Mitigating Variance Caused by Communication in Multi-agent Deep Reinforcement Learning

Changxi Zhu
Utrecht University
Utrecht, Netherlands
c.zhu@uu.nl

Mehdi Dastani
Utrecht University
Utrecht, Netherlands
m.m.dastani@uu.nl

Shihan Wang
Utrecht University
Utrecht, Netherlands
s.wang2@uu.nl

ABSTRACT

Communication can facilitate agents to gain a better understanding of the environment and to coordinate their behaviors in multi-agent deep reinforcement learning (MADRL). However, in certain applications, communication is not available during execution due to factors such as security concerns or limited resources. This paper focuses on a MADRL setting where communication is used only during training, but not during execution, enabling the learning of coordinated behaviors while keeping decentralized execution. As communication may introduce uncertainty, we conduct the first theoretical analysis to study the variance caused by communication in policy gradients using actor-critic methods. Motivated by this analysis, we propose modular techniques to reduce the variance in policy gradients caused by communication. We incorporate these techniques into two existing MADRL with communication methods and evaluate them on multiple tasks in three benchmark environments. The results demonstrate that MADRL with communication methods extended with our techniques not only achieve high-performing agents but also reduce variance in policy gradients.

KEYWORDS

Variance Reduction, Multi-agent Reinforcement Learning, Communication

1 INTRODUCTION

Numerous real-world scenarios involve multiple agents interacting within a shared environment, spanning domains like autonomous driving [32], robotics [17], and game playing [1, 33]. Multi-agent Deep Reinforcement Learning (MADRL) has been widely used to develop cooperative behaviors of agents in partially observable environments [13, 28, 42]. MADRL agents can communicate with one another to share information, thereby broadening their perception of the environment and improving the coordination of their behaviors [13, 44, 46]. In recent years, there has been growing research interest in MADRL focusing on learnable communication protocols, in which agents exchange encoded messages rather than sharing massive local information [46]. These research works are known as MADRL with learning communication (Comm-MADRL), which aims to learn adaptive and flexible communication among agents. Within the Comm-MADRL field, several settings have been utilized, focusing on whether agents are trained in a decentralized or centralized manner, and whether communication is possible during training or policy execution [13, 46].

In practical applications such as UAVs, concerns about security or limited resources often necessitate that agents operate independently without communication, yet in a coordinated manner [2, 35]. In such applications, achieving decentralized and coordinated behavior among agents without information sharing during execution becomes essential. To support such applications, communication can be introduced only during training, ensuring and enhancing learning coordinated behaviors efficiently and effectively. Previous work has explored the use of actor-critic methods and proposed to incorporate communication into individual critics but not actors [15, 20]. In these research works, each critic has access only to local information (e.g., an agent's own history and actions), augmented with encoded low-dimensional messages from other agents, enabling information sharing while maintaining efficient and secure policy learning. During execution, the critics can be discarded, and only the actors are used. In the rest of this paper, we build on these communication methods and coin the term Communicating Critics and Decentralized Actors (CCDA) to refer to the settings where individual critics communicate during training while actors cannot communicate neither during training nor during execution.

In CCDA, critics exchange and leverage low-dimensional messages, benefiting from improved computational efficiency due to the reduced input dimensionality compared to centralized critics that rely on complete global information. Despite the promising applications of CCDA, communication can introduce challenges when incorporated into the critics. Specifically, communicated messages are often generated in a stochastic manner [9, 16] such that, from the perspective of receiver agents, using messages as inputs to their critics introduces uncertainty during value estimation. As a result, the policy gradients of actors guided by communicating critics may exhibit high variance, leading to low sample efficiency and performance degradation. Despite this, previous research has focused on variance analysis in policy gradients without communication [22, 23], and thus not measuring variance caused by communication.

In this work, we conduct the first theoretical analysis of the variance in policy gradients within Comm-MADRL under the CCDA setting. Variance analysis is a vigorous method that allows us to investigate the variability and dispersion of policy learning. Through our variance analysis, we prove that in both idealistic communication setting (where critics communicate sound & complete information) and non-idealistic communication setting (where sound & complete information is corrupted with noise), policy gradients under communicating critics (in the CCDA setting) have equal or higher variance than that of the centralized critic. Our theoretical analysis motivates us to reduce the variance in policy gradients using communicating critics under CCDA. A widely used approach for variance reduction

Proc.@ of the Adaptive and Learning Agents Workshop (ALA 2026), Aydeniz, Delgrange, Mohammedalamen, Yang (eds.), May 25 – 26, 2026, Paphos, Cyprus, https://alaworkshop2026.github.io/. 2026.

Table 1: Comparison of critics and actors, where the on-policy critics Q^π (and Q_i^π) and the actor π_i of agent i may condition on agents' histories (h_i), actions (a_i), and received messages (m_{-i}).

Paradigms	Critics	Actors
Setting 1	$Q^\pi(h_1, \ldots, h_N, a_1, \ldots, a_N)$	$\pi_i(a_i \mid h_i)$
Setting 2	$Q^\pi(h_1, \ldots, h_N, a_1, \ldots, a_N)$	$\pi_i(a_i \mid h_i, m_{-i})$
Setting 3	$Q_i^\pi(h_i, a_i)$	$\pi_i(a_i \mid h_i)$
Setting 4	$Q_i^\pi(h_i, a_i, m_{-i})$	$\pi_i(a_i, \mid h_i, m_{-i})$
Setting 5	$Q_i^\pi(h_i, a_i, m_{-i})$	$\pi_i(a_i \mid h_i)$

is the *baseline* technique [12, 18, 39, 41]. Existing baseline techniques are designed to reduce variance arising from states or actions, while do not account for the variance induced by communication among agents. Moreover, applying existing baseline techniques to MADRL with communication methods is not straightforward, which may not optimally reduce the variance caused by communication.

In this paper, we propose a message-dependent baseline technique that mitigates the variance caused by communicated messages. We derive the optimal form of our proposed baseline and theoretically prove that it reduces the variance in policy gradients. In practice, we seek to estimate the proposed baseline based on communicating critics and the policy distributions of actors. To improve the estimation of the baseline, we propose to use KL divergence as a regularization technique that encourages the actors to remain as close as possible to the communicating critics. Our proposed variance reduction techniques can be applied to any Comm-MADRL method under CCDA. To show the effectiveness and efficiency of the proposed techniques, we extend two existing MADRL methods under the CCDA setting and evaluate the two methods with and without our techniques on multiple tasks in three benchmark environments, StarCraftII Multi-Agent Challenge (SMAC) [30], Traffic Junction [34], and Predator-Prey [34]. The results show that our proposed techniques can reduce the variance in policy gradients caused by communication under CCDA and improve learning performance.

2 RELATED WORKS

To position our focused CCDA setting within the broader literature on MADRL, we distinguish various settings, with and without learning communication, across training and execution phases. In Table 1, we have summarized 5 settings in the MADRL literature: (**Setting** 1) Centralized Training and Decentralized Execution (CTDE) without learning communication, (**Setting** 2) CTDE with communicating actors, (**Setting** 3) Decentralized Training and Decentralized Execution (DTDE), (**Setting** 4) Communicating Critics and Communicating Actors (CCCA), and (**Setting** 5) CCDA where only critics communicate. We also present an graphic view of these settings in Appendix 4. Among them, Settings 1 and 2 use predefined full information during centralized training. Settings 2 and 4 require communicating actors during both training and execution, which may not satisfy practical requirements of security and limited resource as motivated in the introduction. Settings 1 and 3 are different from other settings as communication is not learned. Setting 5 is the CCDA, where communication is learned and utilized in the critics but not in the actors. CCDA can become comparable to Setting 1 when the critics in CCDA share full information deterministically, without any stochasticity in their messages. However, the key difference between CCDA and Setting 1 is that, in CCDA, critics learn what, when, and with whom to communicate low-dimensional messages during training, whereas this is not the case in Setting 1. Due to the similarity between CDDA and Setting 1, we compare CCDA with Setting 1 in our theoretical analysis to show that they have different variance properties.

