^{index_i}. This operation is performed independently for each agent, i.e. each agent selects a policy from the policy pool.

5.3 AGENT AUTONOMY MAXIMIZATION

In order to encourage exploration and improve transfer and generalization capabilities we minimize the dependence of each agent's policy on the information that is available from the SAF's state. To do this, we define a posterior $p(z|s_t^{\text{SAF}}, s_t)$ and a prior distribution $p(z|s_{t-1})$, both parameterized using MLPs. Then, we compute the KL divergence between posterior and prior $D_{KL}(p(z|s_t^{\text{SAF}}, s_t)|p(z|s_{t-1}))$ and we use it as regularizer. As a result we induce a bottleneck on the information provided by the SAF, enhancing agent's autonomy and improving both transfer and generalization capabilities.

for $t \leftarrow 1$ **to** T **do**

Step 1: Each agent i having state information $o_{i,t}$ (encoded partial observation), generates a message.

$\forall i \in \{1, \ldots, N\}, m'_{i,t} = g_\theta(o_{i,t})$
$\boldsymbol{M}'_t = (m'_{1,t}, m'_{1,t}, m'_{2,t} ..., m'_{N,t})$

Step 2: SAF integrates information from all agents

$\widetilde{\boldsymbol{Q}} = \boldsymbol{F}_t \widetilde{\boldsymbol{W}}^q$
$\boldsymbol{F}_t \leftarrow \text{softmax}\left(\frac{\widetilde{\boldsymbol{Q}}(\boldsymbol{M}'_t \widetilde{\boldsymbol{W}}^e)^{\mathrm{T}}}{\sqrt{d_e}}\right) \boldsymbol{M}'_t \widetilde{\boldsymbol{W}}^v$
SAF state F_t is then updated trough self-attention

Step 3: Information from SAF is made available to each agent

$q^s_{i,t} = \boldsymbol{W}^q_{\text{read}} s_{i,t}, \forall i \in \{1, \ldots, N\}$
$\boldsymbol{\kappa} = (\boldsymbol{F}_t \boldsymbol{W}^e)^T$
$\boldsymbol{M}_t = \text{softmax}\left(\frac{\boldsymbol{Q}^s_t \boldsymbol{\kappa}}{\sqrt{d_e}}\right) \boldsymbol{F}_t \boldsymbol{W}^v$

Step 4: Values and Actions computation

$$\forall i \in \{1, \ldots, N\} \begin{cases} s^{\text{SAF}}_{i,t} & = g_\phi([s_{i,t}, m_{i,t}]) \\ V_{i,t} & = Critic(s^{\text{SAF}}_{i,t}) \\ a_{i,u,t} & \sim \pi_{i,u}(s_{i,t}), \forall u \in \{1, \ldots, U\} \end{cases}$$

Step 5: Policy Selection from the pool

$$\forall i \in \{1, \ldots, N\} \begin{cases} q^{policy}_{i,t} & = g_{psel}(s_{i,t}) \\ \text{index}_i & = \text{GumbelSoftmax}\left(\frac{q^{policy}_{i,t}(\boldsymbol{K}^{\Pi}_t)^T}{\sqrt{d_m}}\right) \\ a_{i,t} & = a_{i,\text{index}_i,t} \end{cases}$$

Step 6: Minimizing dependence on information made available by SAF

$$\mathbb{E}_{\pi_\theta}\left[r - \beta D_{\text{KL}}(p_{\text{posterior}}(z|s^{\text{SAF}}_t, s_t)|p_{\text{prior}}(z|s_{t-1}))\right]$$

end

Algorithm 1: MARL with *HECO-PPO*,SAF, Maximizing Agent Autonomy and Policy Selection

