QPLEX: Duplex Dueling Multi-Agent Q-Learning

Anonymous Author(s)
Affiliation
Address
email

Abstract

We explore value-based multi-agent reinforcement learning (MARL) in the popular paradigm of centralized training with decentralized execution (CTDE). A key assumption of CTDE is the IGM (*Individual-Global-Max*) principle, which ensures the consistency of the optimal joint action selection with optimal individual action selections. However, in order to achieve scalability, existing MARL methods either limit representation expressiveness of their value function classes or relax the IGM consistency, which may lead to poor policies or even divergence. In this paper, we present a novel MARL approach, called *duPLEX dueling multi-agent Q-learning* (QPLEX), that takes a duplex dueling network architecture to factorize the joint action-value function. This duplex dueling architecture transforms the IGM principle to easily realized constraints on advantage functions and thus enables simple value function learning with a linear decomposition structure. As a result, QPLEX achieves the first scalable multi-agent Q-learning algorithm with a full expressiveness power induced by IGM. Empirical experiments on StarCraft II unit micromanagement benchmark tasks demonstrate that QPLEX significantly outperforms state-of-the-art baselines in both online and offline data collection.

1 Introduction

Cooperative multi-agent reinforcement learning (MARL) has great promise for addressing many complex real-world problems, such as sensor networks [1], coordination of robot swarms [2] and autonomous cars [3]. However, cooperative MARL encounters two major challenges of scalability and partial observability in practical applications. The joint action space will grow exponentially as the number of agents increases. Due to partial observations, each agent needs to make its decisions based on the local action-observation history. To address these challenges, a popular MARL paradigm, called *centralised training with decentralised execution* (CTDE) [4, 5], has recently attracted attention, where agents' policies are trained with access to global information in a centralized way and executed only based on local observations in a decentralized way.

Many CTDE learning approaches have recently been proposed, among which value-based MARL algorithms [6, 7, 8, 9] have shown state-of-the-art performance on the challenging unit micromanagement benchmark tasks of StarCraft II[1] [10]. A key assumption of CTDE in multi-agent Q-learning is that the optimal joint action should be equivalent to the collection of individual optimal actions of agents, which is called the IGM (*Individual-Global-Max*) principle [8]. Due to the exponential growth of joint action space, the max operator for the greedy joint action selection in Q-learning becomes intractable. To address this scalability issue, VDN [6] and QMIX [7] propose two sufficient conditions of IGM to factorize the joint action-value function. However, these two decomposition methods suffer from structural constraints and severely limit the joint action-value function class they can represent, which may result in poor performance and even unbounded divergence in the off-policy

[1] StarCraft and StarCraft II are trademarks of Blizzard Entertainment™.

Submitted to 34th Conference on Neural Information Processing Systems (NeurIPS 2020). Do not distribute.

training [11]. QTRAN [8] uses a linear decomposition as VDN with two regularization losses as soft constraints for the consistency of greedy action selection. This relaxation with soft constraints may deviate from the exact IGM consistency principle and result in poor empirical performance on complex domains [9]. Thus, how to efficiently achieve the full expressiveness power induced by IGM with high scalability remains an open question.

To address this question, this paper proposes a new MARL approach, called *duPLEX dueling multi-agent Q-learning* (QPLEX), that takes a duplex dueling network architecture to factorize the joint action-value function. QPLEX reformalizes the IGM principle as an *Advantage-based IGM* through the dueling structure $Q = V + A$ proposed by Dueling DQN [12]. This dueling architecture transforms the IGM consistency to easily realized constraints on the value range of advantage functions and thus enables simple value learning with a linear decomposition structure. To our best knowledge, QPLEX achieves the first off-policy multi-agent Q-learning algorithm with efficiency and a full expressiveness power induced by IGM.

We evaluate the performance of QPLEX in both didactic problems proposed by [8, 11] and a range of unit micromanagement benchmark tasks in StarCraft II [10]. In these didactic problems, QPLEX demonstrates its full representation expressiveness and learns the optimal joint action-value function. Empirical results on more challenging StarCraft II tasks show that QPLEX significantly outperforms other multi-agent Q-learning baselines in both online and offline data collection. In particular, the off-policy training experiment demonstrates that QPLEX has the same off-policy nature as that of DQN [13], which is not possessed by other baselines.

2 Preliminaries

2.1 Decentralized Partially Observable MDP (Dec-POMDP)

We model a fully cooperative multi-agent task as a Dec-POMDP [14] defined by a tuple $\mathcal{M} = \langle \mathcal{N}, \mathcal{S}, \mathcal{A}, P, \Omega, O, r, \gamma \rangle$, where $\mathcal{N} \equiv \{1, 2, \ldots, n\}$ is a finite set of agents and $s \in \mathcal{S}$ is a finite set of global states. At each time step, every agent $i \in \mathcal{N}$ chooses an action $a_i \in \mathcal{A} \equiv \{\mathcal{A}^{(1)}, \ldots, \mathcal{A}^{(|\mathcal{A}|)}\}$ on a global state s, which forms a joint action $\boldsymbol{a} \equiv [a_i]_{i=1}^n \in \boldsymbol{\mathcal{A}} \equiv \mathcal{A}^n$. This results in a joint reward $r(s, \boldsymbol{a})$ and a transition to the next global state $s' \sim P(\cdot|s, \boldsymbol{a})$. $\gamma \in [0, 1)$ is the discount factor. We consider a *partially observable* setting, where each agent i receives an individual partial observation $o_i \in \Omega$ according to the observation probability function $O(o_i|s, a_i)$. Each agent i has an action-observation history $\tau_i \in \mathcal{T} \equiv (\Omega \times \mathcal{A})^*$ and constructs its individual policy $\pi_i(a|\tau_i)$ to jointly maximize team performance. We use $\boldsymbol{\tau} \in \boldsymbol{\mathcal{T}} \equiv \mathcal{T}^n$ to denote joint action-observation histories. The formal objective function is to find a joint policy $\boldsymbol{\pi} = \langle \pi_1, \ldots, \pi_n \rangle$ that maximizes a joint value function $V^{\boldsymbol{\pi}}(s) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r_t | s_0 = s, \boldsymbol{\pi}\right]$. Another quantity of interest in policy search is the joint action-value function $Q^{\boldsymbol{\pi}}(s, \boldsymbol{a}) = r(s, \boldsymbol{a}) + \gamma \mathbb{E}_{s'}[V^{\boldsymbol{\pi}}(s')]$.

