TOWARDS FEW-SHOT COORDINATION: REVISITING AD-HOC TEAMPLAY CHALLENGE IN THE GAME OF HANABI

Hadi Nekoei[*]
Mila, Université de Montréal

Xutong Zhao
Mila, Polytechnique Montréal

Janarthanan Rajendran
Mila, Université de Montréal

Miao Liu
IBM Research

Sarath Chandar
Mila, Polytechnique Montréal

ABSTRACT

Cooperative Multi-agent Reinforcement Learning (MARL) algorithms with Zero-Shot Coordination (ZSC) have gained significant attention in recent years. ZSC refers to the ability of agents to coordinate with independently trained agents. While ZSC is crucial for cooperative MARL agents, it might not be possible for complex tasks and changing environments. Agents also need to adapt and improve their performance with minimal interaction with other agents. In this work, we show empirically that state-of-the-art ZSC algorithms have poor performance when paired with agents trained with different methods, and they require millions of samples to adapt to these new partners. To investigate this issue, we formally defined a framework based on a popular cooperative multi-agent game called Hanabi to evaluate the adaptability of MARL methods. In particular, we created a diverse set of pre-trained agents and defined a new metric called adaptation regret that measures the agent's ability to efficiently adapt and improve its coordination performance when paired with some held-out pool of partners on top of its ZSC performance. After evaluating several SOTA algorithms using our framework, our experiments reveal that naive Independent Q-Learning (IQL) agents in most cases adapt as quickly as the SOTA ZSC algorithm Off-Belief Learning (OBL). This finding raises an interesting research question: How to design MARL algorithms with high ZSC performance and capability of fast adaptation to unseen partners. As a first step, we studied the role of different hyper-parameters and design choices on the adaptability of current MARL algorithms. Our experiments show that two categories of hyper-parameters controlling the data diversity and optimization process have a significant impact on the adaptability of Hanabi agents. We hope this initial analysis will inspire more work on designing both general and adaptive MARL algorithms.

1 INTRODUCTION

Our everyday lives are filled with numerous cooperative multi-agent interactions. This includes our everyday regular activities such as crossing a traffic light, driving, buying and selling goods and services, activities at schools, offices and governments to name a few. Artificial Intelligence (AI) agents and systems hold great promise in assisting and helping us with many of these day-to-day activities. It is of paramount importance for these AI agents to have the capabilities to coordinate with multiple humans and other agents (AI or otherwise). Building such AI agents is challenging, as successful cooperation requires an agent to be able to predict and anticipate other agents' behaviors. Often agents have to do this with limited information of the state of world including that of the other agents, and also while other agents' actions being stochastic and changing over time.

Reinforcement Learning (RL) provides a general and scalable framework to model this challenging partially observable, non-stationary, multi-agent learning problem. There has been a recent surge in interest within the AI research community toward building cooperative Multi-Agent RL (MARL) agents. In particular, the community has focused mainly on designing methods for the game of Hanabi (Bard et al., 2020) that enable trained RL agents to perform Zero-Shot Coordination (ZSC) with novel unseen agents (Hu et al., 2020a; 2021; Lupu et al., 2021; Cui et al., 2021; Nekoei et al., 2021; Lucas & Allen, 2022). Hanabi, a partially-observable cooperative multi-agent benchmark, has been a particularly popular game in recent years to study MARL in a cooperative setting. However, most of these efforts are limited to the case of cooperating with novel agents trained independently but with the *same underlying algorithm*. Nekoei et al. (2021) and Lucas & Allen (2022) are among the few approaches trying to evaluate agents in a more general scenario of cooperating with completely novel agents with maybe even different underlying algorithms.

[*] Correspondence to: nekoeihe@mila.quebec.

While ZSC is an important and valuable feature that we would like our cooperative MARL agents to have, just focusing on ZSC is inherently restrictive. In complex tasks and environments (such as in the real world), it may not be possible to learn everything relevant about the environment and the other unseen agents without ever interacting with them. It may not be possible to learn a universal coordination strategy that works well with all novel agents zero-shot. Moreover, the world keeps changing. The environment, including the other agents in it and the task of interest, could change over time. We believe that along with the ability of ZSC, cooperative MARL agents should also have the ability to rapidly adapt with minimal interactions with other agents and improve their performance on top of their ZSC performance whenever possible. The ability to adapt is quite important even after learning to best cooperate at any given moment in the agent's life (Van Seijen et al., 2020). This work aims to formally define adaptation to novel partners in the context of cooperative MARL and propose ways to measure it.

Hanabi requires the agent to possess theory of mind to understand the intent of other agents and cooperate. Bard et al. (2020) designed the Hanabi ad-hoc team play challenge to evaluate a Hanabi agent's ability to play and coordinate with a wide range of teammates the agent has never encountered before.

> *"Good strategies are not unique, and a robust player must learn to recognize intent in other agents' actions and adapt to a wide range of possible strategies."* (Bard et al., 2020)

In particular, during the evaluation phase, the agent's performance is measured via the score achieved by the agent when it is paired with teammates chosen from a held-out pool of agents. However, the details of the ad-hoc teamplay challenge are left open for future work. Recently, there has been a transition in emphasis from addressing the ad-hoc coordination problem to addressing the specific challenge of creating algorithms that can coordinate with independently trained agents, while using the same high-level algorithm also known as Label Free Coordination (LBF) problem (Treutlein et al., 2021). In this work, our goal is to bring back the focus to the ad-hoc coordination challenge which is a more general problem. In particular, we propose to extend this evaluation to measure not just the ZSC of a Hanabi agent, but also the Hanabi agent's ability to efficiently adapt and improve its coordination performance when paired with a held-out pool of agents on top of its ZSC performance. For this, we define metrics such as the *adaptation regret*, which measures the sum of the difference between the best-response score and the score achieved by the RL agent over time in the adaptation phase. Moreover, we investigate the choice of the partners which can greatly affects both zero-shot and adaptation performance of the learner.

We carried out extensive experiments to fine-tune Hanabi agents using various pre-trained partners, and measured their adaptation regret, in order to better understand the current state-of-the-art (SOTA) ZSC algorithms' adaptation capabilities. We included various MARL algorithms from fully specialized Self-Play (SP) agents to the most generalist ZSC algorithms. It was not surprising to find that all of the SOTA algorithms needed millions of interactions with the partners to adapt well, as they lacked any mechanism to learn to adapt quickly during pre-training phase. However, we also discovered that naive Independent Q-Learning (IQL)(Tan, 1993) agents adapted to various partners as quickly as the SOTA Off-Belief Learning (OBL) (Hu et al., 2021) algorithm, which is known to be excellent at ZSC. Therefore, a promising research direction would be to develop MARL algorithms that can perform both ZSC and Few-Shot Coordination (FSC), i.e., minimizing adaptation regret.

Finally, we investigated several hyper-parameters and architecture choices that potentially could influence the adaptability of the baselines. Our experiments show that two categories of hyperparameters (HPs) have a significant impact on the adaptation regret. First, HPs that affect data diversity such as the number of distributed threads and replay buffer size. The second type of HPs is the one influencing the optimization process directly such as finetuning learning rate and batch size. We hope these initial investigations help to give some intuition to others who want to build both general and adaptive agents.

Our main contributions are summarized as follows

- We conceptualize the few-shot coordination (FSC) setting for the ad-hoc teamplay challenge in multi-agent reinforcement learning. We accordingly propose the adaptation regret metric that evaluates how fast an agent adapts to unseen partners.
- We benchmark the adaptation performance of SOTA self-play and zero-shot coordination methods in the game of Hanabi. We discuss the inherent flaws of existing methods when paired with unseen partners.
- We study the effects of partners' diversity and hyper-parameters on the adaptation performance.