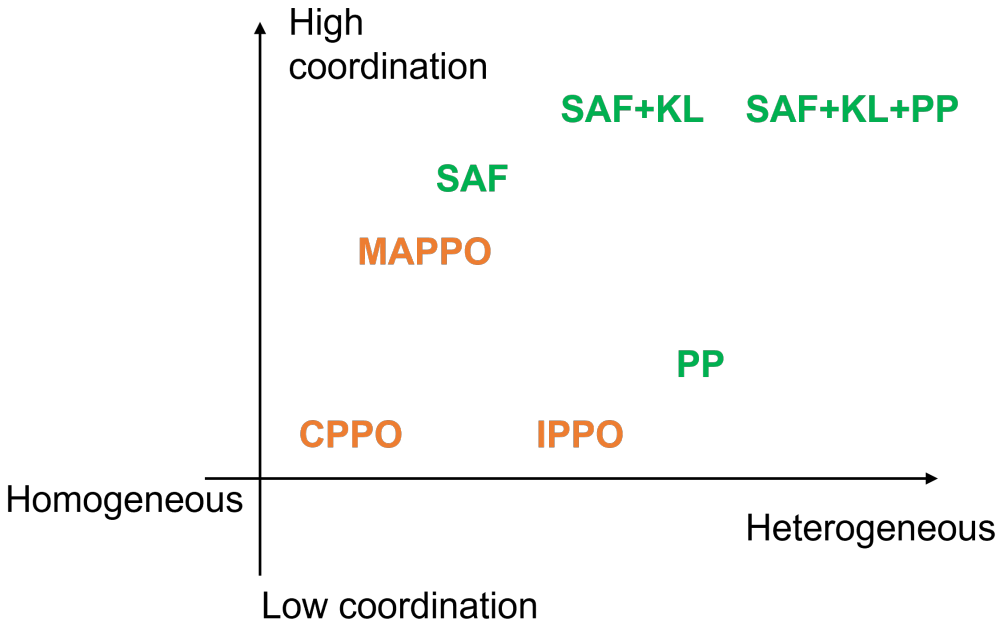
5.4 Algorithm

6 Experiments

In order to understand the impact of levels of coordination and environment heterogeneity on the team behaviours of cooperative agents, we propose the following hypotheses and design empirical experiments to test each of them:

- Cooperative tasks with higher required coordination levels are more difficult
- Cooperative tasks with higher environmental heterogeneity levels are more difficult
- Inter-agent communication with a bottleneck information facilitator helps in cooperative task in general
- Inter-agent communication with a pool of policies to dynamically choose from helps in cooperative team with high environmental heterogeneity

Figure 2: Hypothesis (NOT results) that how different components of our method be helpful



In this study, evaluate performance of popular independent PPO (IPPO) Yu et al. (2021a) and multi-agent PPO (MAPPO) methods and compare their performance with different versions of our HECO-PPO method. We summarize our hypothesis as in figure 2.

6.1 Higher levels of coordination and environmental heterogeneity lead to reduced performance of cooperative agents

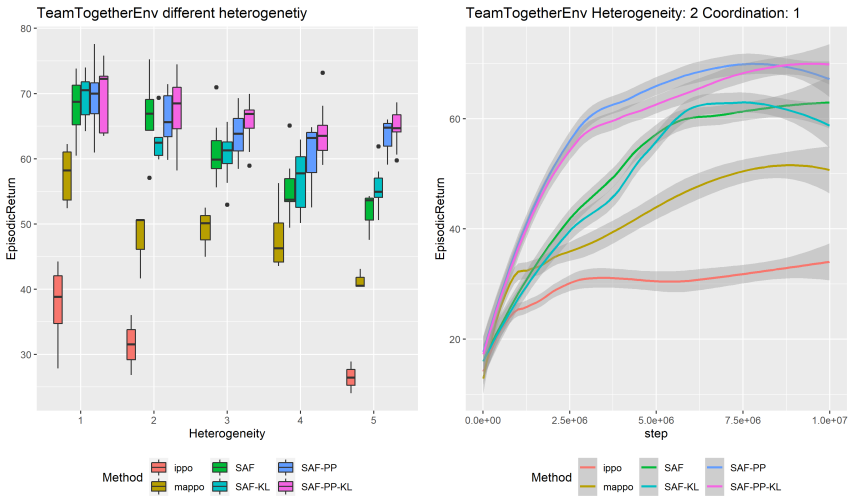
We compare performance of IPPO and MAPPO in tasks with with different coordination levels. The results show that, in all four HECO-Grid tasks, both algorithms show decreased performance when coordination levels are high (figure 4). This is especially true in ”TeamTogether” environment in which the agents can hardly obtain any reward when coordination level is larger than 2.

We also compare the performance of different algorithms in tasks with different environmental heterogeneity levels. The results show that, in all four HECOGrid tasks, both algorithms show decreased performance when environmental heterogeneity is high (figure 3).

