APPENDIX

Anonymous authors
Paper under double-blind review

A RELATED WORK

We mainly introduce recent value based MARL works with CTDE paradigm. Value decomposition is a popular approach for credit assignment in fully cooperative MARL methods with CTDE paradigm. VDN (Sunehag et al., 2017) learns a joint Q value function based on a share reward function. In VDN, where the joint Q value function is linearly factorized into individual utility functions. By contrast, QMIX (Rashid et al., 2018) substitutes the linear factorization with a monotonic factorization, where the weights and bias are produced from the global state through a mixing network. Based on QMIX, SMIX (Wen et al., 2020) replaces the TD(0) Q-learning target with a TD(λ) SARSA target. Qatten (Yang et al., 2020b) adds an attention network before the mixing network of QMIX. QPD (Yang et al., 2020a) decomposes the joint Q value function with the integrated gradient attribution technique, which directly decomposes the joint Q-values along trajectory paths to assign credits for agents. However, due to the representation limitation of the joint Q value function, these methods suffer from the relative overgeneralization. As a result, they can not guarantee the optimal coordination.

Some of recent works try to solve the representation limitation directly through joint Q value function with complete expressiveness capacity. QTRAN learns a joint Q value function with complete expressiveness capacity and introduces two soft regularizations to approximate the IGM condition. QPLEX (Wang et al., 2020) achieves the complete expressive under IGM condition theoretically through a dueling mixing network, where the complete expressiveness capacity is introduced by the mixing of individual advantage functions. 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.

The other works improve the coordination from different perspectives. WQMIX (Rashid et al., 2020) tries to solve the underestimation of the optimal joint values that arise from the representation limitation, where an auxiliary network with complete expressiveness capacity is applied to distinguishes samples with low expressive values. By placing a predefined weight on these samples, WQMIX can alleviate the underestimation of optimal joint Q values. According to Appendix E, a relative higher weight on the superior samples helps to eliminate non-optimal stable points. Therefore, WQMIX is effective to overcome relative overgeneralization to some degree, which is verified by our experiments on predator-prey. However, the joint Q value function under monotonic factorization depends heavily on the reward function, which is unavailable and task-specific. As a result, a heuristic weight has to be adopted in WQMIX, which can not guarantee the optimal coordination.

MAVEN (Mahajan et al., 2019) focuses on the poor exploration that arises from the representation limitation and introduces a latent space for hierarchical control, which achieves temporally extended exploration. UneVEn (Gupta et al., 2021) solves the target task by learning a set of related tasks simultaneously with a linear decomposition of universal successor features, which improves the joint exploration. Both methods raise the proportion of superior samples through an improved joint exploration, which helps to eliminate non-optimal stable points according to Eq.8. However, it requires an enormous exploration to raise the proportion of superior samples under large joint action space, which may reduce the sample efficiency. In practice, both methods apply small noise for joint exploration, where the proportion of superior samples increases slightly. As a result, both methods are insufficient to ensure the optimal coordination.

related works	IGM condition	TGM condition
IQL	No	No
VDN	Yes	No
QMIX	Yes	No
SMIX	Yes	No
Qatten	Yes	No
QDP	No	No
QTRAN	No	Yes
MAVEN	Yes	No
UneVEn	Yes	No
WQMIX	Yes	No
QPLEX	Yes	Yes
GVR(ours)	Yes	Yes

Table 1: Whether related works ensure the IGM and TGM condition.

B JOINT Q VALUES REPRESENTED BY TRUE Q VALUES FOR TWO-AGENT LINEAR VALUE DECOMPOSITION

B.1 DERIVATION

Consider a two-agent fully cooperative task without experience replay. the joint Q value function $Q(u_i^1, u_j^2, \boldsymbol{\tau})$ is linearly factorized into two utility functions $\mathcal{U}^1(u_i^1, \tau^1)$ and $\mathcal{U}^2(u_j^2, \tau^1)$.

$$Q(u_i^1, u_j^2, \boldsymbol{\tau}) = \mathcal{U}^1(u_i^1, \tau^1) + \mathcal{U}^2(u_j^2, \tau^2) \tag{1}$$

where $u_i^1, u_j^2 \in \{u_1, \cdots, u_m\}$ denote the individual actions of agent 1,2 respectively. $\{u_1, \cdots, u_m\}$ is the discrete individual action space. Specially, we denote the individual greedy action of agent 1,2 with $u_{i^*}^1$, $u_{j^*}^2$ respectively. For briefness, we denote $Q(u_i^1, u_j^2, \boldsymbol{\tau}), \mathcal{U}^a(u_i^a, \tau^a)$ with $Q_{ij}, \mathcal{U}_i^a (a \in \{1, 2\})$ respectively. Under $\epsilon-$greedy visitation, we have

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

where $\mathcal{Q}_{ij}$ is the true Q value. The sum of two utility functions over all actions equals to

$$\begin{aligned} \sum_{i=1}^{m} \mathcal{U}_i^1 + \sum_{j=1}^{m} \mathcal{U}_j^2 = & \frac{\epsilon}{m} \left[\sum_{i=1}^{m} \sum_{k=1}^{m} \mathcal{Q}_{ik} + \sum_{j=1}^{m} \sum_{k=1}^{m} \mathcal{Q}_{kj} - m \sum_{k=1}^{m} (\mathcal{U}_k^1 + \mathcal{U}_k^2) \right] \\ & + (1-\epsilon) \left[\sum_{i=1}^{n} (\mathcal{Q}_{ij^*} + \mathcal{Q}_{i^*j}) - m(\mathcal{U}_{i^*}^1 + \mathcal{U}_{j^*}^2) \right] \end{aligned} \tag{3}$$

Notice that $\mathcal{U}_{i^*}^1 + \mathcal{U}_{j^*}^2 = Q_{i^*j^*}$, and $\sum_{i=1}^{m} \sum_{k=1}^{m} \mathcal{Q}_{ik} = \sum_{j=1}^{m} \sum_{k=1}^{m} \mathcal{Q}_{kj} = \sum_{i=1}^{m} \sum_{j=1}^{m} \mathcal{Q}_{ij}$, we have

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

According to Eq.2 and Eq.4, for $\forall i, j \in [1, m]$, the joint Q value function equals to

$$\begin{aligned} Q_{ij} &= \mathcal{U}_i^1 + \mathcal{U}_j^2 \\ &= \frac{\epsilon}{m}\left[\sum_{k=1}^{m}(\mathcal{Q}_{ik}+\mathcal{Q}_{kj}) - \sum_{k=1}^{m}(\mathcal{U}_k^1+\mathcal{U}_k^2)\right] + (1-\epsilon)(\mathcal{Q}_{i^*j}+\mathcal{Q}_{ij^*}-Q_{i^*j^*}) \\ &= \frac{\epsilon}{m}\sum_{k=1}^{m}(\mathcal{Q}_{ik}+\mathcal{Q}_{kj}) + (1-\epsilon)(\mathcal{Q}_{i^*j}+\mathcal{Q}_{ij^*}) - \frac{2\epsilon^2}{m^2(1+\epsilon)}\sum_{i=1}^{m}\sum_{j=1}^{m}\mathcal{Q}_{ij} - \frac{1-\epsilon}{1+\epsilon}Q_{i^*j^*} \\ &\quad - \frac{\epsilon(1-\epsilon)}{m(1+\epsilon)}\sum_{k=1}^{m}(\mathcal{Q}_{i^*k}+\mathcal{Q}_{kj^*}) \end{aligned} \tag{5}$$