2 RELATED WORK

The ad-hoc teamplay challenge in multi-agent reinforcement learning (MARL) requires agents to coordinate with unknown partners who are capable of contributing to the task. Bowling & McCracken (2005) and Stone et al. (2000) were among the first to propose this challenge, while the authors of the Hanabi challenge (Bard et al., 2020) explicitly propose it as a primary benchmark for future progress in cooperative MARL. However, ad-hoc teamplay challenge's details are left to be defined more precisely in the future work.

Hanabi is a popular challenging cooperative game in which multiple different strategies are able to achieve strong performance. These strategies may not be able to coordinate well when paired with each other (Hu et al., 2020a; Nekoei et al., 2021). Since ad-hoc coordination assumes the task to be fully cooperative, and Hanabi elevates theory of mind reasoning about beliefs and intentions of other agents (Bard et al., 2020), Hanabi becomes a highly suitable choice for studying ad-hoc coordination. Other popular benchmark environments either do not exhibit the aforementioned characteristics (e.g., Google football (Kurach et al., 2020), most card games), and/or they have not been evaluated from a coordination perspective (e.g., Starcraft (Samvelyan et al., 2019), MPE (Lowe et al., 2017)), or they are not commonly studied in the literature (e.g., small Hanabi with fewer colors/cards). Nevertheless, the concept of few-shot coordination can be studied in other domains as well, as long as they require different strategies to adapt to novel partners.

Previous work on ad-hoc team play has involved learning diverse sets of policies and using Bayesian optimization, such as in Canaan et al. (2019) which uses the MAP-Elites algorithm (Mouret & Clune, 2015) and an iterative Bayesian optimization approach (Brochu et al., 2010). However, these approaches require meta-information and human knowledge that may not generalize to other problems. Wu et al. (2021) proposes Bayesian Delegation to perform efficient ad-hoc coordination by rapidly inferring the sub-tasks of others. Other related work includes MARL agents with access to observed behavior (Barrett et al., 2017; Peysakhovich & Lerer, 2017; Lerer & Peysakhovich, 2019), communication conventions between agents (Sukhbaatar et al., 2016; Mordatch & Abbeel, 2018), and zero-shot coordination (ZSC) (Hu et al., 2020a; 2021; Nekoei et al., 2021).

In particular, the zero-shot coordination (ZSC) problem has received a surge of interest. In this problem, an algorithm is required to run independently and generate agents with high cross-play performance, i.e., the performance obtained by coordinating with other independently trained agents. To achieve this, methods such as leveraging symmetries of the problem (Hu et al., 2020a), training the best response to a diverse population of agents (Nekoei et al., 2021; Lupu et al., 2021), or making assumptions about prior actions (Hu et al., 2021) have been proposed. In spite of the general definition of ZSC, the MARL community's focus has shifted towards designing agents to produce reproducible policies within the same algorithm, but it does not impose any requirements on these agents to play well with unknown agents such as humans. Despite this, ZSC methods have yielded more versatile agents that have had limited success in transferring to the ad-hoc teamplay setting (Hu et al., 2021).

More recently, Zand et al. (2022) approaches the ad-hoc coordination challenge in the game of Hanabi by considering the problem of selecting a strategy from a finite set of previously trained agents using a posterior belief over the other agents' strategy, to play with an unknown partner. This approach requires having access to a pool of pre-trained policies and more importantly, a policy similar to the partner should be in the pool to get a good ad-hoc coordination performance. Lifelong Hanabi (Nekoei et al., 2021) and Anyplay (Lucas & Allen, 2022) propose inter-crossplay evaluation of hanabi agents which is similar to ad-hoc teamplay challenge. Nekoei et al. (2021) also mentions briefly few-shot evaluation of agents to their partners while learning continually. However, none of these approaches consider a comprehensive evaluation of the adaptation capability of pre-trained Hanabi agents. In the context of the ad-hoc coordination in Hanabi, Canaan et al. (2019) also proposes Quality Diversity algorithms as a class of algorithms to generate populations of diverse agents. These approaches can be complimentary to our benchmark to increase the diversity of our held-out pool of patners.

3 BACKGROUND

Dec-POMDPs: In this paper we consider fully-cooperative Markov games. We model this setting with a Decentralized Partially-Observable Markov Decision Process (Dec-POMDP) (Bernstein et al., 2002; Nair et al., 2003), formally defined as a tuple $G = \{S, A, P, R, \Omega, O, N, \gamma\}$, with the set of states S, the set of actions A, the transition function P, the reward function R, the set of observations for each agent Ω, the observation function O, the number of agents N, and γ as the discount factor. The game is partially observable, with $o^i \sim O(o|i, s)$ as agent i's observation of the global state, sampled from the (stochastic) observation function O. The game is also fully cooperative, thus agents share the same reward $r = R(s, \boldsymbol{a})$, conditioned on the joint action $\boldsymbol{a} = \left[a^i\right]_{i=1}^{N}$ and the global state s. At each timestep t, all agents are at the state s_t. Each agent has an action-observation history (*AOH*) $\tau_t^i = \{o_0^i, a_0^i, r_0^i, \ldots, o_t^i\}$, and

selects action a_t^i using a stochastic policy of the form $\pi_\theta^i(a^i|\tau_t^i)$. The transition function $P(s'|s_t, \boldsymbol{a}_t)$, conditioned on the joint action and the global state, transitions to the next state s_{t+1}. The goal is to maximize the expected return, $J = \mathbf{E}_\tau[R(\tau)]$, where $R(\tau) = \sum_t \gamma^t r_t$ is the discounted cumulative reward calculated using the discount factor γ.

Deep RL: In single-agent deep Q-learning (Mnih et al., 2015), the agent predicts the anticipated total return for each action based on the state information. Given the partially observable setting, the techniques utilized in this study all utilize Recurrent Replay Distributed Deep Q-Networks (R2D2) (Kapturowski et al., 2019) as their foundation. The Q function estimates action values based on the *AOH* instead: $Q(\tau_t, a_t) = \mathbb{E}_\tau\left[R_t(\tau_t)\right]$. The R2D2 algorithm incorporates several modern best practices on top of deep Q-learning, including double-DQN (van Hasselt et al., 2016), dueling network architecture (Wang et al., 2016), prioritized experience replay (Schaul et al., 2016), distributed training setup with parallel running environments (Horgan et al., 2018), and recurrent neural network for dealing with partial observability.

Deep MARL has been successfully employed in various Dec-POMDP settings, as demonstrated by Oliehoek & Amato (2016). The conventional approach for employing deep Q-learning in Dec-POMDP settings is independent Q-learning (IQL) (Tan, 1993). In IQL, each agent treats other agents as part of the environment and learns an independent estimate of the expected return without incorporating the actions of other agents. For the sake of simplicity, this work uses the IQL setup with shared neural network weights θ. We also pre-train agents with Value Decomposition Networks (VDN) algorithm (Sunehag et al., 2017) that learns a joint-action Q-function that consists of the sum of per-agent Q-values to allow for off-policy learning in the multi-agent setting.

ZSC: The prevalent method for learning in Dec-POMDPs is self-play (SP), which involves training RL agents with copies of themselves. However, optimal policies learned through self-play often rely on arbitrary conventions that are agreed upon during training, which can be problematic in real-world scenarios where agents need to coordinate with other unknown AI agents and humans during testing. To address this issue, Treutlein et al. (2021) introduced the Zero-Shot Coordination (ZSC) setting, which requires learning algorithms that produce robust and unique solutions across multiple independent runs, and excludes arbitrary conventions as optimal solutions. Other-play (OP) (Hu et al., 2020b) is a method proposed to prevent agents from learning an arbitrary joint policy out of a set of equivalent but incompatible ones by enforcing equivariance to symmetries in the Dec-POMDP. On the other hand, Hu et al. (2021) proposes Off-belief Learning (OBL) in the ZSC setting, which guarantees convergence to a unique policy by assuming prior actions were taken by a fixed random policy but future actions will be taken by the policy in training. This process can be iterated to train a new, higher-level policy using the trained policy from the previous iteration as the fixed policy.