2.2 Deep Multi-Agent Q-Learning in Dec-POMDP

Q-learning algorithms is a popular algorithm to find the optimal joint action-value function $Q^*(s, \boldsymbol{a}) = r(s, \boldsymbol{a}) + \gamma \mathbb{E}_{s'}\left[\max_{\boldsymbol{a'}} Q^*\left(s', \boldsymbol{a'}\right)\right]$. Deep Q-learning represents the action-value function with a deep neural network parameterized by $\boldsymbol{\theta}$. Mutli-agent Q-learning algorithms [6, 7, 8, 9] use a replay memory D to store the transition tuple $(\boldsymbol{\tau}, \boldsymbol{a}, r, \boldsymbol{\tau'})$, where r is the reward for taking action $\boldsymbol{a}$ at joint action-observation history $\boldsymbol{\tau}$ with a transition to $\boldsymbol{\tau'}$. Due to partial observability, $Q\left(\boldsymbol{\tau}, \boldsymbol{a}; \boldsymbol{\theta}\right)$ is used in place of $Q\left(s, \boldsymbol{a}; \boldsymbol{\theta}\right)$. Thus, parameters $\boldsymbol{\theta}$ are learnt by minimizing the following expected TD error:

$$\mathcal{L}(\boldsymbol{\theta}) = \mathbb{E}_{(\boldsymbol{\tau}, \boldsymbol{a}, r, \boldsymbol{\tau'}) \in D}\left[\left(r + \gamma V\left(\boldsymbol{\tau'}; \boldsymbol{\theta^-}\right) - Q\left(\boldsymbol{\tau}, \boldsymbol{a}; \boldsymbol{\theta}\right)\right)^2\right], \tag{1}$$

where $V\left(\boldsymbol{\tau'}; \boldsymbol{\theta^-}\right) = \max_{\boldsymbol{a'}} Q\left(\boldsymbol{\tau'}, \boldsymbol{a'}; \boldsymbol{\theta^-}\right)$ is the one-step expected future return of the TD target and $\boldsymbol{\theta^-}$ are the parameters of the target network, which will be periodically updated with $\boldsymbol{\theta}$.

2.3 Centralized Training with Decentralized Execution (CTDE)

CTDE is a popular paradigm of cooperative multi-agent deep reinforcement learning [6, 7, 8, 9]. Agents are trained in a centralized way and granted access to other agents' information or the global states during the centralized training process. However, due to partial observability and

communication constraints, each agent makes its own decision based on local action-observation history during the decentralized execution phase. A key assumption of CTDE in multi-agent Q-learning is that the optimal joint action induced from the optimal centralized action-value function is equivalent to the collection of individual optimal actions of agents, which is called IGM (*Individual-Global-Max*) principle [8]. This principle asserts that a joint action-value function $Q_{tot}(\boldsymbol{\tau}, \boldsymbol{a})$ is factorizable if and only if there exists $[Q_i : \mathcal{T} \times \mathcal{A} \mapsto \mathbb{R}]_{i=1}^{n}$ such that $\forall \boldsymbol{\tau} \in \boldsymbol{\mathcal{T}}$:

$$\arg\max_{\boldsymbol{a} \in \boldsymbol{\mathcal{A}}} Q_{tot}(\boldsymbol{\tau}, \boldsymbol{a}) = \begin{pmatrix} \arg\max_{a_1 \in \mathcal{A}} Q_1(\tau_1, a_1), \ldots, \arg\max_{a_n \in \mathcal{A}} Q_n(\tau_n, a_n) \end{pmatrix}. \tag{2}$$

Two factorization structures, **additivity** and **monotonicity**, has been proposed by VDN [6] and QMIX [7], respectively, as shown below:

$$Q_{tot}^{\text{VDN}}(\boldsymbol{\tau}, \boldsymbol{a}) = \sum_{i=1}^{n} Q_i(\tau_i, a_i) \quad \text{and} \quad \forall i \in \mathcal{N}, \frac{\partial Q_{tot}^{\text{QMIX}}(\boldsymbol{\tau}, \boldsymbol{a})}{\partial Q_i(\tau_i, a_i)} \geq 0 \tag{3}$$

These two structures are sufficient conditions for the IGM constraint. However, they are not necessary conditions and limits their representation expressiveness of joint action-value functions. There exist tasks whose factorizable joint action-value functions can not be represented by these decomposition methods, as shown in Section 4 in this paper. In contrast, QTRAN [8] transforms IGM to a linear constraint and uses it as soft regularization constrains. However, this relaxation may violate the exact IGM consistency and result in poor performance in complex problems.

3 QPLEX: Duplex Dueling Multi-Agent Q-Learning

In this section, we will first introduce advantage-based IGM, equivalent to the regular IGM principle, and, with this new definition, convert the IGM consistency of greedy action selection to simple constraints on advantage functions. We then present a novel deep MARL model, called *duPLEX dueling multi-agent Q-learning algorithm* (QPLEX), that realizes these constraints without sacrificing its scalability.

3.1 Advantage-Based IGM

The IGM principle ensures the consistency of greedy action selection on the joint and local action-value functions, and constrains the relative order of Q values. We observe that the IGM principle has no constraints on the joint state-value function, which constructs the one-step TD target for deep Q-learning, as shown by Eq. (1). This observation motivates us to naturally reformalize the IGM principle as an advantage-based IGM, which free the joint value function and transform the consistency constraint on advantage functions through the dueling structure $Q = V + A$ proposed by Dueling DQN [12].