Notice that Q_{ij} is related to the joint greedy Q value $Q_{i^*j^*}$. In order to remove it, we put i^* and j^* into Eq.5.

$$Q_{i^*j^*} = \frac{\epsilon^2}{m}\sum_{k=1}^{m}(\mathcal{Q}_{i^*k}+\mathcal{Q}_{kj^*}) - \frac{\epsilon^2}{m^2}\sum_{i=1}^{m}\sum_{j=1}^{m}\mathcal{Q}_{ij} + (1-\epsilon^2)\mathcal{Q}_{i^*j^*} \tag{6}$$

Substituting Eq.6 into Eq.5, the joint Q values can be represented by true Q values as

$$\begin{aligned} Q_{ij} =& \frac{\epsilon}{m}\sum_{k=1}^{m}(\mathcal{Q}_{ik}+\mathcal{Q}_{kj}) + (1-\epsilon)(\mathcal{Q}_{i^*j}+\mathcal{Q}_{ij^*}) - \frac{\epsilon^2}{m^2}\sum_{i=1}^{m}\sum_{j=1}^{m}\mathcal{Q}_{ij} \\ &- \frac{\epsilon(1-\epsilon)}{m}\sum_{k=1}^{m}(\mathcal{Q}_{i^*k}+\mathcal{Q}_{kj^*}) - (1-\epsilon)^2\mathcal{Q}_{i^*j^*} \end{aligned} \tag{7}$$

B.2 VERIFICATION

We verify the expression of Eq.7 in a two-agent matrix game, where the payoff matrix is shown in Table2(a). Since the episode length is 1, an mlp shared by two agents is adopted as the policy network. The policy network is trained for 200 iterations (100 episodes per iteration) over 5 seeds. According to Table2(b) and 2(c). There are two stable points, which consists with our calculation. The error of joint Q values between calculation and test is lower than 3%.

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

(a)

7.40 (7.38±0.02)	-8.33 (-8.27±0.12)	-7.93 (-7.86±0.13)
-8.33 (-8.33±0.18)	-24.06 (-23.87±0.09)	-23.66 (-23.56±0.17)
-7.93 (-7.93±0.09)	-23.66 (-23.53±0.12)	-23.26 (-23.19±0.20)

(b)

-24.38 (-24.34±0.25)	-14.52 (-14.52±0.11)	-9.32 (-9.43±0.15)
-14.52 (-14.47±0.18)	-4.65 (-4.65±0.11)	0.55 (0.54±0.08)
-9.32 (-9.28±0.21)	0.55 (0.56±0.12)	**5.75** (5.75±0.09)

(c)

Table 2: Verification of calculated stable points for two-agent LVD. (a) The payoff matrix. (b),(c) Comparison between calculation and test, where the test results are shown in parentheses. The numbers in bold denote the max joint Q values, and the greedy policy is marked with a pink background.

C JOINT Q VALUES REPRESENTED BY TRUE Q VALUES FOR TWO-AGENT MONOTONIC VALUE DECOMPOSITION

For two-agent monotonic value decomposition, the joint Q value function is decomposed as $Q_{ij} = \omega_1(s)\mathcal{U}_i^1 + \omega_2(s)\mathcal{U}_j^2 + V(s)$, where ω_1 and ω_2 are the coefficients of $\mathcal{U}_i^1$ and $\mathcal{U}_j^2$ respectively. $V(s)$ is the bias, which is the same for all joint actions under a given state. For brevity, we omit the input. Referring to Eq.2, the individual utility functions with coefficients equal to

$$\begin{aligned}
\omega_1\mathcal{U}_i^1 &= \frac{\epsilon}{m}\sum_{k=1}^{m}\left[\mathcal{Q}_{ik} - \omega_2\mathcal{U}_k^2 - V\right] + (1-\epsilon)\left[\mathcal{Q}_{ij^*} - \omega_2\mathcal{U}_{j^*}^2 - V\right] \\
&= \frac{\epsilon}{m}\sum_{k=1}^{m}\left[\mathcal{Q}_{ik} - \omega_2\mathcal{U}_k^2\right] + (1-\epsilon)\left[\mathcal{Q}_{ij^*} - \omega_2\mathcal{U}_{j^*}^2\right] - V \\
\omega_2\mathcal{U}_j^2 &= \frac{\epsilon}{m}\sum_{k=1}^{m}\left[\mathcal{Q}_{kj} - \omega_1\mathcal{U}_k^1 - V\right] + (1-\epsilon)\left[\mathcal{Q}_{i^*j} - \omega_1\mathcal{U}_{i^*}^1 - V\right] \\
&= \frac{\epsilon}{m}\sum_{k=1}^{m}\left[\mathcal{Q}_{ik} - \omega_2\mathcal{U}_k^2\right] + (1-\epsilon)\left[\mathcal{Q}_{ij^*} - \omega_2\mathcal{U}_{j^*}^2\right] - V
\end{aligned} \tag{8}$$

Referring to the derivation of Eq.4, we have

$$\sum_{k=1}^{m}(\omega_1\mathcal{U}_k^1 + \omega_2\mathcal{U}_k^2) = \frac{2\epsilon}{m(1+\epsilon)}\sum_{i=1}^{m}\sum_{j=1}^{m}\mathcal{Q}_{ij} + \frac{1-\epsilon}{1+\epsilon}\sum_{k=1}^{m}(\mathcal{Q}_{i^*k} + \mathcal{Q}_{kj^*}) - \frac{m(1-\epsilon)}{1+\epsilon}\mathcal{Q}_{i^*j^*} - mV \tag{9}$$

According to Eq.8 and Eq.9, we have

$$\begin{aligned}
Q_{ij} =& \frac{\epsilon}{m}\sum_{k=1}^{m}(\mathcal{Q}_{ik} + \mathcal{Q}_{kj}) + (1-\epsilon)(\mathcal{Q}_{i^*j} + \mathcal{Q}_{ij^*}) - \frac{2\epsilon^2}{m^2(1+\epsilon)}\sum_{i=1}^{m}\sum_{j=1}^{m}\mathcal{Q}_{ij} \\
&- \frac{1-\epsilon}{1+\epsilon}\mathcal{Q}_{i^*j^*} - \frac{\epsilon(1-\epsilon)}{m(1+\epsilon)}\sum_{k=1}^{m}(\mathcal{Q}_{i^*k} + \mathcal{Q}_{kj^*})
\end{aligned} \tag{10}$$

In order to remove $Q_{i^*j^*}$ from Eq.10, let $i = i^*$ and $j = j^*$.

$$Q_{i^*j^*} = \frac{\epsilon^2}{m}\sum_{k=1}^{m}(\mathcal{Q}_{i^*k} + \mathcal{Q}_{kj^*}) - \frac{\epsilon^2}{m^2}\sum_{i=1}^{m}\sum_{j=1}^{m}\mathcal{Q}_{ij} + (1-\epsilon^2)\mathcal{Q}_{i^*j^*} \tag{11}$$

Substituting Eq.11 into Eq.10, we have

$$\begin{aligned}
Q_{ij} =& \frac{\epsilon}{m}\sum_{k=1}^{m}(\mathcal{Q}_{ik} + \mathcal{Q}_{kj}) + (1-\epsilon)(\mathcal{Q}_{i^*j} + \mathcal{Q}_{ij^*}) - \frac{\epsilon^2}{m^2}\sum_{i=1}^{m}\sum_{j=1}^{m}\mathcal{Q}_{ij} \\
&- \frac{\epsilon(1-\epsilon)}{m}\sum_{k=1}^{m}(\mathcal{Q}_{i^*k} + \mathcal{Q}_{kj^*}) - (1-\epsilon)^2\mathcal{Q}_{i^*j^*}
\end{aligned} \tag{12}$$