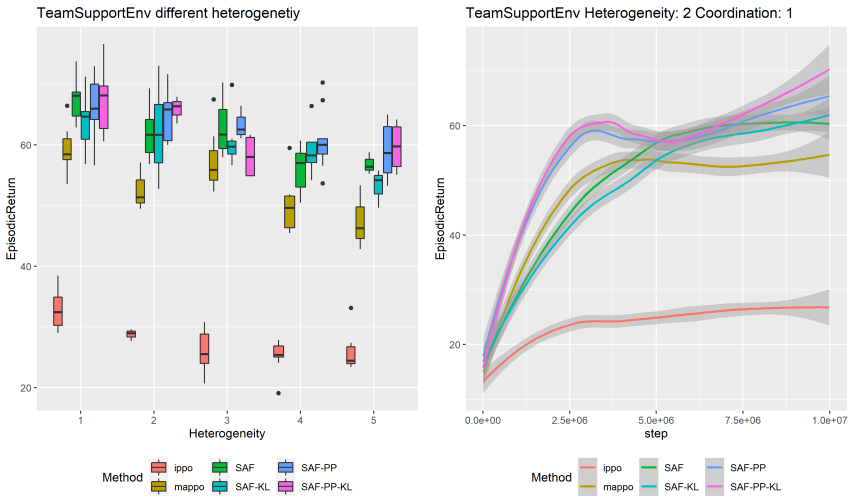
In addition, performance of cooperative agents in ”TeamSupport” task with various combination of heterogeneity levels and coordination levels are compared. The results suggest difficulties of the agents to obtain rewards when both levels are high.

6.2 Stateful Active Facilitator gives performance advantages for higher coordination levels

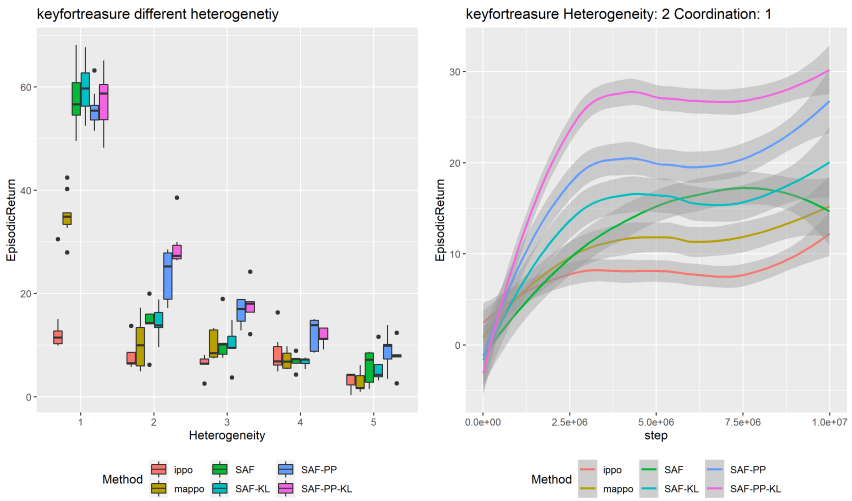
In the previous section, we hypothesized that inter-agent communication through a bottlenecked information facilitator (SAF) would help agents in tasks with high coordination levels. By conducting empirical experiments on HECOGrid environments with different levels of coordination, our results suggest the usage of SAF for inter-agent communication indeed improves performance, especially in the more difficult ”TeamTogether” environment, when compared with traditional IPPO and MAPPO methods.



(a) Different heterogeneity levels ”team together”

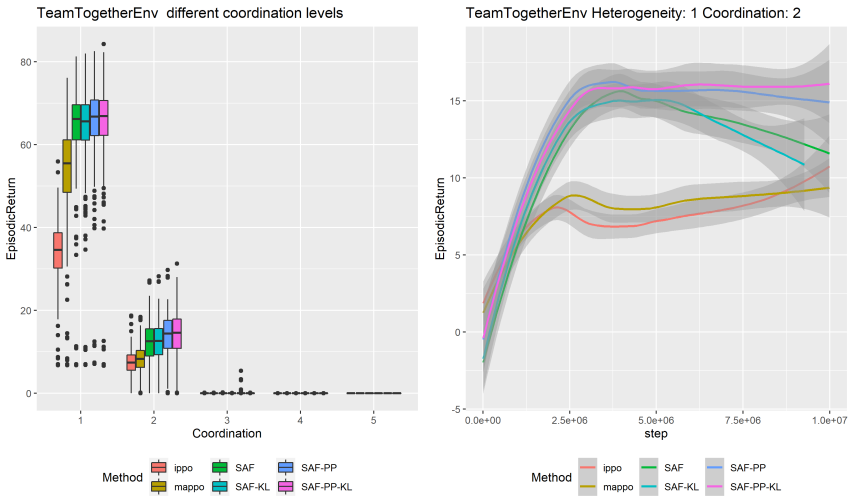


(b) Different heterogeneity levels ”team support”

