COMMUNICATION-EFFICIENT ACTOR-CRITIC METHODS FOR HOMOGENEOUS MARKOV GAMES

Anonymous authors
Paper under double-blind review

ABSTRACT

Recent success in cooperative multi-agent reinforcement learning (MARL) relies on centralized training and policy sharing. Centralized training eliminates the issue of non-stationarity MARL yet induces large communication costs, and policy sharing is empirically crucial to efficient learning in certain tasks yet lacks theoretical justification. In this paper, we formally characterize a subclass of cooperative Markov games where agents exhibit a certain form of homogeneity such that policy sharing provably incurs no suboptimality. This enables us to develop the first consensus-based decentralized actor-critic method where the consensus update is applied to both the actors and the critics while ensuring convergence. We also develop practical algorithms based on our decentralized actor-critic method to reduce the communication cost during training, while still yielding policies comparable with centralized training.

1 INTRODUCTION

Cooperative multi-agent reinforcement learning (MARL) is the problem where multiple agents learn to make sequential decisions in a common environment to optimize a shared reward signal, which finds a wide range of real-world applications such as traffic control (Chu et al., 2019), power grid management (Callaway & Hiskens, 2010), and coordination of multi-robot systems (Corke et al., 2005). Efficient learning for large and complex cooperative MARL tasks is challenging. Naively reducing cooperative MARL to single-agent RL with a joint observation-action space imposes significant scalability issues, since the joint space grows exponentially with the number of agents. Approaches that treat each agent as an independent RL learner, such as Independent Q-Learning (Tan, 1993), overcome the scalability issue yet fail to succeed in complicated tasks due to the non-stationarity caused by other learning agents' evolving policies. To address these challenges, the paradigm of Centralized Training and Decentralized Execution (CTDE) is then proposed, where a *centralized trainer* is assumed to access to information of all agents during training to approximate the global (action-)value function, whereas each agent only needs local information for its action selection during decentralized policy execution (Lowe et al., 2017; Foerster et al., 2017). The centralized critic eliminates non-stationarity during training, while the policy decentralization ensures scalability during execution. Besides, existing CTDE methods almost always enable *policy parameter sharing* to further improve learning scalability and efficiency, where agents also share the parameters of their decentralized policies.

However, in many real-world scenarios, there is not a readily available centralizer that conveniently gathers the global information from all agents, and therefore agents need to rely on all-to-all communication for centralized training, incurring enormous communication overheads for large numbers of agents. This motivates us to think about whether it is possible to train agents in a decentralized and communication-efficient manner, while still keeping the benefits of the centralized training of CTDE. Moreover, despite its wide adoption, little theoretical understanding has been provided to justify policy parameter sharing. Agents should at least exhibit a certain level of homogeneity before it is feasible to share their policies. For example, if the observation and/or action spaces vary across agents, then their decentralized policies cannot even have the same architecture. Even if it is feasible, it is unclear whether restricting the agents to share their policy parameters will introduce any suboptimality.

In this paper, we address these aforementioned issues centered around the CTDE framework. We begin by formally characterizing a subclass of Markov games where the cooperative agents exhibit a certain form of homogeneity such that it is not only feasible but also incurs no suboptimality to share their decentralized policies, thus providing a first theoretical justification for policy parameter sharing. We then develop a decentralized actor-critic algorithm for homogeneous MGs where agents share their critic and actor parameters with consensus-based updates, for which we prove an asymptotic convergence guarantee with linear critics, full observability, and other standard assumptions. To our knowledge, this is the first decentralized actor-critic algorithm that enjoys provable convergence guarantees with policy (i.e., actor) consensus. To account for communication efficiency, we develop a simple yet effective bandit-based process that wisely selects when and with whom to perform the parameter census update based on the feedback of policy improvement during training. To further account for partial observability, we develop an end-to-end learnable gating mechanism for the agents to selectively share their observations and actions for learning the decentralized critics. This series of innovations are capable of transforming any CTDE algorithm into its decentralized and communication-efficient counterpart, with policy consensus in homogeneous MGs for improved efficiency. Our empirical results demonstrate the effectiveness of these innovations when instantiated with a state-of-the-art CTDE algorithm, achieving competitive policy performance with only a fraction of communication during training.

Our contribution is therefore summarized as three-fold: (1) the characterization of a subclass of cooperative Markov games, i.e. homogeneous Markov games (MGs), where policy sharing provably incurs no loss of optimality; (2) a decentralized MARL algorithm for homogeneous MGs that enjoys asymptotic convergence guarantee with policy consensus; and (3) practical techniques that transform CTDE algorithms to their decentralized and communication-efficient counterparts.

2 Related Work

Communication in MARL. Communication is key to solving the issue of non-stationarity in MARL. The CTDE paradigm (Lowe et al., 2017; Foerster et al., 2017) assumes a centralized unit during training to learn a joint value function. Other methods, such as CommNet (Sukhbaatar et al., 2016) and BiCNet (Peng et al., 2017), do not assume a centralized unit and instead allow agents to share information by all-to-all broadcasting, effectively relying on centralized communication. These methods require centralized/all-to-all communication that impedes their application to large numbers of agents. Although follow-up work such as IC3Net (Singh et al., 2018) and VBC (Zhang et al., 2019) proposes algorithms to learn when to communicate, agents there still perform all-to-all communication before others decide whether to receive. We instead entirely abandon centralized/all-to-all communication, letting each agent decide whom to communicate to purely based on its local observation. There is another line of work, Networked MARL (NMARL) (Zhang et al., 2018), that where agents lie on a predefined network such that neighboring agents can freely communicate. Our approach instead learns sparse communication that is dynamically adjusted during decentralized training, even if the predefined network topology can be dense.

