GREEDY-BASED VALUE REPRESENTATION FOR EFFICIENT COORDINATION IN MULTI-AGENT REINFORCEMENT LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Due to the representation limitation of the joint Q value function, multi-agent reinforcement learning (MARL) methods with linear or monotonic value decomposition can not ensure the optimal consistency (i.e. the correspondence between the individual greedy actions and the maximal true Q value), leading to instability and poor coordination. Existing methods focus on addressing the representation limitation through learning the complete expressiveness, which is impractical and may deteriorate the performance in complex tasks. In this paper, we introduce the True-Global-Max (TGM) condition for linear and monotonic value decomposition to achieve the optimal consistency directly, where the TGM condition can be ensured under the unique stability of the optimal greedy action. Therefore, we propose the greedy-based value representation (GVR), which stabilises the optimal greedy action via inferior target shaping and destabilises non-optimal greedy actions via superior experience replay. We conduct experiments on various benchmarks, where GVR significantly outperforms state-of-the-art baselines. Experiment results demonstrate that our method can ensure the optimal consistency under sufficient exploration and is more efficient than methods with complete expressiveness capability.

1 INTRODUCTION

Many real-world tasks can be naturally modelled as cooperative multi-agent reinforcement learning (MARL) problems, e.g., autonomous cars (Cao et al., 2012), traffic scheduling (Wu et al., 2020; Wang et al., 2020a) and robot systems (Vorotnikov et al., 2018; Nazarova & Zhai, 2020). Recently, by taking advantage of the deep learning technique, cooperative MARL has achieved great success on challenging tasks such as StarCraft unit micromanagement (Foerster et al., 2018). During the training of cooperative MARL, any available information is accessible for agents (Wang et al., 2020b), which is advantageous for centralised training. Meanwhile, due to the communication constraints (Wen et al., 2019; Balch & Arkin, 1994) in the physical world and the intractable joint action space as agent number grows, agents are expected to act in decentralised policies. As a result, the paradigm of centralised training with decentralised execution (CTDE) gains growing attention (Oliehoek et al., 2008; Foerster et al., 2016; Lowe et al., 2017).

Value decomposition is a popular approach to adopt CTDE in value based cooperative MARL. For example, VDN (Sunehag et al., 2017) and QMIX (Rashid et al., 2018) decompose the joint Q value function into individual utility functions through linear and monotonic factorization respectively. In MARL methods with value decomposition, agents are required to act individually according to the decomposed utility functions while achieving the best performance from the perspective of a team, which brings the challenge of optimal consistency: the individual greedy actions are expected to be correspondent with the maximal true Q values. To ensure the optimal consistency, two conditions are required to be met at each state. (1) Individual-Global-

Max (IGM) (Son et al., 2019): the joint greedy action is identical to the collection of individual greedy actions. (2) True-Global-Max (TGM): the true Q value of the joint greedy action is the maximal. It has been proved that the IGM condition holds both in VDN and QMIX (Son et al., 2019). However, VDN and QMIX can not ensure the TGM condition due to the representation limitation of the joint Q value function. As a result, they suffer from an underestimation of the optimal joint Q value function, leading to the potential risk of training instability and poor performance (Rashid et al., 2020).

Recent works concentrate on solving the problems that arise from the representation limitation through networks with complete expressiveness capability (CEC). QTRAN (Son et al., 2019) learns a Q value function with CEC and introduces two soft regularizations to approximate the IGM condition. WQMIX introduces an auxiliary network with CEC (Rashid et al., 2020) to distinguish samples with low expressive value and place less importance on them. QPLEX empowers the joint Q value function with CEC through a dual mixing network (Wang et al., 2020b). However, as the state space and the joint action space increase exponentially as the number of agents grows, it is impractical to learn the complete expressiveness in complicated MARL tasks, which may result in convergence difficulty and performance deterioration (Rashid et al., 2020). More details about related works are given in Appendix A.

In this work, we propose the Greedy-based Value Representation (GVR), which directly ensures the IGM and TGM conditions without CEC. The investigation of linear and monotonic value decomposition indicates that the TGM condition can be ensured under the unique stability of the optimal greedy action, where the stability of greedy actions depends both on reward shaping and sample distribution. Therefore, we reshape the target of valueless samples to stabilise the optimal greedy action and then propose a novel prioritized experience replay method to destabilise non-optimal greedy actions. Here, the sample distribution denotes the distribution of joint action samples in the training batch. Our approach outperforms state-of-the-art baselines in experiments of various benchmarks. The experiment results demonstrate that under sufficient exploration, GVR can ensure the optimal consistency in tasks with extreme reward shaping and large joint action spaces.

We have two contributions in this work. Firstly, we investigate and verify the requirements of the TGM condition for linear and monotonic value decomposition. Secondly, we propose the GVR method. Under sufficient exploration, GVR ensures the optimal consistency more efficiently than methods with CEC.

2 PRELIMINARIES

2.1 DEC-POMDP

We model a fully cooperative multi-agent task as a decentralised partially observable Markov decision process (Dec-POMDP) described by a tuple $\mathcal{G} =< S, U, P, r, Z, O, n, \gamma >$ (Guestrin et al., 2001; Oliehoek & Amato, 2016). $s \in S$ denotes the true state of the environment. At each time step, each agent $a \in A \equiv \{1, 2, \cdots, n\}$ receives a local observation $z^a \in Z$ produced by the observation function $O : S \times A \rightarrow Z$, and then chooses an individual action $u^a \in U$ according to a local policy $\pi^a(u^a|\tau^a) : T \times U \rightarrow [0, 1]$, where $\tau^a \in T \equiv (Z \times U)^*$ denotes the local action-observation history. The joint action of n agents $\mathbf{u}$ results in a shared reward $r(s, \mathbf{u})$ and a transition to the next state $s' \sim P(\cdot|s, \mathbf{u})$. $\gamma \in [0, 1)$ is a discount factor.