D STABLE POINTS UNDER ITS

D.1 PROOF 1

Given the greedy action $\boldsymbol{u}^* = \{u^{1*}, \cdots, u^{n*}\}$ and any action $\boldsymbol{u}_s = \{u_s^1, \cdots, u_s^n\} (\boldsymbol{u}_s \neq \boldsymbol{u}^*)$, assuming $\mathcal{Q}(s, \boldsymbol{u}^*) > 0$ (i.e. $\mathcal{Q}_{its}(s, \boldsymbol{u}) = (1-\alpha)Q(\boldsymbol{u}^*, \boldsymbol{\tau})$ for inferior samples), under the hardest exploration case where $u_s^a \neq u^{a*} (\forall a \in [1, n])$, the utility function of individual action $u_s^a (a \in [1, n])$ is consist of two parts

$$\begin{aligned}\mathcal{U}^a(u_s^a, \tau^a) =& (1-\eta_1)\left[(1-\alpha)Q(\boldsymbol{u}^*, \boldsymbol{\tau}) - \sum_k^{m^{n-1}-1}\left[\frac{p(u_s^a, u_k^{-a})}{p(u_s^a) - p(\boldsymbol{u}_s)}\sum_i^{-a}\mathcal{U}^i(u_k^i, \tau^i)\right]\right] \\ &+ \eta_1\left[\mathcal{Q}_{its}(s, \boldsymbol{u}_s) - \sum_i^{-a}\mathcal{U}^i(u_s^i, \tau^i)\right]\end{aligned} \tag{13}$$

where $\eta_1 = (\frac{\epsilon}{m})^{n-1}$, and $-a$ represents the collection of all agents expect agent a. η_1 and $1-\eta_1$ are the proportion (in all samples containing u_s^a) of sample $\boldsymbol{u}_s$ and inferior samples respectively. $\sum_k^{m^{n-1}}\{u_s^a, u_k^{-a}\}$ is the collection of all samples containing u_s^a, and $p(u_s^a, u_k^{-a})$ is the corresponding probability of each sample. Notice the superscript of the first $\sum$ in Eq.24 is $m^{n-1}-1$, where the sample $\boldsymbol{u}_s$ is excluded. $p(u_s^a) - p(\boldsymbol{u}_s) = \sum_k^{m^{n-1}} p(u_s^a, u_k^{-a}) = \frac{\epsilon}{m} - (\frac{\epsilon}{m})^n$ is the normalization coefficient. We ignore the other potential superior samples. Notice

$$\begin{aligned}(1-\eta_1)\sum_k^{m^{n-1}-1}\left[\frac{p(u_s^a, u_k^{-a})}{p(u_s^a) - p(\boldsymbol{u}_s)}\sum_i^{-a}\mathcal{U}^i(u_k^i, \tau^i)\right] + \eta_1\sum_i^{-a}\mathcal{U}^i(u_k^i, \tau^i) \\ = \sum_k^{m^{n-1}}\left[\frac{p(u_s^a, u_k^{-a})}{p(u_s^a)}\sum_i^{-a}\mathcal{U}^i(u_s^i, \tau^i)\right]\end{aligned} \tag{14}$$

Therefore,

$$\mathcal{U}^a(u_s^a, \tau^a) = (1-\eta_1)(1-\alpha)Q(\boldsymbol{u}^*, \boldsymbol{\tau}) - \sum_k^{m^{n-1}}\left[\frac{p(u_s^a, u_k^{-a})}{p(u_s^a)}\sum_i^{-a}\mathcal{U}^i(u_k^i, \tau^i)\right] + \eta_1\mathcal{Q}_{its}(s, \boldsymbol{u}_s) \tag{15}$$

The joint Q value function $Q(\boldsymbol{u}_s, \boldsymbol{\tau})$ can be acquired

$$\begin{aligned}Q(\boldsymbol{u}_s, \boldsymbol{\tau}) &= \sum_{a=1}^n \mathcal{U}^a(u_s^a, \tau^a) \\ &= n(1-\eta_1)(1-\alpha)Q(\boldsymbol{u}^*, \boldsymbol{\tau}) - \sum_{a=1}^n\sum_k^{m^{n-1}}\left[\frac{p(u_s^a, u_k^{-a})}{p(u_s^a)}\sum_i^{-a}\mathcal{U}^i(u_k^i, \tau^i)\right] + n\eta_1\mathcal{Q}_{its}(s, \boldsymbol{u}_s)\end{aligned} \tag{16}$$

Similarly, for the greedy action $\boldsymbol{u}^*$, we have

$$\begin{aligned}Q(\boldsymbol{u}^*, \boldsymbol{\tau}) &= \sum_{a=1}^n \mathcal{U}^a(u^{a*}, \tau^a) \\ &= n(1-\eta_2)(1-\alpha)Q(\boldsymbol{u}^*, \boldsymbol{\tau}) - \sum_{a=1}^n\sum_k^{m^{n-1}}\left[\frac{p(u^{a*}, u_k^{-a})}{p(u^{a*})}\sum_i^{-a}\mathcal{U}^i(u_k^i, \tau^i)\right] + n\eta_2\mathcal{Q}(s, \boldsymbol{u}^*)\end{aligned} \tag{17}$$

where $\eta_2 = (1 - \epsilon + \frac{\epsilon}{m})^{n-1}$. Notice $\frac{p(u^a, u^{-a})}{p(u^a)}$ is independent to action u^a for decentralized execution, therefore $\frac{p(u^{a*}, u_k^{-a})}{p(u^{a*})} = \frac{p(u_s^a, u_k^{-a})}{p(u_s^a)}$. Let $\mathcal{Q}(s, \boldsymbol{u}_s) = (1 + e_Q)\mathcal{Q}(s, \boldsymbol{u}^*)$,according to Eq.16 and Eq.17, we have

$$\Delta Q(\boldsymbol{u}_s, \boldsymbol{\tau}) = Q(\boldsymbol{u}_s, \boldsymbol{\tau}) - Q(\boldsymbol{u}^*, \boldsymbol{\tau}) = n(\eta_1 - \eta_2)\left[\mathcal{Q}(s, \boldsymbol{u}^*) - (1 - \alpha)Q(\boldsymbol{u}^*, \boldsymbol{\tau})\right] + n\eta_1 e_Q \mathcal{Q}(s, \boldsymbol{u}^*) \tag{18}$$

For monotonic value decomposition, Eq.23 also holds since the expressions of $Q(\boldsymbol{u}^*, \boldsymbol{tau})$ and $Q(\boldsymbol{u}_s, \boldsymbol{tau})$ do not change. Verification of Eq.18 is provided in the experimental part of the main body.

