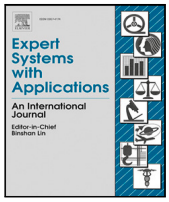
Expert Systems With Applications 245 (2024) 123158

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa



A Contrastive-Enhanced Ensemble Framework for Efficient Multi-Agent Reinforcement Learning☆

Xinqi Du [a,b], Hechang Chen [a,b,c,*], Yongheng Xing [d], Philip S. Yu [e], Lifang He [f]

[a] School of Artificial Intelligence, Jilin University, Changchun 130012, China
[b] Engineering Research Center of Knowledge-Driven Human–Machine Intelligence, Ministry of Education, Changchun 130012, China
[c] Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education, Changchun 130012, China
[d] College of Computer Science and Technology, Jilin University, Changchun 130012, China
[e] Department of Computer Science, University of Illinois at Chicago, Chicago, IL 60607, USA
[f] Department of Computer Science and Engineering, Lehigh University, Bethlehem, PA 18015, USA

ARTICLE INFO

Keywords:
Multi-agent reinforcement learning
Sample efficiency
Ensemble learning
Contrastive learning
Multi-agent system

ABSTRACT

Multi-agent reinforcement learning is promising for real-world applications as it encourages agents to perceive and interact with their surrounding environment autonomously. However, sample efficiency is still a concern that prevents the application of multi-agent reinforcement learning in practice. A well-performing agent typically needs an abundance of interaction data for training, while obtaining numerous interaction data in a 'trial-and-error' manner is usually overhead-expensive or even infeasible for real-world tasks. In this paper, we propose a data-efficient framework, Contrastive-Enhanced Ensemble framework for Multi-Agent Reinforcement Learning (C2E-MARL), with the aim of training better-performing agents in the multi-agent system with fewer interaction data. Specifically, the proposed framework deploys an ensemble of centralized critic networks for action value estimation, i.e., it combines the outputs of multiple critic networks to estimate the action value. It makes full use of data from various perspectives to reduce the estimation error, which is helpful for efficient policy updating. Moreover, contrastive learning, a prevailing self-supervised technology, is employed to enhance the learning efficiency of submodels in C2E-MARL by augmenting the interaction data. Extensive experimental results compared with the state-of-the-art methods on three multi-agent benchmark scenarios demonstrate the superiority of C2E-MARL in terms of efficiency and performance.

1. Introduction

Reinforcement learning (RL) has achieved notable success across various domains over the last few years, including Go (Silver et al., 2017), robotics control (Lillicrap, Hunt, Pritzel, Heess, Erez, Tassa, Silver, & Wierstra, 2016; Schulman, Levine, Abbeel, Jordan, & Moritz, 2015), financial markets (Shavandi & Khedmati, 2022), and so forth (Elhaki, Shojaei, & Mehrmohammadi, 2022; Kiran et al., 2021). Its main idea is to encourage an agent to interact with the environment through trial and error and maximize the accumulated reward. Since an agent trained by RL can perceive and interact with its surrounding environment autonomously, employing the RL method to tackle real-world stochastic tasks is advocated. However, real-world tasks often involve multiple agents with partial observation, which exacerbates the difficulty and complexity of training agents, preventing us from practically using reinforcement learning in multi-agent systems. Therefore, there is an urgent need to design well-performing multi-agent reinforcement learning methods for practical applications.

A considerable literature has studied RL algorithms regarding multi-agent scenarios. The original algorithms train multiple agents individually, subject to their private observation (Tan, 1993), but the learned policies are unstable due to the partially observed environment. Fully-centralized methods are proposed to alleviate the non-stationary problem by incorporating joint information from all agents (Claus & Boutilier, 1998). However, these methods have poor scalability because the dimension of the joint action space expands rapidly as the number of agents grows. Centralized training and decentralized

☆ This work is partially supported in part by the International Cooperation Project of Jilin Province (20220402009GH); the National Natural Science Foundation of China (U2341229, 61976102 and U19A2065); the National Key R&D Program of China (2021ZD0112501 and 2021ZD0112502); Lehigh's Accelerator Foundation under grant No. S00010293, and the National Science Foundation (MRI 2215789 and IIS 1909879).

* Corresponding author at: School of Artificial Intelligence, Jilin University, Changchun 130012, China.
E-mail addresses: duxq18@mails.jlu.edu.cn (X. Du), chenhc@jlu.edu.cn (H. Chen), xingyh18@mails.jlu.edu.cn (Y. Xing), psyu@cs.uic.edu (P.S. Yu), lih319@lehigh.edu (L. He).

https://doi.org/10.1016/j.eswa.2024.123158
Received 24 February 2023; Received in revised form 12 December 2023; Accepted 3 January 2024
Available online 6 January 2024

execution framework (CTDE), a compromise approach, has attracted more attention and been used in many state-of-the-art algorithms, such as QMIX (Rashid et al., 2018), VDN (Sunehag et al., 2018), MADDPG (Lowe et al., 2017), etc. In CTDE, the agents only incorporate joint information during training to broaden their perspectives and stabilize the training while selecting their actions based on local observations. In addition, several advanced techniques are used to enhance model performance. For example, MAAC employs a multi-head attention mechanism to extract relevant information to improve model efficiency (Iqbal & Sha, 2019), G2ANet uses a graph neural network to enhance its performance (Liu, Wang et al., 2020), and VGN (Wei, Li, Zhang, & Wang, 2022) deploys a graph attention network for value decomposition in a multi-agent system.

Although most existing studies have significantly improved in terms of effectiveness, the issue of sample efficiency is still a concern in MARL. Sample efficiency denotes the amount of data required to attain a certain level of performance during training (Nguyen, Nguyen, & Nahavandi, 2020). As pointed out by Duan et al. (2016), it takes several hours or even days to learn RL models to manipulate a simple game, e.g., Atari games (Xu, Zhu, Liu, & Zhao, 2021) and Poker (Li, Wang, Jia, Wu, Zhang, & Qi, 2022), which is more prominent when tackling complicated tasks involving multiple agents (Sukhbaatar, Fergus, et al., 2016). Generally, the reason for sample inefficiency is twofold. First, MARL learns by trial and error, which requires considerable interaction data. The demand for interaction data increases rapidly as the number of agents grows. Second, MARL relies heavily on the representation of deep learning to approximate the policy and value function, which involves many parameters and requires numerous data for well-formulated training. In summary, sample inefficiency hinders the application of MARL to real-world tasks, where the interaction between agents is usually time-consuming, overhead-expensive, or even infeasible. Motivated by these limitations, we aim to take one step further to settle the issue of sample inefficiency in MARL.

In this paper, we propose a Contrastive-Enhanced Ensemble framework for Multi-Agent Reinforcement Learning (C2E-MARL). In this framework, an ensemble of centralized critic networks is deployed to estimate the action value, which makes full use of the interaction data by combining multiple estimators. In this way, it provides various estimations beneficial to reducing the estimation error. Meanwhile, contrastive learning, is employed to assist the underlying submodels in extracting features. It implements interaction data augmentation by using the *dropout*, which is conducive to learning efficiency and representation learning quality. Besides, we conduct extensive experiments on benchmark scenarios and make visualizations to demonstrate the superiority of the proposed C2E-MARL.

The main contributions of this paper are as follows:

- An efficient model called C2E-MARL is proposed to overcome the problem of poor sample efficiency in MARL. To the best of our knowledge, it is the first effort to collaboratively utilize ensemble learning and contrastive learning to improve the sample efficiency in the setting of MARL.
- C2E-MARL deploys multiple centralized critic networks to provide a comprehensive Q-value estimation, making full use of data. Besides, contrastive learning is leveraged to improve the underlying submodels efficiency, further enhancing the overall performance.
- We conduct extensive experiments compared with six state-of-the-art methods on three multi-agent benchmark scenarios to validate C2E-MARL. Empirical results demonstrate the superiority of C2E-MARL in terms of sample efficiency and effectiveness.