4 Few-Shot Coordination

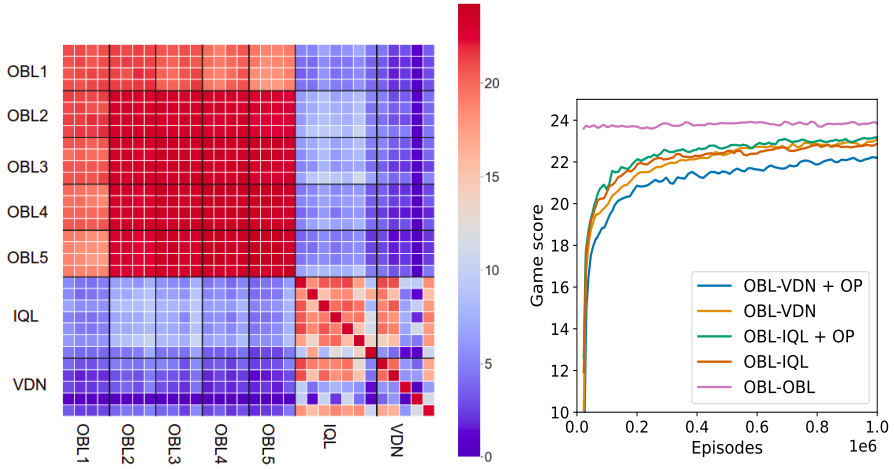
We start by giving the reasons why we need to focus on *Few-Shot Coordination (FSC)* besides Zero-Shot Coordination (ZSC) in section 4.1. Later, in section 4.2, we formalize the FSC setting and explain adaptation regret as a metric to capture both ZSC and FSC capabilities of MARL algorithms.

4.1 Motivation

The ad-hoc teamplay setting (Bard et al., 2020) aims to perform well when the agent is paired with *any* other well-performing policy. Despite its minimal assumptions, as pointed out by Cui et al. (2021), ad-hoc teamplay in Hanabi fails in settings where there is little overlap between good SP policies and those that are suitable for coordination. It can be unreasonable – at least in the game of Hanabi – to expect to have one single algorithm that can perform well with *any* expert partner at test time as required by ad-hoc teamplay challenge.

The difficulty of coordinating with *any* expert partner can be supported by several ZSC works in Hanabi. In ZSC literature, the usual *intra*-crossplay evaluation measures the zero-shot performance of two independent agents learned by the same underlying algorithms. However, previous work has shown a poor zero-shot performance of ZSC algorithms in the *inter*-crossplay criterion, where the agent is paired with other agents pre-trained by different underlying algorithms (Nekoei et al., 2021; Lucas & Allen, 2022). We also confirmed this phenomenon with more diverse agents as shown in Figure 1(a). It is therefore sensible to work beyond zero-shot coordination and aim for fast adaptation towards those expert partners.

Our initial analysis shows that current SOTA MARL methods need an extremely large number of interactions with a new novel partner to adapt and improve their performance on top of their ZSC performance as shown in Figure 1(b). Recognizing the importance of the ability to adapt fast for AI agents, there have been several works recently on this topic in the single-agent RL setting, with a majority of them using techniques of meta-learning to learn to adapt fast. We hope this benchmark plays as a starting point to build such adaptive methods for the cooperative MARL setting.



(a) Cross-Play Matrix (b) adaptation curves

Figure 1: *ZSC and adaptation performance of OBL algorithm with different self-play pre-trained agents. The maximum achievable score in Hanabi is 25. (a) Each row represents an agent with specific hyperparameters/architecture choices with high-level methods separated by black lines. Even though OBL algorithms perform well when paired with other agents trained with OBL, it performs poorly when paired with other SP agents like IQL or VDN. (b) The adaptation performance of an OBL agent when paired with different variants of IQL, VDN, and also an independent OBL agent. It is clear than the OBL agent requires millions of samples to adapt to these novel partners.*

Finally, excellent ZSC algorithms still are far from perfect in coordinating with humans (Cui et al., 2021; Hu et al., 2021). Moreover, even though human players achieve the same performance with rule-based agents and ZSC learned agents, human players still strongly prefer working with the rule-based agent (Kim et al., 2020) viewing the Other-Play agent negatively, citing reasons such as a lack of bilateral understanding, trust, comfort, and perceived performance. Even though there is no guarantee that FSC leads to an increase in subjective metrics without re-conducting the sentiment surveys, intuitively adaptive agents should become more similar to human strategies over time when paired with a human player.

Given these reasons, we believe that the community needs to focus on designing MARL algorithms that are able to adapt quickly with a few interactions to coordinate with AI agents or humans with special strategies while having a generalizable initial strategy capable of ZSC. Now we introduce Few-Shot Coordination (FSC) benchmark as a first step toward designing these algorithms.

4.2 Benchmark Setup

Similar to most MARL settings, our benchmark consists of two phases: the training phase and the evaluation phase. During the training phase, we do not pose any restrictions on the learner agent's learning algorithm, network architecture, or hyperparameter settings. Any method can be used to train the learner agent, including SP, ZSC, or population-based methods.

During the evaluation phase, the objective is to evaluate how quickly the learner can adapt to a pool of unseen partner agents. We define our evaluation metric, adaptation regret, for a learner i and a fixed partner j at evaluation episode T as

$$R_T(i,j) = TC_j^* - \mathbb{E}\left[\sum_{t=1}^{T} C_{ij}^t\right],$$

where C_j^* and C_{ij}^t denote the upper-bound performance for each partner and the performance of learner i adapting to partner j at time-step t, respectively. To aggregate the regret across different partners,

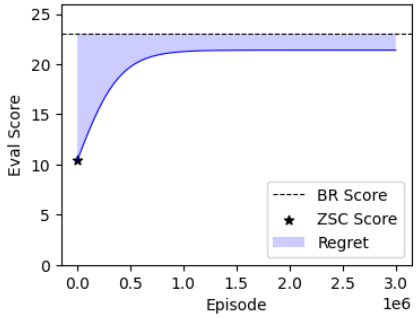
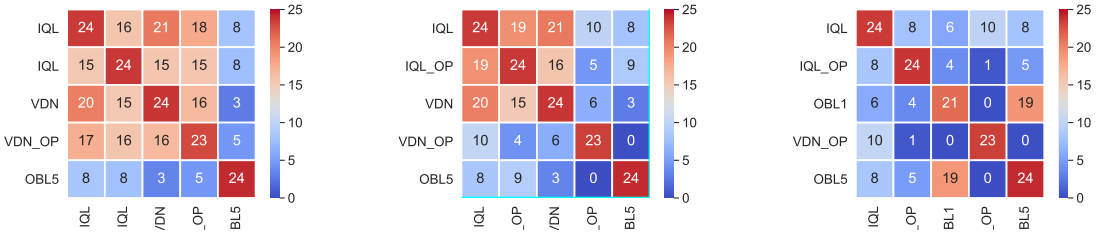


Figure 2: *Adaptation regret metric. This metric measures the sum of the difference between the best-response score and the score achieved by the RL agent over time in the adaptation phase. Adaptation regret captures both ZSC and FSC performance.*



(a) $D : 0.48, S : 0.94$ (b) $D : 0.62, S : 0.94$ (c) $D : 0.78, S : 0.96$

Figure 3: *Three sets of pre-trained Hanabi agents with high strength (S) and different levels of diversity (D). The game score in each cell ranges from 0 to 25.*

we can take an average:

$$R_T(i) = \texttt{aggr}(\{R_T(i,j)\}_{j=1}^{M})$$

where M is the number of partners. Instead of `aggr`, one can use standard averaging across partners or Inter-Quantile Mean (IQM) as suggested by Agarwal et al. (2021) to make the metric robust to outliers. We have several options for setting the upper-bound performance C_j^*. In our current benchmark, we use the maximum achievable score in the game, which is 25. Alternatively, C_j^* could be the best-response (BR) score of partner j, which is the highest possible score that can be obtained by cooperating with partner j. C_j^* could also be the self-play score of the partner j. It is worth noting that the choice of C_j^* does not change the qualitative picture. However, it can lead to large quantitative differences in the results. For instance, if a partner j is difficult to adapt to, using $C_j^* = 25$ will make their contribution to the regret dominate the total regret. The adaptation regret is illustrated in Figure 2. It is worth noting that this metric captures both the ZSC and FSC performance of MARL methods. For each agent, we evaluate its adaptation performance by its mean adaptation regret across a group of partners.