Definition 1 (Advantage-based IGM)**.** *For a joint action-value function Q_{tot}: $\boldsymbol{\mathcal{T}} \times \boldsymbol{\mathcal{A}} \mapsto \mathbb{R}$, if there exists individual action-value functions $[Q_i : \mathcal{T} \times \mathcal{A} \mapsto \mathbb{R}]_{i=1}^{n}$, where $\forall \boldsymbol{\tau} \in \boldsymbol{\mathcal{T}}$, $\forall \boldsymbol{a} \in \boldsymbol{\mathcal{A}}$, $\forall i \in \mathcal{N}$,*

$$\textbf{\textit{(Joint Dueling)}} \quad Q_{tot}(\boldsymbol{\tau}, \boldsymbol{a}) = V_{tot}(\boldsymbol{\tau}) + A_{tot}(\boldsymbol{\tau}, \boldsymbol{a}) \text{ and } V_{tot}(\boldsymbol{\tau}) = \max_{\boldsymbol{a'}} Q_{tot}(\boldsymbol{\tau}, \boldsymbol{a'}), \tag{4}$$

$$\textbf{\textit{(Individual Dueling)}} \quad Q_i(\tau_i, a_i) = V_i(\tau_i) + A_i(\tau_i, a_i) \text{ and } V_i(\tau_i) = \max_{a_i'} Q_i(\tau_i, a_i'), \tag{5}$$

such that the following holds

$$\arg\max_{\boldsymbol{a} \in \boldsymbol{\mathcal{A}}} A_{tot}(\boldsymbol{\tau}, \boldsymbol{a}) = \begin{pmatrix} \arg\max_{a_1 \in \mathcal{A}} A_1(\tau_1, a_1), \ldots, \arg\max_{a_n \in \mathcal{A}} A_n(\tau_n, a_n) \end{pmatrix}, \tag{6}$$

then, we can say that $[Q_i]_{i=1}^{n}$ satisfies advantage-based IGM for Q_{tot}.

As specified in Definition 1, the advantage-based IGM takes a duplex dueling architecture, *Joint Dueling* and *Individual Dueling*, which induces the joint and local advantage fucntions by $A = Q - V$. Compared to the regular IGM, the advantage-based IGM transfers the consistency constraint on action-value functions stated in Eq. (2) to that on advantage functions. This change is an equivalent transformation because the value functions do not affect the action selection, as shown by the following proposition.

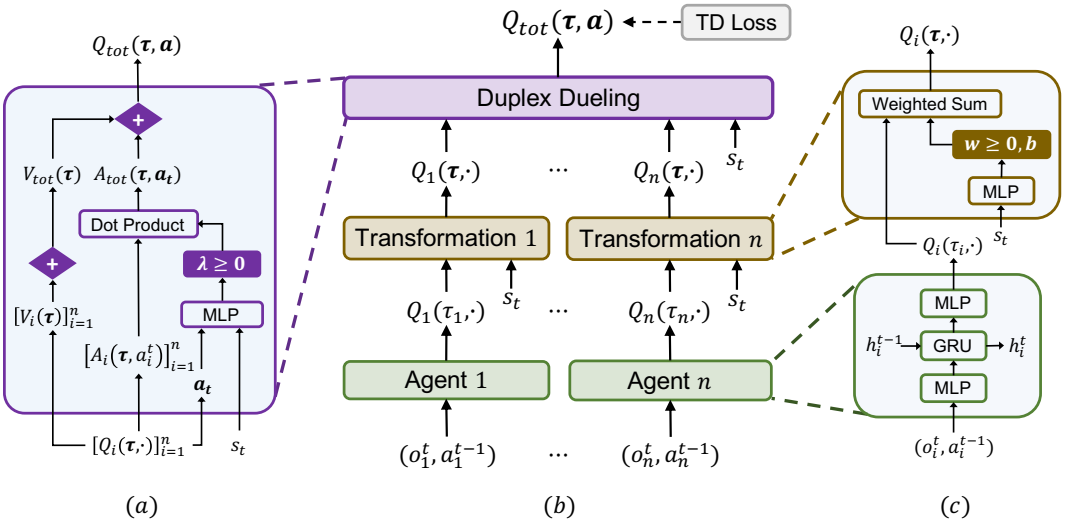


Figure 1: (a) Duplex dueling network structure. (b) The overall QPLEX architecture. (c) Agent network structure (bottom) and Transformation network structure (top). Best viewed in color.

Proposition 1. *The action-value function classes derived from advantage-based IGM and IGM are equivalent.*

One key benefit of using the advantage-based IGM is that its consistency constraint can be easily realized by limiting the value range of advantage functions, as indicated by the following fact.

Fact 1. *The constraint of advantage-based IGM stated in Eq.* (6) *is equivalent to that when* $\forall \boldsymbol{\tau} \in \boldsymbol{\mathcal{T}}$, $\forall \boldsymbol{a}^* \in \mathcal{A}^*(\boldsymbol{\tau})$, $\forall \boldsymbol{a} \in \boldsymbol{\mathcal{A}}$, $\forall i \in \mathcal{N}$,

$$A_{tot}(\boldsymbol{\tau}, \boldsymbol{a}^*) = A_i(\tau_i, a_i^*) = 0 \text{ and } A_{tot}(\boldsymbol{\tau}, \boldsymbol{a}) \leq 0, A_i(\tau_i, a_i) \leq 0, \tag{7}$$

where $\mathcal{A}^*(\boldsymbol{\tau}) = \{\boldsymbol{a} | \boldsymbol{a} \in \boldsymbol{\mathcal{A}}, Q_{tot}(\boldsymbol{\tau}, \boldsymbol{a}) = V_{tot}(\boldsymbol{\tau})\}$.

Fact 1 enables us to develop an efficient MARL algorithm that allows the joint value function learning with any scalable decomposition structure and just imposes simple constraints limiting value ranges of advantage functions in order to achieve a full expressiveness power induced by advantage-based IGM or IGM. The next section will describe such a MARL algorithm.

3.2 The QPLEX Architecture

In this section, we present a novel multi-agent Q-learning algorithm with duplex dueling architecture, called QPLEX, that realizes the advantage-based IGM constraint by exploiting Fact 1. The QPLEX overall architecture is illustrated in Figure 1, which consists of three main modules as follows: (i) an *Individual Action-Value Function* for each agent, (ii) a *Transformation* module that incorporates the information of global state or joint histories into individual action-value functions during the centralized training process, and (iii) the *Duplex Dueling* network module that composes individual action-value functions into a joint value function under the advantage-based IGM constraint. During the centralized training, the whole network is learned in an end-to-end fashion to minimize TD loss as specified in Eq. (1) and, during the decentralized execution, the transformation and duplex dueling modules will be removed and each agent will select their action with its individual Q-function based on its local observation history.