Since $Q(\boldsymbol{u}, \boldsymbol{\tau})$ is an expectation of $\mathcal{Q}(s, \boldsymbol{u})$, for $\forall \boldsymbol{u} \in U^n$, $Q(\boldsymbol{u}, \boldsymbol{\tau}) < max\ \mathcal{Q}(s, \boldsymbol{u})$ holds. If the $\boldsymbol{u}^*$ is exact the optimal action, i.e., $\boldsymbol{u}^* = argmax_{\boldsymbol{u}} \mathcal{Q}(s, \boldsymbol{u})$, for $\forall \boldsymbol{u}_s \in U^n$, $e_Q < 0$ and $Q(\boldsymbol{u}^*, \boldsymbol{\tau}) - \mathcal{Q}(s, \boldsymbol{u}^*) < 0$ hold. Therefore, for $\forall \boldsymbol{u}_s \in U^n$, $\Delta Q(\boldsymbol{u}_s, \boldsymbol{\tau}) < 0$ holds, which indicate when the greedy action is the optimal action, the joint Q value of the optimal action is the maximal, i.e., **there is always an optimal stable point under ITS**.

D.2 PROOF 2

Given the greedy action $\boldsymbol{u}^* = \{u^{1*}, \cdots, u^{n*}\}$ and any action $\boldsymbol{u}_s = \{u_s^1, \cdots, u_s^n\}(\boldsymbol{u}_s \neq \boldsymbol{u}^*)$, assuming $\mathcal{Q}(s, \boldsymbol{u}^*) > 0$ (i.e. $\mathcal{Q}_{its}(s, \boldsymbol{u}) = (1 - \alpha)Q(\boldsymbol{u}^*, \boldsymbol{\tau})$ for inferior samples), under the hardest exploration case where $u_s^a \neq u^{a*}(\forall a \in [1, n])$, the utility function of u_s^a equals to

$$\begin{aligned}
\mathcal{U}_{u_s^a}^a =& \left(\frac{\epsilon}{m}\right)^{n-1}\left[C_{n-1}^0(m^{n-1} - 1)(1 - \alpha)Q(\boldsymbol{u}^*, \boldsymbol{\tau}) + \mathcal{Q}(s, \boldsymbol{u}_s) + f_1(\sum_o^{-a}\sum_{i=1}^m \mathcal{U}_{u_i^o}^o, \sum_{a=1}^n \mathcal{U}_{u^{a*}}^a)\right] \\
&+ \cdots + \left(\frac{\epsilon}{m}\right)^{n-t}(1 - \epsilon)^{t-1}\left[C_{n-1}^{t-1} m^{n-t}(1 - \alpha)Q(\boldsymbol{u}^*, \boldsymbol{\tau}) + f_t(\sum_o^{-a}\sum_{i=1}^m \mathcal{U}_{u_i^o}^o, \sum_{a=1}^n \mathcal{U}_{u^{a*}}^a)\right] + \cdots \\
&+ (1 - \epsilon)^{n-1}\left[C_{n-1}^{n-1}(1 - \alpha)Q(\boldsymbol{u}^*, \boldsymbol{\tau}) + f_n(\sum_o^{-a}\sum_{i=1}^m \mathcal{U}_{u_i^o}^o, \sum_{a=1}^n \mathcal{U}_{u^{a*}}^a)\right] \\
=&(1 - \alpha)Q(\boldsymbol{u}^*, \boldsymbol{\tau}) + \left(\frac{\epsilon}{m}\right)^{n-1}\left[\mathcal{Q}(s, \boldsymbol{u}_s) - (1 - \alpha)Q(\boldsymbol{u}^*, \boldsymbol{\tau})\right] + f_{total}(\sum_o^{-a}\sum_{i=1}^m \mathcal{U}_{u_i^o}^o, \sum_{a=1}^n \mathcal{U}_{u^{a*}}^a)
\end{aligned} \tag{19}$$

where $-a$ represents the collection of all agents except agent a. $f_t(t \in [1, n])$ and f_{total} are mappings from $\{\sum_o^{-a}\sum_{i=1}^m \mathcal{U}_{u_i^o}^o, \sum_{a=1}^n \mathcal{U}_{u^{a*}}^a\}$ to $\mathbb{R}$. We ignore the other potential superior samples. The joint Q value function of $\boldsymbol{u}_s$ equals to

$$\begin{aligned}
Q(\boldsymbol{u}_s, \boldsymbol{\tau}) =& \sum_{a=1}^n \mathcal{U}_{u_s^a}^a \\
=& n(1 - \alpha)Q(\boldsymbol{u}^*, \boldsymbol{\tau}) + n\left(\frac{\epsilon}{m}\right)^{n-1}\left[\mathcal{Q}(s, \boldsymbol{u}_s) - (1 - \alpha)Q(\boldsymbol{u}^*, \boldsymbol{\tau})\right] + nf_{total}(\sum_o^{-a}\sum_{i=1}^m \mathcal{U}_{u_i^o}^o, \sum_{a=1}^n \mathcal{U}_{u^{a*}}^a)
\end{aligned} \tag{20}$$

Next we calculate the joint Q value of the greedy action. The utility function of the individual greedy action $u^{a*}(a \in [1, n])$ equals to

$$\begin{aligned}
\mathcal{U}_{u^{a*}}^a =&(1 - \alpha)Q(\boldsymbol{u}^*, \boldsymbol{\tau}) \\
&+ (1 - \epsilon + \frac{\epsilon}{m})^{n-1}\left[\mathcal{Q}(s, \boldsymbol{u}^*) - (1 - \alpha)Q(\boldsymbol{u}^*, \boldsymbol{\tau})\right] + f_{total}(\sum_o^{-a}\sum_{i=1}^m \mathcal{U}_{u_i^o}^o, \sum_{a=1}^n \mathcal{U}_{u^{a*}}^a)
\end{aligned} \tag{21}$$

The joint Q value of greedy action equals to

$$\begin{aligned} Q(\boldsymbol{u}^*, \boldsymbol{\tau}) = & \sum_{a=1}^{n} \mathcal{U}^a_{u^{a*}} = n(1-\alpha)Q(\boldsymbol{u}^*, \boldsymbol{\tau}) \\ & + n(1-\epsilon+\frac{\epsilon}{m})^{n-1}\left[\mathcal{Q}(s, \boldsymbol{u}^*) - (1-\alpha)Q(\boldsymbol{u}^*, \boldsymbol{\tau})\right] + nf_{total}(\sum_{o}^{-a}\sum_{i=1}^{m}\mathcal{U}^o_{u^o_i}, \sum_{a=1}^{n}\mathcal{U}^a_{u^{a*}}) \end{aligned} \tag{22}$$

Notice $\eta_1 = (\frac{\epsilon}{m})^{n-1}$, $\eta_2 = (1-\epsilon+\frac{\epsilon}{m})^{n-1}$ and $\mathcal{Q}(s, \boldsymbol{u}_s) = (1+e_Q)\mathcal{Q}(s, \boldsymbol{u}^*)$ according to Eq.20 and Eq.22

$$\begin{aligned} \Delta Q(\boldsymbol{u}_s, \boldsymbol{\tau}) &= Q(\boldsymbol{u}_s, \boldsymbol{\tau}) - Q(\boldsymbol{u}^*, \boldsymbol{\tau}) \\ &= n\eta_1\left[(1+e_Q)\mathcal{Q}(s, \boldsymbol{u}^*) - (1-\alpha)Q(\boldsymbol{u}^*, \boldsymbol{\tau})\right] - n\eta_2\left[\mathcal{Q}(s, \boldsymbol{u}^*) - (1-\alpha)Q(\boldsymbol{u}^*, \boldsymbol{\tau})\right] \\ &= n(\eta_1-\eta_2)\left[\mathcal{Q}(s, \boldsymbol{u}^*) - (1-\alpha)Q(\boldsymbol{u}^*, \boldsymbol{\tau})\right] + n\eta_1 e_Q \mathcal{Q}(s, \boldsymbol{u}^*) \end{aligned} \tag{23}$$

which consist with Eq.18 in Proof 1.