Policy parameter sharing and consensus. Policy parameter sharing is widely adopted in MARL where agents share the same action space, yet it has not been theoretically justified. A recent result from Kuba et al. (2021) states that enforcing policy parameter sharing in a general cooperative MG can lead to a suboptimal outcome that is exponentially-worse with the number of agents. We are the first to 1) formally characterize the subclass of homogeneous MGs where enforcing policy parameter incurs no suboptimality and 2) develop an algorithm that performs policy parameter sharing in homogeneous MGs in a soft manner with decentralized consensus-based policy update with convergence guarantees. Zhang & Zavlanos (2019) also develop a policy consensus algorithm for decentralized MARL, yet they do not assume homogeneity and thus need each agent to represent the joint policy for consensus.

3 Homogeneous Markov Game

We consider a cooperative Markov game (MG) $\langle \mathcal{N}, \mathcal{S}, \mathcal{A}, P, R \rangle$ with N agents indexed by $i \in \mathcal{N} = \{1, ..., N\}$, state space $\mathcal{S}$, action space $\mathcal{A} = \mathcal{A}^1 \times \cdots \times \mathcal{A}^N$, transition function $P : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$, and reward functions $R = \{R^i\}_{i \in \mathcal{N}}$ with $R^i : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ for each $i \in \mathcal{N}$. In Section 3, we assume

full observability for simplicity, i.e., each agent observes the state $s \in \mathcal{S}$. Under full observability, we consider joint policies, $\pi : \mathcal{S} \times \mathcal{A} \to [0, 1]$, that can be factored as the product of local policies $\pi^i : \mathcal{S} \times \mathcal{A}^i \to [0, 1]$, $\pi(a|s) = \prod_{i \in \mathcal{N}} \pi^i(a^i|s)$. Let $r(s, a) := \frac{1}{N} \sum_{i \in \mathcal{N}} R^i(s, a)$ denote the joint reward function, and let $\gamma \in [0, 1]$ denote the discount factor. Define the discounted return from time step t as $G_t = \sum_{l=0}^{\infty} \gamma^l r_{t+l}$, where $r_t := r(s_t, a_t)$ is the reward at time step t. The agents' joint policy $\pi = (\pi^1, ..., \pi^N)$ induce a value function, which is defined as $V^\pi(s_t) = \mathbb{E}_{s_{t+1:\infty}, a_{t:\infty}}[G_t|s_t]$, and action-value function $Q^\pi(s_t, a_t) = \mathbb{E}_{s_{t+1:\infty}, a_{t+1:\infty}}[G_t|s_t, a_t]$. The agents are cooperative in the sense that they aim to optimize their policies with respect to the joint reward function, i.e., $\max_\pi J(\pi) = \mathbb{E}_{s_{0:\infty}, a_{0:\infty}}[G_0]$.

3.1 Homogeneous MG: Definition, Properties, and Examples

As along as the action spaces $\{\mathcal{A}^i\}_{i \in \mathcal{N}}$ are homogeneous, policy sharing among $\{\pi^i\}_{i \in \mathcal{N}}$ is feasible. However, such policy sharing can incur suboptimal joint policies for general MGs, as we will see in an example introduced by Kuba et al. (2021) and revisited in this subsection. Here, we characterize a subclass of Markov games in Definition 1 requiring conditions stronger than homogeneous action spaces, where policy sharing provably incurs no suboptimality.

Definition 1 (Homogeneous Markov game)**.** Markov game $\langle \mathcal{N}, \mathcal{S}, \mathcal{A}, P, R \rangle$ is *homogeneous* if:

(i) The local action spaces are homogeneous, i.e., $\mathcal{A}^i = \mathcal{A}^j \ \forall i, j \in \mathcal{N}$. Further, the state is decomposed into local states with homogeneous local state spaces, i.e., $s = (s^1, ..., s^N) \in \mathcal{S} = \mathcal{S}^1 \times \cdots \times \mathcal{S}^N$ with $\mathcal{S}^i = \mathcal{S}^j \ \forall i, j \in \mathcal{N}$.

(ii) The transition function and the joint reward function are permutation invariant. Formally, for any $s_t = (s_t^1, ..., s_t^N), s_{t+1} = (s_{t+1}^1, ..., s_{t+1}^N) \in \mathcal{S}$ and $a_t = (a_t^1, ..., a_t^N) \in \mathcal{A}$, we have

$$P(M_i s_{t+1}|M_i s_t, M_i a_t) = P(M_j s_{t+1}|M_j s_t, M_j a_t), \quad R(M_i s_t, M_i a_t) = R(M_j s_t, M_j a_t)$$

for any $M_i, M_j \in \mathcal{M}$, where $R(s, a) := (R^1(s, a), ..., R^N(s, a))$, Mx denotes a permutation M of ordered list $x = (x^1, ..., x^N)$, and $\mathcal{M}$ is the set of all possible permutations.