Individual Action-Value Function is represented by a recurrent Q-network for each agent i, which takes last hidden states h_i^{t-1}, current local observations o_i^t, and last action a_i^{t-1} as inputs and outputs local $Q_i(\tau_i, a_i)$.

Transformation network module uses the centralized information to transform local action-value functions $[Q_i(\tau_i, a_i)]_{i=1}^n$ to $[Q_i(\boldsymbol{\tau}, a_i)]_{i=1}^n$ conditioned on the joint observation history, as shown below, for any agent i,

$$Q_i(\boldsymbol{\tau}, a_i) = w_i(\boldsymbol{\tau}) Q_i(\tau_i, a_i) + b_i(\boldsymbol{\tau}), \tag{8}$$

where $w_i(\boldsymbol{\tau}) \geq 0$ is the non-negative weight. This non-negative linear transformation maintains the consistency of the greedy action selection and alleviates the partial observability in Dec-POMDP [8, 9]. As used by QMIX [7], QTRAN [8], and Qatten [9], the centralized information can be the global state s, if available, or the joint observation history $\boldsymbol{\tau}$.

Duplex Dueling network module takes the transformation network outputs as input, e.g., $[Q_i]_{i=1}^n$, and produces the values of joint Q_{tot}, as shown in Figure 1a. This duplex dueling network ensures the IGM consistency between individual action-value functions and the joint action-value function. It first derives the dueling structure for each agent i by computing the value function $V_i(\boldsymbol{\tau}) = \max_{a_i} Q_i(\boldsymbol{\tau}, a_i)$ and the advantage function $A_i(\boldsymbol{\tau}, a_i) = Q_i(\boldsymbol{\tau}, a_i) - V_i(\boldsymbol{\tau})$, then uses individual value and advantage functions to compute the joint value $V_{tot}(\boldsymbol{\tau})$ and the joint advantage $A_{tot}(\boldsymbol{\tau}, \boldsymbol{a})$, respectively, and finally outputs $Q_{tot}(\boldsymbol{\tau}, \boldsymbol{a}) = V_{tot}(\boldsymbol{\tau}) + A_{tot}(\boldsymbol{\tau}, \boldsymbol{a})$ by using the joint dueling structure.

Based on Fact 1, the advantage-based IGM principle imposes no constraints on value functions. Therefore, to enable efficient learning, we use a simple sum structure to compose the joint value:

$$V_{tot}(\boldsymbol{\tau}) = \sum_{i=1}^{n} V_i(\boldsymbol{\tau}) \tag{9}$$

To enforce the IGM consistency of the joint advantage and individual advantages, as specified by Eq. (7), QPLEX computes the joint advantage function as follows:

$$A_{tot}(\boldsymbol{\tau}, \boldsymbol{a}) = \sum_{i=1}^{n} \lambda_i(\boldsymbol{\tau}, \boldsymbol{a}) A_i(\boldsymbol{\tau}, a_i), \text{ where } \lambda_i(\boldsymbol{\tau}, \boldsymbol{a}) \geq 0. \tag{10}$$

The joint advantage function A_{tot} is the dot product of local advantage functions $[A_i]_{i=1}^t$ and non-negative importance weights $[\lambda_i]_{i=1}^n$. This non-negativity induced by λ_i will continue to maintain the consistency flow of the greedy action selection. To enable efficient learning of importance weights λ_i with joint observation histories and action, QPLEX uses a small multi-head attention module [15]:

$$\lambda_i(\boldsymbol{\tau}, \boldsymbol{a}) = \sum_{k=1}^{K} \lambda_{i,k}(\boldsymbol{\tau}, \boldsymbol{a}) \phi_{i,k}(\boldsymbol{\tau}) \upsilon_k(\boldsymbol{\tau}), \tag{11}$$

where K is the number of attention heads, $\lambda_{i,k}(\boldsymbol{\tau}, \boldsymbol{a})$ and $\phi_{i,k}(\boldsymbol{\tau})$ are attention weights activated by a sigmoid regularizer, and $\upsilon_k(\boldsymbol{\tau}) \geq 0$ is a non-negative key of each head. This sigmoid activation of λ_i brings sparsity to the credit assignment of the joint advantage function to individuals, which enables efficient multi-agent learning [16].

With Eq. (9) and (10), the joint action-value function Q_{tot} can be reformulated as following:

$$Q_{tot}(\boldsymbol{\tau}, \boldsymbol{a}) = V_{tot}(\boldsymbol{\tau}) + A_{tot}(\boldsymbol{\tau}, \boldsymbol{a}) = \sum_{i=1}^{n} Q_i(\boldsymbol{\tau}, a_i) + \sum_{i=1}^{n} (\lambda_i(\boldsymbol{\tau}, \boldsymbol{a}) - 1) A_i(\boldsymbol{\tau}, a_i). \tag{12}$$

It can be seem that Q_{tot} consists of two terms. The first term is the sum of individual action-value functions $[Q_i]_{i=1}^n$, which is basically the joint action-value function Q_{tot}^{VDN} of VDN [6]. The second term corrects for the discrepancy between the centralized joint action-value function and Q_{tot}^{VDN}, which enables QPLEX with a full expressiveness power.

Proposition 2. *Given enough hidden neurons in the QPLEX architecture, the joint action-value function class that QPLEX can realize is equivalent to what is induced by the IGM principle.*

Proposition 2 assumes that the neural network of QPLEX can be large enough to achieve the full expressiveness of action-value functions through the universal approximation theorem [17]. This full expressiveness of the action-value function class is very critical for multi-agent Q-learning algorithms. As shown by Wang et al. [11], insufficient representation complexity, like linear value decomposition used by VDN [6] and Qatten [9] and monotonic value decomposition used by QMIX [7], may result in learning divergence in some cases. In contrast, QPLEX shows stable and superior performance in both online and offline data collection, as demonstrated in Section 4.