E LVD AND MVD UNDER ITS WITH SUPERIOR SAMPLE WEIGHT

E.1 DERIVATION OF $\Delta Q(s, u_s)$

Given the greedy action $\boldsymbol{u}^*$ and a superior action $\boldsymbol{u}_s$ (i.e. $\mathcal{Q}(s, \boldsymbol{u}_s) > \mathcal{Q}(s, \boldsymbol{u}^*)$) assuming $\mathcal{Q}(s, \boldsymbol{u}^*) > 0$ (i.e. $\mathcal{Q}_{its}(s, \boldsymbol{u}) = (1-\alpha)Q(\boldsymbol{u}^*, \boldsymbol{\tau})$ for inferior samples), under the hardest exploration case where $u^{a*} \neq u^a_s (\forall a \in [1, n])$, the utility function of individual action $u^a_s (a \in [1, n])$ is consist of two parts

$$\begin{aligned} \mathcal{U}^a(u^a_s, \tau^a) = & (1-\eta_{1,w})\left[(1-\alpha)Q(\boldsymbol{u}^*, \boldsymbol{\tau}) - \sum_{k}^{m^{n-1}-1}\left[\frac{p(u^a_s, u^{-a}_k)}{p(u^a_s) - p(\boldsymbol{u}_s)}\sum_{i}^{-a}\mathcal{U}^i(u^i_k, \tau^i)\right]\right] \\ & + \eta_{1,w}\left[\mathcal{Q}_{its}(s, \boldsymbol{u}_s) - \sum_{i}^{-a}\mathcal{U}^i(u^i_s, \tau^i)\right] \end{aligned} \tag{24}$$

where $\eta_{1,w} = \frac{w\eta_1}{1+(w-1)\eta_1}$, $\eta_1 = (\frac{\epsilon}{m})^{n-1}$ and w is a sample weight on the superior samples. Please refer to Eq.24 for more details about the notations. According to Eq.14, we have

$$\begin{aligned} & \sum_{k}^{m^{n-1}-1}\left[\frac{p(u^a_s, u^{-a}_k)}{p(u^a_s) - p(\boldsymbol{u}_s)}\sum_{i}^{-a}\mathcal{U}^i(u^i_k, \tau^i)\right] \\ & = \frac{1}{(1-\eta_1)}\sum_{k}^{m^{n-1}}\left[\frac{p(u^a_s, u^{-a}_k)}{p(u^a_s)}\sum_{i}^{-a}\mathcal{U}^i(u^i_k, \tau^i)\right] - \frac{\eta_1}{(1-\eta_1)}\sum_{i}^{-a}\mathcal{U}^i(u^i_s, \tau^i) \end{aligned} \tag{25}$$

Substituting the left side of Eq.25 into Eq.24

$$\begin{aligned} \mathcal{U}^a(u^a_s, \tau^a) = & (1-\eta_{1,w})(1-\alpha)Q(\boldsymbol{u}^*, \boldsymbol{\tau}) - \frac{1-\eta_{1,w}}{1-\eta_1}\sum_{k}^{m^{n-1}}\left[\frac{p(u^a_s, u^{-a}_k)}{p(u^a_s)}\sum_{i}^{-a}\mathcal{U}^i(u^i_k, \tau^i)\right] \\ & + \frac{\eta_1 - \eta_{1,w}}{1-\eta_1}\sum_{i}^{-a}\mathcal{U}^i(u^i_s, \tau^i) + \eta_{1,w}\mathcal{Q}_{its}(s, \boldsymbol{u}_s) \end{aligned} \tag{26}$$

Notice that

$$\sum_{a=1}^{n}\sum_{i}^{-a}\mathcal{U}^i(u_s^i,\tau^i)=(n-1)\sum_{a=1}^{n}\mathcal{U}^i(u_s^i,\tau^i)=(n-1)Q(\boldsymbol{u}_s,\boldsymbol{\tau}) \tag{27}$$

The joint Q value function $Q(\boldsymbol{u}_s,\boldsymbol{\tau})$ can be acquired

$$\begin{aligned}Q(\boldsymbol{u}_s,\boldsymbol{\tau})=\sum_{a=1}^{n}\mathcal{U}^a(u_s^a,\tau^a)=&n(1-\eta_{1,w})(1-\alpha)Q(\boldsymbol{u}^*,\boldsymbol{\tau})+(n-1)\frac{\eta_1-\eta_{1,w}}{1-\eta_1}Q(\boldsymbol{u}_s,\boldsymbol{\tau})\\&+n\eta_{1,w}\mathcal{Q}_{its}(s,\boldsymbol{u}_s)-\frac{1-\eta_{1,w}}{1-\eta_1}\sum_{a=1}^{n}\sum_{k}^{m^{n-1}}\left[\frac{p(u_s^a,u_k^{-a})}{p(u_s^a)}\sum_{i}^{-a}\mathcal{U}^i(u_k^i,\tau^i)\right]\end{aligned} \tag{28}$$

According to Eq.17, we have

$$\sum_{a=1}^{n}\sum_{k}^{m^{n-1}}\left[\frac{p(u^{a*},u_k^{-a})}{p(u^{a*})}\sum_{i}^{-a}\mathcal{U}^i(u_k^i,\tau^i)\right]=n(1-\eta_2)(1-\alpha)Q(\boldsymbol{u}^*,\boldsymbol{\tau})+n\eta_2\mathcal{Q}(s,\boldsymbol{u}^*)-Q(\boldsymbol{u}^*,\boldsymbol{\tau}) \tag{29}$$

Substituting the left side of Eq.29 into Eq.28

$$\begin{aligned}\left[1-(n-1)\frac{\eta_1-\eta_{1,w}}{1-\eta_1}\right]Q(\boldsymbol{u}_s,\boldsymbol{\tau})=&n(1-\eta_{1,w})(1-\alpha)Q(\boldsymbol{u}^*,\boldsymbol{\tau})+n\eta_{1,w}\mathcal{Q}_{its}(s,\boldsymbol{u}_s)\\&-\frac{1-\eta_{1,w}}{1-\eta_1}\left[n(1-\eta_2)(1-\alpha)-1\right]Q(\boldsymbol{u}^*,\boldsymbol{\tau})-n\eta_2\frac{1-\eta_{1,w}}{1-\eta_1}\mathcal{Q}(s,\boldsymbol{u}^*)\end{aligned} \tag{30}$$

where $\eta_2=(1-\epsilon+\frac{\epsilon}{m})^{n-1}$. Eq.31 can be further simplified as

$$Q(\boldsymbol{u}_s,\boldsymbol{\tau})=\frac{n(1-\alpha)(\eta_2-\eta_1)+1}{1+n(w-1)\eta_1}Q(\boldsymbol{u}^*,\boldsymbol{\tau})+n\frac{w(1+e_Q)\eta_1-\eta_2}{1+n(w-1)\eta_1}\mathcal{Q}(s,\boldsymbol{u}^*) \tag{31}$$

Therefore,

$$\begin{aligned}\Delta Q(\boldsymbol{u}_s,\boldsymbol{\tau})=Q(\boldsymbol{u}_s,\boldsymbol{\tau})-Q(\boldsymbol{u}^*,\boldsymbol{\tau})=&n\frac{(1-\alpha)(\eta_2-\eta_1)-(w-1)\eta_1}{1+n(w-1)\eta_1}Q(\boldsymbol{u}^*,\boldsymbol{\tau})\\&+n\frac{w(1+e_Q)\eta_1-\eta_2}{1+n(w-1)\eta_1}\mathcal{Q}(s,\boldsymbol{u}^*)\end{aligned} \tag{32}$$

When $w=1$, Eq.32 degenerate to Eq.18 (i.e., the case without sample weight). For monotonic value decomposition, Eq.32 also holds since the expressions of $Q(\boldsymbol{u}^*,\boldsymbol{\tau})$ and $Q(\boldsymbol{u}_s,\boldsymbol{\tau})$ do not change.