We denote the joint variable of group agents with bold symbols, e.g., the joint action $\boldsymbol{u} \in \boldsymbol{U} \equiv U^n$, the joint action-observation history $\boldsymbol{\tau} \in \boldsymbol{\mathcal{T}} \equiv T^n$, and the joint policy $\boldsymbol{\pi}(\mathbf{u}|\boldsymbol{\tau})$. The true Q value of $\boldsymbol{\pi}(\mathbf{u}_t|\boldsymbol{\tau}_t)$ is denoted by $\mathcal{Q}^{\boldsymbol{\pi}}(\boldsymbol{u}_t, s_t) = \mathbb{E}_{s_{t+1:\infty}, \boldsymbol{u}_{t+1:\infty}}[R_t|s_t, \boldsymbol{u}_t]$, where $R_t = \sum_{i=0}^{\infty} \gamma^i r_{t+1}$ is the discounted return. The action-state value function of agent a and the group of agents are defined as utility function $\mathcal{U}^a(u^a, o^a)$ and joint Q value function $\boldsymbol{Q}(\boldsymbol{u}, s)$ respectively.

2.2 OPTIMAL CONSISTENCY AND TGM CONDITION

For MARL with linear and monotonic value decomposition, the joint policy is decomposed into decentralised individual policies, where the exploration and Q value representation are significantly simplified. However, the simplification directly leads to the representation limitation (Rashid et al., 2020), where the individual greedy actions may be not correspond with the maximal true Q values, leading to non-optimal coordination. Here we define the correspondence between the individual greedy actions and the maximal true Q value as the optimal consistency.

Definition 1 (Optimal consistency). Given a set of utility functions $\{\mathcal{U}^1(o^1, u^1)), \cdots, \mathcal{U}^n(o^n, u^n)\}$, and the true Q value $\mathcal{Q}^{\boldsymbol{\pi}}(s, \boldsymbol{u})$ under a joint interactive policy $\boldsymbol{\pi}(s, \boldsymbol{u})$, if the following holds

$$\underset{\boldsymbol{u}}{argmax}\ \mathcal{Q}^{\boldsymbol{\pi}}(s, \boldsymbol{u}) = \{\underset{u^1}{argmax}\, \mathcal{U}^1(o^1, u^1), \cdots, \underset{u^n}{argmax}\, \mathcal{U}^n(o^n, u^n)\} \tag{1}$$

then we say the set of utility functions $\{(\mathcal{U}^1(o^1, u^1)), \cdots, \mathcal{U}^n(o^n, u^n)\}$ satisfies the optimal consistency. For simplicity, we ignore situations with non-unique optimal policies.

The optimal consistency ensures the correspondence between individual greedy actions and the team's best performance. To achieve the optimal consistency in MARL, we introduce two conditions: Individual-Global-Max (IGM) and True-Global-Max (TGM).

The IGM condition is proposed by QTRAN (Son et al., 2019), where the definition is $argmax_{\boldsymbol{u}} Q(s, \boldsymbol{u}) = \{argmax_{u^1}\ \mathcal{U}^1(o^1, u^1), \cdots, argmax_{u^n}\ \mathcal{U}^n(o^n, u^n)\}$. The IGM condition ensures the correspondence between individual greedy actions and the joint greedy action. To meet the optimal consistency, the correspondence between the joint greedy action and the maximal true Q value is required, for which we define the TGM condition:

Definition 2 (TGM). Given a joint value function $Q(s, \boldsymbol{u})$, and the true Q value $\mathcal{Q}^{\boldsymbol{\pi}}(s, \boldsymbol{u})$ under a joint interactive policy $\boldsymbol{\pi}(s, \boldsymbol{u})$, if the following holds

$$\underset{\boldsymbol{u}}{argmax}\ \mathcal{Q}^{\boldsymbol{\pi}}(s, \boldsymbol{u}) = \underset{\boldsymbol{u}}{argmax}\ Q(s, \boldsymbol{u}) \tag{2}$$

then we say the joint value function $Q(s, \boldsymbol{u})$ satisfies the TGM condition. For simplicity, we ignore situations with non-unique optimal policies.

3 ANALYSIS OF THE TGM CONDITION FOR LVD & MVD

Linear value decomposition (LVD) and monotonic value decomposition (MVD) are simple and naturally meet the IGM condition. In order to ensure the optimal consistency efficiently, we further investigate the requirements of the TGM condition for LVD and MVD. In this section, we first derive the expression of the joint Q value function for LVD and MVD under $\epsilon-$greedy visitation, where the expression is related to the greedy action. Therefore, we further define the stability of greedy actions, and propose the requirement on the stability to ensure the TGM condition for LVD and MVD.