As the adaptation performance highly depends on the set of partners, we assume each partner meets two basic requirements. First, each partner agent should demonstrate its capability of achieving strong cooperation with some other agent. For instance, a SP agent can reach a high score with itself, or a ZSC agent can reach a high ZSC score with another agent optimized with the same ZSC algorithm. This states that each partner j should have a high C_j^* value. We define $S_j = C_j^*/25 \in [0, 1]$ to represent the strength of individual partner j, and $S = \frac{1}{n}\sum_j S_j$ to indicate the strength of a set of partners, where n is the number of partners. This requirement is important as a random partner is less meaningful to evaluate the learner's adaptability. Second, in order to ensure the learner does not overfit to some arbitrarily specialized partners, the set of partners should have diverse playing strategies. Given that a set of partners have high strength S, intuitively they are diverse if at the same time they have low cross-play scores with each other. We define a soft metric to represent the diversity level of partners:

$$D = 1 - \frac{1}{n^2 - n}\sum_{i \neq j}\frac{C_{ij}}{25},$$

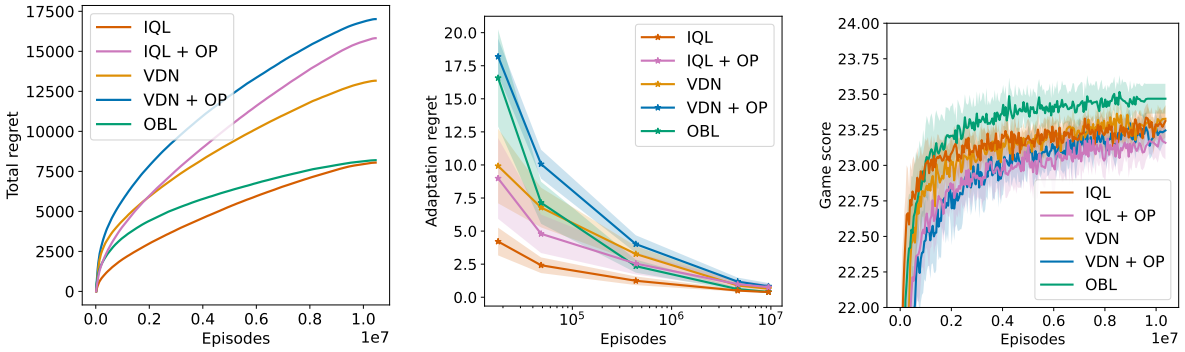
where C_{ij} is the cross-play score of partner i and partner j for a given group of partners. Figure 3 shows three example sets of partners with different levels of diversity. Note that in the case of partners with fully diverse strategies, $D = 1$ and in the case of partners with fully aligned strategies, $D = 0$.

According to these two requirements, each partner could be an agent trained to optimize for SP or ZSC performance. It could also be an offline-learning agent trained with human data or a rule-based agent with hardcoded strategies. We assume each partner remains fixed throughout the evaluation. This assumption is reasonable for the FSC setting in which the goal is to adapt to new partners in a few episodes. Nonetheless, extending our benchmark to include a learning partner is an intriguing avenue that we will leave for future research.

5 Experiments

In this section, we report the benchmarking results of several MARL algorithms following the setting described in section 4.2.

First, we gathered a pool of diverse pre-trained agents including self-play agents such as IQL and VDN and agents trained to perform well in terms of ZSC such as OBL, IQL+OP, and VDN+OP. Then for each method we chose



(a) Total adaptation regret (b) Average adaptation regret (c) Game score

Figure 4: *Results on adapting 5 different agents to partners from Figure 3(b). Refer to Figure 9 in the appendix for the adaptation results of the other two sets of partners from Figure 3. (a) Total regret shows that IQL has lower regret initially than OBL but as OBL finds the best response to the partners, it grows slower than IQL (b) Average adaptation regret over past episodes (c) Game score average across the partners. If zoomed in, IQL adapts faster initially but it converges to a lower final performance. The shaded area is the standard error across 5 different partners.*

randomly an agent as a learner. Finally, based on the aforementioned requirements in section 4.2, we choose the partners as shown in Figure 3 to evaluate adaptation. More details on the agents, their network architecture, etc are described in Table 1 in Appendix A[1]. During adaptation, each partner's policy remains fixed. Moreover, regardless of the training algorithm used to obtain the learner, each learner performs gradient updates as an IQL agent using the same hyperparameter setting as in the pre-training phase. Each learner's adaptation performance against each partner is computed by averaging across 1000 independent games (i.e., seeds). Then for each learner, its adaptation performance is aggregated by averaging the adaptation performance across 5 partners.

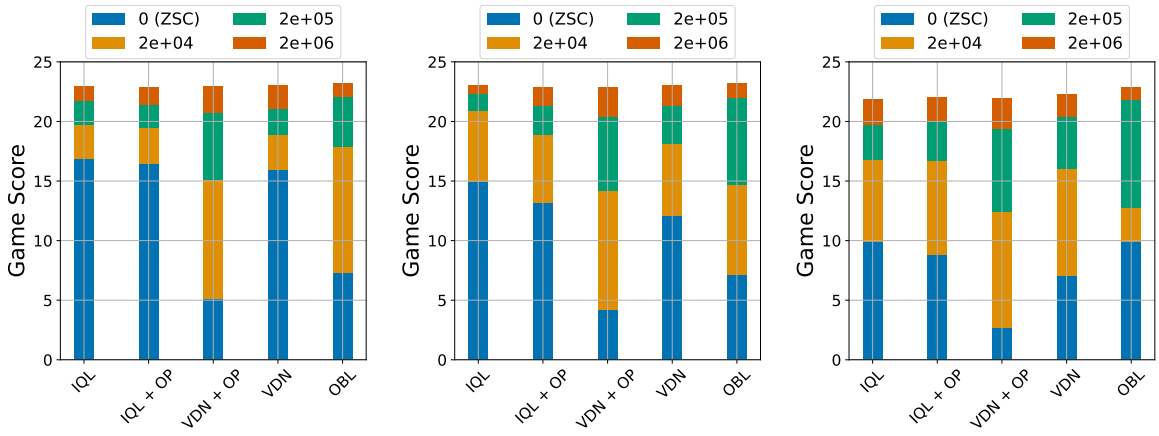
5.1 Benchmark Results

Benchmarking results are summarized in Figure 4, where each curve represents the mean performance of each learner across five partners. The left figure shows the cumulative adaptation regret of each learner over evaluation episodes, the middle figure shows the adaptation regret averaged by the episode number at different points of evaluation, and the right figure shows the actual game score obtained. In Appendix B we provide additional experiments on agents with different levels of strength and diversity.

In general the results demonstrate the lack of efficient adaptability of commonly used methods. To adapt to a partner independently trained with a different algorithm or architecture, they require millions of episodes, which is several orders of magnitude higher than the amount of data needed for few-shot learning in supervised learning.