When $\Delta Q(\boldsymbol{u}_s,\boldsymbol{\tau})>0$, $argmax_{\boldsymbol{u}}Q(\boldsymbol{u},\boldsymbol{\tau})\neq\boldsymbol{u}^*$, which suggests current greedy action is unstable. If $\boldsymbol{u}^*$ is a non-optimal action, to destabilize it, let $\Delta Q(\boldsymbol{u}_s,\boldsymbol{\tau})>0$, assuming $Q(\boldsymbol{u}^*,\boldsymbol{\tau})\approx\mathcal{Q}(s,\boldsymbol{u}^*)$ we have

$$w>\frac{\alpha(\eta_2-\eta_1)}{e_Q\eta_1} \tag{33}$$

When $\boldsymbol{u}_s=argmax_{\boldsymbol{u}}\mathcal{Q}(s,\boldsymbol{u})$, we obtain the lower bound of $\boldsymbol{w}$, which suggest the non-optimal stable points can be eliminated by a large enough weight on the superior sample under ITS.

E.2 VERIFICATION OF THE EFFECT OF SUPERIOR SAMPLE WEIGHTS UNDER ITS.

We carry out experiments in matrix games to evaluate the effect of the weight on the superior sample under ITS, where the payoff matrix is defined as

$$\mathcal{Q}_{its} = \begin{cases} 6(1+e_Q) & \boldsymbol{u} = \{0,0\} \\ 6 & \boldsymbol{u} = \{2,2\} \\ random(-20,6) & others \end{cases} \tag{34}$$

An introduction of the matrix game can be found in the experimental part. An mlp shared by all agents is adopted as the agent network, which is trained for 1000 iterations (100 episodes per iteration) over 5 seeds, where $\alpha = 0.1$, $\epsilon = 0.2$ and $e_Q = 1/3$.

m^n	3^2	5^2	10^2	3^3	3^4
Calculated w_0 (Eq.33)	3.60	6.00	12.00	50.32	**659.50**
Tested $\Delta Q(\boldsymbol{u}_s, \boldsymbol{\tau})$ (Eq.32)	0.01 $\pm$0.06	0.02 $\pm$0.16	0.22 $\pm$0.13	-0.02 $\pm$0.30	-0.48 $\pm$0.75
Tested $Q(\boldsymbol{u}^*, \boldsymbol{\tau})$	5.95 $\pm$0.02	5.97$\pm$0.02	5.98 $\pm$0.01	5.90 $\pm$0.06	5.93 $\pm$0.03

Table 3: Evaluation of the sample weight on superior samples for LVD under ITS. w_0 denotes the lower bound of required sample weight to eliminate the non-optimal stable points when $\Delta Q(\boldsymbol{u}_s, \boldsymbol{\tau}) = 0$. m is the individual action space size and n is the number of agents.

From Table3, the joint Q value of greedy action approximately equals to its true Q value (i.e., $Q(\boldsymbol{u}^*, \boldsymbol{\tau}) \approx \mathcal{Q}(s, \boldsymbol{u}^*) = 6$) under ITS. Besides, the required sample weight to eliminate the non-optimal stable points grows **exponentially** as the number of agent grows, which introduces instability in the joint Q values.

F LVD AND MVD UNDER ITS WITH SUPERIOR EXPERIENCE REPLAY

Given the greedy action $\boldsymbol{u}^*$ and a superior action $\boldsymbol{u}_s$ (i.e. $\mathcal{Q}(s, \boldsymbol{u}_s) > \mathcal{Q}(s, \boldsymbol{u}^*)$) assuming $\mathcal{Q}(s, \boldsymbol{u}^*) > 0$ (i.e. $\mathcal{Q}_{its}(s, \boldsymbol{u}) = (1-\alpha)Q(\boldsymbol{u}^*, \boldsymbol{\tau})$ for inferior samples), under the hardest exploration case where $u_s^a \neq u^{a*} (\forall a \in [1, n])$, the utility function of individual action $u_s^a (a \in [1, n])$ is consist of two parts

$$\begin{aligned} \mathcal{U}^a(u_s^a, \tau^a) =& (1-\eta_{1,ser}) \left[(1-\alpha) Q(\boldsymbol{u}^*, \boldsymbol{\tau}) - \sum_{k}^{m^{n-1}-1} \left[\frac{p(u_s^a, u_k^{-a})}{p(u_s^a) - p(\boldsymbol{u}_s)} \sum_{i}^{-a} \mathcal{U}^i(u_k^i, \tau^i) \right] \right] \\ &+ \eta_{1,ser} \left[\mathcal{Q}_{its}(s, \boldsymbol{u}_s) - \sum_{i}^{-a} \mathcal{U}^i(u_s^i, \tau^i) \right] \end{aligned} \tag{35}$$

where $\eta_{1,ser} = \frac{w+\eta_1}{1+w}$, $\eta_1 = (\frac{\epsilon}{m})^{n-1}$ and w is a sample weight on the superior samples from the superior buffer. Please refer to Eq.24 for more details about the notations. Following the derivation provided in Appendix E.1, we have

$$\begin{aligned} \left[1 - (n-1) \frac{\eta_1 - \eta_{1,ser}}{1-\eta_1} \right] Q(\boldsymbol{u}_s, \boldsymbol{\tau}) =& n(1-\eta_{1,ser})(1-\alpha) Q(\boldsymbol{u}^*, \boldsymbol{\tau}) + n\eta_{1,ser} \mathcal{Q}_{its}(s, \boldsymbol{u}_s) \\ &- \frac{1-\eta_{1,ser}}{1-\eta_1} \left[n(1-\eta_2)(1-\alpha) - 1 \right] Q(\boldsymbol{u}^*, \boldsymbol{\tau}) - n\eta_2 \frac{1-\eta_{1,ser}}{1-\eta_1} \mathcal{Q}(s, \boldsymbol{u}^*) \end{aligned} \tag{36}$$

Eq.36 can be further simplified as

$$Q(\boldsymbol{u}_s, \boldsymbol{\tau}) = \frac{n(1-\alpha)(\eta_2 - \eta_1) + 1}{1+nw} Q(\boldsymbol{u}^*, \boldsymbol{\tau}) + n \frac{(w+\eta_1)(1+e_Q) - \eta_2}{1+nw} \mathcal{Q}(s, \boldsymbol{u}^*) \tag{37}$$

where $\eta_2 = (1 - \epsilon + \frac{\epsilon}{m})^{n-1}$. Therefore,

$$\begin{aligned}\Delta Q(\boldsymbol{u}_s, \boldsymbol{\tau}) = Q(\boldsymbol{u}_s, \boldsymbol{\tau}) - Q(\boldsymbol{u}^*, \boldsymbol{\tau}) =& n\frac{(1-\alpha)(\eta_2 - \eta_1) - w}{1 + nw} Q(\boldsymbol{u}^*, \boldsymbol{\tau}) \\ &+ n\frac{(w + \eta_1)(1 + e_Q) - \eta_2}{1 + nw} \mathcal{Q}(s, \boldsymbol{u}^*)\end{aligned} \tag{38}$$

When $w = 0$, Eq.38 degenerate to Eq.18 (i.e., the case without samples form superior buffer). For monotonic value decomposition, Eq.38 also holds since the expressions of $Q(\boldsymbol{u}^*, \boldsymbol{tau})$ and $Q(\boldsymbol{u}_s, \boldsymbol{tau})$ do not change.