Learning Communication in MADRL. Previous works mainly focus on learning efficient and effective communication to improve learning performance under either CTDE with communicating actors [6] (**Setting** 2), CCCA where both critics and actors communicate [4, 25] (**Setting** 4), and CCDA where only critics communicate [15, 20] (**Setting** 5). In CTDE with communicating actors, existing research works utilize either a shared Q-function [16, 29] or a joint value function [14, 36] for the training of communicating actors. Moreover, encoded messages are often viewed as additional inputs for policies, such as CommNet [36], ATOC [16], TarMAC [6], I2C [7], GACML [24], CACOM [19], RGMComm [4], and Commformer [14]. These methods require explicit message transmission among agents during both training and execution, and assume access to global information for critics. In contrast, the CCDA setting (**Setting** 5) do not involve message exchange among actors, and the critics rely only on local information and encoded messages.

Compared to CTDE with communicating actors, learning communication among individual critics is under-explored. The existing works mainly rely on actor-critic methods [4, 15, 20, 25] (**Settings** 4 **and** 5). When communication is allowed only among critics not actors (i.e., **the CCDA setting**), learning communication relies on MAAC [15] and its latest variant GAAC [20]. In MAAC, each agent's individual critic considers not only its local information but also incorporates and aggregates information from other agents. Based on MAAC, GAAC incorporates a communication graph in the critics to decide when and how to communicate with other agents. In the specific case where agents communicate full information deterministically rather than learning to communicate in a stochastic manner, MADDPG [21] enables each agent to employ an individual but global Q-function as a critic. When communication is learned and used in both critics and actors, Setting 4 is adopted. In this setting, MAGIC [25] proposes to share neural networks for critics and actors, and learn how to schedule and encode messages for both actors and critics. In contrast, CCDA methods (e.g., GAAC) do not involve message sharing among actors, allowing decentralized execution without communication while still benefiting from communication during training.

In addition to learning communication in MADRL, we also notice research works considering predefined communication where agents share experience to enhance training while not using in execution [5, 11]. In comparison, CCDA methods explicitly learns what, when, and how to communicate. As none of the above studies address the issue of high variance arising when communication is learned and integrated into policy gradients, our theoretical analysis provides a first step toward understanding this challenge, focusing on the CCDA setting to offer key insights.

Variance Analysis in MADRL. Variance reduction is an essential topic in MADRL [18, 37]. Previous works have built a theoretical

analysis of the variance in policy gradients without communication. Lyu et al. [22, 23] theoretically contrast policy gradients under CTDE and DTDE settings and claim that the uncertainty of other agents' observations and actions appeared in centralized Q-functions can increase the variance in policy gradients [1]. In contrast, we study variance in policy gradients considering communication.

One of the most successfully applied and extensively studied methods to reduce variance is known as the *baseline* technique [10, 18, 41]. Concretely, Wu et al. [41] utilizes an action-dependent baseline to eliminate the influence of the other agents' policies. Foerster et al. [10] introduces a counterfactual baseline that marginalizes out a single agent's action, while keeping the other agents' actions fixed. More recently, Jakub et al. [18] mathematically analyze the variance of policy gradients under CTDE and quantify how agents contribute to the total variance. They propose a baseline technique to achieve minimal variance when estimating policy gradients under CTDE. In summary, existing baseline techniques consider the source of variance from the uncertainty in other agents' observations or actions, while our baseline technique considers the source of variance from the uncertainty in generated messages.

3 PRELIMINARIES

Multi-Agent Reinforcement Learning. Cooperative multi-agent tasks are often modeled as decentralized partially observable Markov decision processes (Dec-POMDPs) [26]. A Dec-POMDP is defined by a tuple $\left\langle \mathcal{I}, \mathcal{S}, \rho^0, \{\mathcal{A}_i\}, P, \{O_i\}, O, \mathcal{R}, \gamma \right\rangle$, where $\mathcal{I}$ is a set of agents indexed as $\{1, ..., N\}$, $\mathcal{S}$ is a set of environment states, ρ^0 is the state distribution, $\mathcal{A}_i$ is a set of actions of agent i, and $\boldsymbol{O}_i$ is a set of observations of agent i. Then, transition function $P : \mathcal{S} \times \boldsymbol{\mathcal{A}} \rightarrow \Delta(\mathcal{S})$ specifies the transition probability $p(s'|s, \boldsymbol{a})$ from state $s \in \mathcal{S}$ to new state $s' \in \mathcal{S}$ given joint action $\boldsymbol{a} = \langle a_1, ..., a_N \rangle$ and $\boldsymbol{a} \in \boldsymbol{\mathcal{A}} = \times_{i \in \mathcal{I}} \mathcal{A}_i$. With the environment transitioning to new state s', given joint action $\boldsymbol{a}$, the probability of a joint observation $\boldsymbol{o} = \langle o_1, ..., o_N \rangle$ is determined according to the observation function $O : \mathcal{S} \times \boldsymbol{\mathcal{A}} \rightarrow \Delta(\boldsymbol{O})$, where $\boldsymbol{o} \in \boldsymbol{O} = \times_{i \in \mathcal{I}} \boldsymbol{O}_i$. Each agent then receives a shared reward $r = \mathcal{R}(s, \boldsymbol{a})$ according to the reward function $\mathcal{R} : \mathcal{S} \times \boldsymbol{\mathcal{A}} \rightarrow \mathbb{R}$, which is discounted by γ over time steps. The joint policy $\boldsymbol{\pi}$ of agents induces an on-policy joint Q-function: $Q^{\boldsymbol{\pi}}(s, \boldsymbol{a}) = \mathbb{E}_{s_t \sim P, \boldsymbol{a}_t \sim \boldsymbol{\pi}}[\sum_{t=0}^{T} \gamma^t r_t | s_0 = s, \boldsymbol{a}_0 = \boldsymbol{a}]$, which is the expected discounted return within the time horizon T. When the state s is not observable, the joint history $\boldsymbol{h} = \{h_1, ..., h_N\}$ is used, where $h_i = (o_0^i, a_0^i, ..., o_t^i)$ is the individual observation-action history of agent i up to time step t. Therefore, we obtain the history-based joint Q-function $Q^{\boldsymbol{\pi}}(\boldsymbol{h}, \boldsymbol{a})$ [23], which can be implemented by LSTM neural networks [27]. For notational readability, we omit the time step t.

Policy Gradients under CTDE and CCDA. Policy gradient methods under CTDE, e.g., MAPPO [43], use a centralized and joint Q-function to guide the learning of decentralized policies. Following the setting of [22], the CTDE policy gradient of agent i is defined as:

$$g^i_{CTDE} \doteq \mathbb{E}_{\boldsymbol{h}, \boldsymbol{a}}[Q^{\boldsymbol{\pi}}(\boldsymbol{h}, \boldsymbol{a}) \nabla_{\theta_i} \log \pi_i(a_i|h_i, \theta_i)]$$

where $Q^{\boldsymbol{\pi}}(\boldsymbol{h}, \boldsymbol{a})$ is the on-policy joint values and θ_i is the parameters of policy π_i. We further use $\hat{g}^i_{CTDE}$ to denote the (single-sample) estimate of g^i_{CTDE}, i.e., $\hat{g}^i_{CTDE} = Q^{\boldsymbol{\pi}}(\boldsymbol{h}, \boldsymbol{a}) \nabla_{\theta_i} \log \pi_i(a_i|h_i, \theta_i)$. Agent i then utilizes g^i_{CTDE} to update parameter θ_i.

[1] We use CTDE to refer to CTDE without learning communication.

Similar to the policy gradients under CTDE, we formulate the policy gradients under CCDA based on the literature [15, 20]. Essentially, we define messages m_i as being generated from a probabilistic message function based on each agent's history (h_i) and actions from the actor (a_i): $m_i \sim f^{msg}(\cdot|h_i, a_i, \theta^{msg})$ with parameters θ^{msg}. To simplify the theoretical analysis and focus on the effect of received messages on policy gradients, we consider broadcast communication and denote the received messages of agent i from all the other agents $-i$ as $m_{-i} = \{m_1, ..., m_{i-1}, m_{i+1}, ..., m_N\}$. The CCDA policy gradient of receiver agent i given by on-policy values is defined as follows:

$$g^i_{CCDA} \doteq \mathbb{E}_{\boldsymbol{h}, \boldsymbol{a}, \boldsymbol{m}}[Q^{\pi}_i(h_i, a_i, m_{-i}) \nabla_{\theta_i} \log \pi_i(a_i|h_i, \theta_i)]$$

where $Q^{\pi}_i(h_i, a_i, m_{-i})$ is the on-policy Q-values of agent i. Similarly, $\hat{g}^i_{CCDA}$ denotes the (single-sample) estimate of g^i_{CCDA}, i.e., $\hat{g}^i_{CCDA} = Q^{\pi}_i(h_i, a_i, m_{-i}) \nabla_{\theta_i} \log \pi_i(a_i|h_i, \theta_i)$.