The remainder of this manuscript is organized as follows. Section 2 summarizes the previous works related to ours. Section 3 presents the notations used in our method and simply reviews the basic theorems. Section 4 elaborates the details of our model design. Section 5 presents a series of experiments on benchmark scenarios to validate the effectiveness of C2E-MARL and provides the analysis in detail. Section 6 concludes this paper and discusses further work.

2. Related work

In this section, we introduce previous works closely related to our method from three aspects: multi-agent reinforcement learning, ensemble learning, and contrastive learning in RL.

2.1. Multi-agent reinforcement learning

A substantial literature has been proposed for MARL. The earliest method (Tan, 1993) is based on a fully-decentralized framework, which is not feasible due to the partial observation leading to non-stationary. Incorporating joint information can alleviate the non-stationary problem (Claus & Boutilier, 1998), but the dimension tends to grow exponentially with the number of agents and task complexity, which limits the model's scalability and decreases efficiency. Besides, some studies are based on communication, which explicitly assumes the existence of information interaction among agents (Wang, Wang, Zheng, & Zhang, 2019). It is beneficial to improve the stationary (Noaeen, Naik, Goodman, Crebo, Abrar, Abad, Bazzan, & Far, 2022). For example, RIAL (Foerster, Assael, De Freitas, & Whiteson, 2016) and CommNet (Sukhbaatar et al., 2016) are the first to introduce communication learning in multi-agent systems, where each agent enhances its perception capabilities by explicitly communicating with other agents. ATOC (Jiang & Lu, 2018) is proposed to achieve a more flexible communication pattern by the attention mechanism. Unfortunately, they all require an additional communication mechanisms, while the communication resource is limited in real-world tasks. Several prevalent methods are built on the CTDE framework compromising the above methods, such as VDN (Sunehag et al., 2018), QMIX (Rashid et al., 2018), MADDPG (Lowe et al., 2017), MMD-MIX (Xu, Li, Bai and Fan, 2021), etc., and achieved stable and excellent performance (Liu & Tan, 2022).

However, considerable data are required for training a well-performing model, which is time-consuming or even infeasible for real-world tasks. To this end, some methods enhance MARL by combining advanced techniques. For example, MAAC incorporates with an attention mechanism to extract relevant information from high-dimensional input and makes it efficient (Iqbal & Sha, 2019). In this paper, our method is intended to improve the sample efficiency of MARL algorithms.

2.2. Ensemble learning

Ensemble learning is a classical machine learning technique that strategically generates decisions to solve problems by combining multiple underlying submodels. In practice, ensemble learning often yields better results in comparison metrics compared to any of the underlying submodels. Several studies have been proposed that use ensemble learning to address the problems in reinforcement learning, such as approximation error, exploration, and sample efficiency. Specifically, it uses an ensemble of policy or critic networks to propose a final output. Double DQN (Van Hasselt, Guez, & Silver, 2016) and TD3 (Fujimoto, Hoof, & Meger, 2018) simply use two critic networks to alleviate the approximation error. An ensemble of Q-value functions is employed for bootstrapped DQN (Osband, Blundell, Pritzel, & Van Roy, 2016) and EBQL (Peer, Tessler, Merlis, & Meir, 2021) to encourage efficient exploration and reduce estimation error. Maxmin Q-learning (Lan, Pan, Fyshe, & White, 2019) employs multiple critic networks to solve the problem of overestimation. BEAR (Kumar, Fu, Soh, Tucker, & Levine, 2019) utilizes the minimization or average of several estimators to provide more accuracy estimations for policy improvement. Sunrise (Lee, Laskin, Srinivas, & Abbeel, 2021) devises a framework with multiple policy and critic networks to stabilize the training process. However, in multi-agent systems, it is worth noting that there are no multi-agent reinforcement learning methods incorporated with ensemble learning. In this paper, we fill this gap by incorporating ensemble learning into MARL.

2.3. Contrastive learning

Contrastive learning, an advanced self-supervised method, is devoted to learning effective representations and has been successfully applied in computer vision (He, Fan, Wu, Xie, & Girshick, 2020), natural language processing (Gao, Yao, & Chen, 2021; Liang et al., 2021), and so forth. Constrative learning can be regarded as an instance discrimination problem, where one instance should be close to a similar instance while away from dissimilar instances in the embedding space. With its powerful representation learning capabilities, contrastive learning is employed to boost RL for efficiency (Fu et al., 2021; Liu, Zhang et al., 2020). For example, CURL (Laskin, Srinivas, & Abbeel, 2020) and M-CURL (Zhu et al., 2022) accelerate the learning speed of pixel-based RL by incorporating a contrastive learning task, and achieve remarkable performance. As for MARL, it requires global joint information to stabilize training, i.e., even if it is not dealing with a pixel-based task, MARL still needs to deal with high-dimensional inputs, which results in low efficiency. Therefore, we propose employing contrastive learning to enhance the representation learning of joint information in multi-agent systems to improve sampling efficiency.

In general, the core idea behind our work is to deploy an ensemble of Q-value functions, which enables diverse feature extraction. It also provides multiple action value estimation to alleviate the approximation error. Meanwhile, inspired by the "Cannikin Law", we set contrastive learning as an auxiliary task for the underlying models in the ensemble modules to improve their efficiency.

3. Preliminaries

In this section, we detail the notation and review some existing reinforcement learning and contrastive learning studies to provide theoretical support for our method.

3.1. Notation

We model a multi-agent scenario with N agents as Decentralized Partially Observable Markov Decision Process (Dec-POMDP) (Oliehoek & Amato, 2016). It is described by a tuple $G = \langle S, A, T, O, R, \gamma, N\rangle$, where S is a global state set. The action set A and observation set O can be split as $\{A_1, \dots, A_N\}$ and $\{O_1, \dots, O_N\}$ for each agent, respectively. During interaction, agent uses a stochastic policy to choose an action, i.e., $\pi_{\phi_i} : o_i \mapsto a_i, a_i \in A_i$. The joint actions induce a transition to the next state according to state transition function: $T := S \times A_1 \times \cdots \times A_N \times S \mapsto [0, 1]$, and then each agent can observe partial environment $o_i : S \mapsto O_i$ and obtain the reward $r_i := S \times A_i \mapsto \mathbb{R}$. Replay buffer is employed to restore history transitions $H = \left\langle o_i, a_i, o_i', r_i, done\right\rangle$ for training. The objective for each agent is to maximize its accumulated reward:

$$J(\pi_{\phi_i}) = \mathbb{E}[R_T] = \mathbb{E}_{a_i\sim\pi_{\phi_i}(\cdot|o_i)}\left[\sum_{t=0}^{T}\gamma^t r_i^t(o_i, a_i)\right] \tag{1}$$

where γ denotes discount factor to balance immediate reward and long-term gain.

3.2. Soft actor critic

Soft actor–critic (SAC) is an off-policy deep RL algorithm based on maximum entropy (Haarnoja, Zhou, Abbeel, & Levine, 2018), which prevents the agent from overoptimizing the action-value function and encourages exploration. In this work, we adapt SAC to model each agent in multi-agent settings. Specifically, SAC tries to maximize the cumulative return and the entropy of policy π_ϕ:

$$J(\phi) = \mathbb{E}_{o_i\sim p^{\pi_i}}\Big[Q(s, a) + \alpha\mathcal{H}(\pi_\phi(\cdot|s))\Big], \tag{2}$$

where α is a temperature parameter to balance the significance of entropy and the reward. $\mathcal{H}(\pi_\phi(\cdot|s))$ is the entropy of the policy π parameterized by ϕ, which is defined as $\mathcal{H}(\pi_\phi(\cdot|s)) = -\log\pi_\phi(a|s)$.