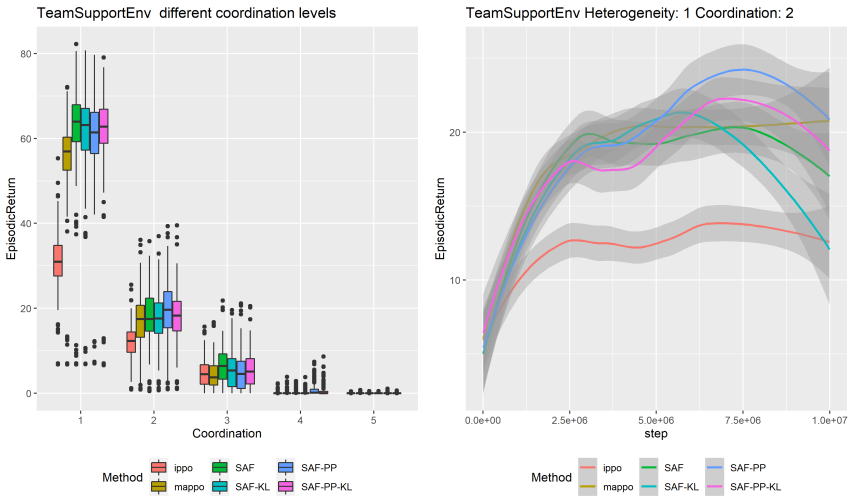


(c) Different heterogeneity levels ”KeyForFeatures”)

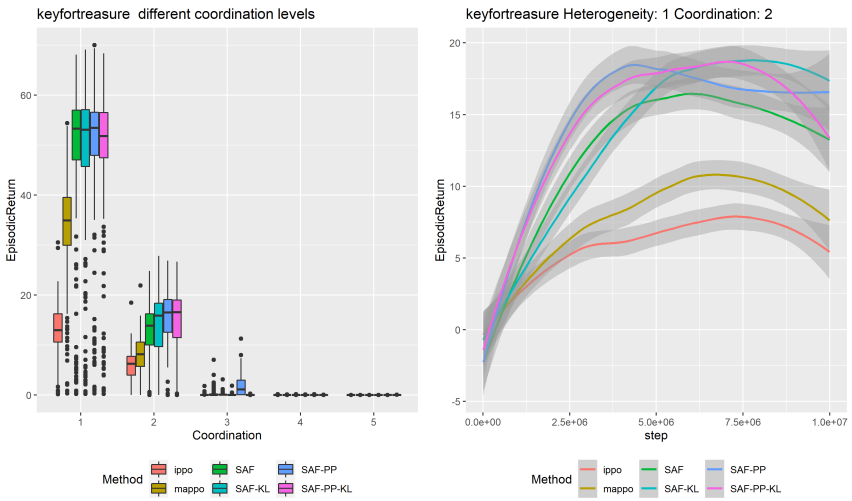
Figure 3: Performance of independent PPO (ippo). multi-agent PPO(mappo) and HECO-PPO methods in environments with different heterogeneity levels. Left column: performance across 3 different random seeds. Right column: Example performance curve in one setting.



(a) Different coordination levels "team together"



(b) Different coordination levels "team support"



(c) Different heterogeneity levels "KeyForTreasures"

Figure 4: Performance of independent PPO. multi-agent PPO and SAF methods in environments with different coordination levels. Left column: performance across 3 different random seeds. Right column: Example performance curve in one setting.

Method	No. of parameters
MAPPO	2,477,296
IPPO	1,740,016
SAF	2,727,536
SAF + PP	2,573,126
HECO-PPO (Ours) (SAF+PP+KL)	2,698,342

6.3 Shared pool of polices improve performance when environmental heterogeneity is high

We hypothesized that usage of a shared pool of policies the agents can dynamically choose from would help in tasks with high levels of environmental heterogeneity. We conducted experiment in tasks with different levels of heterogeneity and found the usage of policy pool bring significant improve when there environments are heterogeneous in all four HECOGrid environments.