The SOTA ZSC algorithm OBL achieves a higher game score among the different methods eventually. Since the partners also include an OBL agent, as a ZSC method it is normal to reach a relatively higher final score. However, OBL has a higher average regret than IQL and VDN at least initially, despite its high performance in the ZSC problem where both agents are trained with OBL. This result highlights the significant impact of the choices of partners on performance. Even though the adaptation regret aims to capture both ZSC and FSC performance, looking solely at the adaptation regret sometimes can be misleading as it can be sometimes dominated by either very small or very large ZSC performance. Therefore, we also visualize the game score at different timesteps during adaptation that we care about in Figure 5 for all three sets of partners with different diversity levels. When examining Figure 5(a) and Figure 5(b), we observe that although OBL has achieved a lower game score after $t = 2e4$ adaptation steps compared to other methods, we should note that it has started from a much lower zero-shot performance at $t = 0$ as indicated by the blue color. On the other hand, in the case of high diversity partners shown in Figure 5(c), both IQL and OBL start from the same ZSC performance. However, IQL adapts faster and achieves better performance at $t = 2e4$ but it is eventually outperformed by OBL after $t = 2e5$. Therefore, Figure 5 alongside Figure 4 provides us with a better understanding of the adaptation performance of various methods. Strong ZSC performance does not suffice to guarantee good results when the learner is paired with other methods in the FSC setting. Another interesting observation is that both IQL and VDN reach a better

[1]All the pre-trained agents and the evaluation code are available at https://github.com/chandar-lab/adaptive-hanabi.git.



(a) Partners' Diversity = 0.48 (b) Partners' Diversity = 0.62 (c) Partners' Diversity = 0.78

Figure 5: *Each color represents the game score of a learner adapting to the partners from Figure 3 after different number of episodes denoted by the legends. These plots can be complementary to adaptation regret plots with ZSC performance included explicitly. The* `aggr` *used for this plots is IQM.*

performance than their other-play counterpart. This observation additionally demonstrates that the direct application of successful ZSC techniques may not lead to performance improvement in few-shot adaptation.

We performed an ablation study on the choice of the upper-bound performance C_j^* and found that it has a negligible impact on the average adaptation regret. (Refer to appendix B.3). Nevertheless, if the SP score of each partner is employed as C_j^*, the differences between algorithms become more noticeable regarding the total regret. This is due to the fact that the average regret is already normalized, and modifying the upper-bound score only slightly shifts the regret curves. Nonetheless, this small shift is reflected more prominently in the total regret curve.

5.2 The role of hyper-parameters in adaptation

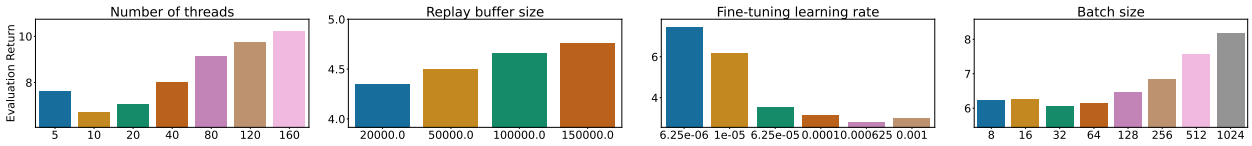
We consider two primary categories of hyper-parameters (HPs) that can impact the adaptation regret. The first category includes HPs that affect data diversity, such as the number of distributed threads and replay buffer size. The second category includes HPs that directly influence the optimization process, such as finetuning learning rate and batch size. We performed the hyper-parameters tuning around the original values used in Hu et al. (2021) and reported the results in Figures 6 and 7. In particular, the effect of the aforementioned HPs on both (a) adaptation regret and (b) the perfect score is studied. While the former measures how sensitive is the adaptability w.r.t these HPs, the latter shows what percentage of the evaluation games are finished with a perfect 25 score for different values of each HP.

Number of threads: `num_threads` and `num_games_per_thread` are two HPs that their multiplication determines the number of parallel independent games running and generating game episodes. A lower number of threads and number of games per thread mean more policy updates per environment step. The original values used by Hu et al. (2021) are `num_theads`=80 and `num_games per_thread`=80. As we decrease the values of these two HPs, the adaptation improves (20 seems to be the best) but if it's too small (10 or 5) it starts to have an adverse effect which can be because of the replay buffer having less diverse experience.

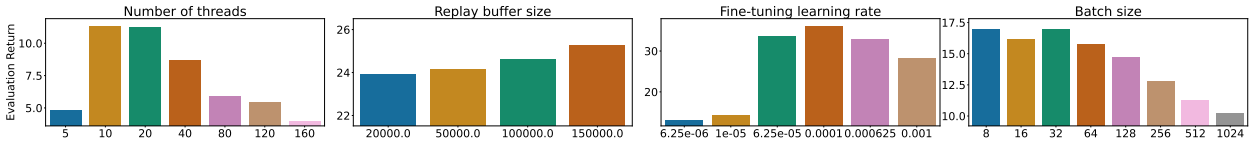
Replay buffer size: `replay_buffer_size` determines the maximum number of episodes stored in the buffer. The original value is $1e5$. From Figure 6, smaller buffer sizes seem to aid adaptation initially, but overly small sizes may reduce data diversity and result in poor performance. The best value is around $5e4$. In Figure 7 however, the smallest buffer size tested seems to perform best.

Fine-tuning learning rate: The optimizer used throughout the experiments is Adam. We tested different values for its learning rate (`lr`). The original value is $6.25e-5$. The best value seems to be a bit higher than the one used to pre-train SP agents ($1e-4$). As expected, very small `lr` leads to small adaptation. However, a too large value of `lr` also might result in overshooting and slow down adaptation.

Batch size: Smaller batch size seems to have a consistent improvement in the case of adapting IQL to the OBL partner. However, a more moderate batch size of around 256 seems to work best when adapting an OBL agent to an IQL agent.

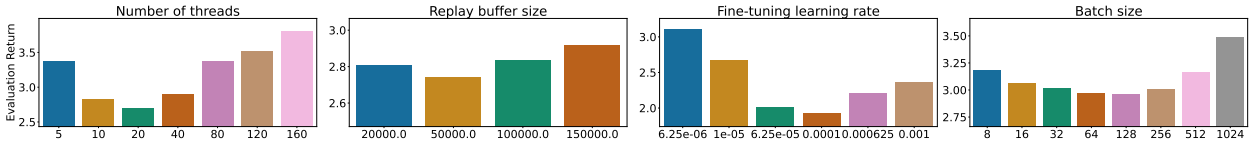


(a) Average adaptation regret, ↓ better

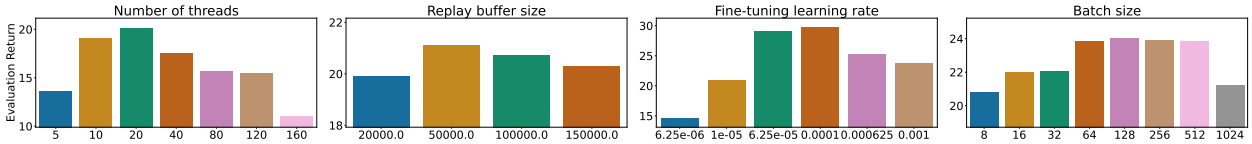


(b) perfect score, ↑ better

Figure 6: *The role of different hyperparameters on adaptability and performance of an IQL agent adapting an OBL agent. (a) Average adaptation regret that is total adaptation regret divided by the number of episodes (b) The percentage of the games finished with a perfect score of 25. All of the HPs show a significant influence on both adaptability and the perfect score.*



(a) Average adaptation regret, ↓ better



(b) perfect score, ↑ better

Figure 7: *The role of different hyperparameters on adaptability and performance of an OBL agent adapting to an IQL agent. (a) Average adaptation regret (b) The percentage of the games finished with a perfect score of 25. All of these HPs show a significant influence on both adaptability and the perfect score.*

While the best-finetuned agent's adaptation performance still falls short of a few-shot learner's capabilities, it is noteworthy to discover the substantial impact of these HPs on adaptation. However, a more intriguing inquiry is how to construct resilient MARL algorithms that can adapt to a new partner without the requirement of careful HP tuning.