3.3. Contrastive learning

Contrastive learning can be described as looking up a dictionary, where positive sample x^+ and negative sample x^- are used as the keys in the dictionary with respect to the anchor x regarding a query. Given a sample x, the common setup of contrast learning is as follows: (1) Constructing a "dictionary" or sample pair (x, x^+) (or (x, x^-)). The simplest method is to take the random transformation of the same data; (2) Defining the loss function, such as the widely used Triplet (Schroff, Kalenichenko, & Philbin, 2015) and InfoNCE (Oord, Li, & Vinyals, 2018):

$$\begin{aligned}\mathcal{L}_{Triplet}(x, x^+, x^-) = \sum_{x\in\mathcal{X}}\max\Big(0, \|f(x) - f(x^+)\|_2^2 \\ - \|f(x) - f(x^-)\|_2^2 + \epsilon\Big),\end{aligned} \tag{3}$$

$$\mathcal{L}_{InfoNCE} = -\log\frac{\exp(score(x, x^+))}{\sum_{x'\in\mathcal{X}}\exp(score(x, x'))}, \tag{4}$$

where $f(x)$ is an encoder function, $\mathcal{X}$ denotes the set of samples, and $score(x, x')$ is used to measure the correlations. Note that $\mathcal{L}_{Triplet}$ heavily relies on the choice of negative samples, easily leading to non-convergence. $\mathcal{L}_{InfoNCE}$ is to distinguish the positive sample from the unrelated sample. In this paper, our proposed method uses InfoNCE as the loss function.

4. Methodology

In this section, we elaborate the proposed C2E-MARL, which deploys an ensemble of Q-value functions and combines an unsupervised contrastive representation technique. The overall architecture is illustrated in Fig. 1, which consists of (1) the basic framework of C2E-MARL; (2) ensemble centralized critic networks; and (3) a contrastive-enhanced module.

4.1. The basic framework of C2E-MARL

The proposed C2E-MARL is based on the CTDE framework, as illustrated in Fig. 1a. It integrates the joint action-observation as extra information during training to broaden the agent's horizon of the environment, which could alleviate the non-stationary problem common in MARL. During interaction with the environment, agents merely use their private observation to avoid communication overhead and poor scalability.

Considering an environment with N agents, the agents interact with the environment using the parameterized policy π_{ϕ_i}, a multilayer perceptron (MLP), to make the action decision a_i only based its own observation o_i. Its objective function with maximum-entropy is defined as:

$$J(\phi_i) = \mathbb{E}_{o_i\sim p^{\pi_i}}\Big[Q_i(\boldsymbol{o}, \boldsymbol{a}) + \alpha\mathcal{H}(\pi_{\phi_i}(\cdot|o_i))\Big], \tag{5}$$

where α denotes the temperature parameter that balance the importance of the entropy term versus the cumulative reward. $\boldsymbol{o} = (o_1, \dots, o_N)$ and $\boldsymbol{a} = (a_1, \dots, a_N)$ are the joint observation vector and action vector respectively. During the training phase, C2E-MARL incorporates the information from other agents to provide action estimation $Q_i(\boldsymbol{o}, (\boldsymbol{a}^{-i}, a_i))$ for policy updating, where $\boldsymbol{a}^{-i}$ denotes the joint vector excluding ith element a_i.

4.2. Ensemble centralized critic networks

The traditional reinforcement learning methods for multi-agent systems have been successful in some challenging domains, but their performance and learning efficiency heavily rely on the estimation accuracy of the action value. C2E-MARL employs an ensemble of Q-value functions to make action value estimations together, as shown in Fig. 1b, which is effective in settling the problem mentioned above.

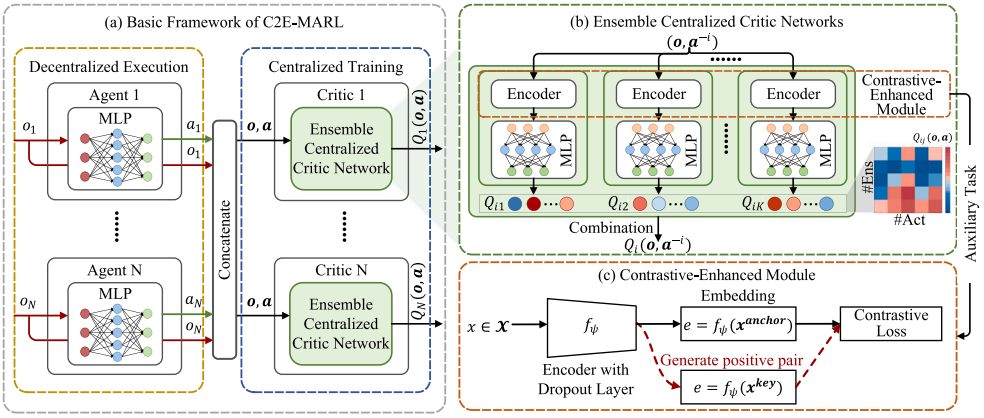


Fig. 1. The overall architecture of C2E-MARL. (a) Basic framework of C2E-MARL: it contains decentralized execution and centralized training. Agents use partial observation to make decisions during interaction with the environment, and the joint information is only incorporated during training to estimate the Q-value. (b) Ensemble centralized critic networks: an ensemble of critic networks is deployed to provide various estimations. (c) Contrastive-enhanced module: it is employed as an auxiliary task to improve the representation learning quality of each encoder in (b).

Formally, we consider an ensemble of K critic networks with the same structure but different parameters, i.e., $\{Q_{\theta_{ij}}, Q_{\bar{\theta}_{ij}}\}_{j=1}^{K}$, where θ_{ij} and $\bar{\theta}_{ij}$ denote the parameters of the jth Q-function and target Q-function for each agent i. Based on this paradigm, C2E-MARL employs the average voting mechanism for the outputs of all critic networks to give a final Q-value estimation, which is used to calculate the policy's objective function:

$$J(\phi_i) = \mathbb{E}_{o_i \sim p^{\pi_i}}\left[\frac{1}{K}\sum_{j=1}^{K} Q_{\theta_{ij}}(\boldsymbol{o}, \boldsymbol{a}) + \alpha \mathcal{H}(\pi_{\phi_i}(\cdot|o_i))\right]. \tag{6}$$

The temporal-difference is used to update the value function, which is based on the Bellman backup equation by establishing the connection between the current Q-value and the value in one-step lookahead. The C2E-MARL utilizes a variant of the Bellman equation, which is defined as:

$$\begin{aligned} \mathcal{L}_Q &= \sum_{j=1}^{K}\sum_{i=1}^{N}\left(Q_{\theta_{ij}}(\boldsymbol{o}_t, \boldsymbol{a}_t) - y_i\right)^2, \\ y_i &= r_t + \gamma\left(Q_{\bar{\theta}_{ij}}(\boldsymbol{o}_{t+1}, \boldsymbol{a}_{t+1}) - \alpha \log \pi_{\bar{\phi}_i}(a^i_{t+1}|o^i_{t+1})\right), \end{aligned} \tag{7}$$

where $\bar{\theta}_{ij}$ and $\bar{\phi}_i$ are parameters of target critic and policy networks; y_i is the target Q-value augmented by the entropy term. However, it regularly suffers from error propagation, i.e., the estimation error of the subsequent state affects the update direction of the current Q-value.

To alleviate this problem, we resort to an ensemble-based paradigm, where each submodel scatters in the solution space and estimates Q-value from multiple perspectives, which is visualized in Section 5. In practice, an ensemble framework usually yields better results in comparison metrics compared to any of the single models in its component. These are several reasons why this model works well. First, each individual model jointly covers the solution space as much as possible to mitigate convergence to the local optimum. Second, each model does not extract the same features from the interaction data and gives action value estimations with differences. It can effectively alleviate the problem of inaccurate estimation compared to only a single critic network setting.