Learning Communication in MADRL. Different CCDA methods may adopt different strategies for deciding when and what to communicate and how to learn communication. In GAAC [20], the messages m_{-i} received by agent i are generated by an embedding layer $\mathcal{E}$ that encodes the history h_j and current actions a_j of each sender agent j and then aggregated through an attention mechanism: $m_{-i} \sim f^{msg}_{-i}(\cdot \mid h_{-i}, a_{-i}, \theta^{msg}_{-i}) = \sum_{j \in -i} c_{ij}(\epsilon)\, \mathcal{E}(h_j, a_j)$, where $c_{ij}(\epsilon) \in \{0, 1\}$ denotes hard attention weights generated by the Gumbel-Softmax with noise variable ϵ for deciding whether to communicate. f^{msg}_{-i} denotes the joint message function of all sender agents $-i$ with parameters θ^{msg}_{-i}. Due to the differentiable Gumbel-Softmax, the parameters θ^{msg}_{-i} are trained end-to-end. Beyond end-to-end training, the literature in Comm-MADRL [6, 16] also adopts policy gradients to optimize communication. Following this line of work, we incorporate communication into the critic using broadcast communication. Specifically, we define the policy gradient for updating the message function f^{msg}_i of each sender agent i as $g^i_{msg} = \mathbb{E}_{\boldsymbol{h}, \boldsymbol{a}, \boldsymbol{m}}[\nabla_{\theta^{msg}_i} \log f^{msg}_i(m_i \mid h_i, a_i, \theta^{msg}_i)\, Q^{\pi}_j(h_j, a_j, m_{-j})]$, where $Q^{\pi}_j(h_j, a_j, m_{-j})$ denotes the Q-value of the receiver agent $j \neq i$ when fixing the other agents' messages.

4 METHODS

We are interested in how communication affects the policy updates of receiver agents. We specifically focus on how policy gradients diverge, in terms of the variance measurement. Inspired by previous variance analysis under CTDE (without learning communication) setting [18, 22, 23], we conduct a variance analysis in CCDA policy gradients, focusing on the variance induced by communication. In our variance analysis, we consider both idealistic communication and non-idealistic communication settings. In both scenarios, we prove that the variance of CCDA policy gradients can be equal to or higher than that of CTDE policy gradients. Motivated by the variance analysis, we further propose variance reduction techniques.

4.1 Variance Analysis

Idealistic Communication Setting. We first consider an idealistic communication setting by assuming the existence of a perfect message decoder. Under such idealistic scenarios, the communicating critics $Q^{\pi}_i(h_i, a_i, m_{-i})$ and the centralized critics $Q^{\pi}(\boldsymbol{h}, \boldsymbol{a})$ become identical because in both cases the complete and sound information are shared among agents. However, the probabilistic

nature of messages (as commonly used by MADRL with communication methods) can lead to variance in policy gradient samples. Hence, we come to the following theorem:

THEOREM 1. *The CCDA sample gradient has a variance greater or equal than that of the CTDE sample gradient in idealistic communication setting:* $Var(\hat{g}^i_{CCDA}) \geq Var(\hat{g}^i_{CTDE})$.

Proof Sketch (full proof in Appendix 1). We leverage the Bellman equation to find the equivalence between $Q^\pi_i(h_i, a_i, m_{-i})$, used as critics in $\hat{g}^i_{CCDA}$, and $Q^\pi(\boldsymbol{h}, \boldsymbol{a})$, used as critics in $\hat{g}^i_{CTDE}$. Essentially, as $Q^\pi(\boldsymbol{h}, \boldsymbol{a})$ and the expected value of $Q^\pi_i(h_i, a_i, m_{-i})$ over messages converge to the same fixed point, we get: $Q^\pi(\boldsymbol{h}, \boldsymbol{a}) = \mathbb{E}_{m_{-i}|\boldsymbol{h},\boldsymbol{a}}[Q^\pi_i(h_i, a_i, m_{-i})]$. Based on this, we find that $\hat{g}^i_{CCDA}$ and $\hat{g}^i_{CTDE}$ are equal in expectation such that the difference between $Var(\hat{g}^i_{CCDA})$ and $Var(\hat{g}^i_{CTDE})$ ends with an expectation of the square of gradients minus the square of the expectation of gradients. According to Jensen's inequality, we conclude that $Var(\hat{g}^i_{CCDA})$ is equal to or higher than $Var(\hat{g}^i_{CTDE})$.

Non-idealistic Communication Setting. We further consider the variance analysis under a non-idealistic communication setting, where messages received by each agent i can be corrupted by noise ϵ_i. The noise term can come from the imperfection of decoders, e.g., due to the use of neural networks. To simplify the analysis, we lift noise in received messages to Q-values, where $m_{-i} =< h_{-i}, a_{-i}, \epsilon_i >$ for agent i, leading to $Q^\pi_i(h_i, a_i, m_{-i}) = Q^\pi_i(h_i, a_i, < h_{-i}, a_{-i}, \epsilon_i >) = Q^\pi_i(\boldsymbol{h}, \boldsymbol{a}, \epsilon_i)$. The individual but joint Q-function with additive noise, $Q^\pi_i(\boldsymbol{h}, \boldsymbol{a}, \epsilon_i)$, is used for decentralized actors, forming a noise version of CCDA policy gradients $\hat{g}^i_{CCDA-noise}$. Inspired by Wang et al. [38], the noise term can affect the rewards of each agent, such as flipping the sign when the reward is binary. We then prove that removing the effect of the noise term (thereby becomes unbiased) can still increase variance, resulting in the following theorem:

THEOREM 2. *The noisy version of CCDA sample gradient has a variance greater or equal than that of the CTDE sample gradient in non-idealistic communication setting:* $Var(\hat{g}^i_{CCDA-noise}) \geq Var(\hat{g}^i_{CTDE})$.

Proof Sketch (full proof in Appendix 2). We first relate the noise term with the probability of changes in rewards. Inspired by Wang et al. [38], a surrogate reward function can be defined to remove the effect of noise in rewards. Based on the surrogate reward function, we define a surrogate Q-function $\hat{Q}^\pi_i(\boldsymbol{h}, \boldsymbol{a}, \epsilon_i)$. By summing up noisy terms ϵ_i, the expected value of $\hat{Q}^\pi_i(\boldsymbol{h}, \boldsymbol{a}, \epsilon_i)$ is shown to be equal to the centralized Q-function $Q^\pi(\boldsymbol{h}, \boldsymbol{a})$ (defined on noise-free and shared rewards in Dec-POMDP), i.e., $Q^\pi(\boldsymbol{h}, \boldsymbol{a}) = \mathbb{E}_{\epsilon_i}[\hat{Q}^\pi_i(\boldsymbol{h}, \boldsymbol{a}, \epsilon_i)]$ by induction proof. The equality greatly simplifies the variance analysis between the noise version of CCDA policy gradients $\hat{g}^i_{CCDA-noise}$ (using $\hat{Q}^\pi_i(\boldsymbol{h}, \boldsymbol{a}, \epsilon_i)$ as critics) and the CTDE policy gradients $\hat{g}^i_{CTDE}$ (using $Q^\pi(\boldsymbol{h}, \boldsymbol{a})$ as critics). By comparing the variance in gradients, we have $Var(\hat{g}^i_{CCDA-noise}) \geq Var(\hat{g}^i_{CTDE})$.

4.2 Variance Reduction Techniques

Motivated by our variance analysis, we propose a novel message-dependent baseline and derive its optimal formulation to minimize the variance induced by communication. We further estimate the proposed baseline and introduce a KL divergence term to improve baseline estimation. The proposed message-dependent baseline and the KL divergence together form our modular techniques, which will be integrated into existing communication methods under the CCDA setting.