4 Experiments

In this section, we will first show two didactic examples proposed by [8, 11] to investigate the optimality and convergence of QPLEX. In addition, we evaluate the performance of QPLEX and other

a_2 \ a_1	$\mathcal{A}^{(1)}$	$\mathcal{A}^{(2)}$	$\mathcal{A}^{(3)}$
$\mathcal{A}^{(1)}$	**8**	-12	-12
$\mathcal{A}^{(2)}$	-12	0	0
$\mathcal{A}^{(3)}$	-12	0	0

(a) Payoff of matrix game.

a_2 \ a_1	$\mathcal{A}^{(1)}$	$\mathcal{A}^{(2)}$	$\mathcal{A}^{(3)}$
$\mathcal{A}^{(1)}$	**8.0**	-12.1	-12.1
$\mathcal{A}^{(2)}$	-12.2	-0.0	-0.0
$\mathcal{A}^{(3)}$	-12.1	-0.0	-0.0

(b) Q_{tot} of QPLEX.

a_2 \ a_1	$\mathcal{A}^{(1)}$	$\mathcal{A}^{(2)}$	$\mathcal{A}^{(3)}$
$\mathcal{A}^{(1)}$	**8.0**	-12.0	-12.0
$\mathcal{A}^{(2)}$	-12.0	-0.0	0.0
$\mathcal{A}^{(3)}$	-12.0	0.0	0.0

(c) Q_{tot} of QTRAN.

Table 1: Payoff matrix of the one-step game and joint action-value functions Q_{tot} of QPLEX and QTRAN. Boldface means the optimal/greedy joint action selection from payoff matrix or Q_{tot}.

multi-agent Q-learning algorithms on complex MARL domains, a range of unit micromanagement benchmark tasks in StarCraft II [10]. We compare our method with five state-of-the-art baselines: QTRAN [8], Qatten [9], QMIX [7], VDN [6], and independent Q-learning (IQL) [18] in the didactic and StarCraft II tasks. Qatten [8] has a linear value decomposition structure with an attention-based *Transformation* module for incorporating centralized information and each agent of IQL [18] considers other agents as part of the environment to realize a single-agent setting with non-stationary. The implementation details of QPLEX and five baselines are discussed in Appendix B.1. For the fair evaluation, all experimental results are illustrated with the median performance as well as the 25-75% percentiles over 6 random seeds. The videos of our experiments on StarCraft II are available online[2].

4.1 Didactic Examples

We first demonstrate our method in the didactic cases. To ensure the sufficient data collection in the joint action space, we fix an uniform exploration strategy (*i.e.*, $\epsilon = 1$ in ϵ-greedy exploration) of more than 100k or 2000k steps in the following two didactic problems, respectively.

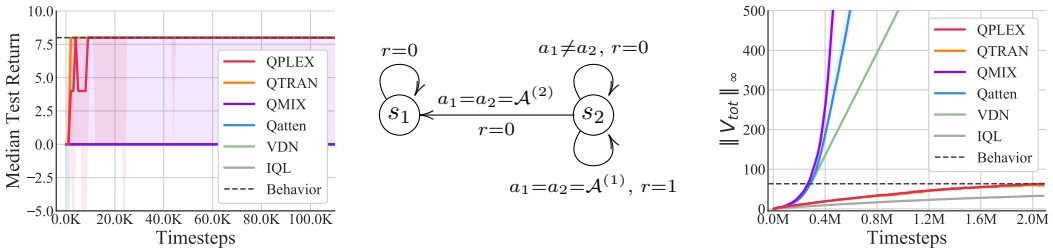
Matrix Game The matrix game proposed by [8] with two agents and three actions is illustrated in Table 1a. This symmetric matrix game describes a simple cooperative multi-agent task with higher miscoordination penalties, whose optimal joint strategy is to perform action $\mathcal{A}^{(1)}$ simultaneously. As shown in Figure 2a, we introduce that only QPLEX and QTRAN with higher expressiveness power can achieve the optimal expected return in the execution stage. Table 1b and 1c demonstrate the optimal joint action-value functions derived from QPLEX and QTRAN, while the non-optimal joint action-value functions of other baselines are deferred to Table 3 in Appendix C.1. The expected return and joint action-value functions empirically reveal that multi-agent Q-learning algorithms with insufficient expressiveness power may fall into the local optimality induced by miscoordination penalties. QTRAN achieves superior performance in this matrix game but may suffer from its relaxation of IGM consistency on StarCraft II benchmark tasks.

Two-State MMDP In this didactic example, we focus on the Multi-agent Markov Decision Process (MMDP) [19], which is a fully cooperative multi-agent setting with full observation. Consider the two-state MMDP proposed by [11] with two agents and two actions (see Figure 2b). Two agents starting at state s_2 explore the extrinsic reward for the 100 environment steps. The optimal policy of this MMDP is simply executing the action $\mathcal{A}^{(1)}$ at state s_2, which is the only coordination pattern to obtain a positive reward. As shown in Figure 2c, QPLEX and QTRAN converge to the optimal infinity norm of joint value function $\|V_{tot}\|_\infty$, while that of other multi-agent Q-learning algorithms with linear or monotonic value decomposition (QMIX, VDN, and Qatten) will diverge to infinity. Moreover, IQL converges to the non-optimal $\|V_{tot}\|_\infty$. This divergence analysis of five baselines has been investigated by Wang et al. [11] through theoretical analysis or empirical verification. Besides QTRAN, we propose another approach QPLEX, which will also converge to the optimal learning performance in this two-state MMDP because of its richer expressiveness power.