We apply the following tips to guarantee the diversity among submodels: (1) random initialization, where each submodel has different initialization parameter values; (2) bootstrapping, which samples data from the replay buffer with replacement for submodels training. Additionally, we use the idea of random minimization and modify the loss function of the critic network as follows:

$$\begin{aligned} \mathcal{L}_Q &= \sum_{j=1}^{K}\sum_{i=1}^{N}\left(Q_{\theta_{ij}}(\boldsymbol{o}_t, \boldsymbol{a}_t) - y_i\right)^2, \\ y_i &= r_t + \gamma\left(\min_{c\in C_i} Q_{\bar{\theta}_{ic}}(\boldsymbol{o}_{t+1}, \boldsymbol{a}_{t+1}) - \alpha \log \pi_{\bar{\phi}_i}(a^i_{t+1}|o^i_{t+1})\right), \\ &\text{where } a^i_{t+1} \sim \pi_{\bar{\phi}_i}(\cdot|o^i_{t+1}), \end{aligned} \tag{8}$$

where C_i is the candidate set of the ensemble critic networks for agent i, which is randomly selected from the ensemble K, and ($|C| \leq K$). Its minimization is used to calculate the loss of the critic networks. In addition, we devise an auxiliary task for C2E-MARL to improve the performance of submodels and detail its implementation in the following subsection.

4.3. Contrastive-enhanced module

MARL confronts the challenge of insufficient learning capability for high-dimensional state representations, even though the states of some multi-agent tasks are represented as vectors (rather than pixel-based states). It is due to the fact that as the number of agents or task complexity increases, the dimension of the joint state information grows accordingly. Considering that joint state information is crucial for stable training, how to effectively learn features from high-dimensional state inputs becomes an urgent problem, which will affect the accuracy of action value estimation by critic networks. Besides, as the famous "Cannikin Law" states, the overall performance of an ensemble model depends on the performance of each submodel. Thus, we attempt to refine the submodels underlying C2E-MARL with advanced techniques contrastive learning, a self-supervised approach for learning a general-purpose representation.

To this end, C2E-MARL splits joint information encoders $f_{\psi_{ij}}(\boldsymbol{o}, \boldsymbol{a}^{-i})$ from the underlying critic networks Q_{ij} and devises contrastive learning for encoders to extract features, which enhances the overall training efficiency. Specifically, C2E-MARL must address two key questions:

- How to construct the data augmentation dataset to generate positive/negative sample pairs?
- How to measure the similarity between samples and define the loss function for training?

Contrastive learning is regarded as a dynamic dictionary look-up process, where how to construct a dictionary is decisive for the overall performance. A common way to construct a dictionary is by applying data augmentation to create noise versions of original samples, such as random cropping, noise injection, shuffle, and so forth. However, the abovementioned data augmentation methods are usually used in

computer vision, i.e., the pixel-based state, which are unsuitable for vector-based state since the joint information (or state vector) contains semantic information. In other words, the modified joint information is likely to be extremely different from its original. In view of this, we make some adjustments to apply it to MARL with vector-based states.

Algorithm 1: Contrastive-Enhanced Ensemble Framework for Multi-Agent Reinforcement Learning (C2E-MARL)

Input: The number of the agents N;
The ensemble size of the critic networks K.
Output: Parameters for policy π_i and critic Q_{ij}.

Initialize $\phi_i, \overline{\phi}_i, \{\theta_{ij}, \overline{\theta}_{ij}, \psi_{ij}, \overline{\psi}_{ij}\}_{j=1}^{K}$ for agent i;
Create replay buffer $RB \leftarrow \{\}$;
for $episode = 1, 2, \cdots, max\_episodes$ **do**
  $s_0 \sim \rho_0$, get the initial o_i for each agent i;
  for $t = 1, 2, \cdots, max\_length$ **do**
    Select actions $a_i \sim \pi_i(\cdot|o_i)$;
    Execute actions $\boldsymbol{a}$, receive $\boldsymbol{o'}$ and $\boldsymbol{r}$;
    Store transitions in RB, and set $\boldsymbol{o} \leftarrow \boldsymbol{o'}$;
    if $episode > threshold$ **then**
      Sample a minibatch B from RB;
      Randomly sample the candidate set C_i from $\{1, 2, \cdots, K\}$ for each agent i;
      Compute the target value y_i: $y_i = r_t + \gamma\big(\min_{c \in C_i} Q_{\overline{\theta}_{ic}}(\boldsymbol{o}_{t+1}, \boldsymbol{a}_{t+1}) - \alpha \log \pi_{\overline{\phi}_i}(a^i_{t+1}|o^i_{t+1})\big)$;
      for $j = 1, 2, \cdots, K$ **do**
        Sample Bootstrap masks for training: $m_{ij} \sim Bernoulli(\beta)$;
        Update ensemble critic parameters: $\theta_{ij} \leftarrow \theta_{ij} - \alpha \nabla_{\theta_{ij}} (Q_{\theta_{ij}}(\boldsymbol{o}, \boldsymbol{a}^{-i}) - y_i)^2$;
      Update policy parameters by Eq. (6): $\phi_i \leftarrow \phi_i - \alpha \nabla_{\phi_i} J(\phi_i)$;
      Update encoder parameters by Algorithm 2;
    Update target network parameters: $\overline{\theta}_{ij} \leftarrow \tau \overline{\theta}_{ij} + (1-\tau)\theta_{ij}, \forall j \in K$; $\overline{\phi}_i \leftarrow \tau \overline{\phi}_i + (1-\tau)\phi_i$.

Algorithm 2: Contrastive-Enhanced Module

Input: A batch of the transition samples.
Output: Parameters for encoders ψ_{ij}.

for $j = 1, 2, \cdots, K$ **do**
  Collect the anchor sample: $e_{anchor} = f_{\psi_{ij}}(\boldsymbol{o}, \boldsymbol{a}^{-i})$;
  Generate the positive sample: $e_{positive} = f_{\psi_{ij}}(\boldsymbol{o}, \boldsymbol{a}^{-i})$;
  Update encoder parameters ψ_{ij} and W in contrastive learning by minimizing Eq. (9).

Overall, C2E-MARL formulates the auxiliary task as instance-level discrimination. Inspired by the data augmentation technique used in Gao et al. (2021), the proposed C2E-MARL propagates the joint information twice through the encoder networks with the *dropout layer*, as shown in Fig. 1c. It naturally produces the anchor sample e_i and its positive sample e_i^+, and other data in the batch are used as negative samples e_j. In this way, C2E-MARL answers the first question.

In terms of the second question, C2E-MARL employs InfoNCE as the loss function, which is defined as follows:

$$\mathcal{L}_{Auxiliary} = -\log \frac{\exp(sim(e_i, e_i^+)/\tau)}{\sum_{j=0}^{B} \exp(sim(e_i, e_j)/\tau)}, \tag{9}$$

where $e_i = f_\psi(x_i)$, which denotes that the joint information x_i is mapped into vector e_i through the encoder f_ψ. And $sim(e_i, e_j) = e_i^T W e_j$, which is a bi-linear inner-product proposed in CURL (Laskin et al., 2020) to measure the similarity of e_i and e_j. W is a learned parameter matrix, and τ and B denote the temperature hyperparameter and batch size, respectively.

Algorithm 1 summarizes the whole process of C2E-MARL in pseudo-code. Besides, the auxiliary task used to update the encoder is summarized in Algorithm 2. In general, C2E-MARL ensembles multiple centralized critic networks scattered in the solution space, which provides more comprehensive action value estimation from various perspectives and facilitates policy updates. Secondly, contrastive representation learning is deployed as an auxiliary task for the underlying models to improve the representation learning capability, thus enhancing the overall performance. In summary, C2E-MARL is an efficient and straightforward approach that is competitive in complex MARL tasks.

5. Experiments

To empirically evaluate C2E-MARL, we conduct extensive experiments on three multi-agent scenarios compared with several state-of-the-art methods. Specifically, we investigate the following four questions.

- **Q1.** How does C2E-MARL perform against baselines in terms of sample efficiency and effectiveness?
- **Q2.** Has the ensemble module learned different characteristics? Does the way of combination in the ensemble module affect the performance?
- **Q3.** How does the auxiliary task, i.e., unsupervised contrastive learning, work for C2E-MARL?
- **Q4.** What influence do the various settings of the ensemble module have on C2E-MARL?
- **Q5.** What impacts do the ensemble size, the learning rate, and the dropout probability have on the performance of C2E-MARL?