We first write out CCDA policy gradients with the message-dependent baseline (denoted as CCDA-OB) as follows:

$$g^i_{CCDA-OB} = \mathbb{E}_{\boldsymbol{h},\boldsymbol{a},\boldsymbol{m}}[(Q_i(h_i, a_i, m_{-i}) - b_i(h_i, m_{-i}))\nabla_{\theta_i} \log \pi_i(a_i|h_i, \theta_i)] \tag{1}$$

where actions are sampled from decentralized policies $\pi_i(\cdot|h_i)$ in practice, and we denote $\hat{g}^i_{CCDA-OB}$ as the estimate of $g^i_{CCDA-OB}$. In Equation 1, we use Q-function $Q_i(h_i, a_i, m_{-i})$ to describe the samples of return of agent i using communication, where message m_{-i} can be either noisy or noise-free. We assume that $Q_i(h_i, a_i, m_{-i})$ can converge to the true on-policy values $Q^\pi_i(h_i, a_i, m_{-i})$. We seek the optimal message-dependent baseline $b^*_i(h_i, m_{-i})$ to minimize the variance $Var(\hat{g}^i_{CCDA-OB})$ of the policy gradient estimate $\hat{g}^i_{CCDA-OB}$. Therefore, we come to the following theorem:

THEOREM 3. *The optimal message-dependent baseline for CCDA-OB gradient estimator is:*

$$b^*_i(h_i, m_{-i}) = \frac{\mathbb{E}_{a_i}[Q_i(h_i, a_i, m_{-i})S]}{\mathbb{E}_{a_i}[S]} \tag{2}$$

where $S = \nabla_{\theta_i} \log \pi_i(a_i|h_i, \theta_i)^T \nabla_{\theta_i} \log \pi_i(a_i|h_i, \theta_i)$.

Proof Sketch (full proof in Appendix 3.1). The key idea is to determine an optimal baseline to minimize the variance of $Var(g^i_{CCDA-OB})$ by analyzing the derivatives of the variance w.r.t. the baseline. In Equation 2, S is the inner product of the gradient $\nabla_{\theta_i} \log \pi_i(a_i|h_i, \theta_i)$, indicating the magnitude of the gradient vector.

The resulting formula $b^*_i(h_i, m_{-i})$ aligns with previous works on baseline techniques considering states and actions [18, 41], while we incorporate partially observable information (histories) and communication (messages) into the Q-function. Based on Theorem 3, we have:

COROLLARY 1. *The variance of CCDA policy gradients is reduced with the optimal message-dependent baseline:* $Var(\hat{g}^i_{CCDA-OB}) \leq Var(\hat{g}^i_{CCDA})$.

Proof Sketch (full proof in Appendix 3.2). The key idea is to integrate the optimal baseline $b^*_i(h_i, m_{-i})$ into the variance $Var(\hat{g}^i_{CCDA-OB})$, which ends with $Var(\hat{g}^i_{CCDA})$ minus a non-negative term. Then, due to the non-negative term, the variance with the baseline is less than or equal to the variance without the baseline. Note that Corollary 1 holds also for non-idealistic communication setting, $Var(\hat{g}^i_{CCDA-OB}) \leq Var(\hat{g}^i_{CCDA-noise})$, by replacing messages with the noisy version $m_{-i} =< h_{-i}, a_{-i}, \epsilon_i >$ and following the same derivations.

The optimal message-dependent baseline defined in Equation 2 is determined by the communicating critics $Q_i(h_i, a_i, m_{-i})$ and the actors' policy distributions $\pi_i(a_i \mid h_i, \theta_i)$, as illustrated in Figure 1, where communication (i.e., messages m_{-i}) is incorporated into the critics but not the actors. Specifically, the communicating critic can derive a policy distribution considering communication, denoted as $\pi_i(a_i \mid h_i, m_{-i})$ in the figure. However, the action distributions of the critics and the actors can be mismatched, resulting in inaccurate baseline estimation. To improve baseline estimation, we align between the actor $\pi_i(\cdot|h_i, \theta_i)$ and the communicating critic $Q_i(h_i, a_i, m_{-i})$.

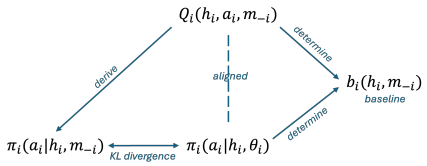

Figure 1: Communicating critics and actors in CCDA are aligned by updating the actors using gradients from the baseline and the KL divergence between the CCDA actors and the derived communicating actors.

Since only the policy $\pi_i(\cdot|h_i, \theta_i)$ is available in training and execution, we estimate the policy derived from the critic using the Boltzmann softmax distribution [3] of local Q-values. Then, we propose to minimize the following KL divergence between decentralized actors $\pi_i(\cdot|h_i, \theta_i)$ and the derived policy with communication for each agent i:

$$\begin{aligned}\mathcal{L}_{\mathrm{KL}}(\theta_i) &= D_{\mathrm{KL}}\big(\pi_i(\cdot|h_i;\theta_i) \,\big\|\, SoftMax(\beta Q_i(h_i, \cdot, m_{-i}))\big) \\ &\overset{def.}{=} \mathbb{E}_a[\log \pi_i(a|h_i;\theta_i) - \beta Q_i(h_i, a, m_{-i}) + \log \sum_{a'} \exp(\beta Q_i(h_i, a', m_{-i}))] \\ &= -\mathcal{H}(\pi_i(\cdot|h_i;\theta_i)] - \mathbb{E}_a[\beta Q_i(h_i, a, m_{-i})] + \log \sum_{a'} \exp(\beta Q_i(h_i, a', m_{-i}))\end{aligned} \tag{3}$$

where the second line follow from the definitions of the KL divergence and the softmax function. In the last line, the first term follows the definition of policy entropy $\mathcal{H}(\pi_i(\cdot|h_i;\theta_i)] = -\mathbb{E}_a[\log \pi_i(a|h_i;\theta_i)]$. The third term is a normalization term that remains constant regardless of the policy distribution. In the second term, the temperature parameter β controls the sharpness of the softmax distribution, where large values make the policy more greedy and deterministic, while small values make it more uniform, encouraging exploration. As a result, minimizing the KL divergence $\mathcal{L}_{KL}(\theta_i)$ is equivalent to maximizing the policy entropy together with the expected Q-values (scaled by β), while the normalization term can be neglected. With the KL divergence, the policy $\pi_i(a|h_i;\theta_i)$ is encouraged to maintain high entropy while being guided by communicating critics toward optimal expected behavior under communication, thereby helping to align decentralized policies $\pi_i(a_i|h_i;\theta_i)$ (without communication) with the desired policy $\pi_i(a_i \mid h_i, m_{-i})$ (using communication).

4.3 The Overall Learning Procedures

The optimal message-dependent baseline (OB) and the KL divergence term (KL) jointly constitute our proposed techniques (OB-KL) regarding the variance reduction in policy gradients. The final ascent gradient for agent i's actor is:

$$g^i_{CCDA-OB-KL} = g^i_{CCDA-OB} - \lambda \nabla_{\theta_i} \mathbb{E}_{\boldsymbol{h},\boldsymbol{m}}[\mathcal{L}_{KL}(\theta_i)] \tag{4}$$

where λ is a regularization coefficient. In $g^i_{CCDA-OB}$, we apply the same clipping strategy as in IPPO [43], but remove the entropy bonus since exploration is considered in the KL regularization term. In practice, the expectations in baseline (Equation 2) and the KL term (Equation 3) are computed using samples of experience. We first compute the baseline based on the analytical form of the softmax policy [18] and sampled Q-values. Then, we compute the KL divergence using Q-values and policy distribution under sampled histories and messages.