3.1 EXPRESSION OF THE JOINT Q VALUE FUNCTION FOR LVD & MVD

Firstly, take two-agent full cooperation with linear value decomposition as an example. Through linear factorization, the joint Q value function $Q(s, u^1, u^2)$ is decomposed into two utility functions $Q(s, u_i^1, u_j^2) = \mathcal{U}^1(o^1, u_i^1) + \mathcal{U}^2(o^2, u_j^2)$, where $i, j \in [1, m]$, and $\{u_1, \cdots, u_m\}$ is the individual action space of each agent. For briefness, we denote $Q(s, u_i^1, u_j^2), \mathcal{U}^a(o^a, u_i^a)$ with $Q_{ij}, \mathcal{U}_i^a (a \in \{1, 2\})$ respectively. Under $\epsilon-$greedy

visitation, we have

$$\begin{aligned} \mathcal{U}_i^1 &= \frac{\epsilon}{m}\sum_{k=1}^{m}(\mathcal{Q}_{ki} - \mathcal{U}_k^2) + (1-\epsilon)(\mathcal{Q}_{ij^*} - \mathcal{U}_{j^*}^2) \\ \mathcal{U}_j^2 &= \frac{\epsilon}{m}\sum_{k=1}^{m}(\mathcal{Q}_{jk} - \mathcal{U}_k^1) + (1-\epsilon)(\mathcal{Q}_{i^*j} - \mathcal{U}_{i^*}^1) \end{aligned} \tag{3}$$

where i^*, j^* denote the individual greedy actions of agent 1, 2 respectively, and $\mathcal{Q}_{ij}$ is the true Q value. Through derivation provided in Appendix B, the joint Q values represented by true Q values is obtained

$$\begin{aligned} Q_{ij} =& \frac{\epsilon}{m}\sum_{k=1}^{m}(\mathcal{Q}_{ik} + \mathcal{Q}_{kj}) + (1-\epsilon)(\mathcal{Q}_{i^*j} + \mathcal{Q}_{ij^*}) - \frac{\epsilon^2}{m^2}\sum_{i=1}^{m}\sum_{j=1}^{m}\mathcal{Q}_{ij} \\ &- \frac{\epsilon(1-\epsilon)}{m}\sum_{k=1}^{m}(\mathcal{Q}_{i^*k} + \mathcal{Q}_{kj^*}) - (1-\epsilon)^2\mathcal{Q}_{i^*j^*} \end{aligned} \tag{4}$$

Verification of the expression can be found in Appendix G. For monotonic value decomposition, the expression of Q_{ij} is identical to Eq.4, which indicates the coefficient and bias introduced by monotonic factorization has no effect on the joint Q function. The derivation of Q_{ij} for monotonic value decomposition is provided in Appendix C. For situations with more than two agents, referring to the derivation process in Appendix B and C, the joint Q value function represented by the true Q values can also be obtained.

3.2 Requirements of the TGM Condition for LVD & MVD

From Eq.4, the joint Q values are related to the greedy action, which may change during training, leading to the instability of the joint Q values. Here we define the stability of the greedy action.

Definition 3 (Stability of greedy action). Given a joint interactive policy $\boldsymbol{\pi}(s, \boldsymbol{u})$, and a stable joint Q value function $Q(s, \boldsymbol{u})$ which has converged to the true Q value $\mathcal{Q}^{\boldsymbol{\pi}}(s, \boldsymbol{u})$ under $\boldsymbol{\pi}(s, \boldsymbol{u})$, if the following holds

$$\underset{\boldsymbol{u}}{argmax}\ Q(s, \boldsymbol{u}) = \underset{\boldsymbol{u}}{argmax}\ \boldsymbol{\pi}(s, \boldsymbol{u}) \tag{5}$$

then we say the greedy action $\boldsymbol{u}^* := argmax_{\boldsymbol{u}}\boldsymbol{\pi}(s, \boldsymbol{u})$ is stable, and $\boldsymbol{u}^* = argmax_{\boldsymbol{u}}Q(s, \boldsymbol{u})$.

8	-12	-12
-12	0	0
-12	0	6

(a) Payoff of the matrix game, where the optimal action is $\boldsymbol{u}_{opt} = argmax_{\boldsymbol{u}}\mathcal{Q}^{\boldsymbol{\pi}}(s, \boldsymbol{u}) = \{1, 1\}$.

-19.90	-10.04	-9.64
-10.04	-0.17	0.23
-9.64	0.23	**0.63**

(b) Q_{ij} under an unstable $\boldsymbol{u}^*$ ($\boldsymbol{u}^* = \{2, 2\} \neq argmax_{\boldsymbol{u}}Q_{ij}$).

7.40	-8.33	-7.93
-8.33	-24.06	-23.66
-7.93	-23.66	-23.26

(c) Q_{ij} which meets the TGM condition under the optimal and stable $\boldsymbol{u}^*$ ($\boldsymbol{u}^* = argmax_{\boldsymbol{u}^*}Q_{ij} = \{1, 1\} = \boldsymbol{u}_{opt}$).

-24.38	-14.52	-9.32
-14.52	-4.65	0.55
-9.32	0.5	**5.75**

(d) Q_{ij} which violates the TGM condition under a non-optimal but stable $\boldsymbol{u}^*$ ($\boldsymbol{u}^* = argmax_{\boldsymbol{u}}Q_{ij} = \{3, 3\} \neq \boldsymbol{u}_{opt}$).

Table 1: Calculation results of $Q_{ij}(i, j \in \{1, 2, 3\})$ in two-agent one-step matrix game with linear value decomposition, where $\epsilon = 0.2$. The introduction of the matrix game can be found in the experimental part. The numbers in bold represent the maximal joint Q values, and the greedy policy is marked with a pink background. The TGM condition is not ensured because there is a non-optimal but stable $\boldsymbol{u}^* = \{3, 3\}$.