6.4 Regularisation to encourage agents' Independence improve performance in heterogeneous environments

In order to further enhance performance of agents in heterogeneous environments, a KL divergence penalty is used to encourage each agent only use information from other agents when needed (see Method section for details). Our experimental results suggest the KL divergence improve agents' performance when heterogeneity levels are high in the more difficult "TeamTogether" task. However, the effects are not observed in other environments.

6.5 out-of-distribution generalization

In some scenarios, agents trained at one level of coordination or heterogeneity will be used in a task with different levels or even unknown levels. We like to understand if HECO-PPO has any advantages in this out-of-distribution(ODD) settings. We conduct OOD test experiments and show the HECO-PPO has advantages over traditional ippo and mappo in OOD test environments with different coordination, heterogeneity levels or different number of cooperative agents in the team.

6.6 Behavior analysis of HECO-PPO

In the previous sections, we show HECO-PPO has performance advantages for task with high heterogeneity and coordination levels. We like to understand how the saf facilitate communication and agents and how different policies in the pool are dynamically used.

Acknowledgments

References

Pierre-Luc Bacon, Jean Harb, and Doina Precup. The option-critic architecture. *Proceedings of the AAAI Conference on Artificial Intelligence*, 31(1), 2017. URL https://ojs.aaai.org/index.php/AAAI/article/view/10916.

Akhil Bagaria and George Konidaris. Option discovery using deep skill chaining. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=B1gqipNYwH.

Christopher Berner, Greg Brockman, Brooke Chan, Vicki Cheung, Przemysław Debiak, Christy Dennison, David Farhi, Quirin Fischer, Shariq Hashme, Chris Hesse, et al. Dota 2 with large scale deep reinforcement learning. *arXiv preprint arXiv:1912.06680*, 2019.

Jhelum Chakravorty, Patrick Nadeem Ward, Julien Roy, Maxime Chevalier-Boisvert, Sumana Basu, Andrei Lupu, and Doina Precup. Option-critic in cooperative multi-agent systems. In *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems*, AAMAS '20, pp. 1792–1794, Richland, SC, 2020. International Foundation for Autonomous Agents and Multiagent Systems. ISBN 9781450375184.

Maxime Chevalier-Boisvert and Lucas Willems. Minimalistic gridworld environment for openai gym. https://github.com/maximecb/gym-minigrid, 2018.

Abhishek Das, Théophile Gervet, Joshua Romoff, Dhruv Batra, Devi Parikh, Mike Rabbat, and Joelle Pineau. Tarmac: Targeted multi-agent communication. In *International Conference on Machine Learning*, pp. 1538–1546. PMLR, 2019.

Jakob N Foerster, Yannis M Assael, Nando De Freitas, and Shimon Whiteson. Learning to communicate with deep multi-agent reinforcement learning. *arXiv preprint arXiv:1605.06676*, 2016.

Jakob N. Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. Counterfactual multi-agent policy gradients. In *AAAI*, 2018.

A. Goyal, A. Lamb, P. Gampa, P. Beaudoin, S. Levine, C. Blundell, Y. Bengio, and M. C. Mozer. Object files and schemata: Factorizing declarative and procedural knowledge in dynamical systems. In *International Conference on Learning Representations*, 2021a.

Anirudh Goyal, Aniket Didolkar, Nan Rosemary Ke, Charles Blundell, Philippe Beaudoin, Nicolas Heess, Michael Mozer, and Yoshua Bengio. Neural production systems. *arXiv preprint arXiv:2103.01937*, 2021b.

Anirudh Goyal, Alex Lamb, Jordan Hoffmann, Shagun Sodhani, Sergey Levine, Yoshua Bengio, and Bernhard Schölkopf. Recurrent independent mechanisms. *ArXiv*, abs/1909.10893, 2021c.

Jayesh K Gupta, Maxim Egorov, and Mykel Kochenderfer. Cooperative multi-agent control using deep reinforcement learning. In *International Conference on Autonomous Agents and Multiagent Systems*, pp. 66–83. Springer, 2017.

Dongge Han, Wendelin Boehmer, Michael Wooldridge, and Alex Rogers. Multi-agent hierarchical reinforcement learning with dynamic termination. AAMAS '19, pp. 2006–2008, Richland, SC, 2019. International Foundation for Autonomous Agents and Multiagent Systems. ISBN 9781450363099.