Algorithm 1 CCDA methods using OB and KL

1: Initialize θ_i, θ^{msg}_i and ϕ_i for actors, communication, and critics
2: **for** each episode **do**
3: Initialize replay buffer D_i for each agent i
4: Get initial observations $\boldsymbol{o}_0 = \{o^1_0, ..., o^N_0\}$
5: **for** $t = 0$ to *max_steps_per_episode* **do**
6: **for** each agent i **do**
7: Decide an action $a^i_t \sim \pi(\cdot|h_i, \theta_i)$
8: Generate messages $m^i_t \sim f^{msg}_i(\cdot|h_i, a_i, \theta^{msg}_i)$
9: Send messages m^i_t and aggregate messages m^{-i}_t
10: Compute $b^i_t(h^i_t, m^{-i}_t)$ according to Equation 2
11: Compute advantage $\hat{A}^i_t = Q_i(h^i_t, a^i_t, m^{-i}_t) - b^i_t(h^i_t, m^{-i}_t)$
12: **end for**
13: Get observations $\boldsymbol{o}_{t+1} = \{o^1_{t+1}, ..., o^N_{t+1}\}$ and rewards r_t
14: Insert experience $(o^i_t, m^{-i}_t, a^i_t, r_t, o^i_{t+1}, \hat{A}^i_t)$ to buffer D_i
15: **end for**
16: **for** each agent i **do**
17: Sample a batch $d_i \in D_i$ from buffer
18: Calculate $\mathcal{L}_{KL}(\theta_i)$ according to Equation 3
19: Update θ_i using $\mathcal{L}_{KL}(\theta_i)$ and sampled advantage values
20: Update θ^{msg}_i evaluated by $Q_i(h^i_t, a^i_t, m^{-i}_t)$
21: Update the critic parameter ϕ_i using TD-learning
22: **end for**
23: **end for**

We further demonstrate how CCDA methods are integrated with our proposed OB and KL techniques in Algorithm 1. As shown in the algorithm, during execution, agents first select actions and generate messages (lines 7-8). Then, agents exchange and aggregate their messages (line 9). Upon received messages, the baseline is computed based on Equation 2 and the advantage is computed using the baseline for later training (lines 10-11). After receiving rewards and new observations, agents store the experience of observations, actions, received messages, rewards, new observations, advantage values in their buffer (line 14). The training is enabled at the end of each episode, where agents first sample a batch of experience from the buffer. Then, each agent updates their policy based on the KL divergence (defined in Equation 3) and sampled advantage values (lines 18-19). Agents update their message functions using the Q-values of communicating critics. Then, communicating critics are updated by minimizing the TD-errors.

As a result, Algorithm 1 can be achieved by any CCDA method through implementing the communication process (line 9) and the learning strategy of communication (line 20). As introduced in Section 3, we adopt two representative communication processes (broadcast and graph-based) and learning strategies of communication (policy gradients and end-to-end) to demonstrate the flexibility and adaptability of the proposed techniques. Benefiting from the plug-in nature, our proposed techniques are model-agnostic and can be seamlessly integrated with existing Comm-MADRL methods under CCDA. We also provide the schematic diagram of integrating OB and KL into CCDA methods and discuss limitations in Appendix 4.

5 EXPERIMENTS

We evaluate our proposed OB-KL techniques in three well-established and challenging multi-agent benchmark environments, StarCraftII Multi-Agent Challenge (SMAC) [8, 30], Traffic Junction [6], and Predator-Prey [34] in MADRL [2]. These environments consist of a varying number of cooperative agents with shared rewards and stochasticity, showing difficulties in coordinating agents' behaviors and achieving cooperative goals. We compare with the following actor-critic methods:

- **CTDE methods (centralized critics)**: MAPPO [43] and MAT [40] are strong baseline methods that have achieved SOTA performance across several MARL benchmarks [40, 43]. Notably, MAPPO and MAT are PPO-based methods using centralized critics under CTDE learning paradigm. We compare CCDA and CTDE methods to empirically show that using communicating critics (under CCDA) can exhibit similar or higher variance than that of using centralized critics (under CTDE), supporting our theoretical analysis in Section 4.1.
- **CCDA methods (communicating critics)**: GAAC [20] and IPPO [43] extended with communication (IPPO-Comm). GAAC is the SOTA method under the CCDA setting using end-to-end training and communication graphs. Due to the scarcity of communication methods under CCDA, we further adapt the well-known decentralized MADRL method, IPPO, with a communication modular, named IPPO-Comm, which allows broadcast communication between critics. Specifically, IPPO-Comm encodes each agent's local histories and actions into a range of integers as messages, using policy gradients to train the messages function (as defined in Section 3). GAAC and IPPO-Comm represent two representative choices for deciding when to communicate (communication graphs or broadcast communication) and how to learn communication (end-to-end or policy gradients).
- **CCDA methods with OB-KL**: We extend the CCDA methods GAAC and IPPO-Comm with our proposed technique (OB-KL), forming GAAC-OB-KL and IPPO-Comm-OB-KL. We aim to show that our proposed techniques can reduce variance compared to CCDA methods without OB-KL, while achieving similar or higher learning performance than other methods. Notably, GAAC-OB-KL and IPPO-Comm-OB-KL not only show the effectiveness of our techniques but also highlight the possibility of integrating these techniques with various communication methods.

We illustrate the essential components of all methods and how they differ from each other in critics and policy regularization techniques in Appendix 4. Notably, all methods use the same formulation of actors $\pi_i(a_i|h_i, \theta_i)$. Compared to IPPO-Comm-OB-KL and GAAC-OB-KL, MAPPO and MAT employ baseline techniques based on joint-history value functions and state-value functions, respectively, which do not account for communicated messages. Moreover, IPPO-Comm-OB-KL and GAAC-OB-KL use our proposed KL regularization for improving the baseline estimation while other methods use entropy regularization for encouraging exploration. To illustrate the efficacy of our proposed variance reduction techniques, we remove the baseline function from IPPO-Comm and GAAC. All results are reported as averages over 6 random seeds. To enable fair comparison, critical hyperparameters are either chosen based on empirical practice or kept consistent across all methods. Importantly, in all evaluation domains, we adopt $\lambda = 0.01$ for our introduced KL regularization, consistent with the entropy coefficient commonly used in the literature [14]. In addition, we tune the Boltzmann softmax temperature $\beta \in \{0.5, 1, 5\}$ and analyze how the choice of temperature β affects variance reduction and learning performance in Sections 5.2 and 5.3. We report the shared and map-specific hyperparameters in Appendix 4, where all methods use identical parameters to ensure fair comparison. Specifically, following the literature [18], we disable GAE [31] to separate the effect of different baseline functions and enable a better comparison of learning performance. We also report the consumed time for all methods in Appendix 4, where incorporating OB-KL does not significantly increase running time.

[2]The source code is available at https://github.com/changxizhu/OBKL.

5.1 Evaluation Domains

In SMAC, following recent literature [14, 45], we select hard maps in which communication plays an important role in broadening agents' views of the environment or coordinating their strategies effectively: `10m_vs_11m` (hard), `6h_vs_8z` (hard+), `1o_10b_vs_1r` (hard+), and `27m_vs_30m` (hard+). These hard maps feature diverse roles of agents and terrains. For example, in `1o_10b_vs_1r` map, an overseer agent (`1o`) that detects the enemy must communicate with 10 baneling agents (`10b`) that act to eliminate the enemy (`1r`). The variety and richness of these tasks provide a comprehensive evaluation of different aspects of communication. All methods are evaluated by the win rate of destroying all enemies.

In Traffic Junction, agents (cars) move along two-way roads with one or more junctions following predefined routes [14, 34]. To encourage timely completion, each agent receives a step penalty of -0.01 multiplied by the number of steps taken. Agents must avoid collisions, which incur a shared penalty of -1 when collisions occur. To promote coordinated behavior, all agents receive a reward of 1 if no collision occurs during an episode. We consider two maps with many agents, different routes, and limited observability. In the `Medium` map, 15 agents must complete routes of length 10 with a single junction. In the `Hard` map, 20 agents must complete routes of length 12 with four junctions. We evaluate the win rate of completed episodes without collisions for all maps.

In Predator-Prey, predators (agents) with limited vision (i.e., a range of 2) search for two prey in a grid world [34]. To increase the stochasticity and difficulty of the environment, the prey move randomly until being caught, and a reward of 1 is given to the predators only when all predators reach the prey; otherwise, a step penalty of -1 is given to the predators. We further consider maps of different scales: `PP9a` where 9 predators operate in a 10×10 grid within 50 steps, and `PP12a` where 12 predators move a 12×12 grid within 100 steps. We evaluate the win rate of all predators catching the prey.