6 Conclusions and Future Work

In this work, we motivate the MARL community to focus on the few-shot adaptation problem besides the well-studied zero-shot coordination problem by giving empirical and intuitive reasons including the failure of current SOTA ZSC algorithms in adapting to new partners. Therefore, we propose a benchmark that evaluates the adaptability of MARL algorithms. Our benchmark targets a realistic scenario where the pre-trained agent is paired with unseen partners that are independently trained with potentially different algorithms or architectures. We accordingly introduce adaptability metrics that quantify agents' performance during adaptation and perform an empirical study on a diverse of pre-trained Hanabi agents. Besides, benchmarking several SOTA methods, we performed extensive experiments to investigate the hyper-parameters influencing adaptability.

We believe that our work paves the way for various promising directions for further investigation in this domain. Currently, none of the SOTA methods can adapt well to a group of partners in less than thousands of episodes even with careful hyperparameter tuning. Therefore, developing methods that perform well on this benchmark is an important future work for the community. It would also be valuable to include more recent SOTA methods such as K-level reasoning to determine their impact on adaptation performance. Additionally, investigating the correlation between

adaptability and partner diversity would provide insight into the generalizability of the MARL algorithms. Another potential direction for future research is to include other rule-based and human-cloned bots to examine the algorithms' adaptability to different types of partners. Finally, studying the case with learning partners could provide valuable information on the adaptability of MARL algorithms to agents that are continually evolving and learning. Addressing these areas of research could lead to significant advancements in the development of robust and adaptive MARL algorithms.

As one of the limitations of our work, we defined adaptation regret and discussed the choice of partners only for the case of two-player game. However, our benchmark can also be extended to more than the two-player Hanabi game. The definition of adaptation regret remains unchanged, except that C_j^* and C_{ij}^t represent the upper-bound performance and the learner i's current performance with a group of partners, rather than with a single partner. However, the process of selecting partners becomes more complex as we now have a multi-dimensional cross-play matrix. For example, how to say in a three-player game, two independent groups of partners of size two are diverse? One natural way would be to evaluate all $2 \times \binom{|P|}{3}$ combinations of the partners where $|P|$ is the pool size. Then we can use the same notation of diversity as described before. Although this approach can be extended effortlessly to games with more players, the number of partner combinations grows exponentially with the number of players in the game. Given the popularity of the two-player Hanabi game in the literature, we focus our attention on it in this study. Nevertheless, exploring the setting with more than two players presents an exciting direction for future research.

ACKNOWLEDGEMENTS

This work is supported by the IBM-Mila grant. We acknowledge the computational resources provided by the Digital Research Alliance of Canada. Janarthanan Rajendran acknowledges the support of the IVADO postdoctoral fellowship. Sarath Chandar acknowledges the support of the Canada CIFAR AI Chair program and an NSERC Discovery Grant.

REFERENCES

Rishabh Agarwal, Max Schwarzer, Pablo Samuel Castro, Aaron C Courville, and Marc Bellemare. Deep reinforcement learning at the edge of the statistical precipice. *Advances in neural information processing systems*, 34:29304–29320, 2021.

Nolan Bard, Jakob N Foerster, Sarath Chandar, Neil Burch, Marc Lanctot, H Francis Song, Emilio Parisotto, Vincent Dumoulin, Subhodeep Moitra, Edward Hughes, et al. The hanabi challenge: A new frontier for ai research. *Artificial Intelligence*, 280:103216, 2020.

Samuel Barrett, Avi Rosenfeld, Sarit Kraus, and Peter Stone. Making friends on the fly: Cooperating with new teammates. *Artificial Intelligence*, 242:132–171, 2017.

Daniel S Bernstein, Robert Givan, Neil Immerman, and Shlomo Zilberstein. The complexity of decentralized control of markov decision processes. *Mathematics of operations research*, 27(4):819–840, 2002.

Michael Bowling and Peter McCracken. Coordination and adaptation in impromptu teams. In *AAAI*, volume 5, pp. 53–58, 2005.

Eric Brochu, Vlad M Cora, and Nando de Freitas. A tutorial on bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *arXiv preprint arXiv:1012.2599*, 2010.

Remi Canaan, Julian Togelius, Andrew Nealen, and Stephan Menzel. Diverse agents for ad-hoc cooperation in hanabi. In *2019 IEEE Conference on Games (CoG)*, pp. 1–8. IEEE, 2019.

Brandon Cui, Hengyuan Hu, Luis Pineda, and Jakob Foerster. K-level reasoning for zero-shot coordination in hanabi. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 8215–8228. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper/2021/file/4547dff5fd7604f18c8ee32cf3da41d7-Paper.pdf.

Daniel Horgan, John Quan, David Budden, Gabriel Barth-Maron, Matteo Hessel, and Hado Van Hasselt. Distributed prioritized experience replay. *arXiv preprint arXiv:1803.00933*, 2018.

Hengyuan Hu, Adam Lerer, Alex Peysakhovich, and Jakob Foerster. " other-play" for zero-shot coordination. *arXiv preprint arXiv:2003.02979*, 2020a.

Hengyuan Hu, Adam Lerer, Alex Peysakhovich, and Jakob Foerster. "Other-play" for zero-shot coordination. In Hal Daumé III and Aarti Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 4399–4410. PMLR, 13–18 Jul 2020b.

Hengyuan Hu, Adam Lerer, Brandon Cui, Luis Pineda, Noam Brown, and Jakob Foerster. Off-belief learning. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 4369–4379. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/hu21c.html.

Szymon Kapturowski, Tom Schaul, Bilal Piot, Matteo Hessel, Hado van Hasselt, and Marc Lanctot. Recurrent experience replay in distributed reinforcement learning. *arXiv preprint arXiv:1910.01741*, 2019.

Jinkyoo Kim, Hua Yang, Sungwook Lee, and Kyunghyun Cho. Evaluation of human-ai teams for learned and rule-based agents in hanabi. *IEEE Transactions on Games*, 12(4):411–423, 2020.

Karol Kurach, Anton Raichuk, Piotr Stańczyk, Michał Zając, Olivier Bachem, Lasse Espeholt, Carlos Riquelme, Damien Vincent, Marcin Michalski, Olivier Bousquet, et al. Google research football: A novel reinforcement learning environment. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 4501–4510, 2020.

Adam Lerer and Alexander Peysakhovich. Learning existing social conventions via observationally augmented self-play. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, pp. 107–114, 2019.

Ryan Lowe, Yi I Wu, Aviv Tamar, Jean Harb, OpenAI Pieter Abbeel, and Igor Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. *Advances in neural information processing systems*, 30, 2017.

Keane Lucas and Ross E. Allen. Any-play: An intrinsic augmentation for zero-shot coordination, 2022. URL https://arxiv.org/abs/2201.12436.

Andrei Lupu, Brandon Cui, Hengyuan Hu, and Jakob Foerster. Trajectory diversity for zero-shot coordination. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 7204–7213. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/lupu21a.html.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, ..., and Stig Petersen. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015.

Igor Mordatch and Pieter Abbeel. Emergence of grounded compositional language in multi-agent populations. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

Jean-Baptiste Mouret and Jeff Clune. Illuminating search spaces by mapping elites. *arXiv preprint arXiv:1504.04909*, 2015.

Ranjit Nair, Milind Tambe, Makoto Yokoo, David Pynadath, and Stacy Marsella. Taming decentralized pomdps: Towards efficient policy computation for multiagent settings. In *IJCAI*, volume 3, pp. 705–711, 2003.

Hadi Nekoei, Akilesh Badrinaaraayanan, Aaron Courville, and Sarath Chandar. Continuous coordination as a realistic scenario for lifelong learning. In *International Conference on Machine Learning*, pp. 8016–8024. PMLR, 2021.

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

Alexander Peysakhovich and Adam Lerer. Prosocial learning agents solve generalized stag hunts better than selfish ones. *arXiv preprint arXiv:1709.02865*, 2017.

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*, 2016.

Peter Stone, Gal A Kaminka, Sarit Kraus, and Jeffrey S Rosenschein. Ad hoc autonomous agent teams: Collaboration without pre-coordination. In *Proceedings of the first international conference on Autonomous agents*, pp. 209–216. ACM, 2000.