5.1. Environment

We perform the experiments on three benchmark environments, i.e., Rover Tower, Cooperative Communication and Cooperative Treasure Collection. All environments are composed of agents and landmarks.

- **Rover Tower** (Iqbal & Sha, 2019): It consists of landmarks, rovers, and towers, in which the agents are randomly paired, e.g., the red circle in Fig. 2(a). Specifically, the tower guides the rover to navigate to its destination, i.e., a specific landmark. When an agent reaches the goal, the environment will give agents a '+10' reward.
- **Cooperative Communication (Co-Comm)** (Mordatch & Abbeel, 2018): It consists of landmarks and two cooperative agents, i.e., the listener and the speaker in Fig. 2(b). The speaker delivers landmark information to assist listener in navigating to destination. The reward is related to the distance between listener and landmark.
- **Cooperative Treasure Collection (Co-TC)** (Iqbal & Sha, 2019): It consists of collectors, treasures, and banks, as shown in Fig. 2(c). The collector first looks for treasures and then deposits them in the corresponding colored banks.

5.2. Baselines

To make a comparison, we consider the following six state-of-the-art methods as baselines:

- **MAAC** (Iqbal & Sha, 2019): Based on the CTDE framework, it incorporates a multi-head attention mechanism for extracting relevant information for action value estimation, which can reduce the interference of irrelevant information and further stabilize the training.

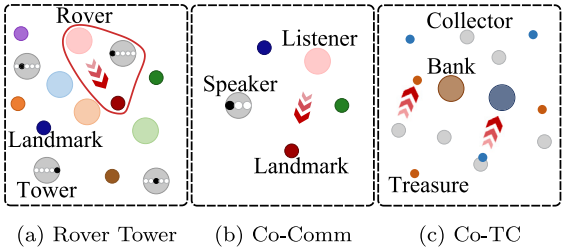


Fig. 2. Three multi-agent benchmark scenarios.

- **MAD3PG** (Li, Wang, Tian, Jia, & Zheng, 2020): It sets up the discrete distributional value function for MARL. In this way, it can reduce the Q-value estimation error, which is essential to guide efficient policy search.
- **MADDPG** (Lowe et al., 2017): It deploys DDPG, a single-agent RL method, upon the CTDE framework for multi-agent systems. The performance is further improved when using an ensemble of policies.
- **MMD-MIX** (Xu, Li et al., 2021): It combines distributional reinforcement learning for MARL by modifying the mixing network proposed in QMIX. Additionally, it utilizes the random ensemble mixture to approximate the Q-value.
- **QMIX** (Rashid et al., 2018): It serves the same purpose as VDN to decompose the team reward but provides an end-to-end joint action-value decomposition. Meanwhile, QMIX uses the LSTM to approximate the value function.
- **VDN** (Sunehag et al., 2018): It decomposes the team reward into the sum of individual rewards. To solve the challenge caused by partial observation, it uses the recurrent neural network to refer to historical information.
- **IQL** (Tan, 1993): It is the most straightforward method used for multi-agent systems, i.e., each agent is trained individually using advanced RL approaches.

5.3. Setup details

We conduct a variety of experiments on these scenarios to verify the validity of our method. Table 1 presents the detailed parameter settings of C2E-MARL, which are consistent in different experiments.

Evaluation Metrics. We mainly use three metrics for comparing the sample efficiency and overall performance of the proposed method, which are: (1) Convergence steps ($\#Ep$): It is used to assess the sample efficiency of the model. With comparable model performance, fewer steps during training indicate higher sample efficiency. (2) Average reward r_{avg}: This metric evaluates the overall performance of the model, with larger values indicating better performance. (3) Standard deviation of average reward r_{std}: It primarily measures the stability during model training; smaller values suggest less fluctuation during training.

5.4. Results and analysis (Q1,Q2)

To investigate **Q1**, we conduct the experiments in three multi-agent scenarios compared with the state-of-the-art approaches. We make the comparison in terms of sample efficiency and model effectiveness. In particular, model performance is measured by the online per-step average reward (r_{avg}) achieved by the agents. Sample efficiency is measured by how many episodes ($\#Ep$) it takes to achieve a particular level of performance (or how much reward can be obtained through a certain number of interactions). For a fair comparison, we set the model with the same hyperparameter for all methods. We perform experiments with five random seeds, whose learning curves are shown in Fig. 3 for intuitive comparison. The detailed results with the standard deviation are plotted individually in Fig. 4 for clarity.

Table 1
The hyper-parameter settings of C2E-MARL.

Category	Hyper-parameter	Setting
Actor	Hidden units of actor network	[128, 128]
	Learning rate of actor	0.001
	The optimizer of actor network	Adam
Critic	Hidden units of encoder	[128]
	Learning rate of critic	0.001
	The optimizer of critic network	Adam
	The size of linear layer in critic network	128
Others	Ensemble size K of critic network	5
	Dropout probability d_p	0.01
	Discount factor γ	0.99
	Soft update factor τ	0.995

Fig. 3a shows a salient efficiency gain when the agents are trained by C2E-MARL. This is what we expected: the contrastive-enhanced ensemble framework is beneficial for improving efficiency. Additionally, it is obvious that other baselines perform less well, and the possible reasons are as follows: (1) MAD3PG uses a distributional value function for efficient policy update, which relies on the setting of the distribution and is difficult to converge; (2) MADDPG suffers approximation error, which slows down the training and even fails in complex tasks; (3) MMD-MIX, QMIX and VDN may not be able to perform in tasks without global team reward and, therefore, perform poorly in Rover-Tower; and (4) IQL lacks information to make accurate estimates and therefore obstructs training.

Fig. 3b shows the overall performance in Cooperative Communication. Overall, C2E-MARL outperforms the existing methods in terms of sample efficiency and model performance. Specifically, the performance of MAAC and MADDPG is comparable to ours. MMD-MIX performs better and more efficient than QMIX and VDN because of the distribution and random ensemble mixture. The IQL performs instability due to the restricted observation. Besides, we zoom in the local of Fig. 3b and find that the sample efficiency of C2E-MARL (converge at $3K$) is nearly twice that of MAAC and MADDPG. The possible reason is that: C2E-MARL makes full use of data and provides multiple Q-value estimations which is conducive to reducing the estimation error. Moreover, data augmentation used in the contrastive learning auxiliary task is also beneficial to its efficiency. For the Cooperative Treasure Collection, C2E-MARL is still efficient, and its performance is comparable to that of the state-of-the-art methods. To summarize, C2E-MARL has consistently outperformed other methods in terms of sample efficiency and performance, which proves its effectiveness in tackling various tasks. Table 2 presents the statistical experiment results of C2E-MARL and its variants, i.e., C2E-MARL w/o En, C2E-MARL w/o CL, and C2E-MARL w/o En&CL (detailed in Section 5.5), compared with the baseline. We report their average reward r_{avg}, the standard deviation of reward r_{std}, and the convergence steps $\#Ep$ for a fair comparison. It is obvious that C2E-MARL is trained with the smaller number of episodes $\#Ep$ and obtains a higher reward r_{avg} with lower r_{std} in all scenarios, which demonstrates the superiority of C2E-MARL in terms of sample efficiency and effectiveness.

To answer **Q2**, we make a visualization on the ensemble critic network. Specifically, for a given state in the environment, e.g., Rover Tower, Fig. 5(a) visualizes the heat map of action value estimated by an ensemble of Q-value functions ($K = 5$), where the color corresponds to the Q-value. Note that the estimation of the same state–action pair is different. Besides, the correlation between each Q-value estimator is visualized in Fig. 5(b), where it can be observed that Q_{i1} has a stronger correlation with Q_{i2} and is largely irrelevant to Q_{i4}. It indicates that these underlying models scatter in the solution space and can approximate the true Q-value function from multiple perspectives.

Table 2
Each performance value are the average of 5 runs with random seeds, respectively. The statistic of reward r_{avg}, r_{std} and the number of the convergence steps $\#Ep$ are employed for quantitative comparison in terms of performance and sample efficiency. The best results are highlighted in boldface.