5.2 Variance in Policy Gradients

We first compare the variance of policy gradients under CCDA and CTDE to verify our theoretical analysis that using communicating critics (in CCDA) can lead to equal or higher variance than using centralized critics (in CTDE), as proved in Section 4.1. We use MAPPO as a representative CTDE method, compared to CCDA methods

Table 2: The std of gradient norms (×0.01) for representative CTDE and CCDA methods in all maps.

	Map	MAPPO w/o baseline	IPPO-Comm	GAAC
SMAC	10m_vs_11m	2.65	3.35	1.52
	6h_vs_8z	7.47	8.74	7.31
	1o_10b_vs_1r	1.11	1.55	6.47
	27m_vs_30m	0.51	1.09	0.90
Traffic Junction	Medium	1.93	2.28	2.27
	Hard	1.71	2.02	2.11
Predator-Prey	PP9a	1.73	2.05	1.81
	PP12a	1.01	1.26	1.12

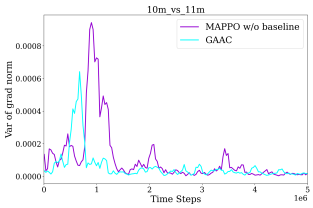

Figure 2: The change of variance over time.

IPPO-Comm and GAAC, and these three methods differ only in their critics. To enable a fair comparison, we remove the baseline function used in MAPPO (denoted as MAPPO w/o baseline) and adopt the same training strategies and hyperparameters for MAPPO w/o baseline, IPPO-Comm and GAAC. Following the literature [18], we compare the standard deviation of policy gradient norms across different seeds during training, as reported in Table 2.

As we can see, IPPO-Comm consistently exhibits higher gradient variance than MAPPO w/o baseline across all domains, empirically supporting our theoretical analysis. GAAC exhibits higher gradient variance than MAPPO w/o baseline in all maps except for 10m_vs_11m, which show much lower variance than other two methods. To understand this behavior, we plot the change of variance over time steps for MAPPO w/o baseline and GAAC in map 10m_vs_11m in Figure 2. As we can see, the variance of GAAC policy gradients increases quickly at the beginning of training. However, after approximately 0.1M time steps, the variance drops and becomes stable, indicating premature convergence. In contrast, MAPPO w/o baseline exhibits fluctuations in the gradient variance even after 3M time steps, which may be attributed to exploration that can potentially lead to better performance.

To further investigate how the KL regularization term influences variance reduction, we compute the relative change in variance when applying OB or OB-KL compared to the base methods (IPPO-Comm and GAAC) under different values of temperature $\beta \in \{0.5, 1, 5\}$. For example, for IPPO-Comm-OB-KL, we have: $\textit{relative change} = \frac{\text{Var(IPPO-Comm-OB-KL)} - \text{Var(IPPO-Comm)}}{\text{Var(IPPO-Comm)}}$, where the KL uses a certain value of β. As shown in Figure 3, the solid lines show the relative change in variance under different β values, while the dotted lines correspond to the relative change in variance achieved by OB alone. In SMAC, IPPO-Comm-OB-KL consistently reduces variance across all tested β values and achieves better variance reduction (i.e., lower *relative change*) than IPPO-Comm-OB in three out

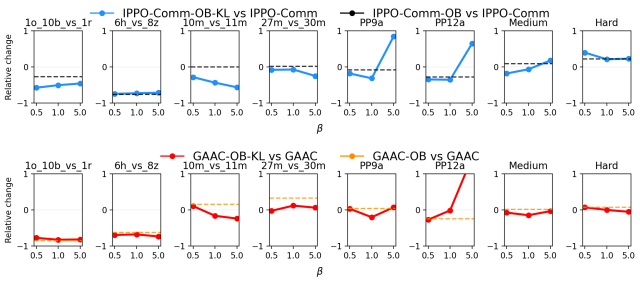

Figure 3: Relative change in variance of gradient norms. Solid lines represent the relative change using OB-KL. Dotted lines represent the relative change using OB.

of four maps. In the Predator-Prey and Traffic Junction domains, IPPO-Comm-OB-KL also outperforms IPPO-Comm-OB in terms of variance reduction for most choices of β. For GAAC-OB-KL, however, the variance reduction can be sensitive to the choice of β. This is likely because GAAC-OB itself can exhibit higher variance than GAAC, causing the effectiveness of GAAC-OB-KL to depend more on a tuned β. Moreover, we find that these behaviors of different β values can be related to policy entropy, reflecting differences in exploration and environment-specific properties. A detailed analysis is presented in Appendix 4.

5.3 Win Rate of Tasks

The win rate of comparing methods across all domains are presented in Figure 4. As shown, compared to CCDA methods without OB-KL (i.e., IPPO-Comm and GAAC), IPPO-Comm-OB-KL and GAAC-OB-KL achieve much higher win rates across all maps, demonstrating the effectiveness of the proposed variance reduction techniques in improving learning performance. Due to the similarity between CCDA and CTDE settings, we also compare IPPO-Comm-OB-KL and GAAC-OB-KL with CTDE methods (using standard implementations). As shown, IPPO-Comm-OB-KL and GAAC-OB-KL achieve higher win rates in most maps than MAT and MAPPO, except for 27m_vs_30m, PP9a, and PP12a. In these three maps, IPPO-Comm-OB-KL achieves similar final performance as MAT. In PP12a, IPPO-Comm-OB-KL can achieve much higher win rates during the early stage of training, i.e., before 1M time steps. On the other hand, GAAC-OB-KL achieves similar performance to MAT in 27m_vs_30m and PP12a. Then, using our proposed variance techniques in CCDA does not compromise learning performance when compared with strong CTDE methods on most maps.

We further conduct ablation studies to investigate how the proposed OB-KL affects learning performance in CCDA setting. Specifically, we perform ablations by removing the KL divergence term and the optimal baseline (OB) from the full versions of IPPO-Comm-OB-KL and GAAC-OB-KL. We first report the win rates obtained under the ablations of OB and KL in Figure 5. As shown, compared to using OB alone, jointly applying OB and KL leads to substantially higher win rates in 10m_vs_11m for GAAC-OB-KL, 6h_vs_8z for IPPO-Comm-OB-KL, 1o_10b_vs_1r for both IPPO-Comm-OB-KL and GAAC-OB-KL, and PP9a for both IPPO-Comm-OB-KL and GAAC-OB-KL. In the remaining maps, jointly applying

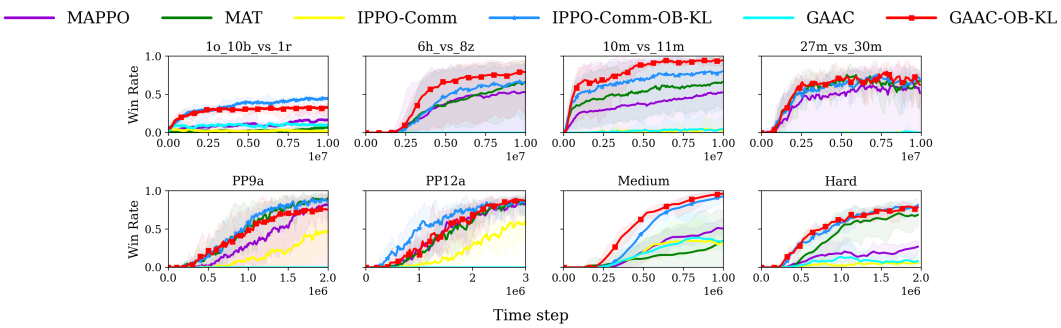

Figure 4: Averaged win rate of all methods in SMAC, Traffic Junction, and Predator-Prey.