If $\boldsymbol{u}^*$ is a non-optimal action, to destabilize it, let $\Delta Q(\boldsymbol{u}_s, \boldsymbol{\tau}) > 0$. A sufficient condition for $\Delta Q(\boldsymbol{u}_s, \boldsymbol{\tau}) > 0$ is both terms in the right side of Eq.38 are no less than 0. As a result,

$$\frac{\eta_2}{1 + e_Q} - \eta_1 \leq w \leq (1 - \alpha)(\eta_2 - \eta_1) \tag{39}$$

To ensure $\frac{\eta_2}{1+e_Q} - \eta_1 < \frac{\eta_2}{1+e_Q} - \eta_1$, let $\alpha = \frac{e_Q}{1-e_Q}$. According to Eq.39, **SER can eliminate the non-optimal stable points by a selecting a suitable value of w for superior samples**.

G Working Principle of GVR and the Algorithm

The working principle of GVR is shown in Fig.1, and the algorithm is given in Algo.1. for details about notations please refer to Appendix F.

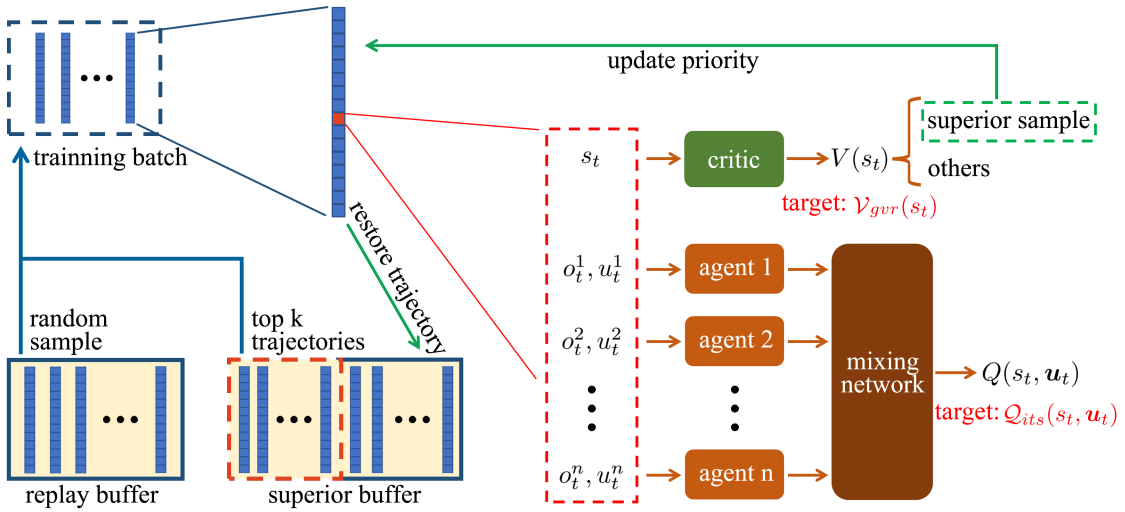

Figure 1: The working principle of GVR. In the training stage of each iteration, we acquire the training batch by concating the trajectories sampled from the replay buffer and the superior buffer. Then we calculate the loss of $Q(\boldsymbol{u}, \boldsymbol{\tau})$ and $V(s)$ referring to their targets. After that, we update the priority of the trajectories in training batch and restore the trajectories to the superior buffer according to the updated priority.

Algorithm 1 Greedy-based Value Representation

Initialize parameters θ_a for agents and θ_c for critic
Initialize replay buffer D_r and superior buffer D_s
Initialize hyperparameter $\alpha \in (0, 1]$, weight $w = (1 - \alpha)(\eta_2 - \eta_1)$
for iteration $i = 1, 2, 3, \cdots$ **do**
  Interact with environment and store transitions to D_r
  Sample batch b_r from D_r
  Take out the top-k trajectories b_s from D_s
  Concat batches $b_{total} = b_r + b_s$
  for trajectory $\tau = 1, 2, 3, \cdots$ in b_{total} **do**
    Reset priority $p_\tau = 0$
    for step $t = 1, 2, 3, \cdots$ **do**
      if trajectory is from b_s **then**
        Calculate agent loss for superior samples $loss_a = w|\mathcal{Q}_{its}(s_t, \boldsymbol{u}_t) - Q_{\theta_a}(\boldsymbol{u}_t, \boldsymbol{\tau}_t)|_2 * \mathcal{I}(\mathcal{Q}(s_t, \boldsymbol{u}_t) > V_{\theta_c}(s))$
      else
        Calculate agent loss $loss_a = |\mathcal{Q}_{its}(s_t, \boldsymbol{u}_t) - Q_{\theta_a}(\boldsymbol{u}_t, \boldsymbol{\tau}_t)|_2$
      end if
      if $V_{\theta_c}(s) < \mathcal{Q}(s, \boldsymbol{u})$ or $\boldsymbol{u} = \boldsymbol{u}^*$ **then**
        Calculate critic loss $loss_c = |\mathcal{Q}(s_t, \boldsymbol{u}_t) - V_{\theta_c}(s)|_2$
      end if
      if $V_{\theta_c}(s) < \mathcal{Q}(s, \boldsymbol{u})$ **then**
        Update priority $p_\tau = p_\tau + 1$
      end if
    end for
    Restore τ to D_b according to p_τ
  end for
  Update θ_a and θ_c
end for

H EFFECT OF REWARD FUNCTION AND ϵ ON STABLE POINTS.

We conduct experiments in two-agent matrix games to verify the effect of reward function and ϵ on stable points. An mlp shared by two agents is adopted as the agent network. The ratios of different stable points are counted with ϵ increasing from 0 to 0.99. At each value of ϵ, 100 times of independent training and test are executed. Each training includes 2000 episodes. The experiments are carried out over 5 seeds. According to Fig.2, the stable points changes with both true Q value and ϵ. As ϵ grows, there becomes only one stable point. We ignore situations with non-unique optimal stable points (e.g., [[1,0],[0,1]]), where the stable points can also be calculated referring to Eq.7.