Sainbayar Sukhbaatar, Rob Fergus, et al. Learning multiagent communication with backpropagation. *Advances in neural information processing systems*, 29, 2016.

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.

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

Johannes Treutlein, Michael Dennis, Caspar Oesterheld, and Jakob Foerster. A new formalism, method and open issues for zero-shot coordination. In *International Conference on Machine Learning*, pp. 10413–10423. PMLR, 2021.

Hado van Hasselt, Arthur Guez, and David Silver. Deep reinforcement learning with double q-learning. In *AAAI*, pp. 2094–2100, 2016.

Harm Van Seijen, Hadi Nekoei, Evan Racah, and Sarath Chandar. The loca regret: a consistent metric to evaluate model-based behavior in reinforcement learning. *Advances in Neural Information Processing Systems*, 33:6562–6572, 2020.

Ziyu Wang, Tom Schaul, Matteo Hessel, Hado van Hasselt, Marc Lanctot, and Nando de Freitas. Dueling network architectures for deep reinforcement learning. In *Proceedings of the 33rd International Conference on Machine Learning (ICML-16)*, pp. 1995–2003, 2016.

Sarah A Wu, Rose E Wang, James A Evans, Joshua B Tenenbaum, David C Parkes, and Max Kleiman-Weiner. Too many cooks: Bayesian inference for coordinating multi-agent collaboration. *Topics in Cognitive Science*, 13(2): 414–432, 2021.

Jaleh Zand, Jack Parker-Holder, and Stephen J Roberts. On-the-fly strategy adaptation for ad-hoc agent coordination. *arXiv preprint arXiv:2203.08015*, 2022.

A EXPERIMENTAL SETUP

In this section, we provide the details of our experimental setup including the type of pre-trained agents in the pool, hyper-parameteres used in the experiments, etc.

A.1 PRE-TRAINED AGENTS TYPES

To create a pool of diverse agents, one way is to per-train agents with different architecture choices. We provide the list of these choices in table 1.

Table 1: All agent types used in the pool.

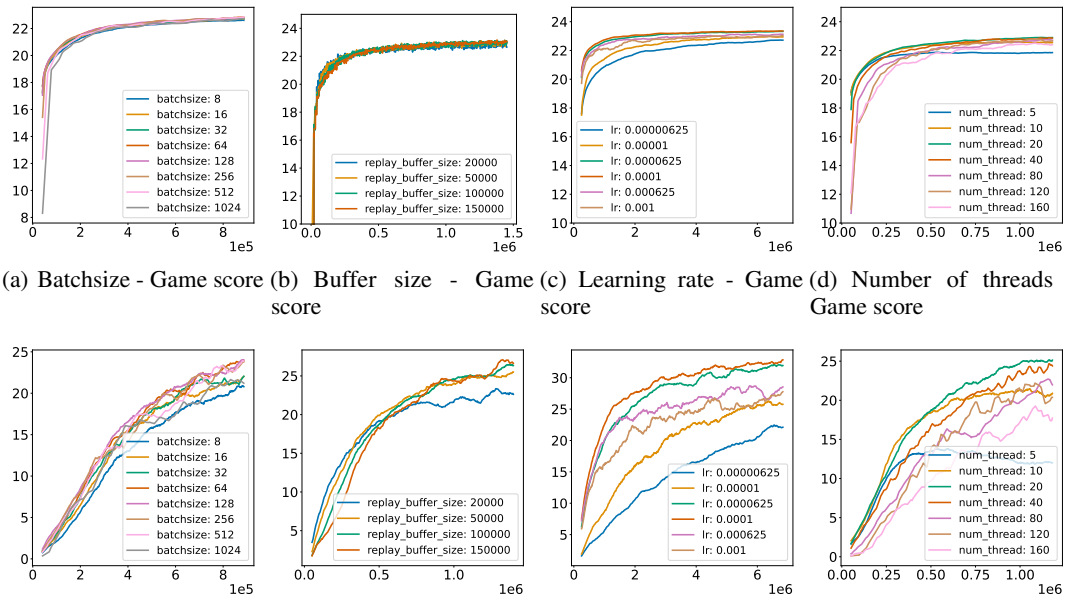
AGENT	RNN TYPE	NUM OF FEED-FORWARD LAYERS	NUM OF RNN LAYERS	RNN HID DIM
TYPE-1	LSTM	1	1	256
TYPE-2	LSTM	1	1	512
TYPE-3	LSTM	1	2	256
TYPE-4	LSTM	1	2	512
TYPE-5	LSTM	2	1	256
TYPE-6	LSTM	2	1	512
TYPE-7	LSTM	2	2	256
TYPE-8	LSTM	2	2	512
TYPE-9	GRU	1	1	256
TYPE-10	GRU	1	1	512
TYPE-11	GRU	1	2	256
TYPE-12	GRU	1	2	512
TYPE-13	GRU	2	1	256
TYPE-14	GRU	2	1	512
TYPE-15	GRU	2	2	256
TYPE-16	GRU	2	2	512

B ADDITIONAL RESULTS

In this section, we discuss the additional results supporting the claims in the main paper.

B.1 FINETUNING CURVES OF SECTION 5.2

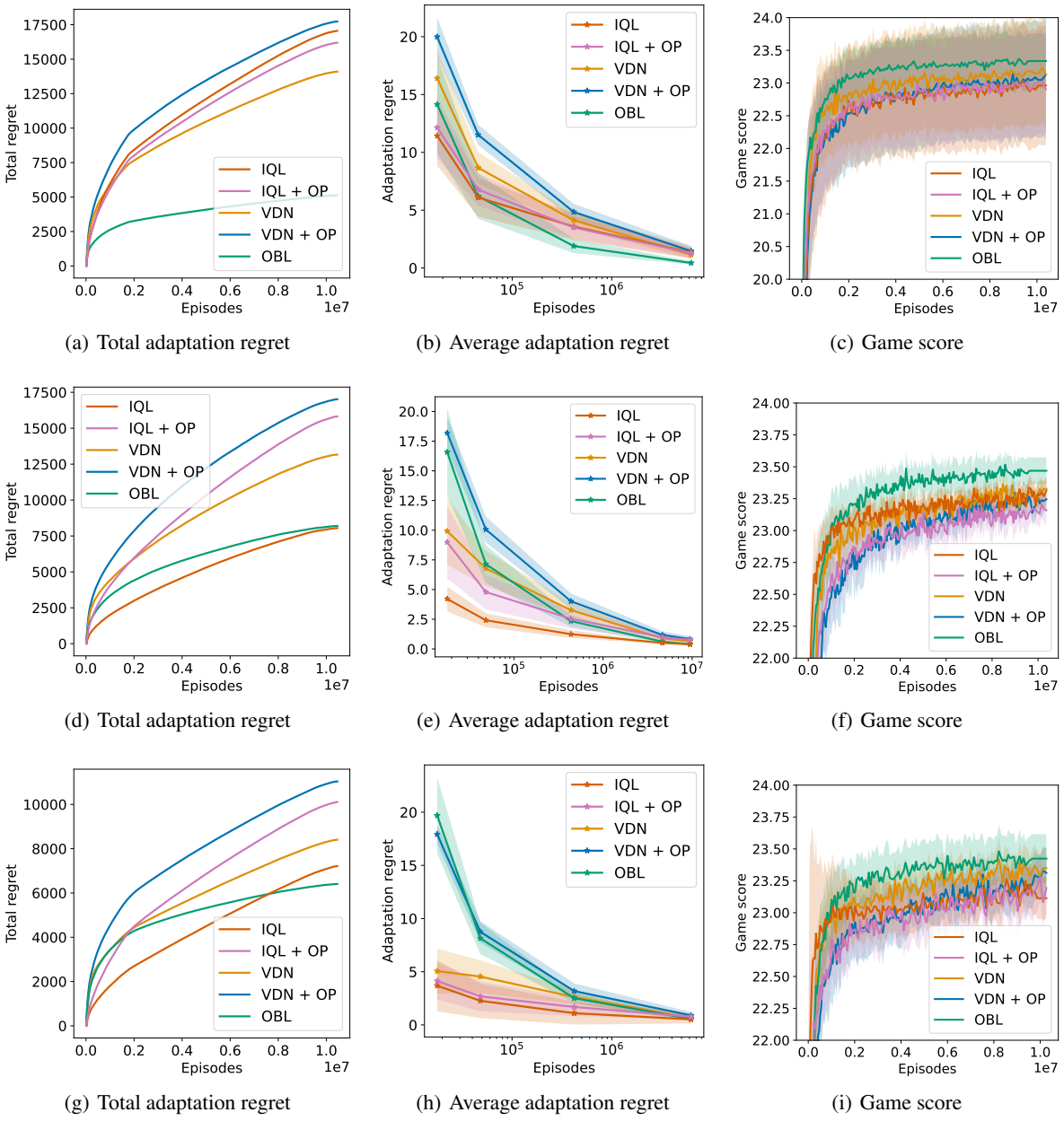
We discussed the role of HPs on both adaptivity and final performance of Hanabi agents in section 5.2. Here we report the adaptation curves and perfect score curves used to generate Figures 6 and 7.