Besides, for a greedy action $\boldsymbol{u}^*$, if $\boldsymbol{u}^* = argmax_{\boldsymbol{u}} \mathcal{Q}^{\boldsymbol{\pi}}(s, \boldsymbol{u})$, then we say the greedy action $\boldsymbol{u}^*$ is optimal. To ensure the TGM condition, the unique stability of the optimal $\boldsymbol{u}^*$ (i.e. **the optimal $\boldsymbol{u}^*$ is the only stable $\boldsymbol{u}^*$ under a given state**) is required. However, for LVD and MVD, there may be non-optimal but stable $\boldsymbol{u}^*$, and the optimal $\boldsymbol{u}^*$ may be unstable. An example is shown in Table1. The TGM condition is not ensured due to a non-optimal but stable $\boldsymbol{u}^* = \{3, 3\}$, where $\boldsymbol{u}^* = argmax_{\boldsymbol{u}} Q(s, \boldsymbol{u}) \neq argmax_{\boldsymbol{u}} \mathcal{Q}^{\boldsymbol{\pi}}(s, \boldsymbol{u})$.

The stability of $\boldsymbol{u}^*$ mainly depends on the true Q values and the exploration rate ϵ, as verified in Appendix G. The true Q values are determined by reward shaping, and the exploration rate is related to sample distribution. Therefore, we can achieve the unique stability of the optimal $\boldsymbol{u}^*$ through well-shaped rewards and improved sample distribution.

4 METHOD

In this section, we introduce the Greedy-based Value Representation (GVR). We first propose inferior target shaping (ITS) to stabilise the optimal greedy actions (optimal $\boldsymbol{u}^*$) under random reward shaping. Then we describe superior experience replay, which destabilises the non-optimal greedy actions (non-optimal $\boldsymbol{u}^*$) under ITS with sufficient exploration. Achieving the unique stability of optimal greedy action for linear and monotonic value decomposition, GVR ensures both the IGM and TGM condition under sufficient exploration.

4.1 INFERIOR TARGET SHAPING

For linear and monotonic value decomposition, the stability of the optimal $\boldsymbol{u}^*$ depends on reward shaping, which is determined by the environment and uncontrollable. To ensure the stability of the optimal $\boldsymbol{u}^*$ under random reward shaping, we consider reshaping the target of the joint Q value function without affecting the optimality. Since the true Q values of non-optimal samples are uninformative, we reshape the target of these samples through inferior target shaping (ITS). The ITS target $\mathcal{Q}^{its}$ is defined as

$$\mathcal{Q}^{its}(s, \boldsymbol{u}) = \begin{cases} \mathcal{Q}(s, \boldsymbol{u}^*) - \alpha|\mathcal{Q}(s, \boldsymbol{u}^*)| & \mathcal{Q}(s, \boldsymbol{u}) < \mathcal{Q}(s, \boldsymbol{u}^*) \ \ and \ \ \boldsymbol{u} \neq \boldsymbol{u}^* \\ \mathcal{Q}(s, \boldsymbol{u}) & others \end{cases} \tag{6}$$

where $\alpha \in (0, 1]$. The samples in the first case are called inferior samples. ITS target only remains the minimum information of non-optimality for inferior samples, which simplifies the representation. Besides, given the greedy action $\boldsymbol{u}^*$ and a random action $\boldsymbol{u}_s (\boldsymbol{u}_s \neq \boldsymbol{u}^*)$, assuming $\mathcal{Q}(s, \boldsymbol{u}^*) > 0$, we have

$$\Delta Q^{its}(s, \boldsymbol{u}_s) = Q^{its}(s, \boldsymbol{u}_s) - Q^{its}(s, \boldsymbol{u}^*) = n\mathcal{Q}(s, \boldsymbol{u}^*) \left[(e_Q + \alpha) \left(\frac{\epsilon}{m}\right)^{n-1} - \alpha(1 - \epsilon + \frac{\epsilon}{m})^{n-1} \right] \tag{7}$$

where $e_Q = \frac{\mathcal{Q}(s, \boldsymbol{u}_s) - \mathcal{Q}(s, \boldsymbol{u}^*)}{\mathcal{Q}(s, \boldsymbol{u}^*)}$. The proof can be found in Appendix D, and the calculation result is verified in the experimental part.

For the optimal $\boldsymbol{u}^*$, we have $\boldsymbol{u}^* = argmax_{\boldsymbol{u}} \mathcal{Q}^{\boldsymbol{\pi}}(s, \boldsymbol{u})$. Therefore, $e_Q < 0$ and $\Delta Q^{its}(s, \boldsymbol{u}_s) < 0$ holds for $\forall \boldsymbol{u}_s \neq \boldsymbol{u}^*$, i.e., **the optimal greedy action is always stable under ITS target**. For a non-optimal $\boldsymbol{u}^*$, to destabilise it, let $\Delta Q^{its}(s, \boldsymbol{u}_s) > 0$, we have

$$\epsilon > \frac{m}{\sqrt[n-1]{\frac{e_Q}{\alpha} + 1} + m - 1} \tag{8}$$

which indicates the non-optimal $\boldsymbol{u}^*$ become unstable under large exploration. When $\boldsymbol{u}_s = argmax_{\boldsymbol{u}} \mathcal{Q}^{\boldsymbol{\pi}}(s, \boldsymbol{u})$, we obtain the lower bound of ϵ (denoted by ϵ_0), which equals to the right side of Eq.8. However, as the number of agents and the size of individual action spaces increases, ϵ_0 grows close to 1, which is inapplicable in tasks with long episodes. To destabilise the non-optimal greedy actions under a reasonable ϵ_0, we further propose the superior experience replay.