(iii) Each agent $i \in \mathcal{N}$ has access to a bijective function $o^i : \mathcal{S} \to \mathcal{O}$ that maps states to a common observation space $\mathcal{O}$. These observation functions $\{o^i\}_{i \in \mathcal{N}}$ are permutation preserving with respect to the state, i.e., for any $s \in \mathcal{S}$ and any $M \in \mathcal{M}$,

$$\left(o^1(Ms), ..., o^N(Ms)\right) = M\left(o^1(s), ..., o^N(s)\right).$$

By Definition 1, besides requiring homogeneous action spaces, our characterization of homogeneous MGs further requires that the global state can be factored into homogeneous local states (condition (i)) such that the transition and reward functions are permutation invariant (condition (ii)). Moreover, condition (iii) requires each agent to have an observation function to form its *local representation* of the global state. The main property of the homogeneous MG is that, after representing the global state with the observation functions, policy sharing incurs no suboptimality. This is formally stated in Theorem 1. Proofs of all the theorems are included in the appendix.

Theorem 1. *Let Π be the set of state-based joint policies, i.e., $\Pi = \{\pi = (\pi^1, ..., \pi^N) : \pi^i : \mathcal{S} \times \mathcal{A}^i \to [0, 1]\}$, and let Π_o be the set of observation-based joint policies, i.e., $\Pi_o = \{\pi_o = (\pi_o^1, ..., \pi_o^N) : \pi_o^i : \mathcal{S} \times \mathcal{O} \to [0, 1]\}$. In homogeneous MGs, we have*

$$\max_{\pi = (\pi^1, ..., \pi^N) \subset \Pi} J(\pi) = \max_{\pi_o = (\pi_o^1, ..., \pi_o^N) \subset \Pi_o} J(\pi_o) = \max_{\pi_o = (\pi_o^1, ..., \pi_o^N):\ \pi_o^1 = ... = \pi_o^N \in \Pi_o} J(\pi_o).$$

To provide more intuition for homogeneous MGs, we here give an example from Multi-Agent Particle Environment (MPE) (Lowe et al., 2017) and a non-example from Kuba et al. (2021).

Example: cooperative navigation. In a cooperative navigation task in MPE, N agents move as a team to cover N landmarks in a 2D space. Under full observably where each agent can observe the information (locations and/or velocities) of all agents and landmarks, we can cast a cooperative navigation task as a homogeneous MG by verifying the three conditions in Definition 1: (i) The local state of agent i consists of its absolute location $l^i = (l_x^i, l_y^i) \in \mathbb{R}^2$ and its absolute velocity $v^i = (v_x^i, v_y^i) \in \mathbb{R}^2$ with respect to the common origin, as well as the absolute locations of all

N landmarks, $\{c^k = (c_x^k, c_y^k)\}_{k=1}^N$. Therefore, the location state spaces are homogeneous, and the concatenation of all the location states preserves the global state of the task. Since local action is the change in velocity, the local action spaces are also homogeneous. (ii) The transition function determines the next global state by the current state and all agents' actions according to physics, and thus it is permutation invariant. The reward function R^i determines the reward for agent i according to the distances between all the agents and the landmarks to encourage coverage, as well as penalties to discourage collisions if any, resulting in a permutation invariant joint reward function. (iii) In MPE, agents' observations are based on *relative*, instead of absolute, locations and/or velocities of other agents and/or landmarks. For cooperative navigation, such observations happen to define observation functions that are bijective and permutation preserving. Specifically, function o^i yields the observation for agent i that consists of its absolute location l^i and velocity v^i, the relative location $l_j^i := l^j - l^i$ and velocity $v_j^i := v^j - v^i$ of other agents $j \in \mathcal{N} \setminus \{i\}$, and the relative location $c_k^i := c^k - l^i$ of all the landmarks $k = 1, .., N$.

Non-Example: a stateless MG. Kuba et al. (2021) recently shows that enforcing policy parameter sharing is exponentially-worse than the optimality without such a restriction in the following stateless MG: Consider a cooperative MG with an even number of N agents, one state s, and the joint action space $0, 1^N$, where the reward function is given by $R(s, 0^{1:N/2}, 1^{1+N/2:N}) = 1$ and $R(s, a^{1:N}) = 0$ for all other joint actions. It is obvious that the optimal value of this MG is 1, while Kuba et al. (2021) prove that the optimal value under policy parameter sharing is $1/2^{N-1}$. This MG is not a homogeneous MG: the agents are relying on the raw state to represent their policies, and therefore their observation functions are identity mappings $o^i(s) = s$, which is not permutation preserving and violates condition (iii) in Definition 1.

3.2 Policy Consensus for Homogeneous MGs

Theorem 1 theoretically justifies the parameter sharing among the actors with observation-based representations, which enables us to develop the first decentralized actor-critic algorithms with consensus update among local (observation-based) actors.