(a) Batchsize - Game score (b) Buffer size - Game score (c) Learning rate - Game score (d) Number of threads - Game score

(e) Batchsize - Perfect score (f) Buffer size - Perfect score (g) Learning rate - Perfect score (h) Number of threads - Perfect score

Figure 8: *The role of different hyperparameters on adaptivity and performance of an OBL agent adapting an IQL agent.*



(a) Total adaptation regret (b) Average adaptation regret (c) Game score

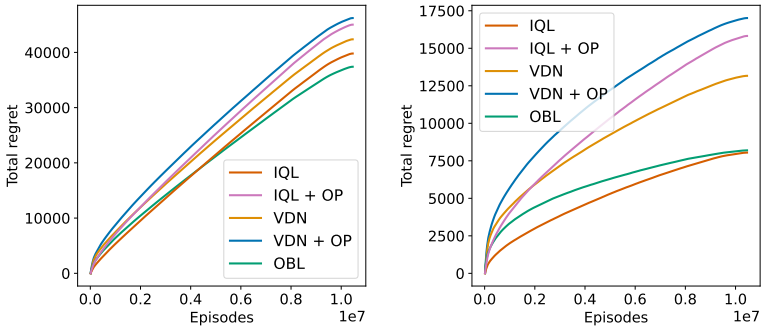
(d) Total adaptation regret (e) Average adaptation regret (f) Game score

(g) Total adaptation regret (h) Average adaptation regret (i) Game score

Figure 9: *Results on adapting 5 different agents to partners from Figure 3(a) (top row) and Figure 3(c) (bottom row).*

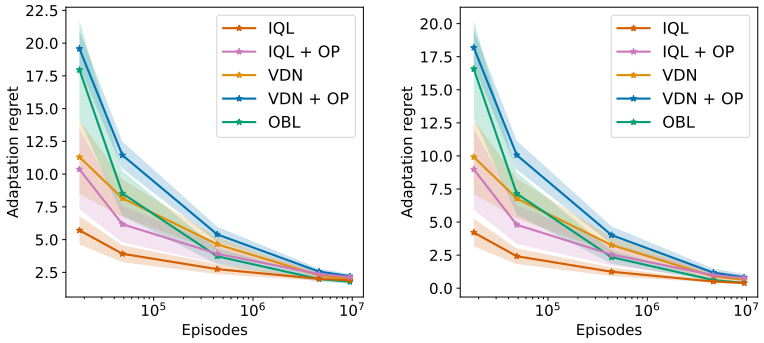
B.2 THE CHOICE OF PARTNERS

B.3 THE CHOICE OF UPPER-BOUND PERFORMANCE



(a) Total adaptation regret - C_j^* = max game score

(b) Total adaptation regret - C_j^* = Self-Play score

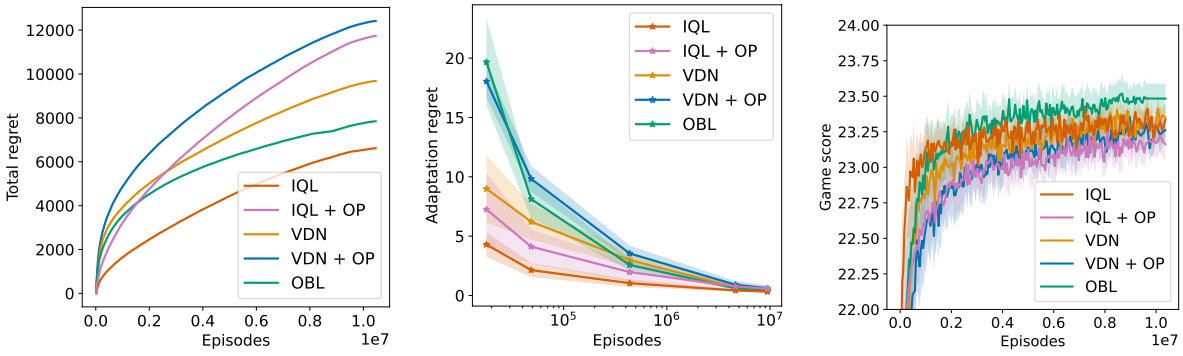


(c) Average adaptation regret - C_j^* = max game score

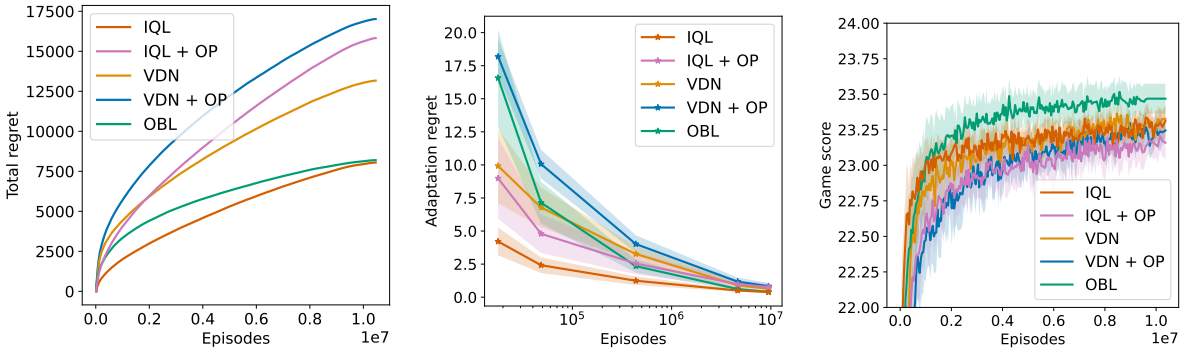
(d) Average adaptation regret - C_j^* = Self-Play score

Figure 10: *The comparison between using different upper-bound performance C_j^* in the adaptation regret definition on benchmark results on adapting 5 different agents to partners from Figure 3(b). It has a negligible impact on the average adaptation regret. Nevertheless, if the SP score of each partner is employed as C_j^*, the differences between algorithms become more noticeable regarding the total regret.*

B.4 THE CHOICE OF THE AGGREGATOR



(a) Total adaptation regret - IQM `aggr` (b) Average adaptation regret - IQM `aggr` (c) Game score - IQM `aggr`



(d) Total adaptation regret - Mean `aggr` (e) Average adaptation regret - Mean `aggr` (f) Game score- Mean `aggr`

Figure 11: *The comparison between mean and IQM aggregator on the benchmark results on adapting 5 different agents to partners from Figure 3(b). While the selection of IQM instead of mean aggregator does not alter the ranking in our particular experiments, it is reflected in the total regret chart by being more resilient to a partner like OBL, which has a considerably low ZSC performance. Consequently, the superiority of IQL as the learner over OBL is more evident when employing IQM aggregator.*