4.2 StarCraft II

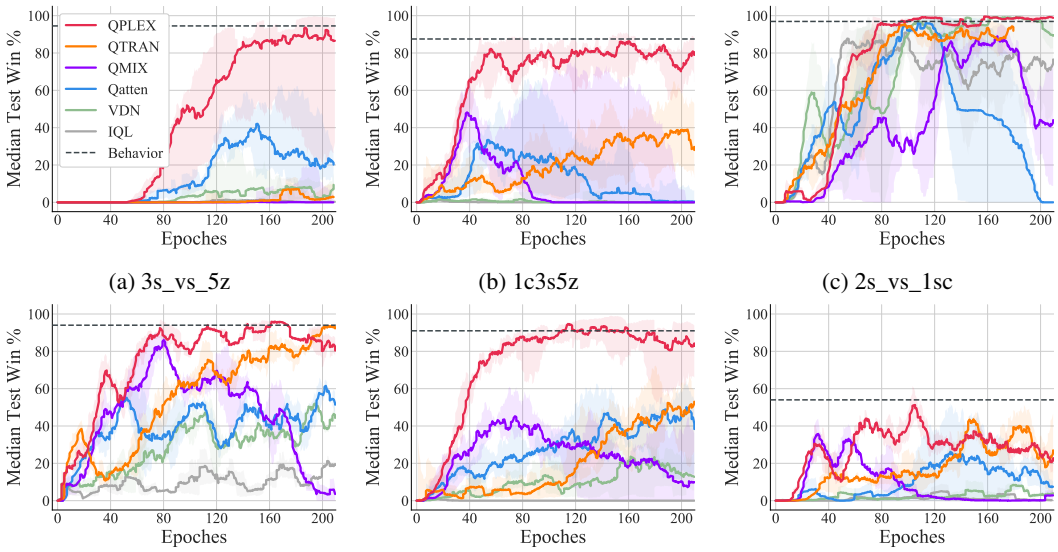
Empirical experiments on more challenging StarCraft II tasks are based on StarCraft Multi-Agent Challenge (SMAC) benchmark [10]. To demonstrate the off-policy nature like that of DQN [13], we adopt the offline data collection procedure of [20], which can only be granted access to a given

[2]https://sites.google.com/view/marl-qplex/



(a) Test return of matrix game. (b) Two-state MMDP. (c) Infinity norm of V_{tot}.

Figure 2: (a) The learning curve of expected return of QPLEX and other baselines in matrix game. (b) An MMDP where mutli-agent Q-learning algorithms with linear or monotonic value decomposition may diverge to infinity. r is a shorthand for $r(s, \boldsymbol{a})$. (c) The learning curve of $\|V_{tot}\|_\infty$ while running several deep multi-agent Q-learning algorithms in the given MMDP.



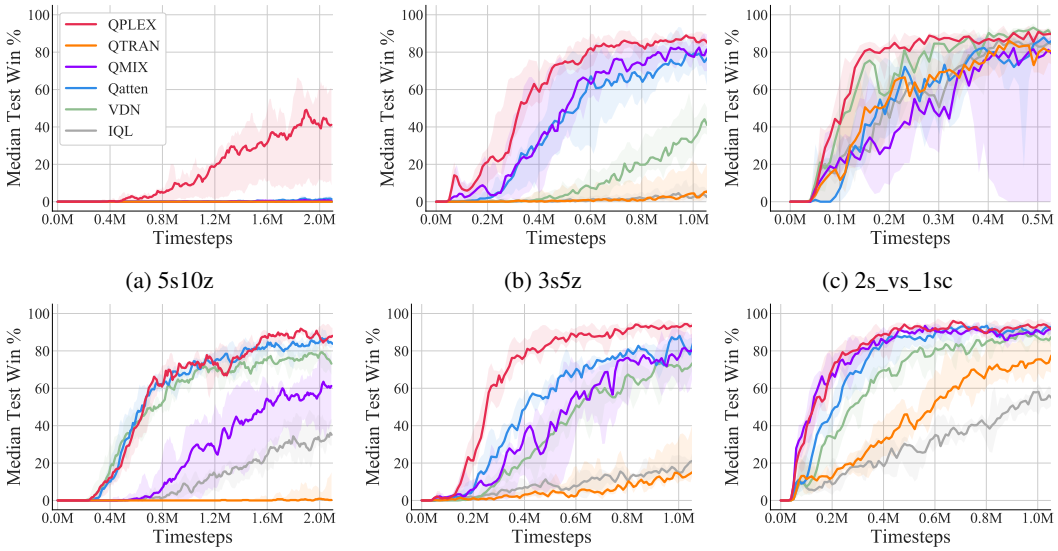
(a) 3s_vs_5z (b) 1c3s5z (c) 2s_vs_1sc

(d) 2s3z (e) 3s5z (f) 2c_vs_64zg

Figure 3: Learning curve of StarCraft II with offline data collection on six different maps.

dataset without additional exploration. We also analyze the empirical performance in another popluar experimental setting with ϵ-greedy exploration and a limited first-in first-out (FIFO) buffer [10], called that with online data collection. Ablation study of QPLEX investigates the influence of *Transformation* module mentioned in section 3.2, which is a popular alleviation trick of partial observability in Dec-POMDP [8, 9]. We pause training every 10k timesteps and evaluate 32 episodes with decentralised greedy action selection to measure the *test win rate*. The detailed experimental setting of StarCraft II refers to Appendix B.2.

Training with Offline Data Collection We illustrate the learning curve of StarCraft II with offline data collection in Figure 3. To construct a diverse dataset, we train a behaviour policy and collect all its experienced transitions throughout the training process (see the details in Appendix B.2). Notably, our method QPLEX significantly outperforms other multi-agent Q-learning baselines including QTRAN and may reach the *Behavior* line. Most of the baselines cannot utilize the off-policy dataset collected by an unfamiliar behaviour policy due to their limited expressiveness power. This argument has been justified by [11] through theoretical and empirical analysis of unexpected projection error induced by the expressiveness limitation. Thus, our full representation expressiveness associated with IGM will protect the off-policy nature of QPLEX in the Q-learning iteration. On the six maps stated in Figure 3, QMIX and Qatten cannot always maintain stable learning performance, while VDN and IQL also suffer from offline data collection and lead to poor empirical results. QTRAN may perform well on certain maps when its soft constraints of relaxation can sometimes be achieved.



(a) 5s10z (b) 3s5z (c) 2s_vs_1sc

(d) 3s_vs_5z (e) 1c3s5z (f) 2s3z

Figure 4: Learning curve of StarCraft II with online data collection on six different maps.