Formally, the critic class $Q(\cdot, \cdot; \omega)$ is parameterized with ω to approximate the global action-value function $Q^\pi(\cdot, \cdot)$. Upon on-policy transition $(s_t, a_t, \{r_t^i\}_{i\in\mathcal{N}}, s_{t+1}, a_{t+1})$ sampled by the current policy, the critic parameter ω^i for each agent $i \in \mathcal{N}$ is updated using its local temporal difference (TD) learning followed by a consensus update (Zhang et al., 2018):

$$\tilde{\omega}_t^i = \omega_t^i + \beta_{\omega,t} \cdot \delta_t^i \cdot \nabla_\omega Q(s_t, a_t; \omega_t^i), \qquad \omega_{t+1}^i = \textstyle\sum_{j\in\mathcal{N}} c_{\omega,t}(i,j) \cdot \tilde{\omega}_t^j \tag{1}$$

where $r_t^i = R^i(s_t, a_t)$ is the local reward of agent i, $\delta_t^i = r_t^i + \gamma Q(s_{t+1}, a_{t+1}; \omega_t^i) - Q(s_t, a_t; \omega_t^i)$ is the local TD error of agent i, $\beta_{\omega,t} > 0$ is the critic stepsize, and $C_{\omega,t} = [c_{\omega,t}(i,j)]_{i,j\in\mathcal{N}}$ is the critic consensus matrix. The observation-based actor for each agent $i \in \mathcal{N}$ is parameterized as $\pi^i(a^i|o^i(s); \theta^i)$ with parameter θ^i, which is updated based on the multi-agent policy gradient derived from the critic followed by a consensus update:

$$\tilde{\theta}_{t+1}^i = \theta_t^i + \beta_{\theta,t} \cdot Q(s_t, a_t; \omega_t^i) \cdot \nabla_{\theta^i} \log \pi^i(a_t^i|o^i(s_t); \theta_t^i), \qquad \theta_{t+1}^i = \textstyle\sum_{j\in\mathcal{N}} c_{\theta,t}(i,j) \cdot \tilde{\theta}_t^j \tag{2}$$

where the observation-based actor class $\pi(\cdot|\cdot; \theta)$ is assumed to be differentiable, $\beta_{\theta,t} > 0$ is the actor stepsize, and $C_{\theta,t} = [c_{\theta,t}(i,j)]_{i,j\in\mathcal{N}}$ is the actor consensus matrix. Theorem 2 states the convergence of $\{\omega_t^i\}$ and $\{\theta_t^i\}$ generated by (1)(2) with the linear critic class and under standard assumptions on the stepsizes, consensus matrices, and stability (refer to the appendix for details).

Theorem 2. *Under standard assumptions for linear actor-critic methods with consensus update, with $\{\omega_t^i\}$ and $\{\theta_t^i\}$ generated from Equations (1) and (2), we have $\lim_t \omega_t^i = \omega_*$ and $\lim_t \theta_t^i = \theta_*$ almost surely for any $i \in \mathcal{N}$, where θ_* is a stationary point associated with update (2), and θ_*^i is the minimizer of the mean square projected Bellman error for the joint policy parameterized by θ_*.*

4 Practical Algorithm

The convergence of our decentralized actor-critic algorithm in Section 3.2 relies on the assumptions of linear function approximators, full observability, and well-connected consensus. In this section, we develop a practical algorithm that relaxes these assumptions and achieves communication efficiency. Specifically, we use neural networks to represent the our actors and critics. With agents

only partially observing the global state, we in Section 4.1 develop an architecture, *communication network*, for the agents to learn to exchange their local observations and actions in a communication efficient manner. To achieve communication efficiency on parameter consensus, in Section 4.2 we develop an effective *bi-level* multi-armed bandit for the agents to learn to exchange the parameters only when it benefits learning.

4.1 EFFICIENT OBSERVATION-ACTION COMMUNICATION

We consider the partial observability setting where the agents cannot directly observe the global state s_t and their observation functions $\{o^i\}$ are not bijective. Further, similar to Network MARL (Zhang et al., 2018), we assume the agents can communicate through a time-variant network $\mathcal{G}_t := (\mathcal{N}, \mathcal{E}_t)$ with vertex set $\mathcal{N}$ and directed edge set $\mathcal{E}_t \subseteq \{(i,j) : i,j \in \mathcal{N}, i \neq j\}$. Denote the neighbors of agent i at time step t as $\mathcal{N}_t^i := \{j : (i,j) \in \mathcal{E}_t\}$. Every time step t, each agent i uses the observations and actions of its own as well as from its neighbors for its critic to approximate the global action-value. To encourage communication efficiency, we propose the communication network that selects a subset $\mathcal{C}_t^i \subseteq \mathcal{N}_t^i$ for observation-action communication, such that agent i's critic becomes $Q^i(\{(o_t^k, a_t^k)\}_{k \in \{i\} \cup \mathcal{C}_t^i})$. For the texts below, we abuse notation $o_t^i := o^i(s_t)$ to denote the observation and omit the subscript of time step t when the context is clear.

Communication network. The communication network C^i of agent i outputs $c_j^i \in \{0,1\}$ indicating whether to communicate with neighbor $j \in \mathcal{N}^i$, i.e., $\mathcal{C}^i = \{j \in \mathcal{N}^i : c_j^i = 1\}$. Specifically, we choose an L-layer graph convolutional networks (GCN) to implement C^i, which can deal with arbitrary input size determined by $|\mathcal{N}^i|$ achieve permutation invariance. Specifically, the input to this GCN is a fully connected graph with one vertex $(o^i, e_j^i(o^i))$ per neighbor $j \in \mathcal{N}^i$, where $e_j^i(o^i)$ embeds information of neighbor j that can be extracted from o^i. For example, when agents' identities are observable, $e_j^i(o^i)$ can be (the embedding of) the ID of neighbor j. When identities are not observable, $e_j^i(o^i)$ preserves information specific to neighbor j, such as the physical distance from j to i in MPE. The GCN's last layer output the logits from which $\{c_j^i\}_{j \in \mathcal{N}^i}$ are sampled. To enable differentiability of the sampling, we use the reparameterization trick Straight-Through Gumbel-Softmax (Jang et al., 2016).