Andrew Jaegle, Sebastian Borgeaud, Jean-Baptiste Alayrac, Carl Doersch, Catalin Ionescu, David Ding, Skanda Koppula, Daniel Zoran, Andrew Brock, Evan Shelhamer, Olivier J Henaff, Matthew Botvinick, Andrew Zisserman, Oriol Vinyals, and Joao Carreira. Perceiver IO: A general architecture for structured inputs & outputs. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=fILj7WpI-g.

Jiechuan Jiang and Zongqing Lu. Learning attentional communication for multi-agent cooperation. In *NeurIPS*, 2018.

Hao Jin, Yang Peng, Wenhao Yang, Shusen Wang, and Zhihua Zhang. Federated reinforcement learning with environment heterogeneity. In *International Conference on Artificial Intelligence and Statistics*, pp. 18–37. PMLR, 2022.

Khimya Khetarpal, Martin Klissarov, Maxime Chevalier-Boisvert, Pierre-Luc Bacon, and Doina Precup. Options of interest: Temporal abstraction with interest functions. In *AAAI*, 2020.

Woojun Kim, Jongeui Park, and Youngchul Sung. Communication in multi-agent reinforcement learning: Intention sharing. In *International Conference on Learning Representations*, 2020.

Martin Klissarov, Pierre-Luc Bacon, Jean Harb, and Doina Precup. Learning options end-to-end for continuous action tasks. 2017. doi: 10.48550/ARXIV.1712.00004. URL https://arxiv.org/abs/1712.00004.

Joel Z. Leibo, Edgar A. Duéñez-Guzmán, Alexander Sasha Vezhnevets, John P. Agapiou, Peter Sunehag, R. Koster, Jayd Matyas, Charlie Beattie, Igor Mordatch, and Thore Graepel. Scalable evaluation of multi-agent reinforcement learning with melting pot. *ArXiv*, abs/2107.06857, 2021.

Chenghao Li, Chengjie Wu, Tonghan Wang, Jun Yang, Qianchuan Zhao, and Chongjie Zhang. Celebrating diversity in shared multi-agent reinforcement learning. *arXiv preprint arXiv:2106.02195*, 2021.

Ryan Lowe, Yi Wu, Aviv Tamar, Jean Harb, Pieter Abbeel, and Igor Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. *Neural Information Processing Systems (NIPS)*, 2017a.

Ryan Lowe, Yi Wu, Aviv Tamar, Jean Harb, Pieter Abbeel, and Igor Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. *arXiv preprint arXiv:1706.02275*, 2017b.

Anuj Mahajan, Tabish Rashid, Mikayel Samvelyan, and Shimon Whiteson. Maven: Multi-agent variational exploration. In *NeurIPS*, 2019.

Ofir Nachum, Shixiang Gu, Honglak Lee, and Sergey Levine. Data-efficient hierarchical reinforcement learning. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, pp. 3307–3317. Curran Associates Inc., 2018.

Kamal Ndousse, Douglas Eck, Sergey Levine, and Natasha Jaques. Emergent social learning via multi-agent reinforcement learning. In *ICML*, 2021.

Shubham Pateria, Budhitama Subagdja, Ah-hwee Tan, and Chai Quek. Hierarchical reinforcement learning: A comprehensive survey. *ACM Comput. Surv.*, 54(5), jun 2021. ISSN 0360-0300. doi: 10.1145/3453160. URL https://doi.org/10.1145/3453160.

Nasim Rahaman, Muhammad Waleed Gondal, Shruti Joshi, Peter V. Gehler, Yoshua Bengio, Francesco Locatello, and Bernhard Scholkopf. Dynamic inference with neural interpreters. In *NeurIPS*, 2021.

Tabish Rashid, Mikayel Samvelyan, Christian Schroeder, Gregory Farquhar, Jakob Foerster, and Shimon Whiteson. Qmix: Monotonic value function factorisation for deep multi-agent reinforcement learning. In *International Conference on Machine Learning*, pp. 4295–4304. PMLR, 2018a.

Tabish Rashid, Mikayel Samvelyan, C. S. D. Witt, Gregory Farquhar, Jakob N. Foerster, and Shimon Whiteson. Qmix: Monotonic value function factorisation for deep multi-agent reinforcement learning. *ArXiv*, abs/1803.11485, 2018b.