Method	Rover tower			Co-Comm			Co-TC		
	$r_{avg}\uparrow$	$r_{std}\downarrow$	$\#Ep\downarrow$	$r_{avg}\uparrow$	$r_{std}\downarrow$	$\#Ep\downarrow$	$r_{avg}\uparrow$	$r_{std}\downarrow$	$\#Ep\downarrow$
C2E-MARL	**4.89**	0.84	**≈ 9.8K**	**−0.26**	**0.05**	≈ 3K	**0.99**	**0.04**	**≈ 18K**
MAAC	4.56	1.25	≈ 20K	−0.30	0.14	≈ 6K	0.96	0.05	≈ 18K
MAD3PG	−0.06	1.21	–	−2.48	0.40	≈ 6.5K	0.16	0.14	–
MADDPG	0.19	0.91	–	−0.39	0.15	≈ 6K	−0.11	0.02	–
MMD-MIX	−0.28	0.95	–	−0.85	0.14	≈ 4.5K	−1.36	0.16	–
QMIX	0.22	1.07	≈ 12.5K	−1.50	0.34	≈ 10K	−1.04	0.23	–
VDN	−0.03	0.88	–	−1.87	0.47	≈ 3K	0.05	0.07	–
IQL	−1.68	**0.65**	–	−1.50	0.20	**≈ 2K**	0.06	0.07	–
C2E-MARL w/o En	4.33	1.12	≈ 15K	−0.42	0.10	≈ 2K	0.94	0.06	≈ 18K
C2E-MARL w/o CL	3.93	1.14	≈ 10K	−0.35	0.09	≈ 4K	0.96	0.05	≈ 18K
C2E-MARL w/o En&CL	3.71	1.15	≈ 10K	−0.45	0.09	≈ 2K	0.89	0.07	≈ 22.5K

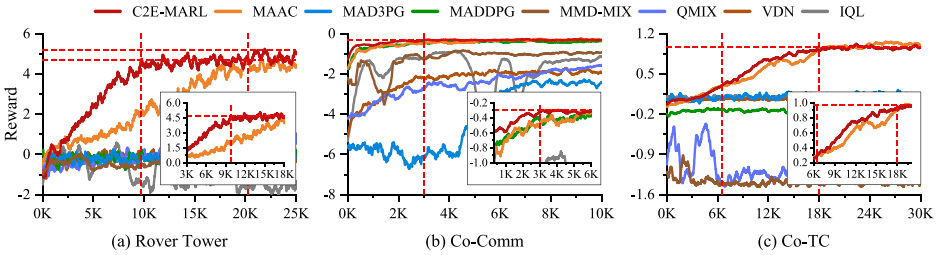


Fig. 3. The overall results on three multi-agent scenarios. The X-axis denotes the episode number and Y-axis denotes the per-step average reward (online). The partial magnifications are set up to present the differences more clearly.

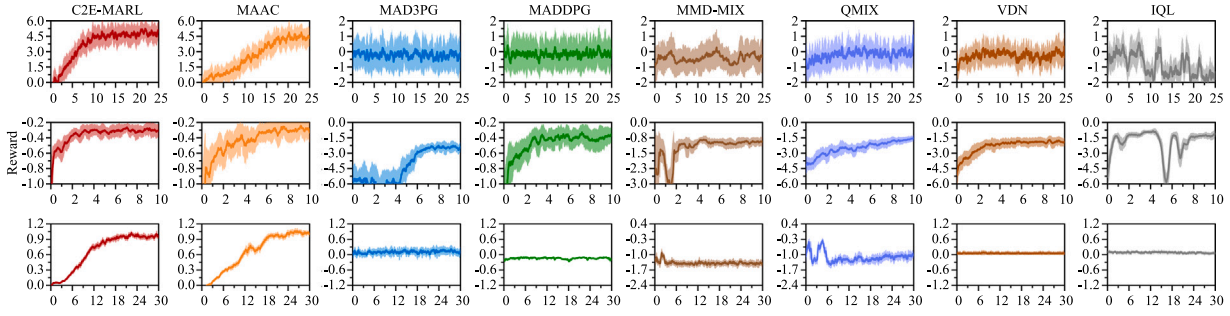


Fig. 4. The overall results on three multi-agent scenarios, i.e., Rover Tower (top panel), Cooperative Communication (middle panel) and Cooperative Treasure Collection (bottom panel). The X-axis denotes the episode number and Y-axis denotes the per-step average reward (online).

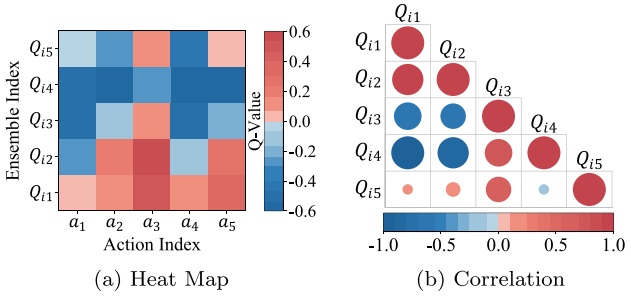


Fig. 5. The left panel is the heat map of the action value (given a state) estimating by multiple Q-value functions. The right panel is the correlation between an ensemble of critic networks. The color denotes correlation value, and the size of circle denotes the degree of correlation/irrelevance.

Besides, we investigate whether the way of calculating the target Q-value in the ensemble module impacts the performance. Specifically, the variants of C2E-MARL used in the investigation are presented as follows:

- **C2E-MARL w/ Maxmin**: It minimizes over the ensemble critic networks as the target Q-value.
- **C2E-MARL w/ Average**: It takes the average of the ensemble critic networks as the target Q-value.

Both the average reward r_{avg} and the deviation r_{std} are listed in Table 3. We find that our methods, i.e., selecting the candidate module randomly from the ensemble critic network to calculate the target Q-value, can obtain the highest r_{avg}, i.e., achieve the best performance. C2E-MARL w/ Maxmin and C2E-MARL w/ Average are easily influenced by the "weak" critic network and further degrade the overall performance.

5.5. Ablation study (Q3, Q4)

Our proposed C2E-MARL has two key components: ensemble critic networks and contrastive-enhanced modules. In this section, we will validate their impacts on the performance via ablation experiments. We detail the variants of C2E-MARL as follows:

- **C2E-MARL w/o En**: We replace an ensemble of critic networks with only one, and still deploy a contrast learning auxiliary task at the upstream network.

Table 3
The performance comparisons with C2E-MARL's variants, which are different in the way of calculating the Q-target. Each performance value are the average of 5 runs with different random seeds, respectively. It is expressed in the form of mean ± standard deviation, i.e., $r_{avg}(\uparrow) \pm r_{std}(\downarrow)$. The best results are highlighted in boldface.

$r_{avg} \pm r_{std}$ Environment / Method	Rover Tower	Co-Comm	Co-TC
C2E-MARL w/ Maxmin	4.61 ± 1.14	-0.32 ± 0.10	-0.03 ± 0.02
C2E-MARL w/ Average	4.59 ± 1.22	-0.31 ± 0.10	0.86 ± 0.06
C2E-MARL (Ours)	**4.89 ± 0.84**	**-0.26 ± 0.05**	**0.99 ± 0.04**

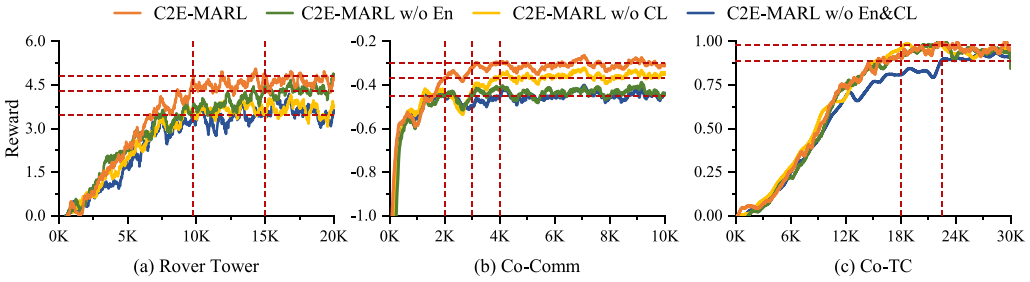


Fig. 6. The impact of different variants of C2E-MARL on three multi-agent scenarios. The X-axis denotes the episode number, and Y-axis denotes the per-step average reward (online). For clarity, we appropriately adjust the range of the X-axis.