Actor and critic networks. Our proposed communication network is compatible with any multi-agent actor-critic architecture. Our experiments mainly explore deterministic actors $a^i = \pi^i(o^i)$. Similar to the communication network, the critic network $Q^i(\{(o^k, a^k)\}_{k \in \{i\} \cup \mathcal{C}^i})$ is also implemented by an L-layer GCN to deal with arbitrary input size and achieve permutation invariance, where the input to the first layer is the fully connected graph with vertices $\{(o^k, a^k)\}_{k \in \{i\} \cup \mathcal{C}^i}$.

Training. Critic Q^i directly guides agent i's actor update using the deterministic policy gradient. Critic Q^i itself is updated to minimize the TD loss $\mathcal{L}_{\text{TD}}^i = \mathbb{E}_{o_t, a_t, r_t, o_{t+1}}[(Q_t^i - y_t^i)^2]$, where $o_t := (o_t^1, ..., o_t^N)$ is the joint observation, $a_t := (a_t^1, ..., a_t^N)$ is the joint action, $Q_t^i := Q^i(\{(o_t^k, a_t^k)\}_{k \in \{i\} \cup \mathcal{C}_t^i})$ is the abbreviated notation for the critic value of agent i at timestep t, and $y_t^i := r_t^i + \gamma Q_{t+1}^i$ is the TD target. Due to the differentiability enabled by Gumbel-Softmax, the gradient can flow from Q^i to communication network C^i. Commonly used in the literature (Jang et al., 2016), the update of C^i is also guided by a regularization term $|\frac{1}{|\mathcal{N}_t^i|} \sum_{j \in \mathcal{N}_t^i} c_j^i - \eta|$, which places a restriction on the amount of communication allowed defined by rate η.

4.2 A BI-LEVEL BANDIT FOR PARAMETER CONSENSUS

The parameter consensus defined in Equations (1,2) requires each agent i to communicate with all other agents j such that the (i,j) entry of the consensus matrix is non-zero. To achieve communication efficiency, existing literature mainly considers gossip algorithms where each agent only communicated with one neighbor j per communication round (Boyd et al., 2006), i.e., the consensus matrix entries satisfy $c(i,j) = c(i,i) = 1/2$. Here, we develop a novel bi-level multi-armed bandit to further improve communication efficiency over gossip algorithms, where at each round each agent chooses *whether or not* to perform consensus at the high level, and if yes, chooses which neighbor to perform gossip consensus update. For ease of exposition, we assume that 1) the agents

perform a gradient step and decide whether and how to perform parameter consensus every episode indexed by $m = 1, 2, ...$, and 2) every agent can always choose from all other agents for parameter consensus, i.e., $\mathcal{N}_t^i = \mathcal{N} \setminus \{i\}$. Extensions of our bi-level bandit to more general settings are straightforward. We next formally describe this bi-level bandit for an arbitrary agent i, dropping the superscript i for convenience.

Arms: The high-level is a 2-armed bandit determining whether to perform consensus at each round m. Denote the selected high-level arm at round m as x_m^1, where $x_m^1 = 0, 1$ corresponds to performing and not-performing consensus, respectively. The low-level is a $(N-1)$-armed bandit, and we let $x_m^2 \in \mathcal{N} \setminus \{i\}$ denote the selected low-level arm at round m.

Rewards: We design different reward functions for the arms in the two levels, as they have different goals. The high-level bandit aims to 1) reduce communication while 2) maintaining a reasonable performance of the learned consensus matrices. Requirement 2) can be captured by the difference of the episodic rewards at different time steps, and requirement 1) can be measured by the frequency of selecting to perform consensus. Let G^m be the total rewards of episode m. To fulfill requirement 1), We first normalize return of the current episode m using the latest l episodes, $r_m \leftarrow \frac{G^m - \text{mean}(G^{m-l+1:m})}{\text{std}(G^{m-l+1:m})}$. To fulfill requirement 2), we make the transformation: $r_m \leftarrow r_m / \sum_{l'=1}^{l} \mathbb{1}[x_{l'}^1 = 0]$ if $r_m \geq 0$, and $r_m \leftarrow r_m / \sum_{l'=1}^{l} \mathbb{1}[x_{l'}^1 = 1]$ otherwise. The final bandit reward for round m is then mapped to $[-1, 1]$ by $r_m \leftarrow 2\sigma(r_m) - 1$, where σ is the logistic function. The low-level bandit only considers the performance of the learned policy, so the reward low-level is designed as the episodic reward normalized by a number of latest episodes, before being further normalized by the logistic function.

The reward function we design above is in nature non-stationary, since the agents continuously updating their policies, which directly influence the values of the episodic reward difference. Here we choose the adversarial bandit algorithm Exponentially Weighted Average Forecasting (Cesa-Bianchi & Lugosi, 2006) to learn the bi-level bandit.