Training with Online Data Collection Figure 4 shows the results of StarCraft II under the online data collection procedure. This popular experimental setting proposed by [10] uses ϵ-greedy exploration strategy with a limited FIFO buffer to construct a dataset online and train the model based on it (see the details in Appendix B.2). Obviously, our method QPLEX also outperforms other baselines by a large margin during the online data collection. On the super hard map 5s10z, the outperformance gap between QPLEX and five baselines exceeds 40% win rate after 2 million steps of training. With the online data collection, most multi-agent Q-learning baselines except QTRAN can achieve reasonable performance. Compared with the offline data collection, we conjecture that the limited representation expressiveness of such baselines may not cause a huge effect empirically. The theoretical benefits of online data collection in multi-agent Q-learning with linear value decomposition [11] may support our speculation. QTRAN shows poor empirical performance in this setting, which may be because its relaxation of IGM consistency affects the online data collection process and leads to unexpected deviations in the training dataset.

Ablation Study We conduct the ablation study on several StarCraft II tasks to evaluate the importance of QPLEX's *Transformation* module described in section 3.2. We call the QPLEX algorithm without *Transformation* module as QPLEX-NT for simplicity, whose implementation details are discussed in Appendix B.1. Figure 5 demonstrates the win rates for QPLEX and QPLEX-NT with offline data collection on 1c3s5z map and other ablation experiments are deferred to Figure 6 in Appendix C.2. These plots show that QPLEX has obvious superior performance than QPLEX-NT. These empirical performance gaps confirm that centralized information during the training phase is indeed beneficial to improve the sample efficiency and final performance, which has been widely used by [7, 8, 9]. This ablation study reveals that the *Transformation* module is critical for QPLEX to achieve the highly scalable and efficient multi-agent Q-learning algorithm.

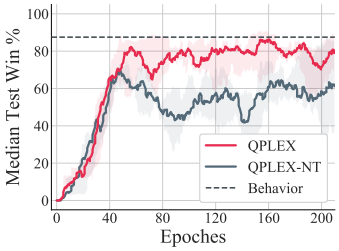


Figure 5: Win rates for QPLEX and its ablation QPLEX-NT with offline data collection on 1c3s5z map.

5 Conclusion

In this paper, we presented a novel multi-agent Q-learning framework within the paradigm of centralized training with decentralized execution. A key assumption of CTDE is the IGM (*Individual-Global-Max*) principle, which ensures the consistency of the optimal joint action selection with

optimal individual action selections. How to efficiently achieve the full expressiveness power induced by IGM with high scalability still remains an open question in the MARL area. Our learning framework takes a duplex dueling network architecture to achieves the first scalable multi-agent Q-learning algorithm with a full expressiveness power induced by IGM. This duplex dueling architecture is used to factorize the joint action-value function, which transforms the IGM principle to easily realized constraints on advantage functions and supports scalable value function learning through a linear decomposition structure. Empirical experiments in StarCraft II tasks demonstrate that our method significantly outperforms state-of-the-art baselines in both online and offline data collection.

Broader Impact

This paper considers a general problem setting and focuses on the theoretical analysis, thus the discussion for the potential broader impact is not applicable.

References

[1] Chongjie Zhang and Victor Lesser. Coordinated multi-agent reinforcement learning in networked distributed pomdps. In *Twenty-Fifth AAAI Conference on Artificial Intelligence*, 2011.

[2] Maximilian Hüttenrauch, Adrian Šošić, and Gerhard Neumann. Guided deep reinforcement learning for swarm systems. *arXiv preprint arXiv:1709.06011*, 2017.

[3] 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.

[4] 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.

[5] Landon Kraemer and Bikramjit Banerjee. Multi-agent reinforcement learning as a rehearsal for decentralized planning. *Neurocomputing*, 190:82–94, 2016.

[6] 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 based on team reward. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, pages 2085–2087, 2018.

[7] Tabish Rashid, Mikayel Samvelyan, Christian Schroeder Witt, Gregory Farquhar, Jakob Foerster, and Shimon Whiteson. Qmix: Monotonic value function factorisation for deep multi-agent reinforcement learning. In *International Conference on Machine Learning*, pages 4292–4301, 2018.

[8] 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. *arXiv preprint arXiv:1905.05408*, 2019.

[9] 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*, 2020.

[10] 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. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*, pages 2186–2188. International Foundation for Autonomous Agents and Multiagent Systems, 2019.

[11] Jianhao Wang, ZhiZhou Ren, Beining Han, and Chongjie Zhang. Towards understanding linear value decomposition in cooperative multi-agent q-learning. *arXiv preprint arXiv:2006.00587*, 2020.

[12] Ziyu Wang, Tom Schaul, Matteo Hessel, Hado Van Hasselt, Marc Lanctot, and Nando De Freitas. Dueling network architectures for deep reinforcement learning. *arXiv preprint arXiv:1511.06581*, 2015.

[13] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015.

[14] Frans A Oliehoek, Christopher Amato, et al. *A concise introduction to decentralized POMDPs*, volume 1. Springer, 2016.

[15] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[16] Tonghan Wang, Jianhao Wang, Chongyi Zheng, and Chongjie Zhang. Learning nearly decomposable value functions via communication minimization. *arXiv preprint arXiv:1910.05366*, 2019.

[17] Balázs Csanád Csáji et al. Approximation with artificial neural networks. *Faculty of Sciences, Etvs Lornd University, Hungary*, 24(48):7, 2001.

[18] Ming Tan. Multi-agent reinforcement learning: Independent vs. cooperative agents. In *Proceedings of the tenth international conference on machine learning*, pages 330–337, 1993.

[19] Craig Boutilier. Planning, learning and coordination in multiagent decision processes. In *Proceedings of the 6th conference on Theoretical aspects of rationality and knowledge*, pages 195–210. Morgan Kaufmann Publishers Inc., 1996.

[20] Scott Fujimoto, David Meger, and Doina Precup. Off-policy deep reinforcement learning without exploration. *arXiv preprint arXiv:1812.02900*, 2018.