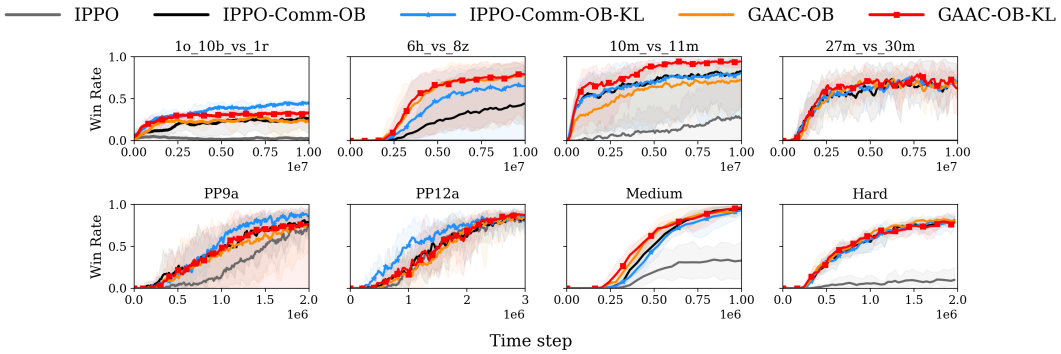

Figure 5: Averaged win rate when ablating OB and KL in SMAC, Traffic Junction, and Predator-Prey.

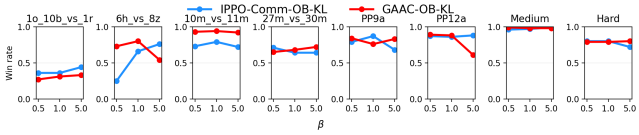

Figure 6: Win rate of IPPO-Comm-OB-KL and GAAC-OB-KL under different β values in all maps.

OB and KL does not lead to significant performance degradation compared to using OB alone. We further ablate communication, i.e., the DTDE method IPPO, and show that CCDA with our proposed variance reduction techniques consistently outperform the DTDE method that does not use communication.

We further investigate the effect of different β values on learning performance, as shown in Figure 6. The performance of IPPO-Comm-OB-KL and GAAC-OB-KL is generally similar across different β values in most maps, except for `6h_vs_8z` and `PP12a`. In `6h_vs_8z`, IPPO-Comm-OB-KL using larger β values (e.g., $\beta = 5$) tends to be deterministic at the early stage of training, while eventually maintaining similar amount of entropy compared to other β values. This pattern of IPPO-Comm-OB-KL with $\beta = 5$ can lead to much lower dead ratio of agents, which can finally achieve higher win rate. In contrast, for GAAC-OB-KL in `PP12a`, we observe a significant drop in policy entropy at the early stage of training for large β value, i.e., 5, which leads to convergence to suboptimal policies. Nevertheless, $\beta = 1$ provides a robust choice, achieving performance comparable to CTDE methods in all maps.

6 CONCLUSIONS

We investigate the variance of policy gradients caused by communication in multi-agent deep reinforcement learning. Specifically, we focus on the Communicating Critics and Decentralized Actors (CCDA) setting, where communication is allowed only among critics during training, while actors do not communicate during training and execution. By variance analysis, we prove that CCDA policy gradients have a higher or equal variance than the policy gradients under CTDE without communicating critics. We further propose a message-dependent baseline technique and a regularization technique for variance reduction in policy gradients. We theoretically prove that the optimal message-dependent baseline can reduce the variance in CCDA policy gradients. The experiments on several tasks show that our proposed techniques achieve not only a similar or higher learning performance but also reduced variance in policy gradients. In the future, we would like to investigate how continuous values of messages are quantized or bandwidth-limited, and utilizing adaptive baselines affect the variance in policy gradients. We will also investigate whether communication in general Comm-MARL settings (e.g., those with communicating actors) causes increased variance, and whether our proposed technique can reduce such variance in the future.

REFERENCES

[1] Noam Brown and Tuomas Sandholm. 2019. Superhuman AI for multiplayer poker. *Science* 365, 6456 (2019), 885–890.

[2] Rodolfo Carneiro Cavalcante, Ig Ibert Bittencourt, Alan Pedro da Silva, Marlos Silva, Evandro de Barros Costa, and Robério José R. dos Santos. 2012. A survey of security in multi-agent systems. *Expert Syst. Appl.* 39, 5 (2012), 4835–4846.

[3] Nicolò Cesa-Bianchi, Claudio Gentile, Gergely Neu, and Gábor Lugosi. 2017. Boltzmann Exploration Done Right. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett (Eds.). 6284–6293.

[4] Jingdi Chen, Tian Lan, and Carlee Joe-Wong. 2024. RGMComm: Return Gap Minimization via Discrete Communications in Multi-Agent Reinforcement Learning. In *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2014, February 20-27, 2024, Vancouver, Canada*, Michael J. Wooldridge, Jennifer G. Dy, and Sriraam Natarajan (Eds.). AAAI Press, 17327–17336. https://doi.org/10.1609/AAAI.V38I16.29680

[5] Filippos Christianos, Lukas Schäfer, and Stefano V. Albrecht. 2020. Shared Experience Actor-Critic for Multi-Agent Reinforcement Learning. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin (Eds.).

[6] Abhishek Das, Théophile Gervet, Joshua Romoff, Dhruv Batra, Devi Parikh, Mike Rabbat, and Joelle Pineau. 2019. TarMAC: Targeted Multi-Agent Communication. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*. 1538–1546.

[7] Ziluo Ding, Tiejun Huang, and Zongqing Lu. 2020. Learning Individually Inferred Communication for Multi-Agent Cooperation. In *Advances in Neural Information Processing Systems 33 (NeurIPS)*, Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin (Eds.). https://proceedings.neurips.cc/paper/2020/hash/fb2fcd534b0ff3bbed73cc51df620323-Abstract.html

[8] Benjamin Ellis, Jonathan Cook, Skander Moalla, Mikayel Samvelyan, Mingfei Sun, Anuj Mahajan, Jakob Foerster, and Shimon Whiteson. 2023. Smacv2: An improved benchmark for cooperative multi-agent reinforcement learning. *Advances in Neural Information Processing Systems* 36 (2023), 37567–37593.

[9] Jakob N. Foerster, Yannis M. Assael, Nando de Freitas, and Shimon Whiteson. 2016. Learning to Communicate with Deep Multi-Agent Reinforcement Learning. In *Advances in Neural Information Processing Systems 29 (NIPS)*. 2137–2145.

[10] Jakob N. Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. [n.d.]. Counterfactual Multi-Agent Policy Gradients. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, Sheila A. McIlraith and Kilian Q. Weinberger (Eds.). 2974–2982.

[11] Matthias Gerstgrasser, Tom Danino, and Sarah Keren. 2023. Selectively Sharing Experiences Improves Multi-Agent Reinforcement Learning. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (Eds.).

[12] Evan Greensmith, Peter L. Bartlett, and Jonathan Baxter. 2004. Variance Reduction Techniques for Gradient Estimates in Reinforcement Learning. *J. Mach. Learn. Res.* 5 (2004), 1471–1530.

[13] Sven Gronauer and Klaus Diepold. 2022. Multi-agent deep reinforcement learning: a survey. *Artif. Intell. Rev.* 55, 2 (2022), 895–943. https://doi.org/10.1007/S10462-021-09996-W

[14] Shengchao Hu, Li Shen, Ya Zhang, and Dacheng Tao. 2024. Learning Multi-Agent Communication from Graph Modeling Perspective. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net. https://openreview.net/forum?id=Qox9rO0kN0

[15] Shariq Iqbal and Fei Sha. [n.d.]. Actor-Attention-Critic for Multi-Agent Reinforcement Learning. In *Proceedings of the 36th International Conference on Machine Learning (ICML) (Proceedings of Machine Learning Research, Vol. 97)*, Kamalika Chaudhuri and Ruslan Salakhutdinov (Eds.). 2961–2970.

[16] Jiechuan Jiang and Zongqing Lu. 2018. Learning Attentional Communication for Multi-Agent Cooperation. In *Advances in Neural Information Processing Systems 31 (NIPS)*. 7265–7275.

[17] Jens Kober, J. Andrew Bagnell, and Jan Peters. 2013. Reinforcement learning in robotics: A survey. *Int. J. Robotics Res.* 32, 11 (2013), 1238–1274. https://doi.org/10.1177/0278364913495721

[18] Jakub Grudzien Kuba, Muning Wen, Linghui Meng, Shangding Gu, Haifeng Zhang, David Mguni, Jun Wang, and Yaodong Yang. 2021. Settling the Variance of Multi-Agent Policy Gradients. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan (Eds.). 13458–13470.