- **C2E-MARL w/o CL**: We remove the auxiliary task for the underlying model of an ensemble of critic networks.
- **C2E-MARL w/o En&CL**: It is a version of the MARL algorithm that does not integrate ensemble learning and contrastive learning.

As shown in Fig. 6, C2E-MARL outperforms the other variants throughout these multi-agent scenarios. Table 2 presents the statistical experiment results of these variants. Specifically, contrastive learning can improve the efficiency (C2E-MARL w/o En), but the dropout layer used to generate positive samples may lose some features and degrade performance. The problem becomes more prominent, particularly as the task complexity increases. The ensemble learning framework (C2E-MARL w/o CL) is empirically beneficial for improving sample efficiency, and the improvement will be more salient enhanced by the contrastive representation learning auxiliary task. In general, the results imply that both the ensemble paradigm and contrastive learning auxiliary tasks are conducive to sample efficiency and model performance.

5.6. Parameter sensitivity study (**Q5**)

In this subsection, we will conduct parameter sensitivity experiments and analyze the impacts of three primary parameters, i.e., ensemble size K, learning rate lr and dropout probability d_p, on C2E-MARL. Specifically, the ensemble size K influences the complexity of the network structure and further influences the model's performance, the learning rate lr determines the step size for iteration update, and the dropout probability d_p impacts the data augmentation in contrastive learning. To investigate the effect of the parameters, we employ single-parameter sensitivity analysis by varying one parameter while fixing the others each time. Fig. 7 shows the variation in the model performance over different hyperparameter settings.

Figs. 7(a) and 7(b) show the performance of the model for different ensemble sizes and learning rates on Rover Tower and Co-Comm. There is a tendency for the model performance to improve and then degrade as the ensemble size increases, and the optimal ensemble size is approximately 4 to 6 for all scenarios. This suggests that the increase in ensemble size K makes the network structure more complex. When the complexity of the network structure is too great, it will affect the overall performance of the model, so we set the ensemble size of the model to $K = 5$. For learning rate, C2E-MARL achieves the best performance when $lr = 0.001$, and the optimal learning rate is approximately 0.0005 to 0.001 for all scenarios. Fig. 7(c) shows the performance with different dropout probabilities. The dropout layer is used to generate positive samples for contrastive learning by discarding some features, and its improper settings may degrade performance. Fig. 7(c) demonstrates that C2E-MARL performs better when $d_p = 0.1$, ensuring that it generates good positive instances for contrastive learning.

5.7. Further discussion

In this subsection, we analyze C2E-MARL in terms of sample efficiency and effectiveness. The proposed framework takes advantage of both ensemble learning and contrastive learning. It deploys multiple critic networks and forms an ensemble model for action value estimation, which can reduce the estimation error and provide trustworthy action value estimation. In this way, it is helpful to update the policy in the right direction efficiently and obtain a well-performing agent. In addition, contrastive learning is employed as an auxiliary task for the ensemble critic networks to learn the representation of the joint information. Compared with taking the raw embedding as the input for the critic network, C2E-MARL uses the dropout operation to augment the data to learn the joint information representation with contrastive learning, which helps improve the ensemble critic networks. Empirical results demonstrate that C2E-MARL achieves superior performance on benchmark scenarios with high sample efficiency.

6. Conclusion

In this paper, we explore the problem of sample efficiency in MARL and propose an efficient algorithm called C2E-MARL. It employs an ensemble of centralized critic networks for Q-value estimation, which reduces the estimation error by extracting features from multiple perspectives. Inspired by the "Cannikin Law", we deploy the contrastive learning auxiliary task to speed up the learning efficiency of underlying submodels by generating data and improving representation quality, which in turn improves the overall performance. Finally, we conduct substantial experiments and analyses to demonstrate that the proposed method outperforms state-of-the-art MARL methods in terms of both sample efficiency and effectiveness. For future work, we will explore more details of the optimizing process of C2E-MARL and apply it to more complex scenarios.

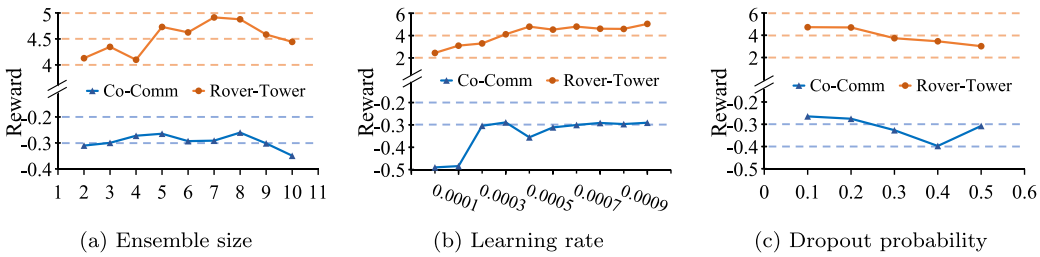


Fig. 7. Effect of ensemble size, learning rate and dropout probability to performance.

CRediT authorship contribution statement

Xinqi Du: Conceptualization, Validation, Methodology, Software, Writing – original draft. **Hechang Chen:** Writing – review & editing, Investigation, Supervision. **Yongheng Xing:** Conceptualization, Investigation. **Philip S. Yu:** Writing – review & editing, Supervision. **Lifang He:** Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

This work is partially supported in part by the International Cooperation Project of Jilin Province (20220402009GH); the National Natural Science Foundation of China (U2341229, 61976102 and U19A2065); the National Key R&D Program of China (2021ZD0112501 and 2021ZD0112502); Lehigh's Accelerator Foundation under grant No. S00010293, and the National Science Foundation (MRI 2215789 and IIS 1909879).

References

Claus, C., & Boutilier, C. (1998). The dynamics of reinforcement learning in cooperative multiagent systems. *AAAI/IAAI, 1998*(746–752), 2.

Duan, Y., Schulman, J., Chen, X., Bartlett, P. L., Sutskever, I., & Abbeel, P. (2016). RL^2: Fast reinforcement learning via slow reinforcement learning. arXiv preprint arXiv:1611.02779.

Elhaki, O., Shojaei, K., & Mehrmohammadi, P. (2022). Reinforcement learning-based saturated adaptive robust neural-network control of underactuated autonomous underwater vehicles. *Expert Systems with Applications*, *197*, Article 116714. http://dx.doi.org/10.1016/j.eswa.2022.116714.

Foerster, J., Assael, I. A., De Freitas, N., & Whiteson, S. (2016). Learning to communicate with deep multi-agent reinforcement learning. *Advances in Neural Information Processing Systems*, *29*.

Fu, H., Tang, H., Hao, J., Chen, C., Feng, X., Li, D., et al. (2021). Towards effective context for meta-reinforcement learning: an approach based on contrastive learning. In *Proceedings of the AAAI conference on artificial intelligence, Vol. 35* (pp. 7457–7465).

Fujimoto, S., Hoof, H., & Meger, D. (2018). Addressing function approximation error in actor-critic methods. In *International conference on machine learning* (pp. 1587–1596). PMLR.

Gao, T., Yao, X., & Chen, D. (2021). SimCSE: Simple contrastive learning of sentence embeddings. In *Empirical methods in natural language processing (EMNLP)*.

Haarnoja, T., Zhou, A., Abbeel, P., & Levine, S. (2018). Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *Proceedings of the 35th international conference on machine learning, Vol. 80* (pp. 1861–1870). PMLR.