4.2 SUPERIOR EXPERIENCE REPLAY

According to Eq.4, the coefficients in the expression of joint Q value function are constants, suggesting that the joint Q value can be viewed as an expectation of true Q values. From this perspective, the stability of $\boldsymbol{u}^*$ is also related to the sample distribution. To destabilise the non-optimal greedy actions under given ϵ, we consider applying weights on samples and improving the sample distribution.

We investigate the effect of sample weight on the stability of greedy actions for two-agent linear and monotonic value decomposition with ITS. The result indicates that a weight $w > 1$ on the optimal sample helps reduce ϵ_0. However, the proportion of the optimal samples continue to decrease as the joint action space grows, leading to the fluctuation in the joint Q value of the optimal sample. More details can be found in Appendix E. For situations with more than two agents, it is hard to obtain the accurate lower bound of w due to the complex expression of the joint Q value function. However, it is proved experimentally in WQMIX Rashid et al. (2020) that a relatively large weight on samples with good performance helps alleviate the underestimation of the optimal sample's joint Q value.

To raise the proportion of the optimal sample stably, we combine sample weight with the superior experience replay. We first introduce a critic $V(s)$, which approximates the true Q value of the joint greedy sample. The target of the critic is defined as

$$\mathcal{V}^{gvr}(s) = \begin{cases} \mathcal{Q}(s, \boldsymbol{u}) & V(s) < \mathcal{Q}(s, \boldsymbol{u}) \ \ or \ \ \boldsymbol{u} = \boldsymbol{u}^* \\ V(s) & others \end{cases} \tag{9}$$

Inspired by prioritized experience replay (Schaul et al., 2015; Zhang & Sutton, 2017), we introduce a superior buffer where the episodic trajectories are stored. The number of superior samples which satisfy $\mathcal{Q}(s, \boldsymbol{u}) > V(s)$ and $\boldsymbol{u} \neq \boldsymbol{u}^*$ within a trajectory is defined as the priority of the trajectory. In the superior buffer, trajectories are ranked according to their priorities. The training batch consists of two parts: trajectories randomly sampled from the replay buffer and the top k trajectories from the superior buffer. At the end of each episode, the trajectories sampled from the superior buffer will be put back after the update of their priorities. The working principle of GVR and the algorithm is given in Appendix F.

5 EXPERIMENTS

In this section, firstly we verify the stability of greedy action $\boldsymbol{u}^*$ for linear value decomposition (LVD) in matrix games, where we also test the improvements of GVR. Secondly, to evaluate the stability and scalability of our method, we test the performance of GVR in predator-prey tasks with extreme reward shaping. Thirdly, we conduct experiments on complex tasks of StarCraft multi-agent challenge (SMAC) (Samvelyan et al., 2019). Finally, we design ablation studies to investigate the improvement of GVR. Our method is compared with state-of-the-art baselines including QMIX (Rashid et al., 2018), QPLEX (Wang et al., 2020b), and WQMIX (Rashid et al., 2020). All experiments are conducted over 5 seeds.

5.1 ONE-STEP MATRIX GAME

Matrix game is a simple fully cooperative multi-agent task, where the shared reward is defined by a payoff matrix. In one-step matrix games, the true Q values are directly accessible from the payoff matrix, which is convenient for the verification of the TGM condition. In this subsection, a mlp network shared by individual agents is adopted for all experiments.

The stability of greedy actions for LVD. We conduct experiments on two-agent one-step matrix game, where we first verify the expression of Eq.4. We also evaluate the effect of reward shaping and ϵ on the stability of greedy actions for LVD. Detailed information and the experiment results are given in Appendix G.

Evaluation of GVR. We first evaluate the effect of ITS. A payoff matrices of size 3^4 is generated for VDN and ITS respectively, where the true Q values of non-optimal samples are random (over 5 seeds) in ITS matrix.

$$\mathcal{Q}_{vdn} = \begin{cases} 6(1+e_Q) & \boldsymbol{u}^* = \{0,0,0,0\} \\ 6 & \boldsymbol{u}^* = \{2,2,2,2\} \\ 6(1-\alpha) & others \end{cases} \quad \mathcal{Q}_{its} = \begin{cases} 6(1+e_Q) & \boldsymbol{u}^* = \{0,0,0,0\} \\ 6 & \boldsymbol{u}^* = \{2,2,2,2\} \\ random(-20,6) & others \end{cases} \tag{10}$$

where $e_Q = 0.3$, $\alpha = 0.1$. We measure $\Delta Q_{opt} = Q(0,0,0,0) - Q(2,2,2,2)$ for VDN and ITS trained with corresponding matrices, where $\epsilon = 0.1$ and the joint greedy action is fixed to $\boldsymbol{u}^* = \{2,2,2,2\}$. As shown in Fig.1(a), the tested $\Delta Q(s, \boldsymbol{u}_{opt})$ consists with the calculation result ($\Delta Q(s, \boldsymbol{u}_{opt}) = -1.951$) from Eq.7. Besides, the converged $\Delta Q(s, \boldsymbol{u}_{opt})$ of VDN+ITS with random reward shaping and VDN with fixed reward shaping is very close, which indicates that ITS target can prevent the optimal joint Q value from the interference of non-optimal samples' reward shaping.