5 Experiments

Our experiments aim to answer the following questions in Sections 5.1-5.3, respectively:

- How communication-efficient is our algorithm proposed in Section 4 against baselines and ablations?
- How empirically effective is policy consensus? Specifically, compared with not using policy consensus, can policy consensus converge to better joint policies faster?
- What are the qualitative properties of the learned communication rules?

Environments. We evaluate our model on three tasks in Multi-Agent Particle Environment (MPE) with the efficient implementation by Liu et al. (2020), each of which has a version with $N = 15$ agents and another with $N = 30$ agents. As described in Section 3.1, these MPE environments can be cast as homogeneous MGs provided full observability and the permutation preserving observation functions. We set the communication range to be $k = 10 < N$ nearest agents for all the environments to introduce partial observability. The details of the observation functions in each environment are as follows. *Cooperative Navigation:* There are 15, 30 landmarks for $N = 15, 30$ respectively. The observation of an agent contains its own absolute location, the relative locations of the nearest 10 agents, and the relative location of the 11 nearest landmarks. *Cooperative Push:* N cooperating agents are tasked to push a large ball to a target position. There are $2, 2$ landmarks for $N = 15, 30$ respectively. The observation of an agent contains its own absolute location, the relative locations of the nearest 10 agents, and the relative location of the 2 landmarks. *Predator-and-Prey:* N cooperating predators (agents) are tasked to capture p preys. The preys are pre-trained and controlled by the environment. There are $5, 10$ preys and $5, 10$ landmarks (blocks) for $N = 15, 30$ respectively. Each predator can see $p = 3, 5$ preys, $l = 3, 5$ landmarks, $k = 10, 10$ other predators for $N = 15, 30$ respectively. The observation of an agent contains its own absolute location, the relative locations of the nearest k agents (predators), the nearest p preys, and the relative location of the l nearest landmark.

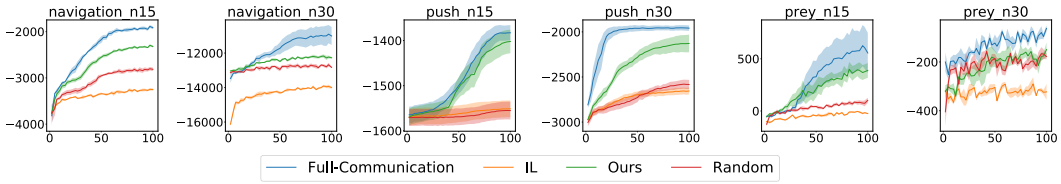

Figure 1: Y-axis: episodic reward. X-axis: training step (1e4). The comparison of our model, which has sparse communication in both observation-action communications and parameter consensus, with baselines full, random, zero communications.

Baselines. We use Permutation Invariant Critic (Liu et al., 2020), the state-of-the-art CTDE actor-critic algorithm on MPE, as our centralized training (**CT**) algorithm to derive the following decentralized training baselines. **Full-Communication** employs all-to-all communication for both observation-action and parameters, i.e., at each time step, each agent receives the observations and actions from all other neighbors for its critic, as well as critic and policy parameters from all other neighbors for its consensus update. Independent learning (**IL**) employs no communication, i.e., each agent uses its local observations and actions only for its critic and performs no consensus update. **Random** selects a subset of neighbors for communication at random. **Rule-based** parameter consensus has a fixed probability (same as the one learned by the bi-level bandit) of non-communication. Each agent saves a copy of the parameters from the latest communications with other agents. When agent i considers doing communication at t, it calculates the $l1$ norm of the parameters of Q^i and $\hat{Q}^j$ which is the latest copy of j's parameters it saves. The norm serves as the score to rank the other agents. Intuitively, the high parameter difference implies dramatic behavioral differences. The agent j with the highest score is selected to do parameter consensus, and the copy of agent j's critic parameter is recorded by agent i. For fair comparisons, the Random and Rule-based baselines incur the same communication cost (i.e., the fraction of neighbors for observation-action communication, and the frequency for parameter consensus) as learned by our algorithm.

5.1 Communication Efficiency Against Baselines and Ablations

Figure 1 shows the learning curves comparing our communication-efficient decentralized actor-critic algorithm described in Section 4 against the aforementioned baselines, with the observation-action communication rate set as 50%. The results clearly verify the importance of observation-action and parameter communication, as Full-Communication outperforms IL by significant margins uniformly in all the tasks. As will be confirmed in Section 5.3, our algorithm complies with the given observation-action communication rate, and its bandit learning chooses to communicate with a neighbor for parameter consensus only roughly 95% less frequently than Full-Communication. Our algorithm significantly outperforms the Random baseline that uses the same amount of communication in almost all the tasks, and achieves performances comparable with Full-Communication

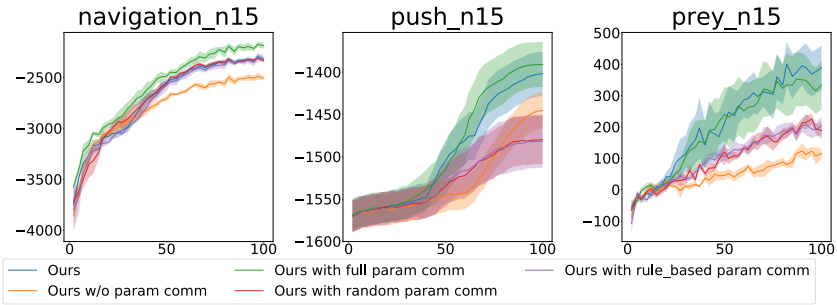

Figure 2: Y-axis: episodic reward. X-axis: training step (1e4). The comparison of parameter consensus strategy learned by our bandit with random parameter consensus and Rule-based parameter consensus.