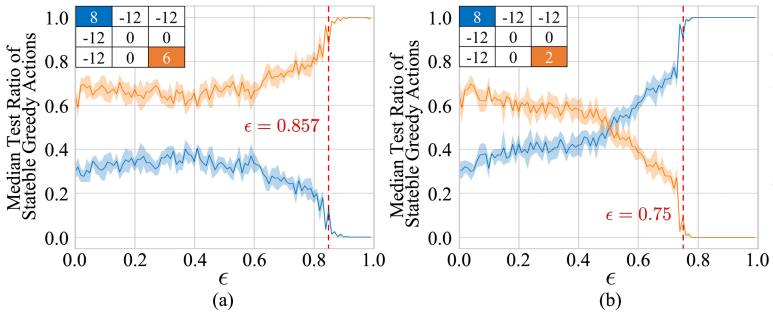

Figure 2: Median test ratios of stable points (i.e., test probability of different convergence) under 100 times of independent training vs ϵ. The payoff matrices are shown in the upper-left of each subgraph. The greedy actions in different stable points are marked with different colors (blue for $\{u^{1*}, u^{2*}\} = \{0, 0\}$ and orange for $\{u^{1*}, u^{2*}\} = \{2, 2\}$). Take subgraph (a) as an example, when $\epsilon = 0.4$, the ratios of two stable greedy actions ($\{0, 0\}$ and $\{2, 2\}$) approximate 0.35 and 0.65 respectively. In the presented examples, when $\epsilon > 0.857$ and $\epsilon > 0.75$, there becomes **almost only one stable point, which consists with the calculated threshold** (denoted by red dash lines) from Eq.7.

I EXPERIMENTAL SETTINGS AND ADDITIONAL EXPERIMENTS

I.1 ABLATION STUDIES

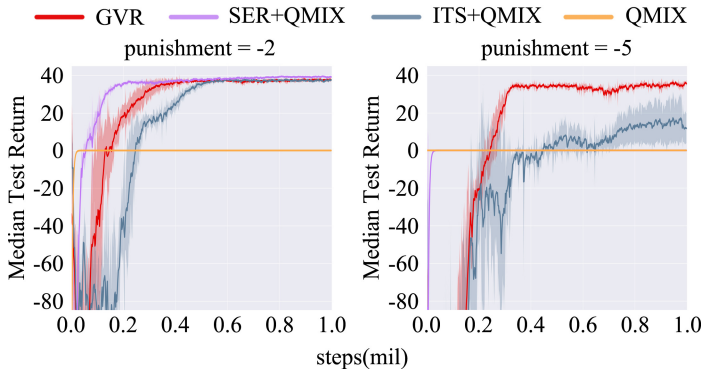

Figure 3: Ablation studies on the effect of ITS and SER.

We conduct ablation studies to investigate the effect of GVR. We first evaluate the effect of inferior target shaping (ITS) and superior experience replay (SER) on QMIX. The experimental settings are the same as the comparative experiments of predator-prey before. The experiments are carried out over 5 seeds.

It can be seen from the Fig.3 that in task with punishment -2, both ITS and SER helps to solve the problem. In task with punishment -5, due to the extreme negative return of inferior samples, SER alone is unable to solve the problem. Meanwhile, in spite of the shaped reward by ITS, the proportion of superior samples is very small, leading to instability during training.

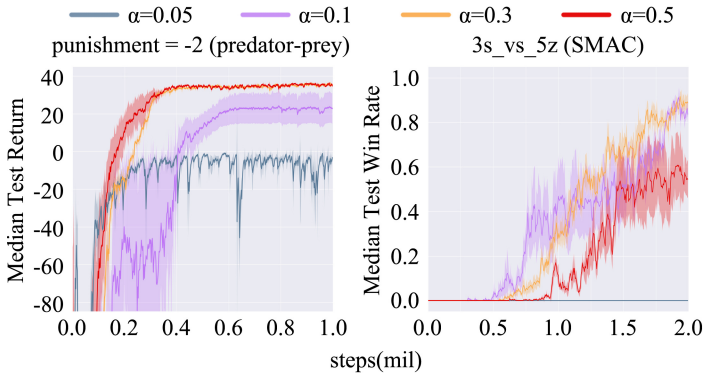

Figure 4: Ablation studies on the parameter α.

We also investigate the effect of the parameter α on GVR. The experimental settings for SMAC and predator-prey are the same as the corresponding comparative experiments before. The experiments are carried out over 5 seeds.

It can be seen from the Fig.4 that a too small or too large value of α leads to poor performance. According to the definition of ITS target, the α determines the gap the joint Q values between greedy samples and inferior samples. As a result, a too-small value of α brings the risk of confusion between these two kinds of samples. Meanwhile, a too-big value of α may prevent the update from a greedy action to a superior action.

I.2 Comparison With Joint Exploration Methods

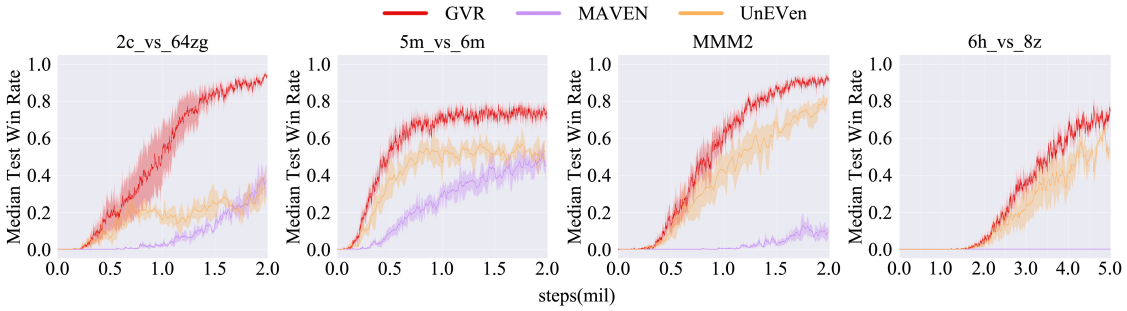

Figure 5: Comparison between GVR, MAVEN and UneVEn.

We compare our method with MAVEN and UneVEn on SMAC. The experiment results are given in Fig.5, where GVR shows the best performance. Besides, to further investigate the scalability of our method, we

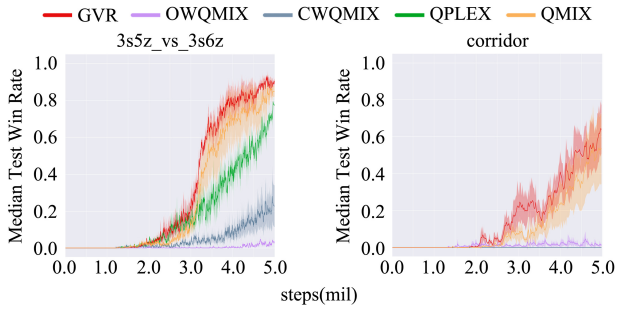

Figure 6: Comparison between GVR and baselines on two super hard SMAC tasks.

investigate the performance of GVR on two other super hard tasks of SMAC, and the experiment results are shown in Fig.6.

REFERENCES

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.

Anuj Mahajan, Tabish Rashid, Mikayel Samvelyan, and Shimon Whiteson. Maven: Multi-agent variational exploration. *arXiv preprint arXiv:1910.07483*, 2019.

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.

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.

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

Chao Wen, Xinghu Yao, Yuhui Wang, and Xiaoyang Tan. Smix (λ): Enhancing centralized value functions for cooperative multi-agent reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 7301–7308, 2020.

Yaodong Yang, Jianye Hao, Guangyong Chen, Hongyao Tang, Yingfeng Chen, Yujing Hu, Changjie Fan, and Zhongyu Wei. Q-value path decomposition for deep multiagent reinforcement learning. In *International Conference on Machine Learning*, pp. 10706–10715. PMLR, 2020a.

Yaodong Yang, Jianye Hao, Ben Liao, Kun Shao, Guangyong Chen, Wulong Liu, and Hongyao Tang. Qatten: A general framework for cooperative multiagent reinforcement learning. *arXiv preprint arXiv:2002.03939*, 2020b.