Matthew Riemer, Miao Liu, and Gerald Tesauro. Learning abstract options. *ArXiv*, abs/1810.11583, 2018.

Mikayel Samvelyan, Tabish Rashid, Christian Schroeder de Witt, Gregory Farquhar, Nantas Nardelli, Tim G. J. Rudner, Chia-Man Hung, Philip H. S. Torr, Jakob Foerster, and Shimon Whiteson. The starcraft multi-agent challenge, 2019.

Amanpreet Singh, Tushar Jain, and Sainbayar Sukhbaatar. Learning when to communicate at scale in multiagent cooperative and competitive tasks. *arXiv preprint arXiv:1812.09755*, 2018.

Sainbayar Sukhbaatar, arthur szlam, Jason Weston, and Rob Fergus. End-to-end memory networks. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 28*, pp. 2440–2448. Curran Associates, Inc., 2015. URL http://papers.nips.cc/paper/5846-end-to-end-memory-networks.pdf.

Sainbayar Sukhbaatar, Arthur D. Szlam, and Rob Fergus. Learning multiagent communication with backpropagation. In *NIPS*, 2016.

Peter Sunehag, Guy Lever, Audrunas Gruslys, Wojciech M. Czarnecki, Vinícius Flores Zambaldi, Max Jaderberg, Marc Lanctot, Nicolas Sonnerat, Joel Z. Leibo, Karl Tuyls, and Thore Graepel. Value-decomposition networks for cooperative multi-agent learning. *ArXiv*, abs/1706.05296, 2018.

Richard S. Sutton, Doina Precup, and Satinder Singh. Between mdps and semi-mdps: A framework for temporal abstraction in reinforcement learning. *Artificial Intelligence*, 112(1):181–211, 1999. ISSN 0004-3702. doi: https://doi.org/10.1016/S0004-3702(99)00052-1. URL https://www.sciencedirect.com/science/article/pii/S0004370299000521.

Ming Tan. Multi-agent reinforcement learning: Independent versus cooperative agents. In *ICML*, 1993.

Ming Tan. *Multi-Agent Reinforcement Learning: Independent vs. Cooperative Agents*, pp. 487–494. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1997. ISBN 1558604952.

Oriol Vinyals, Timo Ewalds, Sergey Bartunov, Petko Georgiev, Alexander Sasha Vezhnevets, Michelle Yeo, Alireza Makhzani, Heinrich Küttler, John Agapiou, Julian Schrittwieser, et al. Starcraft ii: A new challenge for reinforcement learning. *arXiv preprint arXiv:1708.04782*, 2017.

Jianhao Wang, Zhizhou Ren, Terry Liu, Yang Yu, and Chongjie Zhang. Qplex: Duplex dueling multi-agent q-learning. *arXiv preprint arXiv:2008.01062*, 2020.

Yuanfei Wang, Fangwei Zhong, Jing Xu, and Yizhou Wang. Tom2c: Target-oriented multi-agent communication and cooperation with theory of mind, 2021.

Marissa A. Weis, Kashyap Chitta, Yash Sharma, Wieland Brendel, Matthias Bethge, Andreas Geiger, and Alexander S. Ecker. Unmasking the inductive biases of unsupervised object representations for video sequences, 2020.

C. S. D. Witt, Tarun Gupta, Denys Makoviichuk, Viktor Makoviychuk, Philip H. S. Torr, M. Sun, and Shimon Whiteson. Is independent learning all you need in the starcraft multi-agent challenge? *ArXiv*, abs/2011.09533, 2020a.

C. S. D. Witt, Tarun Gupta, Denys Makoviichuk, Viktor Makoviychuk, Philip H. S. Torr, M. Sun, and Shimon Whiteson. Is independent learning all you need in the starcraft multi-agent challenge? *ArXiv*, abs/2011.09533, 2020b.

Chao Yu, Akash Velu, Eugene Vinitsky, Yu Wang, Alexandre Bayen, and Yi Wu. The surprising effectiveness of ppo in cooperative, multi-agent games. *arXiv preprint arXiv:2103.01955*, 2021a.

Chao Yu, Akash Velu, Eugene Vinitsky, Yu Wang, Alexandre M. Bayen, and Yi Wu. The surprising effectiveness of mappo in cooperative, multi-agent games. *ArXiv*, abs/2103.01955, 2021b.

A APPENDIX

You may include other additional sections here.