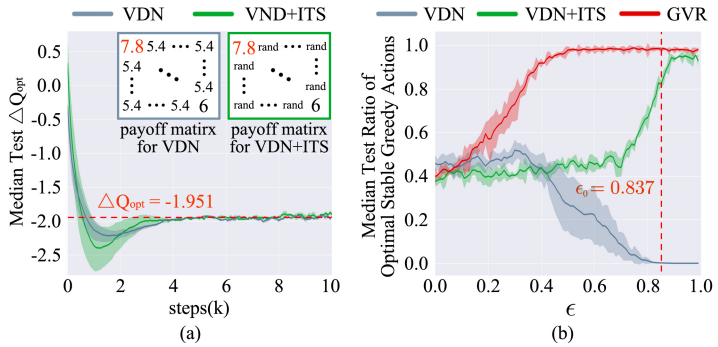

Figure 1: Evaluation of GVR in 4-agent matrix games. (a) Comparison of ΔQ_{opt} between VDN+ITS with random reward shaping and VDN with fixed reward shaping. The red dash line ($\Delta Q(s, \boldsymbol{u}_{opt}) = -1.951$) denotes the calculated result (from Eq.7). (b) Median test ratio of the optimal and stable $\boldsymbol{u}^*$ as ϵ grows. $y = 1$ represents the unique stability of the optimal $\boldsymbol{u}^*$, where The TGM condition is ensured. The red dash line ($\epsilon_0 = 0.837$) denotes the calculated lower bound of ϵ (from Eq.8) when VDN+ITS ensures the TGM condition.

We also evaluate the ratio of optimal and stable $\boldsymbol{u}^*$ for VDN, VDN+ITS, and GVR as ϵ grows. The experiments are conducted under random reward shaping, where the payoff matrices are generated according to Q_{its} over 5 seeds. For GVR, the size of the superior buffer and superior batch is 3 and 1 respectively, and the weight of the superior samples is 5. At each value of ϵ, 100 times of training are executed. Each training includes 2000 episodes. According to Fig.1(b), the optimal $\boldsymbol{u}^*$ is not ensured to be stable in VDN since $y = 0 (\exists \epsilon \in (0,1])$. For VDN+ITS, $y > 0 (\forall \epsilon \in (0,1])$, which indicates ITS helps to stabilise the optimal $\boldsymbol{u}^*$. Besides, VDN+ITS destabilises most of the non-optimal greedy actions when $\epsilon > 0.837$, which consists with the calculation results from Eq.8. GVR achieves the unique stability of optimal greedy action and ensures the TGM condition in a wild range of ϵ.

GVR vs methods with complete expressiveness capacity. To compare the efficiency between GVR and methods with complete expressiveness capacity, we evaluate GVR, QPLEX, and CWQMIX in matrix games with different scales of action space. Similar to Q_{its} in Eq.10, we generate random matrices of size 3^2, 6^3, and 12^4 over 5 seeds, where the first and the last element are set to be 8 and 6 respectively. For each matrix, the methods are trained for 10k steps, where $\epsilon = 0.2$. The experimental settings for GVR is the same as previous. The sample weight $w = 0.5$ for CWQMIX.

From Fig.2, in the last two tasks ($U^n = 6^3$ and $U^n = 12^4$), the representation errors of CWQMIX and QPLEX do not decrease during training, which suggests that they are unable to learn the complete expressiveness

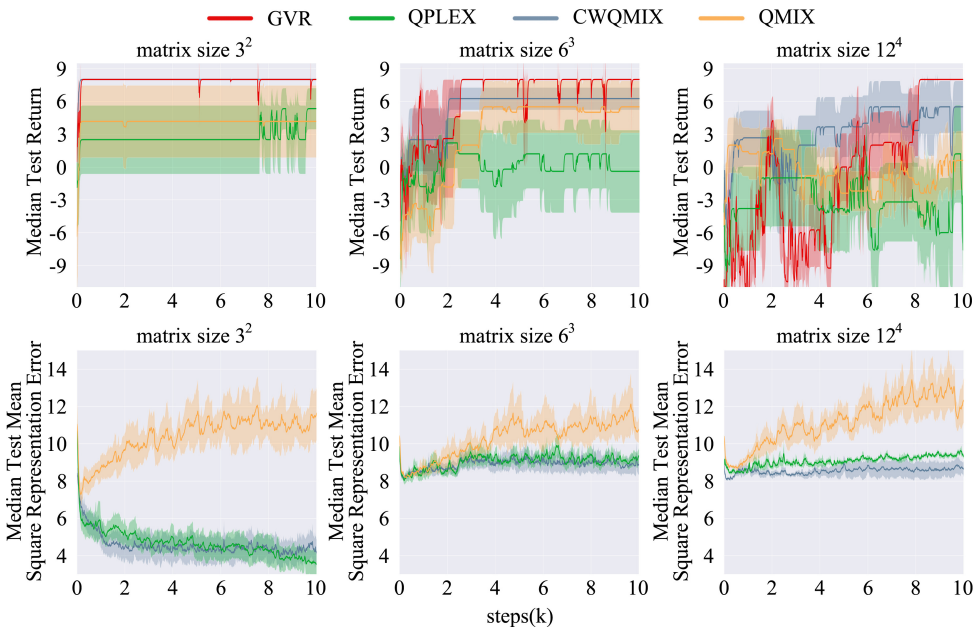

Figure 2: GVR vs methods with complete expressiveness capacity in matrix games.

within given steps. We do not measure the representation error of GVR because the representation target is reshaped by ITS. No need for complete expressiveness, GVR is more efficient and shows better performance than methods with complete expressiveness capability.