Name	Ally Units	Enemy Units
2s3z	2 Stalkers & 3 Zealots	2 Stalkers & 3 Zealots
3s5z	3 Stalkers & 5 Zealots	3 Stalkers & 5 Zealots
1c3s5z	1 Colossus, 3 Stalkers & 5 Zealots	1 Colossus, 3 Stalkers & 5 Zealots
5s10z	5 Stalkers & 10 Zealots	5 Stalkers & 10 Zealots
2s_vs_1sc	2 Stalkers	1 Spine Crawler
3s_vs_5z	3 Stalkers	5 Zealots
2c_vs_64zg	2 Colossi	64 Zerglings

Table 2: SMAC challenges.

A Omitted Proofs in Section 3

Proposition 1. *The action-value function classes derived from advantage-based IGM and IGM are equivalent.*

Fact 1. *The constraint of advantage-based IGM stated in Eq.* (6) *is equivalent to that when* $\forall \boldsymbol{\tau} \in \boldsymbol{\mathcal{T}}$, $\forall \boldsymbol{a^*} \in \mathcal{A}^*(\boldsymbol{\tau})$, $\forall \boldsymbol{a} \in \boldsymbol{\mathcal{A}}$, $\forall i \in \mathcal{N}$,

$$A_{tot}(\boldsymbol{\tau}, \boldsymbol{a^*}) = A_i(\tau_i, a_i^*) = 0 \text{ and } A_{tot}(\boldsymbol{\tau}, \boldsymbol{a}) \leq 0, A_i(\tau_i, a_i) \leq 0, \tag{7}$$

where $\mathcal{A}^*(\boldsymbol{\tau}) = \{\boldsymbol{a} | \boldsymbol{a} \in \boldsymbol{\mathcal{A}}, Q_{tot}(\boldsymbol{\tau}, \boldsymbol{a}) = V_{tot}(\boldsymbol{\tau})\}$.

Proposition 2. *Given enough hidden neurons in the QPLEX architecture, the joint action-value function class that QPLEX can realize is equivalent to what is induced by the IGM principle.*

B Implementation Details and Experimental Settings

B.1 Implmentation Details

In addition, we stop gradients of local advantage function A_i to increase the optimization stability of the max operator of dueling structure. This instability consideration about max operator has been justified by Dueling DQN [12]. We approximate the joint action-value function as

$$Q_{tot}(\boldsymbol{\tau}, \boldsymbol{a}) \approx \sum_{i=1}^{n} Q_i(\boldsymbol{\tau}, a_i) + \sum_{i=1}^{n} \left(\lambda_i(\boldsymbol{\tau}, \boldsymbol{a}) - 1\right) \widetilde{A}_i(\boldsymbol{\tau}, a_i), \tag{13}$$

where $\widetilde{A}_i$ denotes a variant of the local advantage function A_i by stoping gradients.

B.2 StarCraft II

We first describe the scenarios that we consider in details. We consider combat scenarios where the enemy units are controlled by StarCraft II built-in AI (difficulty level is set to medium) and each of the ally unit is controlled by a learning agent. Units of two groups can be asymmetric and the initial placement is random. At each timestep, each agent takes one action from the discrete action space consists of the following action: noop, move[direction], attack[enemy id], and stop. Under the control of these actions, agents move and attack in a continuous map. A global reward that is equal to the total damage dealt on the enemy units is given at each timestep. Killing each enemy unit and winning a combat induces an extra bonus of 10 and 200 respectively. Here we briefly introduce the SMAC challenges in Table 2.

6 seeds is 3 behaviour and two training seeds in each.

C Omitted Figures and Tables in Section 4

C.1 Omitted Tables in Section 4.1

C.2 Omitted Figures in Section 4.2

a_2 \ a_1	$\mathcal{A}^{(1)}$	$\mathcal{A}^{(2)}$	$\mathcal{A}^{(3)}$
$\mathcal{A}^{(1)}$	-7.8	-7.8	-7.8
$\mathcal{A}^{(2)}$	-7.8	-0.0	**-0.0**
$\mathcal{A}^{(3)}$	-7.8	-0.0	-0.0

(a) Q_{tot} of QMIX.

a_2 \ a_1	$\mathcal{A}^{(1)}$	$\mathcal{A}^{(2)}$	$\mathcal{A}^{(3)}$
$\mathcal{A}^{(1)}$	-6.5	-4.9	-4.9
$\mathcal{A}^{(2)}$	-5.0	-3.5	**-3.4**
$\mathcal{A}^{(3)}$	-5.0	-3.5	-3.5

(b) Q_{tot} of Qatten.

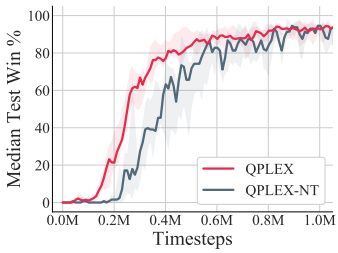
a_2 \ a_1	$\mathcal{A}^{(1)}$	$\mathcal{A}^{(2)}$	$\mathcal{A}^{(3)}$
$\mathcal{A}^{(1)}$	-6.5	-5.0	-5.0
$\mathcal{A}^{(2)}$	-5.0	-3.5	-3.5
$\mathcal{A}^{(3)}$	-5.0	**-3.5**	-3.5

(c) Q_{tot} of VDN.

a_2 \ a_1	$\mathcal{A}^{(1)}$	$\mathcal{A}^{(2)}$	$\mathcal{A}^{(3)}$
$\mathcal{A}^{(1)}$	-5.3	-4.6	-4.7
$\mathcal{A}^{(2)}$	-4.7	-4.0	-4.0
$\mathcal{A}^{(3)}$	-4.7	**-4.0**	-4.0

(d) Averaged individual Q of IQL.

Table 3: The joint action-value functions Q_{tot} or averaged individual Q of QMIX, Qatten, VDN, and IQL. Boldface means greedy joint action selection from individual or joint action-value functions.



(a) Offline data collection of 1c3s5z

Figure 6: Win rates for QPLEX and its ablation QPLEX-NT in both online and offline data collection on three different maps.