He, K., Fan, H., Wu, Y., Xie, S., & Girshick, R. (2020). Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 9729–9738).

Iqbal, S., & Sha, F. (2019). Actor-attention-critic for multi-agent reinforcement learning. In *Proceedings of the 36th international conference on machine learning* (pp. 2961–2970).

Jiang, J., & Lu, Z. (2018). Learning attentional communication for multi-agent cooperation. *Advances in Neural Information Processing Systems*, *31*.

Kiran, B., Sobh, I., Talpaert, V., Mannion, P., Sallab, A., Yogamani, S., et al. (2021). Deep reinforcement learning for autonomous driving: A survey. *IEEE Transactions on Intelligent Transportation Systems*, 1–18.

Kumar, A., Fu, J., Soh, M., Tucker, G., & Levine, S. (2019). Stabilizing off-policy q-learning via bootstrapping error reduction. *Advances in Neural Information Processing Systems*, *32*, 11784–11794.

Lan, Q., Pan, Y., Fyshe, A., & White, M. (2019). Maxmin Q-learning: Controlling the estimation bias of Q-learning. In *International conference on learning representations*.

Laskin, M., Srinivas, A., & Abbeel, P. (2020). Curl: Contrastive unsupervised representations for reinforcement learning. In *International conference on machine learning* (pp. 5639–5650). PMLR.

Lee, K., Laskin, M., Srinivas, A., & Abbeel, P. (2021). Sunrise: A simple unified framework for ensemble learning in deep reinforcement learning. In *International conference on machine learning* (pp. 6131–6141). PMLR.

Li, H., Wang, X., Jia, F., Wu, Y., Zhang, J., & Qi, S. (2022). RLCFR: Minimize counterfactual regret by deep reinforcement learning. *Expert Systems with Applications*, *187*, Article 115953. http://dx.doi.org/10.1016/j.eswa.2021.115953.

Li, R., Wang, R., Tian, T., Jia, F., & Zheng, Z. (2020). Multi-agent reinforcement learning based on value distribution. In *Journal of physics: conference series*. Article 012017.

Liang, X., Wu, L., Li, J., Wang, Y., Meng, Q., Qin, T., et al. (2021). R-Drop: Regularized dropout for neural networks. In *NeurIPS*.

Lillicrap, T., Hunt, J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., et al. (2016). Continuous control with deep reinforcement learning. In *Proceedings of the 33rd international conference on machine learning* (pp. 1501–1506).

Liu, X., & Tan, Y. (2022). Feudal latent space exploration for coordinated multi-agent reinforcement learning. *IEEE Transactions on Neural Networks and Learning Systems*, 1–9. http://dx.doi.org/10.1109/TNNLS.2022.3146201.

Liu, Y., Wang, W., Hu, Y., Hao, J., Chen, X., & Gao, Y. (2020). Multi-agent game abstraction via graph attention neural network. In *Proceedings of the AAAI conference on artificial intelligence, Vol. 33* (pp. 7211–7218).

Liu, G., Zhang, C., Zhao, L., Qin, T., Zhu, J., Jian, L., et al. (2020). Return-based contrastive representation learning for reinforcement learning. In *International conference on learning representations*.

Lowe, R., Wu, Y. I., Tamar, A., Harb, J., Abbeel, O. P., & Mordatch, I. (2017). Multi-agent actor-critic for mixed cooperative-competitive environments. In *Advances in neural information processing systems, Vol. 30* (pp. 6379–6390).

Mordatch, I., & Abbeel, P. (2018). Emergence of grounded compositional language in multi-agent populations. In *Proceedings of the AAAI conference on artificial intelligence, Vol. 32*.

Nguyen, T. T., Nguyen, N. D., & Nahavandi, S. (2020). Deep reinforcement learning for multiagent systems: A review of challenges, solutions, and applications. *IEEE Transactions on Cybernetics*, *50*(9), 3826–3839. http://dx.doi.org/10.1109/TCYB.2020.2977374.

Noaeen, M., Naik, A., Goodman, L., Crebo, J., Abrar, T., Abad, Z. S. H., et al. (2022). Reinforcement learning in urban network traffic signal control: A systematic literature review. *Expert Systems with Applications*, *199*, Article 116830. http://dx.doi.org/10.1016/j.eswa.2022.116830.

Oliehoek, F. A., & Amato, C. (2016). The decentralized POMDP framework. In *A concise introduction to decentralized POMDPs* (pp. 11–32).

Oord, A. v. d., Li, Y., & Vinyals, O. (2018). Representation learning with contrastive predictive coding. arXiv preprint arXiv:1807.03748.

Osband, I., Blundell, C., Pritzel, A., & Van Roy, B. (2016). Deep exploration via bootstrapped DQN. In *Advances in neural information processing systems, Vol. 29* (pp. 4026–4034). Curran Associates, Inc..

Peer, O., Tessler, C., Merlis, N., & Meir, R. (2021). Ensemble bootstrapping for Q-learning. In *International conference on machine learning* (pp. 8454–8463). PMLR.

Rashid, T., Samvelyan, M., Schroeder, C., Farquhar, G., Foerster, J., & Whiteson, S. (2018). QMIX: Monotonic value function factorisation for deep multi-agent reinforcement learning. In *Proceedings of the 35th international conference on machine learning (ICML)* (pp. 4295–4304).

Schroff, F., Kalenichenko, D., & Philbin, J. (2015). Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 815–823).

Schulman, J., Levine, S., Abbeel, P., Jordan, M., & Moritz, P. (2015). Trust region policy optimization. In *Proceedings of the 32nd international conference on machine learning* (pp. 1889–1897).

Shavandi, A., & Khedmati, M. (2022). A multi-agent deep reinforcement learning framework for algorithmic trading in financial markets. *Expert Systems with Applications*, *208*, Article 118124. http://dx.doi.org/10.1016/j.eswa.2022.118124.

Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., et al. (2017). Mastering the game of go without human knowledge. *Nature*, 354–359.

Sukhbaatar, S., Fergus, R., et al. (2016). Learning multiagent communication with backpropagation. *Advances in Neural Information Processing Systems*, *29*, 2244–2252.

Sunehag, P., Lever, G., Gruslys, A., Czarnecki, W., Zambaldi, V., Jaderberg, M., et al. (2018). Value-decomposition networks for cooperative multi-agent learning. In *Proceedings of the 17th international conference on autonomous agents and multiagent systems (AAMAS' 18)* (pp. 2085–2087).

Tan, M. (1993). Multi-agent reinforcement learning: Independent versus cooperative agents. In *Proceedings of the 10th international conference on machine learning* (pp. 330–337).

Van Hasselt, H., Guez, A., & Silver, D. (2016). Deep reinforcement learning with double q-learning. In *Proceedings of the AAAI conference on artificial intelligence, Vol. 30*.

Wang, T., Wang, J., Zheng, C., & Zhang, C. (2019). Learning nearly decomposable value functions via communication minimization. arXiv preprint arXiv:1910.05366.

Wei, Q., Li, Y., Zhang, J., & Wang, F.-Y. (2022). VGN: Value decomposition with graph attention networks for multiagent reinforcement learning. *IEEE Transactions on Neural Networks and Learning Systems*, 1–14. http://dx.doi.org/10.1109/TNNLS.2022.3172572.

Xu, Z., Li, D., Bai, Y., & Fan, G. (2021). Mmd-mix: Value function factorisation with maximum mean discrepancy for cooperative multi-agent reinforcement learning. In *2021 international joint conference on neural networks (IJCNN)* (pp. 1–7). IEEE.

Xu, D., Zhu, F., Liu, Q., & Zhao, P. (2021). Improving exploration efficiency of deep reinforcement learning through samples produced by generative model. *Expert Systems with Applications*, *185*, Article 115680. http://dx.doi.org/10.1016/j.eswa.2021.115680.

Zhu, J., Xia, Y., Wu, L., Deng, J., Zhou, W., Qin, T., et al. (2022). Masked contrastive representation learning for reinforcement learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.