[19] Xinran Li and Jun Zhang. 2024. Context-aware Communication for Multi-agent Reinforcement Learning. In *Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2024, Auckland, New Zealand, May 6-10, 2024*, Mehdi Dastani, Jaime Simão Sichman, Natasha Alechina, and Virginia Dignum (Eds.). International Foundation for Autonomous Agents and Multiagent Systems / ACM, 1156–1164. https://doi.org/10.5555/3635637.3662972

[20] Yong Liu, Weixun Wang, Yujing Hu, Jianye Hao, Xingguo Chen, and Yang Gao. 2020. Multi-Agent Game Abstraction via Graph Attention Neural Network. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence (AAAI)*. 7211–7218.

[21] Ryan Lowe, Yi Wu, Aviv Tamar, Jean Harb, Pieter Abbeel, and Igor Mordatch. 2017. Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett (Eds.). 6379–6390.

[22] Xueguang Lyu, Andrea Baisero, Yuchen Xiao, Brett Daley, and Christopher Amato. 2023. On Centralized Critics in Multi-Agent Reinforcement Learning. *J. Artif. Intell. Res.* 77 (2023), 295–354. https://doi.org/10.1613/JAIR.1.14386

[23] Xueguang Lyu, Yuchen Xiao, Brett Daley, and Christopher Amato. 2021. Contrasting Centralized and Decentralized Critics in Multi-Agent Reinforcement Learning. In *AAMAS '21: 20th International Conference on Autonomous Agents and Multiagent Systems, Virtual Event, United Kingdom, May 3-7, 2021*, Frank Dignum, Alessio Lomuscio, Ulle Endriss, and Ann Nowé (Eds.). 844–852.

[24] Hangyu Mao, Zhengchao Zhang, Zhen Xiao, Zhibo Gong, and Yan Ni. [n.d.]. Learning Agent Communication under Limited Bandwidth by Message Pruning. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence*. 5142–5149.

[25] Yaru Niu, Rohan R. Paleja, and Matthew C. Gombolay. 2021. Multi-Agent Graph-Attention Communication and Teaming. In *20th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*. 964–973.

[26] Frans A. Oliehoek and Christopher Amato. 2016. *A Concise Introduction to Decentralized POMDPs*. Springer.

[27] Shayegan Omidshafiei, Jason Pazis, Christopher Amato, Jonathan P. How, and John Vian. 2017. Deep Decentralized Multi-task Multi-Agent Reinforcement Learning under Partial Observability. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017 (Proceedings of Machine Learning Research, Vol. 70)*, Doina Precup and Yee Whye Teh (Eds.). PMLR, 2681–2690.

[28] Afshin Oroojlooy and Davood Hajinezhad. 2023. A review of cooperative multi-agent deep reinforcement learning. *Appl. Intell.* 53, 11 (2023), 13677–13722. https://doi.org/10.1007/S10489-022-04105-Y

[29] Peng Peng, Quan Yuan, Ying Wen, Yaodong Yang, Zhenkun Tang, Haitao Long, and Jun Wang. 2017. Multiagent Bidirectionally-Coordinated Nets for Learning to Play StarCraft Combat Games. *CoRR* abs/1703.10069 (2017). arXiv:1703.10069 http://arxiv.org/abs/1703.10069

[30] Mikayel Samvelyan, Tabish Rashid, Christian Schröder de Witt, Gregory Farquhar, Nantas Nardelli, Tim G. J. Rudner, Chia-Man Hung, Philip H. S. Torr, Jakob N. Foerster, and Shimon Whiteson. 2019. The StarCraft Multi-Agent Challenge. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS '19, Montreal, QC, Canada, May 13-17, 2019*, Edith Elkind, Manuela Veloso, Noa Agmon, and Matthew E. Taylor (Eds.). 2186–2188.

[31] John Schulman, Philipp Moritz, Sergey Levine, Michael I. Jordan, and Pieter Abbeel. 2016. High-Dimensional Continuous Control Using Generalized Advantage Estimation. In *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, Yoshua Bengio and Yann LeCun (Eds.). http://arxiv.org/abs/1506.02438

[32] Shai Shalev-Shwartz, Shaked Shammah, and Amnon Shashua. 2016. Safe, Multi-Agent, Reinforcement Learning for Autonomous Driving. *CoRR* abs/1610.03295 (2016). arXiv:1610.03295 http://arxiv.org/abs/1610.03295

[33] David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, Yutian Chen, Timothy P. Lillicrap, Fan Hui, Laurent Sifre, George van den Driessche, Thore Graepel, and Demis Hassabis. 2017. Mastering the game of Go without human knowledge. *Nat.* 550, 7676 (2017), 354–359. https://doi.org/10.1038/nature24270

[34] Amanpreet Singh, Tushar Jain, and Sainbayar Sukhbaatar. 2019. Learning when to Communicate at Scale in Multiagent Cooperative and Competitive Tasks. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. https://openreview.net/forum?id=rye7knCqK7

[35] Georgy Skorobogatov, Cristina Barrado, and Esther Salamí. 2020. Multiple UAV Systems: A Survey. *Unmanned Syst.* 8, 2 (2020), 149–169.

[36] Sainbayar Sukhbaatar, Arthur Szlam, and Rob Fergus. 2016. Learning Multiagent Communication with Backpropagation. In *Advances in Neural Information*

Processing Systems 29 (NIPS). 2244–2252.

[37] George Tucker, Surya Bhupatiraju, Shixiang Gu, Richard E. Turner, Zoubin Ghahramani, and Sergey Levine. 2018. The Mirage of Action-Dependent Baselines in Reinforcement Learning. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Workshop Track Proceedings*. OpenReview.net.

[38] Jingkang Wang, Yang Liu, and Bo Li. 2020. Reinforcement Learning with Perturbed Rewards. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*. AAAI Press, 6202–6209. https://doi.org/10.1609/AAAI.V34I04.6086

[39] Lex Weaver and Nigel Tao. 2001. The Optimal Reward Baseline for Gradient-Based Reinforcement Learning. In *UAI '01: Proceedings of the 17th Conference in Uncertainty in Artificial Intelligence, University of Washington, Seattle, Washington, USA, August 2-5, 2001*, Jack S. Breese and Daphne Koller (Eds.). Morgan Kaufmann, 538–545.

[40] Muning Wen, Jakub Grudzien Kuba, Runji Lin, Weinan Zhang, Ying Wen, Jun Wang, and Yaodong Yang. 2022. Multi-Agent Reinforcement Learning is a Sequence Modeling Problem. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh (Eds.).

[41] Cathy Wu, Aravind Rajeswaran, Yan Duan, Vikash Kumar, Alexandre M. Bayen, Sham M. Kakade, Igor Mordatch, and Pieter Abbeel. 2018. Variance Reduction for Policy Gradient with Action-Dependent Factorized Baselines. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

[42] Yaodong Yang and Jun Wang. 2020. An Overview of Multi-Agent Reinforcement Learning from Game Theoretical Perspective. *CoRR* abs/2011.00583 (2020). arXiv:2011.00583 https://arxiv.org/abs/2011.00583

[43] Chao Yu, Akash Velu, Eugene Vinitsky, Jiaxuan Gao, Yu Wang, Alexandre M. Bayen, and Yi Wu. 2022. The Surprising Effectiveness of PPO in Cooperative Multi-Agent Games. In *NeurIPS*.

[44] Mohamed Salah Zaïem and Etienne Bennequin. 2019. Learning to Communicate in Multi-Agent Reinforcement Learning : A Review. *CoRR* abs/1911.05438 (2019). arXiv:1911.05438 http://arxiv.org/abs/1911.05438

[45] Zhuohui Zhang, Bin He, Bin Cheng, and Gang Li. 2025. Bridging Training and Execution via Dynamic Directed Graph-Based Communication in Cooperative Multi-Agent Systems. In *AAAI-25, Sponsored by the Association for the Advancement of Artificial Intelligence, February 25 - March 4, 2025, Philadelphia, PA, USA*, Toby Walsh, Julie Shah, and Zico Kolter (Eds.). AAAI Press, 23395–23403. https://doi.org/10.1609/AAAI.V39I22.34507

[46] Changxi Zhu, Mehdi Dastani, and Shihan Wang. 2024. A survey of multi-agent deep reinforcement learning with communication. *Autonomous Agents and Multi-Agent Systems* 38, 4 (2024). https://doi.org/10.1007/s10458-023-09633-6