5.2 Predator-Prey

In this subsection, we compare GVR with state-of-the-art baselines in the predator-prey tasks (Böhmer et al., 2020). In our experiments, 8 predators are trained to capture 8 preys, where the preys are controlled by random policies. The reward is shared by all agents, and a punishment to evaluate the relative overgeneralisation (Panait et al., 2006; Wei et al., 2018) is applied to it when only a single agent capture a prey. A rnn network shared by individual agents is adopted in this part.

Our experiments are carried out under 3 punishment values. We adopt the default settings for baselines (Rashid et al., 2020). For GVR, $\alpha = 0.3$, the size of the superior buffer and superior batch is 200 and 8 respectively, and the weight of the superior samples is 5. Under all punishments, each algorithm is trained for 1e6 steps, with ϵ damping from 1 to 0.05 during the first 5e4 steps. From Fig.3, VDN and QMIX fail in all experiments, and converge steadily to a non-optimal greedy action, where all agents tend to avoid the preys. QPLEX also fails in spite of its complete expressiveness capacity. WQMIX can only solve the task with a small punishment of -2, while GVR is able to solve the tasks under all punishments.

5.3 StarCraft Multi-agent Challenge

StarCraft multi-agent challenge (SMAC) is a popular benchmark for the evaluation of MARL algorithms. We compare GVR with baselines in various difficult tasks of SMAC. The game version is 69232. We adopt the default settings for baselines. For GVR, $\alpha = 0.2$, the weight of the superior samples is 2, and the settings of the superior buffer is the same as the predator-prey. Each algorithm is trained for 2e6 steps in MMM2,

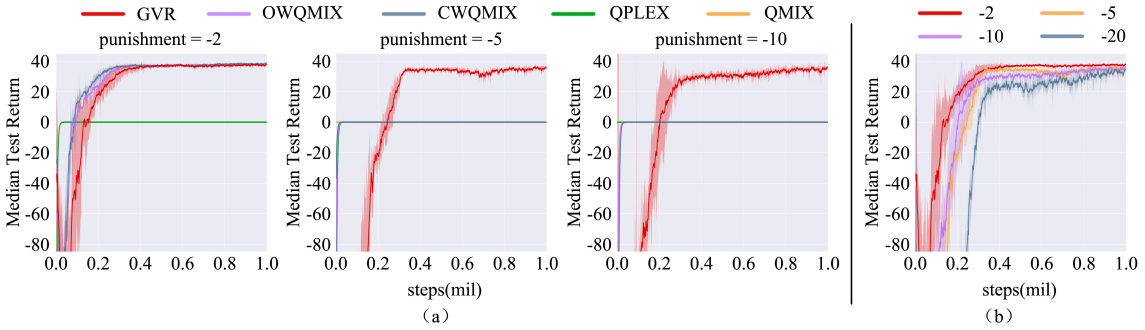

Figure 3: Experiment results on predator-prey. (a) Comparison between GVR and baselines. (b) Performance of GVR with different punishments.

2c_vs_64_zg and 6h_vs_8z, with ϵ damping from 1 to 0.05 during the first 5e4 steps. Especially, in 6h_vs_8z, each algorithm is trained for 5e6 steps, with ϵ damping from 1 to 0.05 during the first 1e6 steps.

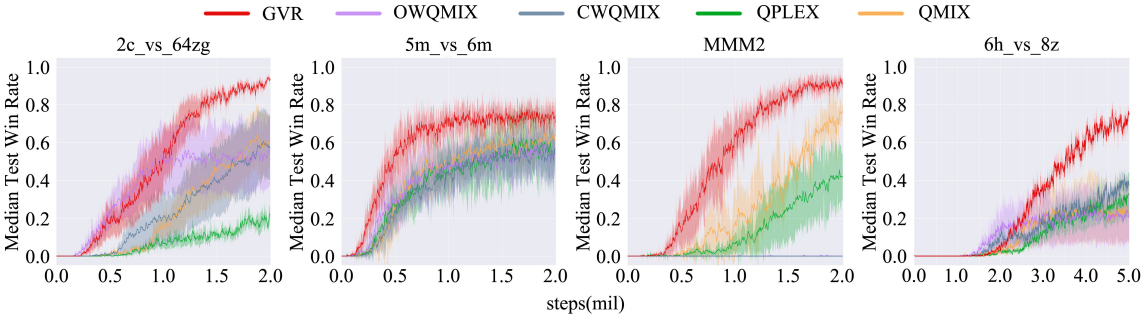

Figure 4: Median test win rate vs training steps.

From Fig.4, GVR shows the best performance. Different from predator-prey with abnormal punishments, the reward shaping in SMAC is more reasonable, where the linear and monotonic value decomposition can meet the TGM condition approximately. However, due to the difficulty of complete expressiveness, the algorithms with full representation expressiveness capability (QPLEX, OWQMIX, CWQMIX) do not perform better than QMIX. Ablation studies on GVR are given in Appendix H.