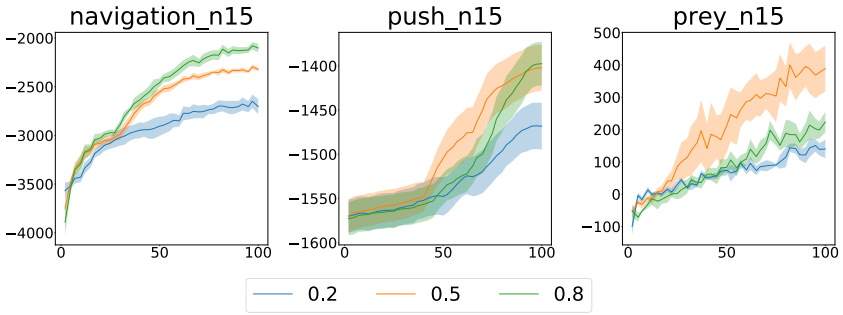

Figure 3: Y-axis: episodic reward. X-axis: training step (1e4). Model learned with different η for the sparsity regularization of the observation-action communication.

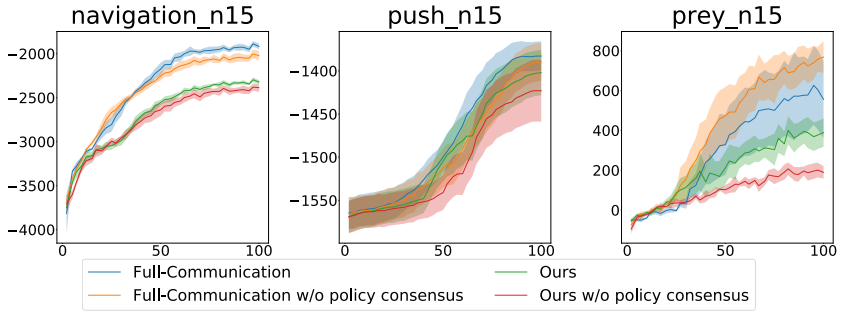

Figure 4: Y-axis: episodic reward. X-axis: training step (1e4). The comparison of model learned with policy consensus with the one learned w/o policy consensus.

in several tasks. Remarkably, Random barely outperforms IL in Cooperative Push, suggesting that efficient communication for decentralized training is challenging.

Ablation: parameter communication. We perform ablations on the three environments with $N = 15$ agents. Fix the observation-action communication pattern to be learned by the proposed communication network, we can see in Figure 2 that parameter consensus strategy learned by our bi-level bandit outperforms the Random parameter consensus baseline, the Rule-based parameter consensus strategy, and even the parameter consensus using Full-Communication in Cooperative Push and Predator-and-Prey, and behaves comparably to random parameter consensus and Rule-based parameter consensus in Cooperative Navigation. Noticeably, the Rule-based algorithm behaves similar to the Random parameter consensus baseline in the three scenarios. A possible explanation is that the parameters of the other agents an agent records are outdated, as the Rule-based algorithm uses the same communication frequency (around 95%) learned by the bi-level bandit model. Another explanation is that parameter consensus harms exploration, and the balance of them cannot be handled by the Rule-based algorithm which only considers marinating homogeneous behaviors between agents.

Ablation: observation-action communication. Figure 3 shows that increasing communications is not always helpful. In particular, fixing our communication network as the communication pattern, the model learned with 20% observation-action communication rate performs the worst among the three environments. The one learned with 80% observation-action communication rate performs the best in Cooperative Navigation, performs similarly to the one learned with 50% observation-action communication rate in Cooperative Push, and even performs worse in Predator-and-Prey than the one learned with learned with 50% observation-action communication rate.

5.2 Effectiveness of policy consensus

Theorem 2 assures that policy consensus in distributed actor-critic methods for Homogeneous MGs provably converges to a local optimum, yet it remains open questions whether such a local is good

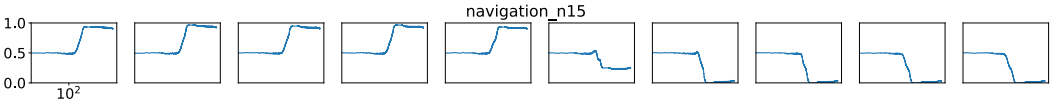

Figure 5: Y-axis: Avg communication rate. X-axis: training step in log scale. The average communication rate for the 10 neighbors in Cooperative Navigation ($N = 15$), with the order increasing in distance from left to right.

and whether the convergence is fast. Our experiments in this subsection empirically investigate these questions by comparing the learning curves with and without policy parameter consensus. We separately consider 1) the setting where the agents employ all-to-all communication for their critics and parameter consensus, such that the assumptions of full observability and well-connected consensus matrices in Theorem 2 are satisfied, and 2) the setting where the agents learn to communicate efficiently with our communication network and the bandit. The results in Figure 4 show that policy parameter consensus is always beneficial with our communication-efficient method and, surprisingly, it can negatively impact the training with all-to-all communication (e.g., Predator-and-Prey). One plausible explanation is that, while it might speed up convergence, policy parameter consensus can harm exploration, leading to worse local optima.