6 Conclusion and Future Work

This paper discusses the conditions of the optimal consistency for value decomposition, which is an important issue in value-based MARL with the CTDE paradigm. To ensure the optimal consistency efficiently, we introduce the TGM condition for linear and monotonic value decompositions, where the TGM condition can be ensures under the unique stability of the optimal greedy action. We further propose the GVR method, which is verified to ensure the optimal consistency under sufficient exploration. In experiments, GVR outperforms state-of-the-art baselines. The results demonstrate GVR is more efficient than methods with complete expressiveness capability. However, GVR is unable to ensure the TGM condition in hard exploration tasks, where the superior samples are difficult to obtain. We are interested in the combination of GVR with efficient exploration approaches (Gupta et al., 2021; Wang et al., 2019) in future work.

REFERENCES

Tucker Balch and Ronald C Arkin. Communication in reactive multiagent robotic systems. *Autonomous robots*, 1(1):27–52, 1994.

Wendelin Böhmer, Vitaly Kurin, and Shimon Whiteson. Deep coordination graphs. In *International Conference on Machine Learning*, pp. 980–991. PMLR, 2020.

Yongcan Cao, Wenwu Yu, Wei Ren, and Guanrong Chen. An overview of recent progress in the study of distributed multi-agent coordination. *IEEE Transactions on Industrial informatics*, 9(1):427–438, 2012.

Jakob Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. Counterfactual multi-agent policy gradients. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

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

Carlos Guestrin, Daphne Koller, and Ronald Parr. Multiagent planning with factored mdps. In *NIPS*, volume 1, pp. 1523–1530, 2001.

Tarun Gupta, Anuj Mahajan, Bei Peng, Wendelin Böhmer, and Shimon Whiteson. Uneven: Universal value exploration for multi-agent reinforcement learning. In *International Conference on Machine Learning*, pp. 3930–3941. PMLR, 2021.

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

Anaid V Nazarova and Meixin Zhai. The application of multi-agent robotic systems for earthquake rescue. In *Robotics: Industry 4.0 Issues & New Intelligent Control Paradigms*, pp. 133–146. Springer, 2020.

Frans A Oliehoek and Christopher Amato. *A concise introduction to decentralized POMDPs*. Springer, 2016.

Frans A Oliehoek, Matthijs TJ Spaan, and Nikos Vlassis. Optimal and approximate q-value functions for decentralized pomdps. *Journal of Artificial Intelligence Research*, 32:289–353, 2008.

Liviu Panait, Sean Luke, and R Paul Wiegand. Biasing coevolutionary search for optimal multiagent behaviors. *IEEE Transactions on Evolutionary Computation*, 10(6):629–645, 2006.

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

Tabish Rashid, Gregory Farquhar, Bei Peng, and Shimon Whiteson. Weighted qmix: Expanding monotonic value function factorisation for deep multi-agent reinforcement learning. *arXiv preprint arXiv:2006.10800*, 2020.

Mikayel Samvelyan, Tabish Rashid, Christian Schroeder De Witt, Gregory Farquhar, Nantas Nardelli, Tim GJ Rudner, Chia-Man Hung, Philip HS Torr, Jakob Foerster, and Shimon Whiteson. The starcraft multi-agent challenge. *arXiv preprint arXiv:1902.04043*, 2019.

Tom Schaul, John Quan, Ioannis Antonoglou, and David Silver. Prioritized experience replay. *arXiv preprint arXiv:1511.05952*, 2015.

Kyunghwan Son, Daewoo Kim, Wan Ju Kang, David Earl Hostallero, and Yung Yi. Qtran: Learning to factorize with transformation for cooperative multi-agent reinforcement learning. In *International Conference on Machine Learning*, pp. 5887–5896. PMLR, 2019.

Peter Sunehag, Guy Lever, Audrunas Gruslys, Wojciech Marian Czarnecki, Vinicius Zambaldi, Max Jaderberg, Marc Lanctot, Nicolas Sonnerat, Joel Z Leibo, Karl Tuyls, et al. Value-decomposition networks for cooperative multi-agent learning. *arXiv preprint arXiv:1706.05296*, 2017.

Sergey Vorotnikov, Konstantin Ermishin, Anaid Nazarova, and Arkady Yuschenko. Multi-agent robotic systems in collaborative robotics. In *International Conference on Interactive Collaborative Robotics*, pp. 270–279. Springer, 2018.

Enshu Wang, Rong Ding, Zhaoxing Yang, Haiming Jin, Chenglin Miao, Lu Su, Fan Zhang, Chunming Qiao, and Xinbing Wang. Joint charging and relocation recommendation for e-taxi drivers via multi-agent mean field hierarchical reinforcement learning. *IEEE Transactions on Mobile Computing*, 2020a.

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

Tonghan Wang, Jianhao Wang, Yi Wu, and Chongjie Zhang. Influence-based multi-agent exploration. *arXiv preprint arXiv:1910.05512*, 2019.

Ermo Wei, Drew Wicke, David Freelan, and Sean Luke. Multiagent soft q-learning. In *2018 AAAI Spring Symposium Series*, 2018.

Jiayan Wen, Peng Xu, Chen Wang, Guangming Xie, and Yuan Gao. Distributed event-triggered circle formation control for multi-agent systems with limited communication bandwidth. *Neurocomputing*, 358: 211–221, 2019.

Tong Wu, Pan Zhou, Kai Liu, Yali Yuan, Xiumin Wang, Huawei Huang, and Dapeng Oliver Wu. Multi-agent deep reinforcement learning for urban traffic light control in vehicular networks. *IEEE Transactions on Vehicular Technology*, 69(8):8243–8256, 2020.

Shangtong Zhang and Richard S Sutton. A deeper look at experience replay. *arXiv preprint arXiv:1712.01275*, 2017.