5.3 Qualitative analyses of the learned communication rule

We first qualitatively examine the observation-action communication rule learned by our communication network. We plot the output of our communication network, i.e., the probabilities of communicating with the $k = 10$ distance-sorted neighbors, as the learning progresses. Interestingly, the rule learned by our communication network encourages communicating with the nearest neighbors. For example, in Cooperative Navigation with the 50% communication rate as shown in Figure 5, the probabilities of agents communicating with nearest 5 neighbors are over 75%, around 25% for the 6th nearest agent, around 0% for the other neighbors. We provide the counterparts of Figure 5 for all the environments in the appendix.

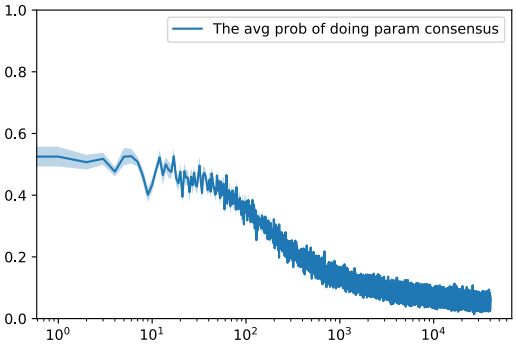

Figure 6: Y-axis: probability of no parameter consensus. X-axis: training step in log scale.

Figure 6 shows the probability of choosing the high-level arm of not performing consensus as the learning progresses, averaged across the runs on all the environments. The result shows that, with the bandit's reward function designed to reduce parameter consensus frequency, the probability of selecting not to communicate increases from around 50% to more than 90%.

6 Conclusion

In this paper, we characterize a subclass of cooperative Markov games where the agents exhibit a certain form of homogeneity such that policy sharing provably incurs no loss of optimality. We develop the first multi-agent actor-critic algorithm for homogeneous MGs that enjoys asymptotic convergence guarantee with decentralized policy consensus. For practical usage, we propose techniques that can efficiently learn to communicate with other agents in exchange of observations, actions, and parameters. The empirical results show that our proposed model performs better than several baselines in terms of communication efficiency.

REFERENCES

Stephen Boyd, Arpita Ghosh, Balaji Prabhakar, and Devavrat Shah. Randomized gossip algorithms. *IEEE transactions on information theory*, 52(6):2508–2530, 2006.

Duncan S Callaway and Ian A Hiskens. Achieving controllability of electric loads. *Proceedings of the IEEE*, 99(1):184–199, 2010.

Nicolò Cesa-Bianchi and Gábor Lugosi. *Prediction, Learning, and Games*. 01 2006. ISBN 978-0-521-84108-5. doi: 10.1017/CBO9780511546921.

Tianshu Chu, Jie Wang, Lara Codecà, and Zhaojian Li. Multi-agent deep reinforcement learning for large-scale traffic signal control. *IEEE Transactions on Intelligent Transportation Systems*, 21(3): 1086–1095, 2019.

Peter Corke, Ron Peterson, and Daniela Rus. Networked robots: Flying robot navigation using a sensor net. In *Robotics research. The eleventh international symposium*, pp. 234–243. Springer, 2005.

Jakob Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. Counterfactual multi-agent policy gradients. *arXiv preprint arXiv:1705.08926*, 2017.

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

Jakub Grudzien Kuba, Ruiqing Chen, Munning Wen, Ying Wen, Fanglei Sun, Jun Wang, and Yaodong Yang. Trust region policy optimisation in multi-agent reinforcement learning. *arXiv preprint arXiv:2109.11251*, 2021.

Iou-Jen Liu, Raymond A Yeh, and Alexander G Schwing. Pic: permutation invariant critic for multi-agent deep reinforcement learning. In *Conference on Robot Learning*, pp. 590–602. PMLR, 2020.

Ryan Lowe, Yi I Wu, Aviv Tamar, Jean Harb, OpenAI Pieter Abbeel, and Igor Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. In *Advances in neural information processing systems*, pp. 6379–6390, 2017.

Peng Peng, Ying Wen, Yaodong Yang, Quan Yuan, Zhenkun Tang, Haitao Long, and Jun Wang. Multiagent bidirectionally-coordinated nets: Emergence of human-level coordination in learning to play starcraft combat games. *arXiv preprint arXiv:1703.10069*, 2017.

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, and Rob Fergus. Learning multiagent communication with backpropagation. *arXiv preprint arXiv:1605.07736*, 2016.

Ming Tan. Multi-agent reinforcement learning: Independent vs. cooperative agents. In *In Proceedings of the Tenth International Conference on Machine Learning*, pp. 330–337. Morgan Kaufmann, 1993.

Kaiqing Zhang, Zhuoran Yang, Han Liu, Tong Zhang, and Tamer Basar. Fully decentralized multi-agent reinforcement learning with networked agents. In *International Conference on Machine Learning*, pp. 5872–5881. PMLR, 2018.

Sai Qian Zhang, Qi Zhang, and Jieyu Lin. Efficient communication in multi-agent reinforcement learning via variance based control. *arXiv preprint arXiv:1909.02682*, 2019.

Yan Zhang and Michael M Zavlanos. Distributed off-policy actor-critic reinforcement learning with policy consensus. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pp. 4674–4679. IEEE, 2019.